
1. Introduction

“Users should approach all software with prudent caution and healthy skepticism, for the history of science and engineering, including the still-young history of software engineering, is littered with failed promises.” Henry Petroski¹

1.1 An Overview: The Role of Computational Aerodynamics

What is computational aerodynamics (CA)? Theoretical aerodynamics has always provided insights to aerodynamicists through solutions of the governing equations of fluid mechanics. However, before computers became widely available the application of theoretical aerodynamics to specific problems was frequently impractical. Nevertheless, theoretical results from simplified model problems provided important insights which aerodynamicists used as a basis for developing aerodynamic concepts and understanding experimental results. However, aerodynamic design was carried out experimentally; primarily in wind tunnels. Starting nearly thirty years ago, and becoming increasingly important in the last decade, computational aerodynamics has become an important precursor and supplement to the use of the wind tunnel. Computational aerodynamics applies specific solutions of the governing equations of fluid mechanics to the design and analysis of vehicle systems. Usually this means the numerical solution of governing equations rather than numerical evaluation of analytically derived solutions. As soon as computers became available aerodynamicists started using them. The first computational aerodynamics computer programs that were reasonably general and easy to use became widely available in the late '60s, and started providing valuable design information for aerodynamics. Typically, they provided three-dimensional solutions for linear aerodynamics problems, and two-dimensional solutions of the nonlinear boundary layer equations. As with any new technology, this capability arose before engineers understood how to integrate it into the existing design process. Initially proponents claimed that computational aerodynamics would replace wind tunnels. It was well into the '70s before the early promise matured into a realization of the difficulties that would have to be overcome for computed solutions to replace wind tunnels. The wind tunnel is still in use, and, NASA has recently announced its intention to build two new wind tunnels. In the ensuing years computational aerodynamics has become an identifiable new technology, making important contributions to flight vehicle design. Now, there is a distinct body of knowledge that provides a foundation for work in the field.

Computational aerodynamics is one of the most important technologies in the development of advanced vehicles. Many engineers are actively involved in design and analysis using computational aerodynamics. Although numerous books have appeared describing the basic theory of computational fluid dynamics (CFD), guidance on the *application* of these methods is scarce. However, most engineers working in computational aerodynamics are applying these methods, and not developing new algorithms. There is a difference between CFD algorithm development and application skills. CFD algorithm developers have their specific interests and organizations. They are trying to solve fundamental algorithm problems and usually do not use their codes to do aerodynamic design and analysis. As a result, they generally have a poor understanding of the needs of and demands on the user*. Users must understand the algorithms and assumptions employed in the methods, and an education in the *effective use of the computational aerodynamics methods in engineering design and analysis*. The ability to approach aerodynamics problems using computational methods, assess the results, and make engineering decisions requires very different skills and attitudes than those associated with fundamental algorithm development.

Although you cannot use a computational aerodynamics code blindly and expect to obtain valid results, skilled engineers can obtain valuable results when computational aerodynamics is used with some skill, knowledge, ingenuity and judgment. The computer power available to every engineer today is greater than the total computing power available to the engineers who put men on the moon in the Apollo program, and even to those who designed the space shuttle. Unfortunately, it is possible for an engineer using this large computational power to make an error and not catch it. Several structural failures arising from faulty use of computational structures methodology have been documented recently.¹ Thus, significant responsibility accompanies the use of these immense computational capabilities.

It is impossible to anticipate the variety of requests that arise for computational aerodynamics analysis. Although we emphasize aircraft here, computational aerodynamics is also used in the analysis and design of missiles, cars, rotorcraft, submarines and ships. In addition to external flows, CA is used for internal flow problems, including inlets, turbomachinery, and nozzles. Although in a global, long-term sense, computational aerodynamics should replace the wind tunnel, for now this is not the case. Indeed, experimental and computational methods form a good complement to allow aerodynamicists to investigate problems and assess designs.

Typical major goals of computational aerodynamics include:

- vehicle design, *i.e.* development of optimum airfoils and wings for external performance, and inlets, diffusers, and nozzles for internal performance and aero-propulsion integration
- performance: estimation of the drag, lift, and moment characteristics of the vehicle
- definition of loads for structural design (including structural deformation under load)

* Although many developers lack interest in computing drag accurately, a few notable exceptions exist.²

- aeroelastic analysis, including flutter and divergence—requiring coupling with structural analysis and control system design analysis methodology)
- definition of aerodynamic characteristics for evaluation of stability, control, and handling characteristics (*i.e.*, provide the math model for flight simulation).

The current capability doesn't allow computational aerodynamics to accurately satisfy all these goals. Several difficulties prevent the use of computational aerodynamics in the most general situations, and engineering judgment must be exercised to obtain useful results. Difficulties preventing complete numerical simulation include both geometric and fluid mechanics complexity (one simple definition of aerodynamics is 50% flowfield, 50% geometry). The simplest fluid mechanics idealizations are available to provide information at the conceptual and preliminary design stages. Advanced computational methods, which are typically difficult to use and don't yet predict drag well, are used in a different role. The advanced methods are perhaps best used to investigate the detailed physics of the flow. The availability of detailed results over the entire surface, and also everywhere in the flowfield, provides a crucial supplement to wind tunnel testing. Used together, with wind tunnel data providing key anchor points to access and understand the accuracy of the computational method, significant advances in aerodynamic design have been demonstrated. Thus, advanced computational aerodynamics is truly an area where Hamming's adage,[†] "the purpose of computing is insight, not numbers" is true.

1.2 Current Status of Computational Aerodynamics

The capability of computational aerodynamics is continually improving. But, the claims of methodology developers not intimately acquainted with the problems of applying advanced methods should be viewed with caution. Algorithm developers frequently make overly optimistic claims. However, significant technology development resources are being directed toward improving the capability of CFD, and we can expect that in the future we will be relying much more heavily on CFD results alone to make engineering decisions. For example, the recent three-stage, air-launched, winged space booster Pegasus^{TM3} was designed using computational methods alone. No wind tunnel tests were done. The initial launches were successful, and it appeared that the accuracy of the analysis was adequate for this unmanned vehicle. However, after a subsequent launch failure, a dispute arose over whether the aerodynamics had been accurately predicted, or whether the control system was too sensitive to imperfections in the aerodynamic model. The problem was in the lateral-directional characteristics, an area often neglected by code developers.

A recent AIAA Progress Series volume edited by Henne⁴ describes the state of the art in 1990 through many examples of applications (especially note the comments of Ray Hicks, a veteran CFD user and early advocate of the use of CFD in aerodynamics). For example, "normal" 2D airfoil analysis and design can now be done reliably using computational methods.

[†] Hamming authored a numerical methods book many years ago. The quotation cited is the frontispiece of the book.

A prospective computational aerodynamics user should understand the limitations. Bradley and Bhateley⁵ have reviewed the situation and in 1983 proposed a classification scheme in terms of the types of flowfield. They divided the flowfield into seven categories, and categorized the capability to compute each type of flow over a variety of geometries of increasing complexity. Their capability chart is given in Table 1. The only capability they rated as good was attached flow over simple shapes. The capability today is better, but the classification idea is still valid and the capability is still the same in relation to each category.

Table 1. One point of view regarding computational aerodynamics capability.

Status of Computational Capability							
	Attached Flow	Separated Flow	Vortex Flow	Mixed Vortex Attached	Mixed Vortex Separated	Dynamic	Complex Geometry Coupling
Axisymmetric and 2-D	good	fair	poor	poor	poor	fair	N/A ¹
Research Wing/Body	good	fair	fair	fair	poor	fair	poor
Transport Aircraft	good	fair	fair	fair	poor	fair	fair
Fighter Aircraft	fair	poor	fair	fair	poor	poor	poor
Special Purpose Aircraft	fair	poor	poor	poor	poor	poor	poor
from Bradley and Bhateley, reference 4							
1 not applicable							

Case studies provide another way to assess computational aerodynamics impact. Shevell⁶ identified several aerodynamic design problems that arose in flight on various transport aircraft. He examined these problems to determine if the use of computational aerodynamics would have avoided these problems. His conclusion was that *uninformed* computational aerodynamics would likely not have prevented these problems. They included subtle aspects of attached flow airfoil and wing aerodynamics, the ability to compute deep stall characteristics of T-tail aircraft, the use of nacelle strakes to improve high lift and fuselage strakes to improve high alpha directional stability. The impact of CFD is being felt however. Rubbert and Goldhammer⁷ have reviewed the situation at Boeing, and Busch⁸ has reviewed the use of CFD in the design of the YF-23. In the case of the YF-23, an Euler analysis was used quite successfully. Thus, inviscid codes are proving to be of significant value at the project level in aerodynamic design.

However, the conceptual design community has voiced frustration with CFD.⁹ At the conceptual design level decisions are made based on rapid evaluation of the performance potential of a variety of configurations, rather than the detailed study and development of a particular design over a number of years. At this level advanced CFD methods have not yet proven useful. The problem can be traced to the inability of the codes to predict drag directly in a conceptual design sense. Part of the problem here is a miscommunication between the conceptual design aerodynamicists and code developers. Conceptual designers want to know what level of performance can be expected from a configuration *after the aerodynamic design is done*. The aerodynamic design of a single configuration may take months (or years). Although work is in progress to improve the design situation, current advanced CFD methodology is essentially an analysis tool. To discriminate between a series of different candidate configurations using CFD a rapid design capability with accurate estimation of the eventual drag level achievable must be available. Linear theory methods provide some of this capability, but nonlinear methodology for complete configurations on the time-scale of a day is not yet available.

Although absolute values of drag are currently considered too difficult to compute using CFD, the AGARD Panel on CFD and Drag¹⁰ suggested that CFD-based drag prediction was very effective when “embedded in an increment/decrement procedure involving experimental results for complete configurations, and CFD results for simplified configurations.” Another useful application of CFD in this context is the assessment of wind tunnel model support effects and wall interference, which was done for the YF-23.⁸

1.3 Objectives and Guiding Principles in Using Computational Aerodynamics

The objective of this text is to provide an overview of computational aerodynamics as currently practiced, and an understanding of the basis for this technology and the terminology. We will emphasize the assumptions used in the various methods. We provide both the foundations and motivation for further study in computational aerodynamics. We will also use the available computational aerodynamics methods to develop an understanding of applied aerodynamics using computational methods. Although the objective is an emphasis on applications, the underlying theory is provided in some detail. Code implementation details are continually changing. However, much of the fundamental theory is now becoming well defined, and an understanding of the foundations of the methods is essential.

What is more important, we include many examples showing what steps users must take to determine if the answers they are obtaining in their applications are reasonable. How will you know if the answer is right when an engineering decision must be made based on computational aerodynamics? As discussed above, blind acceptance of computed results will lead to problems. Similarly, as described by Hancock,¹¹ advances in computational capability have led to increased demands on experimental aerodynamics. More experimental data must be taken and the conditions must be much more exacting than the level of aerodynamic testing frequently conducted in the past. Examples of

the resulting interplay between computational and experimental work were given recently by Neumann.¹² In addition, code validation has become a field in its own right. Code assessment for the range of validity and accuracy is difficult and time consuming. However, the importance of this step cannot be overemphasized. The issues are described in detail in the paper by Bobbitt,¹³ and the importance of code validation was reinforced in the 1993 Dryden Lecture,¹⁴ which addressed code validation and defined the NASA Ames approach to the problem. The sidebar below is from a recent article by Petroski.¹ Each engineer must test a code before using it to make a decision.

“ Perhaps the most damaging limitation is that software can be misused or used inappropriately by an inexperienced or overconfident engineer.”

“ No software can be proven with absolute certainty to be totally error-free, and thus its design, construction and use should be approached as cautiously as that of any major structure, machine or system on which human lives depend. Although the reputation and track record of software producers and their packages can be relied on to a reasonable extent, good engineering always involves checking them out. If the black box cannot be opened, a good deal of confidence in it and understanding of its operation can be gained by testing.

The proof tests to which software is subjected should involve the simple and ordinary as well as the complex and bizarre. A lot more might be learned about a finite element package, for example, by solving a problem whose solution is known rather than one whose answer is unknown. In the former case, something might be inferred about the limitations of the black box; in the latter, the output from the black box might bedazzle rather than enlighten. In the final analysis it is the proper attention to detail—in the human designer’s mind as well as in the computer software—that makes the most complex and powerful applications work properly.”¹

Thus the objective of any computational aerodynamics work must be:

- Is the answer right?
- Assuming the answer is correct, what is computational aerodynamics revealing about the physics of the flowfield?

In this text current codes are described for each class of methods. This provides the reader with a basis for understanding what capability to expect, and a starting point in searching for an appropriate method. Readers should understand that these surveys are subject to rapid change when describing methods currently considered advanced.

1.4 Typical Steps to Using Computational Aerodynamics, the Art of the Analyst

Given a flowfield or aircraft to examine, we start with a physical problem, and then represent the physical situation with a mathematical model. We then obtain a solution for the mathematical problem and use that solution to deduce something about the physical problem. As noted above, skill

and experience are required to carry out this sequence of steps. In particular, judgment has to be used to select the method to be used. Sometimes, within the allotted budget and time, a CFD solution cannot be used to obtain the desired results. That's why you have engineers and not engineering aides performing the analysis.

The process is given by Rubbert and Tinoco,¹⁵ and is illustrated in Fig. 1, as requiring the following steps:

- Start with the real flow around the aircraft.
- Create a physical model of the flowfield, perhaps (and traditionally) considering it as an inviscid transonic flow, a boundary layer flow and a wake.
- Create the simplified mathematical model(s) to be solved.
- Carry out the numerical solution.
- Examine the results.
- Interpret the sequence of physical model, mathematical model, and numerical solution, together with the computed results to provide the final aerodynamic solution.

Notice here that the numerical solution of a computational problem is a small part of the total engineering process. Successful aerodynamicists must master the entire sequence of steps.

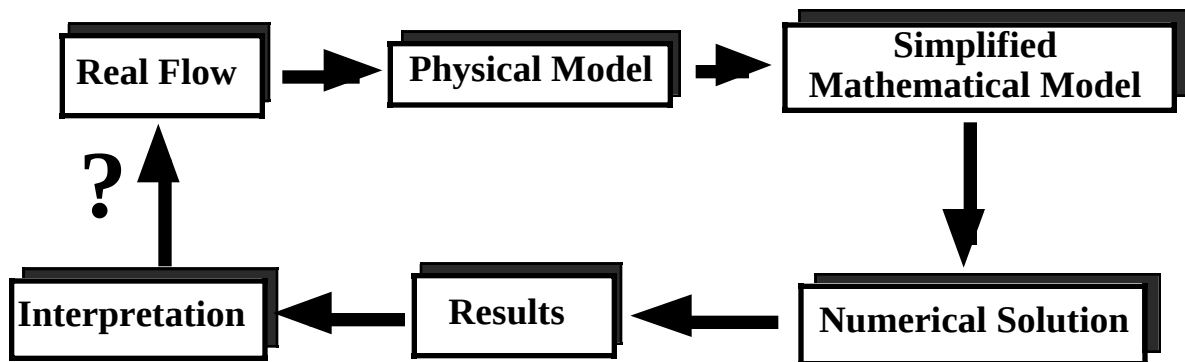


Figure 1. Steps in applying computational analysis to aerodynamics (Ref. 14).

1.5 Design vs Analysis: Computational Aerodynamics in Vehicle Design

Classical: repetitive analysis to design

Although computational fluid dynamics has become a major area of research, its use in the early stages of aircraft configuration development is not generally understood. An incredible variety of problems arise in advanced design, and this precludes the standard use of any simple, uniform procedure. Since the conceptual and preliminary design phases determine the basic configuration architecture, this is the area where improved design methods can make the biggest impact.

New configurations must exploit advanced technology to achieve improved performance over existing designs. In an ideal situation the new aircraft will incorporate new component concepts that have been developed extensively in basic R&D programs. One example is the use of advanced transonic airfoils that in the '60s and early '70s were developed in the wind tunnel. Today they are designed reliably using 2D computational codes such as GRUMFOIL.¹⁶ Another example is the incorporation of the SC3¹⁷ concept in a highly swept fighter design requiring efficient supersonic maneuvering.

A key to the successful development of a configuration is the participation of an experienced team that can project the possibilities for advanced performance without performing the work in detail. This experience base must be the result of having worked extensively with advanced vehicle design and computational methods. Hopefully, with this experience, reliable projections of the performance that can be obtained using computational design methods can be made with confidence.

Linear theory methods are used in conceptual design on a daily and even hourly basis. The aerodynamicist works with the configuration designer to develop a properly balanced design with optimum trimmed performance. This part of the problem can usually be treated well enough for conceptual design using linear methods. However, the high angle of attack characteristics and determination of acceptable control power is still a challenge for all levels of computational aerodynamics codes.¹⁸ Currently, the "timescale" for using advanced codes is too long for conceptual design, approaching the time required for a wind tunnel test. It may require weeks or even months to obtain reliable solutions over a complete new geometry. Figure 2 presents an analogy between wind tunnel testing and computational methods.¹⁹ The geometry definition required for advanced analysis is equivalent to the requirements for fabricating a wind tunnel model. This definition can be time consuming. Thus advanced codes are normally used to assess only a few specific features of a new vehicle concept. That feature might be a unique configuration idea, where you need to evaluate the viability of a new concept to reduce risk, and assure the program managers that the vehicle concept is realistic. Usually there is only time to examine one aspect of the design using advanced codes during the conceptual design phase.

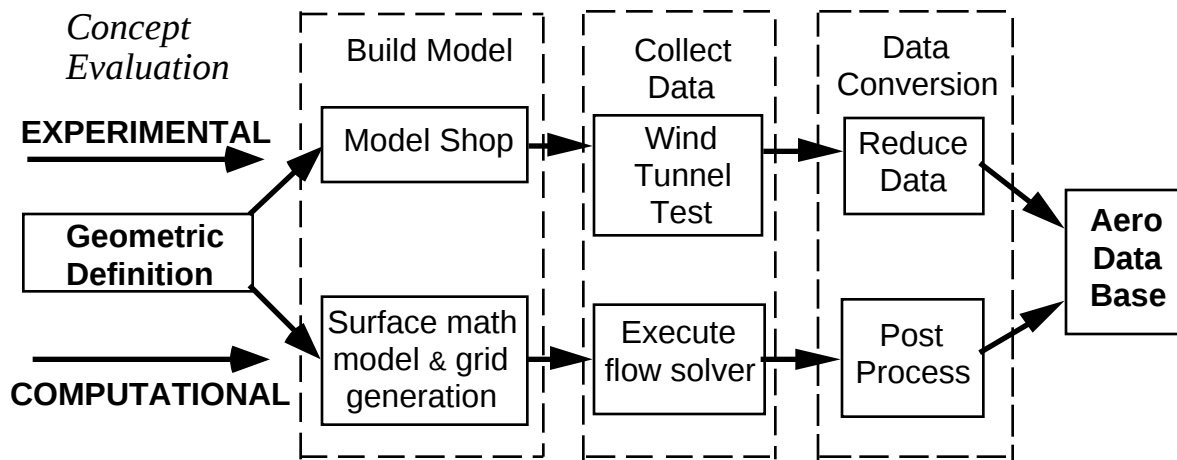


Figure 2. Analogy between computational and experimental aerodynamics, based on a figure by Bengelink.¹⁹

The Grumman X-29 forward swept wing airplane provides an illustrative example. The key idea in the Grumman proposal for a forward swept wing arose due to work on an aft swept wing design using an extremely high performance airfoil, the Grumman “K” foil. That airfoil was developed in 1974, initially by Don MacKenzie and then by MacKenzie working with Paul Bavitz. They used the hodograph method (see Chap. 11), and an early transonic viscous code developed by Bavitz²⁰ when he was on assignment from Grumman to NASA Langley working for Whitcomb. Wind tunnel work by Glenn Spacht (1977) led to the realization that the full performance of this airfoil could be realized only if the wing had a highly swept trailing edge, combined with reduced sweep on the leading edge. An inverse-taper, aft-swept wing or a conventionally tapered forward-swept wing planform provided the only means of meeting these requirements. Today, that conclusion could have been reached using 3D computational methods. The airfoil technology development was done using advanced methods. The rest of the design was done using more conventional (at that time) linear aerodynamic methods, verified and refined during wind tunnel tests.

Thus, computational aerodynamics was directly responsible for the airfoil design which was the core technology that led to the development of the X-29. Nevertheless, the wing design was done with the aid of at least some wind tunnel work. Two other aspects of the design besides the point performance development required testing. To define the aerodynamic model required to design the control system, the aerodynamic characteristics were documented over the complete range of angle of attack and sideslip for every combination of control surface deflections in an extensive wind tunnel test program before the first flight* was possible (note that the control system design is now frequently the reason for programs delays of advanced aircraft—detailed definition of the aerodynamic characteristics is required). Wind tunnel testing was also required to define the load distributions for structural design. The critical design loads usually occur at conditions well away from the design point and involve many loading conditions under separated flow conditions.

* The first flight took place on December 14, 1984, ten years after the airfoil was designed.

Examples of the interplay between configuration design and computational aerodynamics in the near future are the use of tailored forebodies developed using computational aerodynamics to provide specific levels of directional stability at high angle of attack, or components designed to achieve significant regions of laminar flow. The actual preliminary design phase would then integrate these components into the configuration without extremely intense computational aerodynamics work.

The use of computational aerodynamics becomes routine once the configuration geometry is well defined. Computational aerodynamics is heavily used in conjunction with wind tunnel tests once the program moves into a demonstration/validation stage. Often the codes are used to design incremental modifications to the wind tunnel-tested configuration. Many examples of configuration refinements and modifications using computational aerodynamics have been documented at this stage. Perhaps the best example is the design of the nacelle-wing installation on the recent Boeing transports, and especially the new big-engine 737.¹⁵

Advanced: Direct design and optimization

More important, but lagging behind the development of computational aerodynamics as an analysis tool, is the use of the computer to design and optimize the configuration directly. In this role the computer is used in a fundamentally different way than as the computational equivalent of a wind tunnel. This is the most efficient use of computational aerodynamics in vehicle design. Many ideas have been proposed and it is currently an area of active research, but relatively few methods have actually been completely developed. We will discuss and compare these methods in the following chapters. Here, again, the most successful applications have been for airfoils. Further discussion and references can be found in the paper describing “Smart Aerodynamic Optimization,”²¹ and in the recent work in the area by a pioneer of analysis methods, Anthony Jameson.^{22,23}

1.6 A Brief History of Computing Systems and Computational Aerodynamics

The development of computational aerodynamics is closely linked to the development of computers, and more recently computing systems and software. Recall that computers were, at least in part, originally developed for aerodynamics work: the creation of accurate ballistic shell trajectory tables. The other original reasons to develop computers were for cryptography, and subsequently for the nuclear weapons program.

Perhaps the most important early computational aerodynamics work conducted in the '50s and '60s was the work on “the blunt body problem.” At that time the prediction of the heat transfer and flight characteristics of ballistic missiles and manned space capsules entering the atmosphere were the “hot” items in fluid mechanics. Much of this work was done in machine and assembly language. Compilers were not advanced, frequently containing bugs themselves. Programmers thought that compilers produced code that executed much more slowly than code written by professional pro-

grammers in machine or assembly language (the original developers of programmable computers, Von Neuman and Turing, assumed that machine level instruction was sufficient—they never considered high level languages to be necessary). Patching bugs in the executable code was considered possible! These computers were slower and had much less storage than an Apple II Computer. Trying to solve problems that exceed the capability of the computer is a standard feature of computational aerodynamics. Despite rapid advances in computing technology, aerodynamicists always demand more speed and storage; even with the current Cray C-90.

Vortex lattice methods for aircraft applications were reported in 1963 (in Sweden, and at Boeing and Grumman in the US). At that time the vast majority of advanced fluid mechanics work was associated with the space program. Other methods were also under development, with Douglas being a leader.

I first used a computer in 1966 at McDonnell Aircraft in Saint Louis, Mo. We were doing trajectory analysis for hypersonic vehicles. The aerodynamic characteristics were calculated by hand using Newtonian flow theory. Cards were used for input, and some cases were run locally in Saint Louis (we had a “priority” of 5 minutes a night of CPU time, and no one talked about what computer was used). Long jobs were run on an underutilized company computer in Houston and output was flown back on the company plane, which made frequent trips to Houston (these were the days of Project Gemini).

The introduction of the IBM System 360 in the mid '60s revolutionized access to computers for non-specialists. This was the first widely available, easily used computing system. VPI acquired a System 360 at that time, and an addition was made to Burruss Hall to house the computer center. Until then the university computer had been housed in a temporary wood building near the present site of Derring Hall. Initially, the only access to the computer was through submission of a box of cards. Jobs were run in a batch mode, and FORTRAN II was being supplanted by FORTRAN IV. It took hours or even days to get a job back. Students should understand that the computing power available through this process was much less than they have on today's PCs. With the introduction of FORTRAN IV the scientific computing community started using a language that would be stable for many years. CDC introduced the CDC 6000 series computers at about this time, and the CDC 6600 became the computer of choice for scientific computing. Seymour Cray was one of the key designers of that computer. Later the CDC 7600 was introduced, and the CDC 7600 at NASA Ames, using the SCOPE operating system, was the best system I ever used. Our access to the CDC 7600 was still by submission of card decks. We used a CDC development environment known as UPDATE, that was an approach to what is known today as version control. It worked very well.

By 1970 aerodynamicists were solving linearized inviscid three dimensional incompressible flow problems routinely, and two dimensional boundary layer methods were available. The most important problem being tackled in 1970 was the computation of transonic flow. The first solutions

began to be reported around 1970, and the first practical solution procedure was reported in 1971. The first half of the '70s was dominated by the development of solutions for two-dimensional transonic flow. Three-dimensional transonic small disturbance theory solutions also began to appear.

At a major conference at NASA Langley* in 1975 one speaker drew on the rapid advances in capability to present a chart which could be used to project that computational aerodynamics would be fully developed by 1984.

A joke by the CFD researchers at NASA Ames in the mid-70s reflects the attitude of the time:

Question: "What do you use wind tunnels for?"

Answer: "They are places with lots of space, where you store your computer output"

Well, this shows that prediction is tricky, especially when it involves the future. (YB)

The early explosive advance in capability did not continue, and progress slowed. Advances became much more difficult. Why? Computational fluid dynamics (CFD) development became more rigorous, and:

- i) Complexity of three-dimensional flowfields is not just "one more dimension."
 - algorithms for the Euler equations were difficult to develop
 - computer storage requirements made remarkable demands on computers
 - handling this much information, pre- and post- computation is a job in itself
- ii) Separated flow solutions were required.
 - numerical algorithms required further development
 - after the storage requirements began to be overcome, the inevitable limits of turbulence models became apparent
- iii) Real life arbitrary geometry presented surprising challenges.
 - grid generation became a discipline in its own right
- iv) Software development is the "tar pits" of engineering (see Chap. 3)
 - as more people work on a code the productivity decreases dramatically, individuals can no longer single-handedly create a complete new code

The situation today is again fluid. Key new developments in computing revolve around the dramatic advances in workstation technology, graphical interaction with results, and the computation of solutions using massively parallel processing technology. After working in an essentially stable

* "Aerodynamic Analysis Requiring Advanced Computers," March 4-6, 1975. (see NASA SP-347)

computing environment for twenty years, computational aerodynamicists will be using exciting new computer hardware and new software products in the near future.

One advance in productivity is the adoption of UNIX as a common operating system on most current scientific computing systems. Although this can be an emotional subject, computational aerodynamics users benefit greatly. Code developers and computer systems people typically work with a single operating system. But, as a user, I have been in the position of having to compute on many different systems, frequently all in the same day. As an example, consider an actual case where proficiency in IBM CMS and TSO, VAX VMS, and Cray COS were required during the course of a single day's work effort. Each system had a different text editor! UNIX and *vi*, the common text editor, eliminates this problem.

1.7 Typical Method/Code Development Cycle

Code development is a long process. Initial efforts to develop the algorithm are only a portion of the effort. Figure 3 shows the typical process. Initial demonstration of a new algorithm is often done by a single CFD researcher and may take a year or two. Technology development is typically carried out by several engineers, applying the method using pilot codes tailored to solve a specific problem. The code itself is continually adjusted to obtain improved results and handle unforeseen situations. This activity may also last one or two years.

If the method provides a significant improvement compared to existing codes, it may be developed into a production code. Frequently computer science majors direct this activity. An attempt is made to anticipate future requirements, and make a very general code. This may also take years.

Once the production code is finished, the user has a code which typically cannot be changed without major coordination with support groups. Since five years may have elapsed since the method was originated, more advanced methods may already be available in the technology development stage, and the production code *may not be used as often as expected*. Instead, the latest capability may be available in a new pilot code. Thus pilot codes are frequently used in advanced vehicle analysis and design to get the best possible answers. How this problem is best handled requires the use of engineering judgment. The continually changing software problem adds another complication to the practice of computational aerodynamics.

One significant philosophical change emerged in code development over the last decades. Initially, each code had its own input geometry definition and graphics package to examine the results. Because this information is common to most codes, these tasks were separated, and the codes began to be designed so that the geometry/mesh generation became distinct from the flow solver. In addition, the graphical analysis was separated. Essentially, the results were stored as databases to be examined with another program. This allowed for more efficient code development. For the flow solver part of the problem this meant that the code required pre- and post-processing software. This work could be

done independently of the flow solver. This approach, illustrated in Fig. 4, allows much more flexibility. Generally, it's much better for the user, although for some simple calculations it's much simpler if the flow solver handles grid generation and graphical output. This is sometimes possible in two-dimensional analysis, but rare for three-dimensional analysis, where the amount of data generated can be overwhelming.

An important consideration is the teamwork approach. Figure 4 showed the split of the code tasks. The use of a team to develop codes is just as important as the codes themselves. Whitfield has recently described his own experiences.²⁴ He found that using teams with individuals responsible for specific parts of the work is the key to good productivity. In particular, geometry definition and grid generation are areas where considerable skill and dedication are required. In his lab one individual is responsible for all the grid generation work. This is a graduate student or research engineer level activity, and productivity has improved using this approach. Similarly, post processing using computational flow visualization has been done by dedicating people to do this work exclusively. In this case undergraduate students are capable of handling the job. Consistent with my own experience, Whitfield has found that the *codes* do not produce results, but *people* produce results using the codes. The human element is at least important as the software and hardware. Finally, to simplify problems comparing results from different flow models, Whitfield is now using only Euler or Navier-Stokes solutions, and has stopped using the simpler flow models shown in Fig. 4.

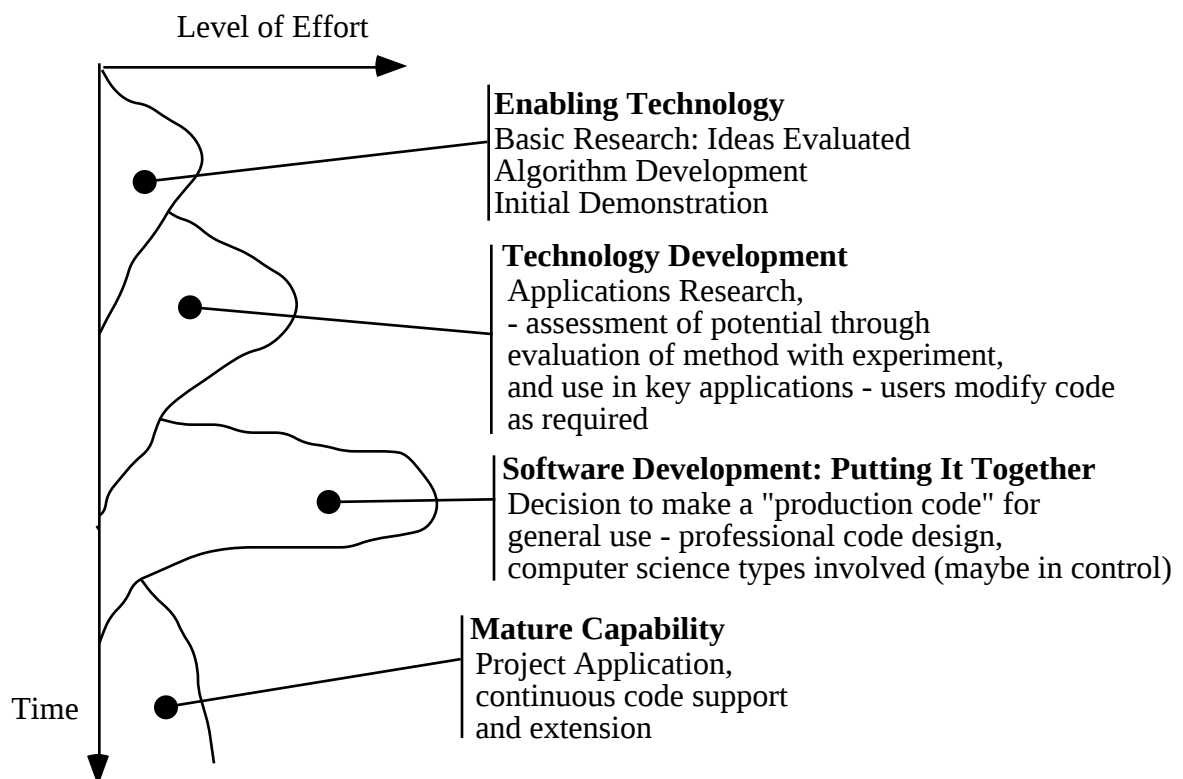


Figure 3. Code Development Process

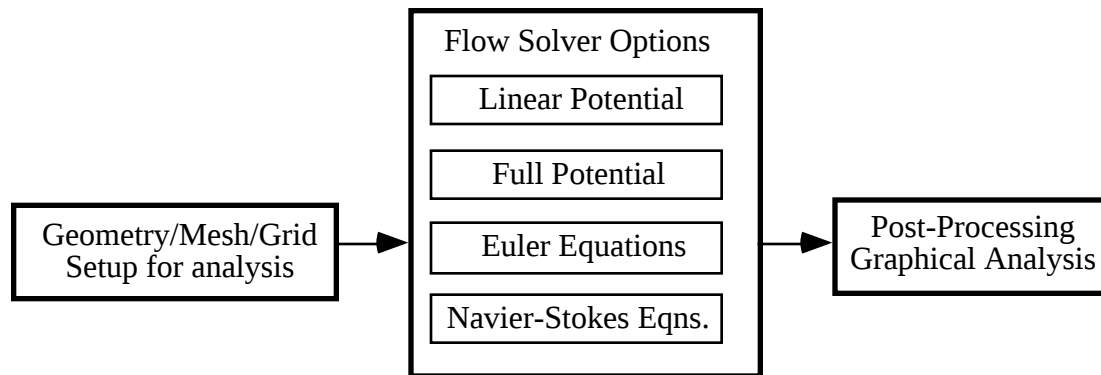


Figure 4. Typical split of functions in a CFD software system.

1.8 Overview of the following chapters

This text provides a systematic development of computational methods starting with early “pre-CFD” methods of computational aerodynamics that are still useful in aerodynamic design and analysis. These methods are used to introduce essential aspects of applied aerodynamics for airfoils and wings and an introduction to drag calculation methods. The basic ideas of CFD and grid generation are then discussed, followed by a presentation of viscous effects and transonic flows in aerodynamics. Finally, a discussion of extensions required to treat high speed aerodynamics problems is presented. In each case the theory is described, fundamental assumptions identified, a numerical implementation is presented, and examples illustrating the use of the method to understand aerodynamic design and analysis are given. Having acquired insight into basic applied aerodynamics, a brief tutorial covering the current advanced methods is presented. We conclude with a review of good computational aerodynamics procedures required to use computational aerodynamics in practice.

Upon the completion of the text you should be able to assess a problem for analysis using computational aerodynamics, formulate the problem, select a method, and obtain a solution. Then you should be able to use engineering judgment to decide if you have a valid engineering answer.

1.9 Exercise

Pick an airfoil. Select any airfoil of your choice for which you can find geometry, and the *experimental* pressure distributions, and force & moment data.

1. Plot the airfoil.
2. Plot the pressure distribution at one angle of attack.
3. Plot the force and moment data over a range of angles of attack. Make sure to include the drag polar.
4. Turn in a cover sheet describing airfoil and data source (make sure to include the test conditions: Reynolds number, Mach number and transition details, i.e., fixed or free transition. If fixed, where is it fixed?).

Caution: You will use these data to compare with results from a computer program, and this data set will play a role in several assignments, so pick an airfoil you can use all semester. See Appendix A for additional information on airfoils, Appendix B for sources of data, and Appendix C for directions on the presentation of results (all of which aren't repeated here).

1.10 References

1. Petroski, H., "Failed Promises," *American Scientist*, Vol. 82, Jan-Feb. 1994, pp.6-9.
2. Jameson, A., "Re-Engineering the Design Process through Computation," AIAA Paper 97-0641, Jan. 1997.
3. Mendenhall, M.R., Lesieutre, D.J., Whittaker, C.H., and Curry, R.E., "Aerodynamic Analysis of Pegasus™ — Computations vs Reality," AIAA Paper 93-0520, Jan. 1993.
4. Henne, P.A., ed., *Applied Computational Aerodynamics*, AIAA, Washington, 1990.
5. Bradley, R.G., and Bhateley, I.C., "Computational Aerodynamic Design of Fighter Aircraft - Progress and Pitfalls," AIAA Paper 83-2063, August 1983.
6. Shevell, R.S., "Aerodynamic Bugs: Can CFD Spray Them Away?" AIAA Paper 85-4067, October, 1985.
7. Rubbert, P., and Goldhammer, M., "CFD in Design: An Airframe Perspective," AIAA Paper 89-0092, Jan. 1989.
8. Busch, R.J., Jr., "Computational Fluid Dynamics in the Design of the Northrop/McDonnell Douglas YF-23 ATF Prototype," AIAA 91-1627, June 1991.
9. Snyder, J.R., "CFD Needs in Conceptual Design," AIAA Paper 90-3209, September, 1990.
10. Sloof, J.W., ed., "Technical Status Review on Drag Prediction and Analysis from Computational Fluid Dynamics: State of the Art," AGARD-AR-256, May 1989.
11. Hancock, G.J., "Aerodynamics - the role of the computer," *Aeronautical Journal*, August/September, 1985, pp. 269-279.
12. Neumann, R.D., "CFD Validation - The Interaction of Experimental Capabilities and Numerical Computations," AIAA Paper 90-3030, August, 1990.
13. Bobbitt, Percy J., "The Pros and Cons of Code Validation," AIAA Paper 88-2535, June, 1988.
14. Marvin, J.G., "Dryden Lectureship in Research, A Perspective on CFD Validation," AIAA Paper 93-0002, Jan. 1993.
15. Rubbert, P.E., and Tinoco, E.N., "Impact of Computational Methods on Aircraft Design," AIAA Paper 83-2060, August, 1983.
16. Melnik, R.E., Chow, R.R., Mead, H.R., and Jameson, A., "A Multigrid Method for the Computation of Viscous/Inviscid Interaction on Airfoils," AIAA Paper 83-0234, 1983.
17. Mason, W.H., "A Wing Concept for Supersonic Maneuvering," NASA CR 3763, 1983.
18. Morrocco, J.D., "Lockheed ATF Team Cities Lessons Learned in Prototyping Effort," *Av. Wk.*, Nov. 5, 1990, pg. 87.

-
19. Bengelink, Ronald L., "The Integration of CFD and Experiment: An Industry Viewpoint," AIAA Paper 88-2043, 1988.
 20. Bavitz, P.C., "An Analysis Method for Two-Dimensional Transonic Viscous Flow," NASA TN D-7718, 1974.
 21. Aidala, P.V., Davis, W.H., Jr., and Mason, W.H., "Smart Aerodynamic Optimization," AIAA Paper 83-1863, July 1983.
 22. Jameson, Anthony, "Successes and Challenges in Computational Aerodynamics," AIAA Paper 87-1184, 1987.
 23. Jameson, A., "Aerodynamic design via control theory " ICASE Report No. 88-64, Nov. 1988.
 24. Whitfield, David L., "Perspective on Applied CFD," AIAA Paper 95-0349, January, 1995.

2. Getting Ready for Computational Aerodynamics: Fluid Mechanics Foundations

from AIAA 82-0315, by D.R. Carlson

We need to review the governing equations of fluid mechanics before examining the methods of computational aerodynamics in detail. Developments in computational methods have resulted in a slightly different approach to the fundamental conservation statements compared with pre-computer classical presentations. The review also establishes the nomenclature to be used in the rest of the chapters. The presentation presumes that the reader has previously had a course

in fluid mechanics or aerodynamics. Many excellent discussions of the foundations of fluid mechanics for aerodynamics application are available. Karamcheti¹ does a good job. Other books containing good discussions of the material include the books by Bertin and Smith,² Anderson,³ and Moran.⁴ The best formal derivation of the equations is by Grossman.⁵

2.1 Governing Equations of Fluid Mechanics

The flow is assumed to be a continuum. For virtually all aerodynamics work this is a valid assumption. One case where this may not be true: rarefied gas dynamics, where the flow has such low density that the actual molecular motion must be analyzed. This is rarely important, even in aero-space plane calculations. Aeroassisted Orbital Transfer Vehicles (AOTV's) are the only current vehicles requiring non-continuum flowfield analysis.

2- 2 Applied Computational Aerodynamics

The fluid is defined by an equation of state and the thermodynamic and transport properties, *i.e.*, the ratio of specific heats, γ , viscosity, μ , and the coefficient of heat conduction, k . Governing equations and boundary conditions control the motion of the fluid. The governing equations are given by conservation laws:

- | | |
|------------|--|
| • mass | continuity |
| • momentum | Newton's 2 nd Law, $\mathbf{F} = m\mathbf{a}$ |
| • energy | 1st Law of Thermodynamics |

Coordinate systems are also important in aerodynamics. The general equations of fluid motion are independent of the coordinate system. However, simplifying assumptions frequently introduce a *directional bias* into approximate forms of the equations, and require that they be used with a specific coordinate system orientation relative to the flowfield.

Cartesian coordinates are normally used to describe vehicle geometry. In this chapter we will work entirely in the Cartesian coordinate system. It is frequently desirable to make calculations in non-Cartesian coordinate systems that are distorted to fit a particular shape. General non-orthogonal curvilinear coordinates are discussed in Chapter 9. Even when using Cartesian coordinates, the x , y , and z coordinates are oriented differently depending on whether the flow is two- or three-dimensional. Figure 2-1 shows the usual two-dimensional coordinate system. The standard aerodynamics coordinate system in three dimensions is illustrated in Fig. 2-2.

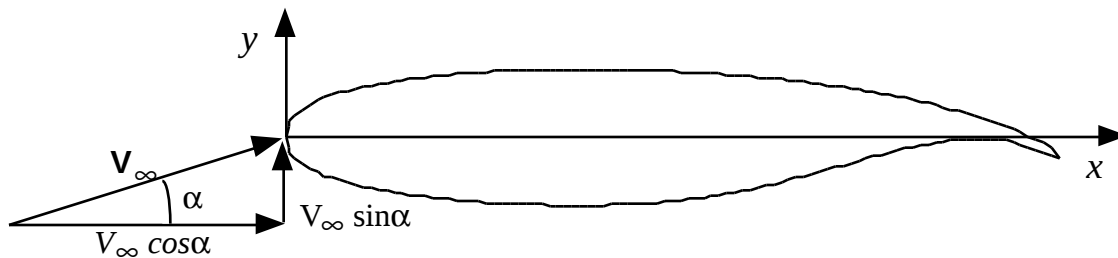


Figure 2-1 Coordinate system for two-dimensional flow.

In general Cartesian coordinates, the independent variables are x , y , z , and t . We want to know the velocities, u , v , w , and the fluid properties; p , ρ , T . These six unknowns require six equations. The six equations used are provided by the following:

continuity	1	equation(s)
momentum	3	"
energy	1	"
equation of state	1	" .

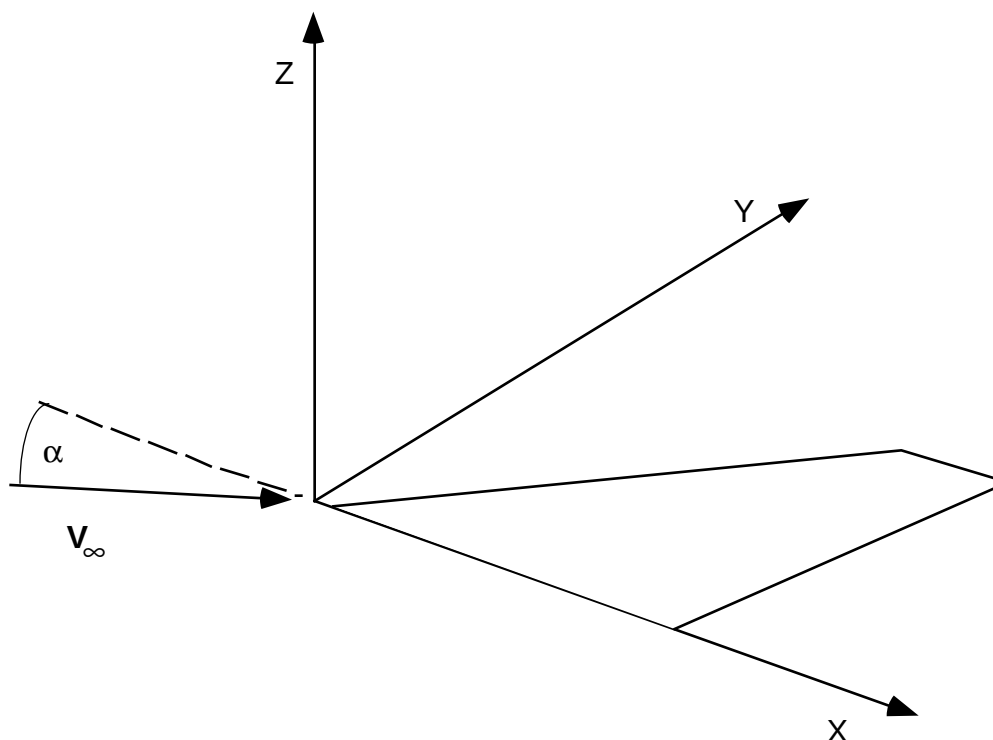


Figure 2-2 Standard coordinate system for three-dimensional flow.

Assumptions frequently reduce the number of equations required. Examples include incompressible, inviscid, irrotational flow, which can be described by a single equation, as shown below. Prior to the 1980s almost all aerodynamics work used a single partial differential equation, possibly coupled with another equation. An example of this approach is the calculation of potential flow for the inviscid portion of the flowfield, and use of the boundary layer equations to compute the flowfield where an estimate of the viscous effects is required.

2.2 Derivation of Governing Equations

We now need to develop a mathematical model of the fluid motion suitable for use in numerical calculations. We want to find the flowfield velocity, pressure and temperature distributions. The mathematical model is based on the conservation laws and the fluid properties, as stated above. Two approaches can be used to obtain the mathematical description defining the governing equations.

- I. *Lagrangian*: In this method each fluid particle is traced as it moves around the body. Even in steady flow, the forces encountered by the particle will be a function of its time history as it moves relative to a coordinate system fixed to the body, as defined in Figs. 2-1 and 2-2. This method corresponds to the conventional concept of Newton's Second Law.

- II. *Eulerian*: In this method we look at the entire space around the body as a field, and determine flow properties at various points in the field while the fluid particles stream past. Once this viewpoint is adopted, we consider the distribution of velocity and pressure throughout the field, and ignore the motion of individual fluid particles.

Virtually all computational aerodynamics methods use the Eulerian approach. The use of this approach requires careful attention in the application of the conservation concepts, and Newton's second law in particular. Since these two approaches describe the same physical phenomena, they can be mathematically related. Karamcheti¹ provides a particularly good explanation of the ideas underlying approaches to the governing equations in his Chapters 4-7. Newton's Law governs the motion of a fixed fluid particle. However, to establish a viable method for computation, aerodynamicists employ the Eulerian approach, and define a control volume, which maintains a fixed location relative to the coordinate system. The connection between the rate of change of the properties of the fixed fluid particle (velocity, density, pressure, *etc.*) and the rate of change of fluid properties flowing through a fixed control volume* requires special consideration. The substantial derivative, discussed below, is employed to define the rate of change of fixed fluid particle properties as the particle moves through the flowfield relative to the fixed coordinate system. An integral approach to the description of the change of properties of a fluid particle relative to the fixed coordinate system is available through the use of the Reynolds Transport Theorem, which is described by Owczarek⁶ and Grossman⁵ (section 1.2).

The conservation equations can be expressed in either a differential or integral viewpoint. The differential form is the most frequently used in fluid mechanics analysis and textbooks. However, many numerical methods use the integral form. Numerically, integrals are more accurately computed than derivatives. The integral form handles discontinuities (shocks) better. The differential form assumes properties are continuous. We will use aspects of each approach.

* The concept of a "control volume" arose as an engineering requirement for a means to formulate the physical description to allow calculations to be made. It differs from the viewpoint adopted by physicists. An explanation of the concept's origins is contained in the book by Walter G. Vincenti, *What Engineers Know, and How They Know It: Analytical Studies from Aeronautical History*, John Hopkins Univ. Press, 1990. The chapter is entitled "A Theoretical Tool for Design: Control Volume Analysis 1912-1953."

2.2.1 Conservation of Mass: the Continuity Equation

In this section we derive the continuity equation from a control volume viewpoint (in 2D), and then we look at the equivalent integral statement and the use of the Gauss Divergence Theorem to establish the connection. Other derivations are given by Moran⁴ (sections 2.2, 2.3, 2.4) Anderson³ (chapters 2 and 6), and Bertin and Smith² (chapter 2).

The statement of conservation of mass is in words simply:

$$\begin{array}{ccc} \text{net outflow of mass} & & \text{decrease of mass} \\ \text{through the surface} & = & \text{within the} \\ \text{surrounding the volume} & & \text{volume.} \end{array}$$

To translate this statement into a mathematical form, consider the control volume given in Fig. 2-3. Here, u is the velocity in the x -direction, v is the velocity in the y -direction, and ρ is the density.

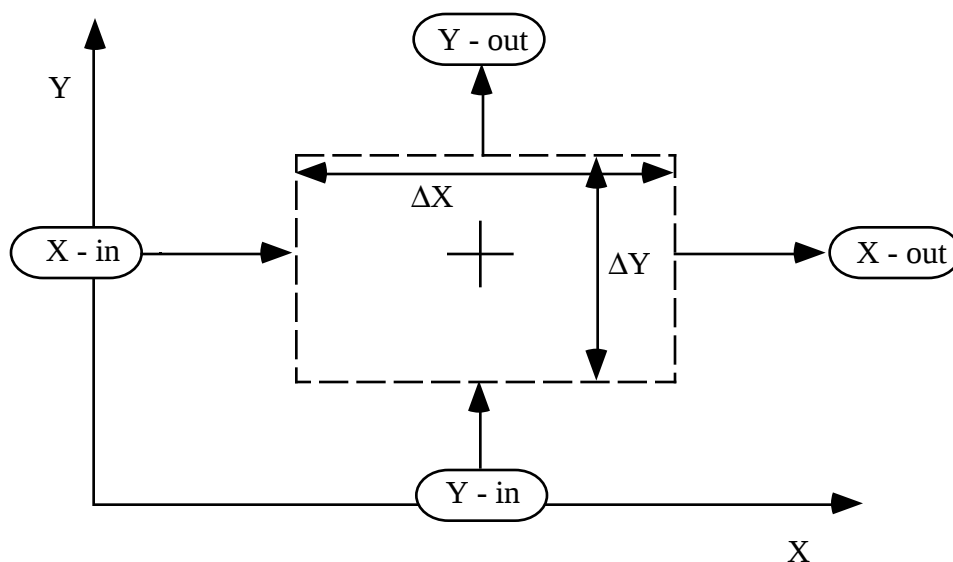


Figure 2-3. Control volume for conservation of mass.*

The net mass flow rate, or flux,** (out of the volume) is:

$$\begin{aligned} [X - \text{out}] - [X - \text{in}] + [Y - \text{out}] - [Y - \text{in}] &= \text{change of mass (decrease)} \\ &= -\frac{\partial \rho}{\partial t} \Delta X \Delta Y. \end{aligned} \quad (2-1)$$

* Note that convention requires that control volumes be described using dashed lines to illustrate that the boundaries are fictitious, and fluid is flowing freely across them.

** A *flux* is a quantity which flows across the boundary of a defined surface. Typically we think of mass, momentum and energy fluxes.

2- 6 Applied Computational Aerodynamics

Use a Taylor series expansion of the mass fluxes into the volume around the origin of the volume. The flux per unit length through the surface is multiplied by the length of the surface to get:

$$\begin{aligned} [X - \text{out}] &= \left[\rho u + \frac{\partial \rho u}{\partial x} \cdot \frac{\Delta X}{2} \right] \Delta Y \\ [X - \text{in}] &= \left[\rho u - \frac{\partial \rho u}{\partial x} \cdot \frac{\Delta X}{2} \right] \Delta Y \\ [Y - \text{out}] &= \left[\rho v + \frac{\partial \rho v}{\partial y} \cdot \frac{\Delta Y}{2} \right] \Delta X \\ [Y - \text{in}] &= \left[\rho v - \frac{\partial \rho v}{\partial y} \cdot \frac{\Delta Y}{2} \right] \Delta X . \end{aligned} \quad (2-2)$$

Adding these terms up we get:

$$\begin{aligned} & \left[\rho u + \frac{\partial \rho u}{\partial x} \cdot \frac{\Delta X}{2} \right] \Delta Y - \left[\rho u - \frac{\partial \rho u}{\partial x} \cdot \frac{\Delta X}{2} \right] \Delta Y \\ & + \left[\rho v + \frac{\partial \rho v}{\partial y} \cdot \frac{\Delta Y}{2} \right] \Delta X - \left[\rho v - \frac{\partial \rho v}{\partial y} \cdot \frac{\Delta Y}{2} \right] \Delta X = -\frac{\partial \rho}{\partial t} \Delta X \Delta Y . \end{aligned} \quad (2-3)$$

Summing up and canceling $\Delta X \Delta Y$ we get:

$$\frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} = -\frac{\partial \rho}{\partial t} \quad (2-4)$$

or in three dimensions:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} + \frac{\partial \rho w}{\partial z} = 0 . \quad (2-5)$$

This is the differential form of the continuity equation. The more general vector form of the equation is:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{V}) = 0 . \quad (2-6)$$

Alternately, consider the arbitrary control volume shown in Fig. 2-4. The conservation of mass can then be written in an integral form quite simply. The surface integral of the flow out of the volume simply equals the change of mass given in the volume:

$$\oiint \rho \mathbf{V} \cdot \hat{n} dS = -\frac{\partial}{\partial t} \iiint_V \rho dV . \quad (2-7)$$

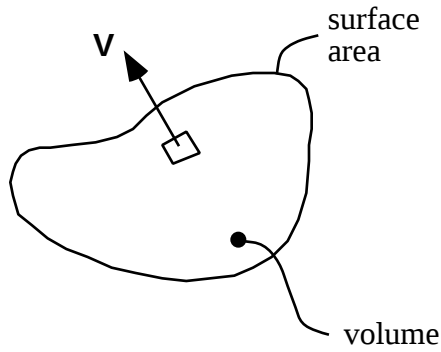


Figure 2-4. Arbitrary fluid control volume.

This is true without making any assumption requiring continuous variables and differentiability. It's for all flows, viscous or inviscid, compressible or incompressible.

To relate this expression to the differential form, we make use of the Gauss Divergence Theorem, which assumes continuous partial derivatives. It is given by:

$$\oint \mathbf{A} \cdot \hat{\mathbf{n}} dS = \iiint_V \nabla \cdot \mathbf{A} dV \quad (2-8)$$

and the equivalent statement for a scalar is:

$$\oint \phi \mathbf{n} dS = \iiint_V \text{grad} \phi dV. \quad (2-9)$$

Using this theorem, the differential and integral forms can be shown to be the same. First, re-write the surface integral in the conservation of mass, Eq. (2-7), as:

$$\oint \rho \mathbf{V} \cdot \hat{\mathbf{n}} dS = \iiint_V \nabla \cdot (\rho \mathbf{V}) dV \quad (2-10)$$

using the divergence theorem, Eq. (2-8). The continuity equation integral form thus becomes:

$$\iiint_V \nabla \cdot (\rho \mathbf{V}) dV = -\frac{\partial}{\partial t} \iiint_V \rho dV \quad (2-11)$$

and since v refers to a fixed volume, we can move $\partial/\partial t$ inside the integral,

$$\iiint_V \left[\nabla \cdot (\rho \mathbf{V}) + \frac{\partial \rho}{\partial t} \right] dV = 0. \quad (2-12)$$

For this to be true in general, the integrand must be zero, which is just the differential form! Further discussion, and other derivations are available in Moran,⁴ sections 2.2, 2.3, and 2.4, Anderson,³ section 2.6, and Bertin and Smith², Chapter 2.

2.2.2 Conservation of Momentum, and the Substantial Derivative

In this section we derive the general equations for the conservation of momentum. This is a statement of Newton's 2nd Law: *The time rate of change of momentum of a body equals the net force exerted on it.* For a fixed mass this is the famous equation

$$\mathbf{F} = m\mathbf{a} = m \frac{D\mathbf{V}}{Dt} . \quad (2-13)$$

Substantial Derivative

We need to apply Newton's Law to a moving fluid element (the "body" in the 2nd Law statement given above) from our fixed coordinate system. This introduces some extra complications. From our fixed coordinate system, look at what D/Dt means. Consider Fig. 2-5 (from Karamcheti¹). Consider any fluid property, $Q(\mathbf{r}, t)$.

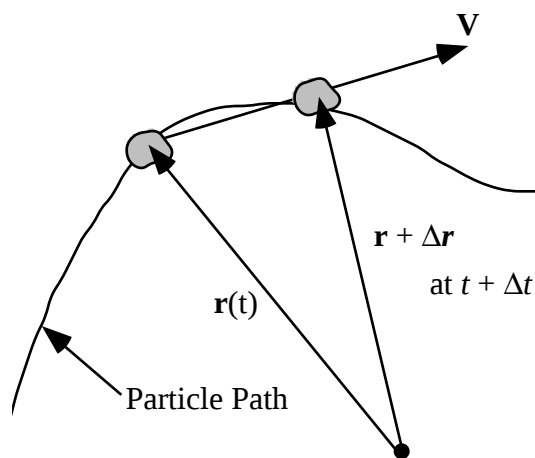


Figure 2-5. Moving particle viewed from a fixed coordinate system.

The change in position of the particle between the position r at t , and $r + \Delta r$ at $t + \Delta t$ is:

$$\Delta Q = Q(\mathbf{r} + \Delta \mathbf{r}, t + \Delta t) - Q(\mathbf{r}, t) . \quad (2-14)$$

The space change Δs is simply equal to $\mathbf{V}\Delta t$. Thus we can write:

$$\Delta Q = Q(\mathbf{r} + \mathbf{V}\Delta t, t + \Delta t) - Q(\mathbf{r}, t) , \quad (2-15)$$

which is in a form which can be used to find the rate of change of Q :

$$\frac{DQ}{Dt} = \lim_{\Delta t \rightarrow 0} \frac{\Delta Q}{\Delta t} = \lim_{\Delta t \rightarrow 0} \frac{Q(\mathbf{r} + \mathbf{V}\Delta t, t + \Delta t) - Q(\mathbf{r}, t)}{\Delta t} . \quad (2-16)$$

Note that the rate of change is in two parts, one for a change in time, and one for a change in space. Thus we write the change of Q as a function of both time and space using the Taylor series expansion as:

$$Q(\mathbf{r} + \mathbf{V}\Delta t, t + \Delta t) = Q(\mathbf{r}, t) + \left. \frac{\partial Q}{\partial t} \right|_{\mathbf{r}, t} \Delta t + \dots + \left. \frac{\partial Q}{\partial s} \right|_{\mathbf{r}, t} V\Delta t + \dots, \quad (2-17)$$

where the direction of s is understood from Fig. 2-5. Substituting into Eq. (2-16) and taking the limit, we obtain:

$$\lim_{\Delta t \rightarrow 0} \frac{\Delta Q}{\Delta t} = \underbrace{\frac{\partial Q}{\partial t}}_{\substack{\text{local time} \\ \text{derivative, or} \\ \text{local derivative}}} + \underbrace{\frac{\partial Q}{\partial s} V}_{\substack{\text{variation with} \\ \text{change of position,} \\ \text{convective derivative}}} \quad (2-18)$$

substantial derivative

This is the important consideration in applying Newton's Law for a moving particle to a point fixed in a stationary coordinate system. The second term in Eq. (2-18) has the unknown velocity V multiplying a term containing the unknown Q . This is important.

The convective derivative introduces a fundamental nonlinearity into the system

We now put this result into a specific coordinate system:

$$\frac{\partial Q}{\partial s} = \mathbf{e}_V \cdot \nabla Q. \quad (2-19)$$

where $\mathbf{e}_V$ denotes the unit vector in the direction of $\mathbf{V}$. Thus, $\mathbf{V} = V\mathbf{e}_V$ and:

$$\frac{\partial Q}{\partial s} V = \mathbf{V} \cdot \nabla Q. \quad (2-20)$$

Thus, we write the substantial derivative, Eq. (2-16), using Eqs.(2-18) and (2-20) as:

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + (\mathbf{V} \cdot \nabla), \quad (2-21)$$

which can be applied to either a scalar as:

$$\frac{DQ}{Dt} = \frac{\partial Q}{\partial t} + (\mathbf{V} \cdot \nabla)Q \quad (2-22)$$

or to a vector quantity as:

$$\frac{D\mathbf{V}}{Dt} = \frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla) \mathbf{V}. \quad (2-23)$$

In Cartesian coordinates, $\mathbf{V} = u, v, w$, and the substantial derivative becomes:

$$\frac{Du}{Dt} = \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z}$$

$$\frac{Dv}{Dt} = \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z}$$

$$\frac{Dw}{Dt} = \frac{\partial w}{\partial t} + u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z}. \quad (2-24)$$

To solve equations containing these nonlinear terms we generally have to either use finesse, where we avoid solutions requiring Eq. (2-24) by using other facts about the flowfield to avoid having to deal with Eq.(2-24) directly, or employ numerical methods. There are only a very few special cases where you can obtain analytic solutions to equations explicitly including the non-linearity.

Forces

Now we need to find the net forces on the system. What are they?

- body forces
- pressure forces
- shear forces

Each of these forces applies to the control volume shown in Fig. 2-6 given below. The τ is a general symbol for stresses. In the figure, the first subscript indicates the direction normal to the surface, and the second subscript defines the direction in which the force acts. Fluids of interest in aerodynamics are *isotropic*. To satisfy equilibrium of moments about each axis:

$$\tau_{xy} = \tau_{yx}, \quad \tau_{yz} = \tau_{zy}, \quad \tau_{zx} = \tau_{xz}. \quad (2-25)$$

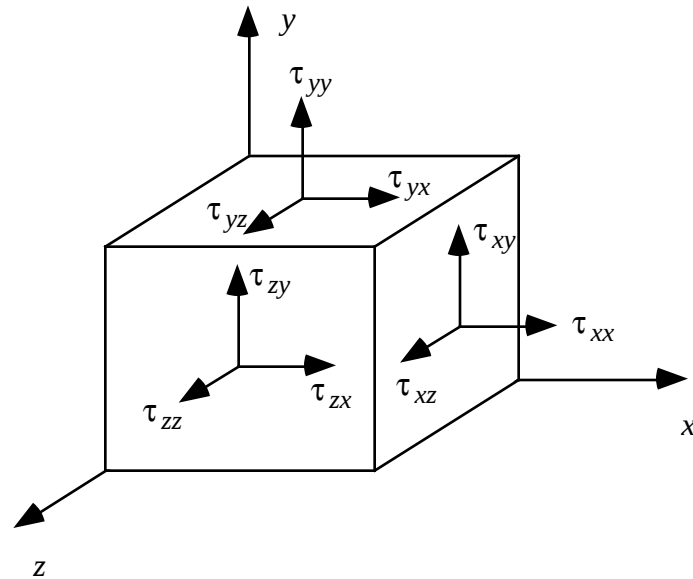


Figure 2-6. Control volume with surface forces shown.

The connection between pressure and stress is defined more specifically when the properties of a fluid are prescribed. Figure 2-7 shows the details of the forces, expanded about the origin using a Taylor Series. The force f is defined to be the body force per unit mass.

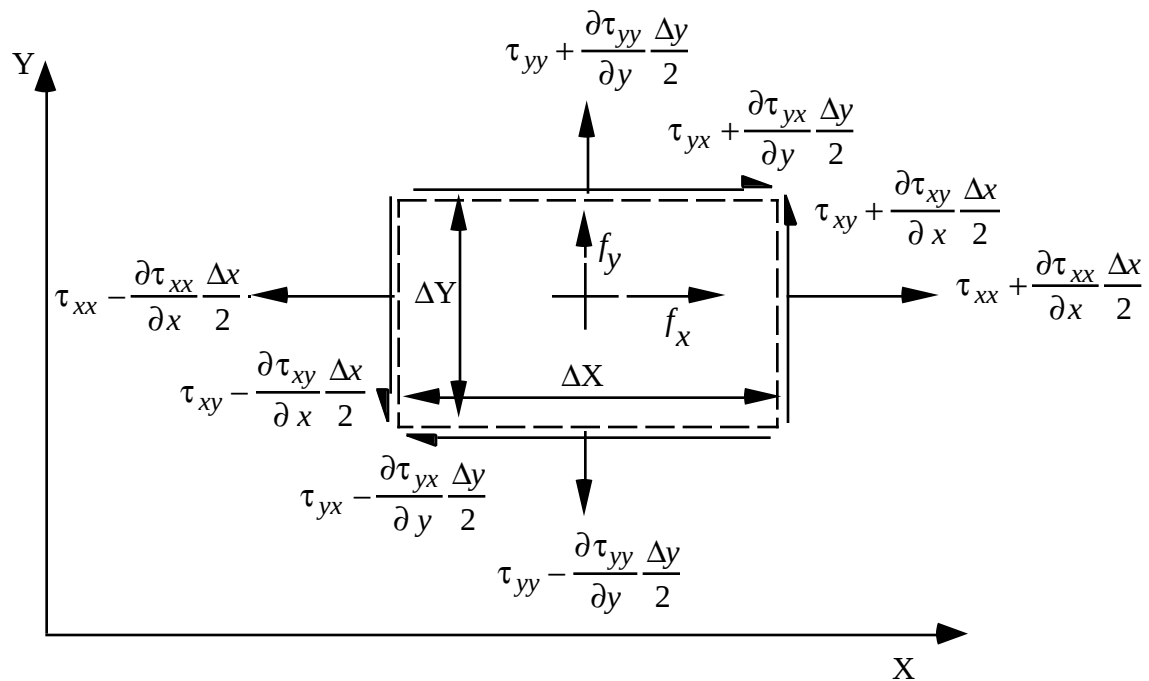


Figure 2-7. Details of forces acting on a two-dimensional control volume.

Considering the x -direction as an example, and using the Taylor series expansion shown in Figure 2-7, the net forces are found in a manner exactly analogous to the approach used in the derivation of the continuity equation. Thus, the net force in the x -direction is found to be:

$$\rho \cdot \Delta x \Delta y f_x + \frac{\partial}{\partial x}(\tau_{xx})\Delta x \Delta y + \frac{\partial}{\partial y}(\tau_{yx})\Delta y \Delta x. \quad (2-26)$$

Now we combine the forces, including the z -direction terms. Substitute for the forces into the original statement, of $\mathbf{F} = m\mathbf{a}$, Eq.(2-13), and use the substantial derivative and the definition of the mass, $m = \rho\Delta x\Delta y\Delta z$. Then the x -momentum equation becomes {writing Eq.(2-13) as $m\mathbf{a} = \mathbf{F}$, the usual fluid mechanics convention, and considering the x component, $ma_x = F_x$ },

$$\rho\Delta x\Delta y\Delta z\frac{Du}{Dt} = \rho\Delta x\Delta y\Delta z f_x + \frac{\partial}{\partial x}(\tau_{xx})\Delta x\Delta y\Delta z + \frac{\partial}{\partial y}(\tau_{yx})\Delta y\Delta x\Delta z + \frac{\partial}{\partial z}(\tau_{zx})\Delta y\Delta x\Delta z. \quad (2-27)$$

The $\Delta x\Delta y\Delta z$'s cancel out and can be dropped. The final equations can now be written. Completing the system with the y - and z - equations we obtain,

$$\begin{aligned} \rho \frac{Du}{Dt} &= \rho f_x + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} \\ \rho \frac{Dv}{Dt} &= \rho f_y + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} \\ \rho \frac{Dw}{Dt} &= \rho f_z + \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z}. \end{aligned} \quad (2-28)$$

These are general conservation of momentum relations, valid for anything!

To make Eq. (2-28) specific, we need to relate the stresses to the motion of the fluid. For gases and water, stress is a *linear* function of the rate of strain. Such a fluid is called a Newtonian fluid, *i.e.*:

$$\tau = \mu \frac{\partial u}{\partial y} \quad (2-29)$$

where μ is the coefficient of viscosity. In our work we consider μ to be a function of temperature only. Note that in air the viscosity coefficient increases with increasing temperature, and in water the viscosity coefficient decreases with temperature increases.

To complete the specification of the connection between stress and rate of strain, we need to define precisely the relation between the stresses and the motion of the fluid. This can become complicated. In general the fluid description requires two coefficients of viscosity. The coefficient of viscosity arising from the shear stress is well defined. The second coefficient of viscosity is not. This coefficient depends on the normal stress, and is only important in computing the detailed structure of shock waves. Various assumptions relating the coefficients of viscosity are made. The set of assumptions which leads to the equations known as the Navier-Stokes equations are:

- The stress-rate-of-strain relations must be independent of coordinate system.
- When the fluid is at rest and the velocity gradients are zero (the strain rates are zero), the stress reduces to the hydrostatic pressure.
- Stoke's Hypothesis is used to eliminate the issue of mean pressure vs thermodynamic pressure (this is the assumption between viscosity coefficients).

Details of the theory associated with these requirements can be found in Schlichting⁷ and Grossman.⁵ Using the conditions given above leads to the following relations:

$$\begin{aligned}\tau_{xx} &= -p - \frac{2}{3}\mu \nabla \cdot \mathbf{V} + 2\mu \frac{\partial u}{\partial x} \\ \tau_{yy} &= -p - \frac{2}{3}\mu \nabla \cdot \mathbf{V} + 2\mu \frac{\partial v}{\partial y} \\ \tau_{zz} &= -p - \frac{2}{3}\mu \nabla \cdot \mathbf{V} + 2\mu \frac{\partial w}{\partial z}\end{aligned}\tag{2-30}$$

and

$$\begin{aligned}\tau_{xy} &= \tau_{yx} = \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \\ \tau_{xz} &= \tau_{zx} = \mu \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \\ \tau_{yz} &= \tau_{zy} = \mu \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right).\end{aligned}\tag{2-31}$$

Combining and neglecting the body force (standard in aerodynamics), we get:

$$\begin{aligned}\rho \frac{Du}{Dt} &= -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left(2\mu \frac{\partial u}{\partial x} - \frac{2}{3} \mu \nabla \cdot \mathbf{V} \right) + \frac{\partial}{\partial y} \left[\mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right] + \frac{\partial}{\partial z} \left[\mu \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \right] \\ \rho \frac{Dv}{Dt} &= -\frac{\partial p}{\partial y} + \frac{\partial}{\partial x} \left[\mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right] + \frac{\partial}{\partial y} \left(2\mu \frac{\partial v}{\partial y} - \frac{2}{3} \mu \nabla \cdot \mathbf{V} \right) + \frac{\partial}{\partial z} \left[\mu \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right) \right] \\ \rho \frac{Dw}{Dt} &= -\frac{\partial p}{\partial z} + \frac{\partial}{\partial x} \left[\mu \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \right] + \frac{\partial}{\partial y} \left[\mu \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \right] + \frac{\partial}{\partial z} \left(2\mu \frac{\partial w}{\partial z} - \frac{2}{3} \mu \nabla \cdot \mathbf{V} \right).\end{aligned}\quad (2-32)$$

These are the classic Navier-Stokes Equations (written in the standard aerodynamics form, which neglects the body force). They are *i)* non-linear {recall that superposition of solutions is not allowed, remember D/Dt }, *ii)* highly coupled, and *iii)* long! As written above it's easy to identify $\mathbf{F} = m\mathbf{a}$, written in the fluid mechanics form $m\mathbf{a} = \mathbf{F}$.

When the viscous terms are small, and thus ignored, the flow is termed inviscid.

The resulting equations are known as the *Euler Equations*.

There are also alternate integral formulations of the equations. Consider the momentum flux through an arbitrary control volume in a manner similar to the integral statement of the continuity equation pictured in Fig. 2-4 and given in Eq.(2-7). Here, the momentum change, $\rho \mathbf{V}$, is proportional to the force. The integral statement is:

$$\oint \rho \mathbf{V} (\mathbf{V} \cdot \hat{n}) ds + \frac{\partial}{\partial t} \iiint_V \rho \mathbf{V} dv = \mathbf{F} = \mathbf{F}_{\text{volume}} + \mathbf{F}_{\text{surface}}. \quad (2-33)$$

and this statement can also be converted to the differential form using the Gauss Divergence Theorem. Note that we use the derivative notation $\partial / \partial t$ to denote the change in the fixed “porous” control volume that has fluid moving across the boundaries.

The derivation of the Navier-Stokes Equations is for general unsteady fluid motion. Because of limitations in our computational capability (for some time to come), these equations are for laminar flow. When the flow is turbulent, the usual approach is to Reynolds-average the equations, with the result that additional Reynolds stresses appear in the equations. Clearly, the addition of new unknowns requires additional equations. This problem is treated through turbulence modeling and is discussed in Chapter 10, Viscous Effects in Aerodynamics.

2.2.3 The Energy Equation

The equation for the conservation of energy is required to complete the system of equations. This is a statement of the 1st Law of Thermodynamics: *The sum of the work and heat added to a system will equal the increase of energy.* Following the derivation given by White:⁸

$$\underbrace{\frac{dE_t}{dt}}_{\text{change of total energy of the system}} = \underbrace{\delta Q}_{\text{change of heat added}} + \underbrace{\delta W}_{\text{change of work done on the system}} \quad (2-34)$$

For our fixed control volume coordinate system, the rate of change is:

$$\frac{DE_t}{Dt} = \dot{Q} + \dot{W} \quad (2-35)$$

where:

$$E_t = \rho \left(e + \frac{1}{2} V^2 - \mathbf{g} \cdot \mathbf{r} \right) \quad (2-36)$$

and e is the internal energy per unit mass. The last term is the potential energy, *i.e.* the body force. In aerodynamics this term is neglected. E_t can also be written in terms of specific energy as:

$$E_t = \rho e_0, \quad (2-37)$$

where:

$$e_0 = e + \frac{1}{2} V^2. \quad (2-38)$$

To obtain the energy equation we need to write the RHS of Eq.(2-35) in terms of flow properties. Consider first the heat added to the system.* The heat flow into the control volume is found in the identical manner to the mass flow. Using Fig. 2-8 for reference, obtain the expression for the net heat flow.

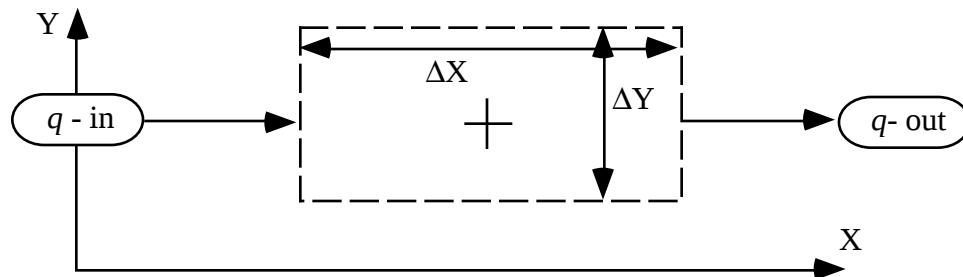


Figure 2-8. x -component of heat flux into and out of the control volume.

* Here we neglect heat addition due to radiation. See Grossman⁵ for the extension to include this contribution.

The heat fluxes are:

$$\begin{aligned} q_{x_{in}} &= \left(q_x - \frac{\partial q}{\partial x} \frac{\Delta x}{2} \right) \Delta y \\ q_{x_{out}} &= \left(q_x + \frac{\partial q}{\partial x} \frac{\Delta x}{2} \right) \Delta y \end{aligned} \quad (2-39)$$

and the net heat flow into the control volume in the x-direction is $q_{x_{in}} - q_{x_{out}}$, or:

$$-\frac{\partial q}{\partial x} \Delta x \Delta y.$$

Similarly, using the same analysis in the y and z directions we obtain the net heat flux into the control volume (realizing that the $\Delta x \Delta y \Delta z$ terms will cancel):

$$\dot{Q} = - \left(\frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} + \frac{\partial q_z}{\partial z} \right) = -\nabla \cdot \mathbf{q}. \quad (2-40)$$

Now relate the heat flow to the temperature field. Fourier's Law provides this connection:

$$\mathbf{q} = -k \nabla T \quad (2-41)$$

where k is the coefficient of thermal conductivity. Eq.(2-41) is then put into Eq.(2-40) to get the heat conduction in terms of the temperature gradient:

$$\dot{Q} = -\nabla \cdot \mathbf{q} = +\nabla \cdot (k \nabla T). \quad (2-42)$$

Next find the work done on the system. Using the definition of *work = force x distance*, the *rate of work* is:

$$\dot{W} = \text{force} \times \text{velocity}. \quad (2-43)$$

Using the control volume again, we find the work, which is equal to the velocity times the stress. The work associated with the x-face of the control volume (for two-dimensional flow) is:

$$w_x = u \tau_{xx} + v \tau_{xy}. \quad (2-44)$$

The complete description of the work on the control volume is shown in Figure 2-9.

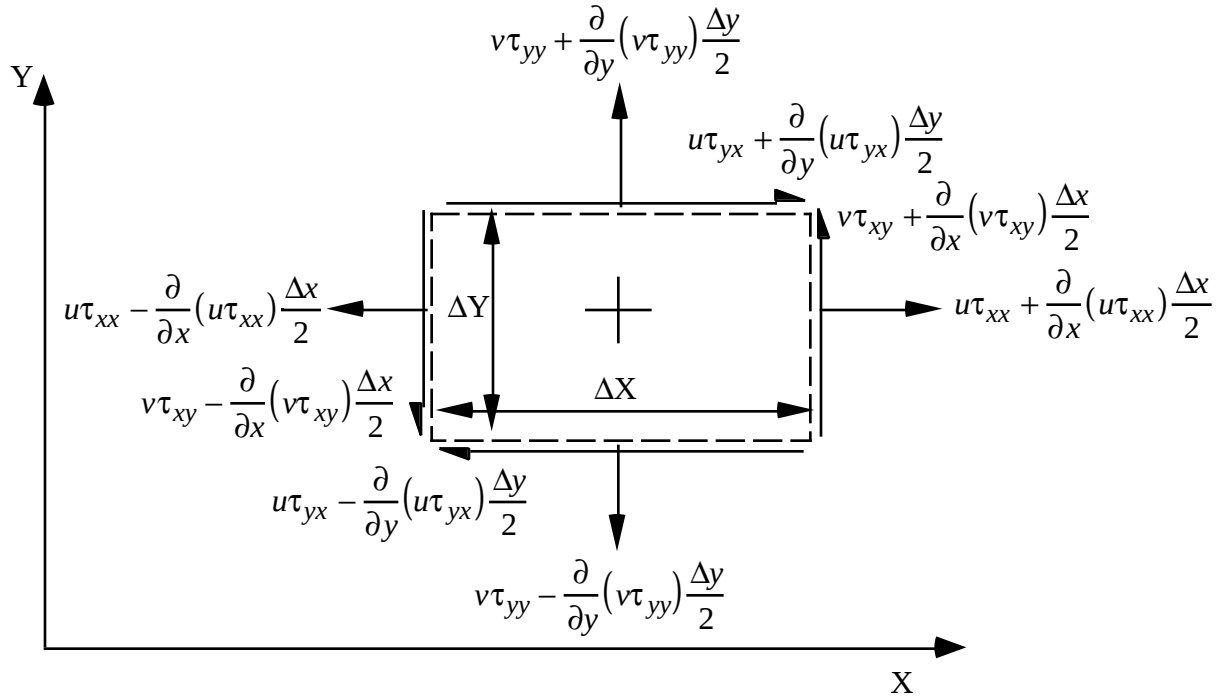


Figure 2-9 Work done on a control volume.

Using the x -component of net work as an example again, the work done on the system is

$w_{x\text{in}} - w_{x\text{out}}$ or:

$$\left(w_x - \frac{\partial w_x}{\partial x} \frac{\Delta x}{2} \right) \Delta y - \left(w_x + \frac{\partial w_x}{\partial x} \frac{\Delta x}{2} \right) \Delta y = -\frac{\partial w_x}{\partial x} \Delta x \Delta y. \quad (2-45)$$

Including the other directions (and dropping the $\Delta x \Delta y \Delta z$ terms, which cancel out)*:

$$\dot{W} = -\text{div} \mathbf{w} = \frac{\partial}{\partial x} (u\tau_{xx} + v\tau_{xy}) + \frac{\partial}{\partial y} (u\tau_{yx} + v\tau_{yy}). \quad (2-46)$$

Substituting Eqs.(2-37) and (2-38) into (2-35) for E_t , Eq.(2-42) for the heat, and Eq.(2-46)

for the work, we obtain:

$$\frac{Dp \left(e + \frac{1}{2} V^2 \right)}{Dt} = \nabla \cdot (k \nabla T) - \text{div} \mathbf{w}. \quad (2-47)$$

* Here we are using White's notation. Realize there is a difference between W and $\mathbf{w}$.

Many, many equivalent forms of the energy equation are found in the literature. Often the equation is thought of as an equation for the temperature. We now describe how to obtain one specific form. Substituting in the relations for the τ 's in terms of μ and the velocity gradients, Eqs. (2-29) and (2-30), we obtain the following lengthy expression (see Bertin and Smith² page 41-45). Making use of the momentum and continuity equations to “simplify” (?), and finally, introducing the definition of enthalpy, $h = e + p/\rho$, we obtain a frequently written form. This is the classical energy equation, which is given as:

$$\rho \frac{Dh}{Dt} - \frac{Dp}{Dt} = \underbrace{\nabla \cdot (k \nabla T)}_{\text{heat conduction}} + \underbrace{\Phi}_{\substack{\text{viscous dissipation} \\ \text{(always positive)}}} \quad (2-48)$$

where

$$\Phi = \mu \left\{ 2 \left[\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 + \left(\frac{\partial w}{\partial z} \right)^2 \right] + \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right)^2 + \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right)^2 \right. \\ \left. + \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right)^2 - \frac{2}{3} \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right)^2 \right\}. \quad (2-49)$$

The energy equation can be written in numerous forms, and many different but entirely equivalent forms are available. In particular, the energy equation is frequently written in terms of the total enthalpy, H , to good advantage in inviscid and boundary layer flows. A good discussion of the energy equation is also given by White.⁸

There is also an integral form of this equation:

$$\oint \rho \left(e + V^2 / 2 \right) (\mathbf{V} \cdot \hat{n}) ds + \frac{\partial}{\partial t} \iiint_V \rho \left(e + V^2 / 2 \right) dv = \dot{Q} + \dot{W}. \quad (2-50)$$

Here again note that we use the derivative notation $\partial / \partial t$ to denote the change in the fixed “porous” control volume that has fluid moving across the boundaries.

2.3 Boundary Conditions

If all flowfields are governed by the same equations, what makes flowfields different? Boundary conditions are the means through which the solution of the governing equations produce differing results for different situations. In computational aerodynamics the specification of boundary conditions constitutes the major part of any effort. Presuming that the flowfield algorithm selected for a particular problem is already developed and tested, the application of the method usually requires the user to specify the boundary conditions.

In general, the aerodynamicist must specify the boundary conditions for a number of different situations. Perhaps the easiest (and most obvious physically) is the condition on the surface. The statement of the boundary conditions is tightly connected to the flowfield model in use. For an inviscid steady flow over a solid surface the statement of the boundary condition is:

$$\mathbf{V}_R \cdot \mathbf{n} = 0 \quad (2-51)$$

which simply says that the difference between the velocity of the component of flow normal to the surface and the surface normal velocity (the relative velocity, $\mathbf{V}_R$) is zero. This simply means that the flow is parallel to the surface, and is known as the non-penetration condition. If $\mathbf{V}$ is the fluid velocity and $\mathbf{V}_S$ is the surface velocity, then this becomes,

$$(\mathbf{V} - \mathbf{V}_S) \cdot \mathbf{n} = 0. \quad (2-52)$$

Finally, if the surface is fixed,

$$\mathbf{V} \cdot \mathbf{n} = 0. \quad (2-53)$$

If the flow is viscous the statement becomes even simpler: $\mathbf{V} = 0$, the no-slip condition. If the surface is porous, and there is mass flow, the values of the surface velocity must be specified as part of the problem definition. Numerical solutions of the Euler and Navier-Stokes solutions require that other boundary conditions be specified. In particular, conditions on pressure and temperature are required, and will be discussed in later chapters.

As an example, recall that to obtain the unit normal the body is defined (in 2D) in the form $F(x,y) = 0$, the traditional analytic geometry nomenclature. In terms of the usual two-dimensional notation, the body shape is given by $y = f(x)$, which is then written as:

$$F(x,y) = 0 = y - f(x) \quad (2-54)$$

and

$$\mathbf{n} = \frac{\nabla F}{|\nabla F|}. \quad (2-55)$$

Conditions also must be specified away from the body. Commonly this means that at large distances from the body the flowfield must approach the freestream conditions. In numerical computations the question of the farfield boundary condition can become troublesome. How far away is infinity? Exactly how should you specify the farfield boundary condition numerically? How to best handle these issues is the basis for many papers currently appearing in the literature.

Another important use of boundary conditions arises as a means of modeling physics that would be neglected otherwise. When an approximate flowfield model is used, the boundary conditions frequently provide a means of including key elements of the physics in the problem without having to include the physics explicitly. The most famous example of this is the Kutta Condition, wherein the viscous effects at the trailing edge can be accounted for in an inviscid calculation without treating the trailing edge problem explicitly. Karamcheti¹ discuss boundary conditions in more detail.

2.4 Standard Forms and Terminology of Governing Equations

To understand the literature in computational aerodynamics, several other aspects of the terminology must be discussed. This section provides several of these considerations.

2.4.1. Nondimensionalization

The governing equations should be nondimensionalized. Considering fluid mechanics theory, nondimensionalization reveals important similarity parameters. In practice, many different nondimensionalizations are used, and for a particular code, care must be taken to understand exactly what the nondimensionalization is.

Sometimes the dimensional quantities are defined by $()^*$'s or $(\tilde{})$'s. In other schemes the non-dimensionalized variables are designated by the special symbols. In the example given here, the non-dimensionalized values are denoted by an $()^*$. In this system, once the quantities are defined, the * 's are dropped, and the nondimensionalization is understood.

Many different values can be used. We give an example here, and use the the freestream velocity and flow properties, together with the reference length as follows:

$$\begin{aligned}
 x^* &= \frac{x}{L} & y^* &= \frac{y}{L} & z^* &= \frac{z}{L} & t^* &= \frac{tV_\infty}{L} \\
 u^* &= \frac{u}{V_\infty} & v^* &= \frac{v}{V_\infty} & w^* &= \frac{w}{V_\infty} & p^* &= \frac{p}{\rho_\infty V_\infty^2} \\
 T^* &= \frac{T}{T_\infty} & \rho^* &= \frac{\rho}{\rho_\infty} & \mu^* &= \frac{\mu}{\mu_\infty} & e_o^* &= \frac{e_o}{U_\infty^2}
 \end{aligned} \tag{2-56}$$

Each code will have a set of reference nondimensionalizations similar to these. A specific example is given below in Section 2.4.3. Frequently, the speed of sound is used as the reference velocity. Making sure that you understand the nondimensionalization is an important part of applying the codes to aerodynamics problems properly.

2.4.2. Use of divergence form

The classical forms of the governing equations normally given in textbooks usually are not used for computations (as we gave them above). Instead the divergence, or conservation, form* is used. This form is found to be required for reliable numerical calculation. If discontinuities in the flowfield exist, this form must be used to account for discontinuities correctly. It is a way to improve the capability of the differential form of the governing equations. For example, across a shock wave the density and velocity both jump in value. However, the product of these quantities, the mass flow, is a constant. Thus we can easily see why it is better numerically to work with the product rather than the individual variables. In this section we show how the divergence forms are obtained from the standard classical form. We use the 2D steady x-momentum equation as the example:

$$\rho u \frac{\partial u}{\partial x} + \rho v \frac{\partial u}{\partial y} = - \frac{\partial p}{\partial x}. \quad (2-57)$$

This equation is written using the following identities:

$$\frac{\partial \rho u u}{\partial x} = \rho u \frac{\partial u}{\partial x} + u \frac{\partial \rho u}{\partial x} \quad (2-58)$$

or:

$$\rho u \frac{\partial u}{\partial x} = \frac{\partial (\rho u^2)}{\partial x} - u \frac{\partial \rho u}{\partial x}, \quad (2-59)$$

and similarly with the second term:

$$\frac{\partial \rho v u}{\partial y} = \rho v \frac{\partial u}{\partial y} + u \frac{\partial \rho v}{\partial y} \quad (2-60)$$

or

$$\rho v \frac{\partial u}{\partial y} = \frac{\partial \rho v u}{\partial y} - u \frac{\partial \rho v}{\partial y}. \quad (2-61)$$

* Be careful here, the continuity, momentum and energy equations are all conservation equations. The terminology can be confusing. Conservation *form* refers to the situation where the variables are inside the derivatives. That's why I prefer the use of divergence form to describe this mathematical arrangement. Conservation form is the more widely used terminology. They are both the same.

Substituting (2-59) and (2-61) into (2-57):

$$\frac{\partial \rho u^2}{\partial x} - u \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v u}{\partial y} - u \frac{\partial \rho v}{\partial y} + \frac{\partial p}{\partial x} = 0 \quad (2-62)$$

which can be written:

$$\frac{\partial \rho u^2}{\partial x} + \frac{\partial \rho v u}{\partial y} - u \underbrace{\left(\frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} \right)}_{=0 \text{ from continuity}} + \frac{\partial p}{\partial x} = 0. \quad (2-63)$$

Finally, the x-momentum equation written in divergence form for 2D steady flow is:

$$\frac{\partial (\rho u^2 + p)}{\partial x} + \frac{\partial (\rho v u)}{\partial y} = 0. \quad (2-64)$$

The equations must be written in divergence form to be valid when shock waves are present.

2.4.3. Standard Form of the Equations

Even after writing the governing equations in divergence form, the equations that you see in the literature won't look like the ones we've been writing down. A standard form is used in the literature for numerical solutions of the Navier-Stokes equations. In this section we provide one representative set. They come from the NASA Langley codes **cfl3d** and **cfl3de**. Professors Walters and Grossman and their students have made contributions to these codes. The Navier-Stokes equations (and the other equations required in the system) are written in vector divergence form as follows:

$$\frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial (\mathbf{F} - \mathbf{F}_v)}{\partial x} + \frac{\partial (\mathbf{G} - \mathbf{G}_v)}{\partial y} + \frac{\partial (\mathbf{H} - \mathbf{H}_v)}{\partial z} = 0 \quad (2-65)$$

where the conserved variables are:

$$\mathbf{Q} = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E_t \end{Bmatrix} = \begin{Bmatrix} \text{density} \\ \text{x - momentum} \\ \text{y - momentum} \\ \text{z - momentum} \\ \text{total energy per unit volume} \end{Bmatrix}. \quad (2-66)$$

The flux vectors in the x-direction are:

$$\begin{array}{cc}
 \text{Inviscid terms} & \text{Viscous terms} \\
 \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ (E_t + p)u \end{bmatrix} & \mathbf{F}_v = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ u\tau_{xx} + v\tau_{xy} + w\tau_{xz} - \dot{q}_x \end{bmatrix}.
 \end{array} \quad (2-67)$$

Similar expressions can be written down for the y- and z-direction fluxes, with the y-direction given as:

$$\begin{array}{cc}
 \text{Inviscid terms} & \text{Viscous terms} \\
 \mathbf{G} = \begin{bmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho vw \\ (E_t + p)v \end{bmatrix} & \mathbf{G}_v = \begin{bmatrix} 0 \\ \tau_{yx} \\ \tau_{yy} \\ \tau_{yz} \\ u\tau_{yx} + v\tau_{yy} + w\tau_{yz} - \dot{q}_y \end{bmatrix},
 \end{array} \quad (2-68)$$

and in the z-direction:

$$\begin{array}{cc}
 \text{Inviscid terms} & \text{Viscous terms} \\
 \mathbf{H} = \begin{bmatrix} \rho w \\ \rho wu \\ \rho wv \\ \rho w^2 + p \\ (E_t + p)w \end{bmatrix} & \mathbf{H}_v = \begin{bmatrix} 0 \\ \tau_{zx} \\ \tau_{zy} \\ \tau_{zz} \\ u\tau_{zx} + v\tau_{zy} + w\tau_{zz} - \dot{q}_z \end{bmatrix}.
 \end{array} \quad (2-69)$$

The equation of state (perfect gas) is written in this formulation as:

$$p = (\gamma - 1) \left[E_t - \rho (u^2 + v^2 + w^2) / 2 \right]. \quad (2-70)$$

To complete the flow equations, we need to define the nondimensionalization, and the shear stress and heat transfer nomenclature.

Shear stress and heat transfer terms are written in indicial (or index*) notation as:

$$\tau_{x_i x_j} = \frac{M_\infty}{\text{Re}_L} \left[\mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) + \lambda \frac{\partial u_k}{\partial x_k} \delta_{ij} \right] \quad (2-71)$$

* Index notation is a shorthand notation. x_i denotes x,y,z for $i = 1,2,3$.

and:

$$\dot{q}_{x_i} = - \left[\frac{M_\infty \mu}{\text{Re}_L \text{Pr}(\gamma - 1)} \right] \frac{\partial(a^2)}{\partial x_i} = - \left[\frac{M_\infty \mu}{\text{Re}_L \text{Pr}(\gamma - 1)} \right] \frac{\partial T}{\partial x_i}. \quad (2-72)$$

The molecular viscosity is found using Sutherland's Law:

$$\mu = \tilde{\mu} / \tilde{\mu}_\infty = \left(\frac{\tilde{T}}{\tilde{T}_\infty} \right)^{3/2} \left[\frac{\tilde{T}_\infty + \tilde{c}}{\tilde{T} + \tilde{c}} \right] = (T)^{3/2} \left[\frac{(1 + \tilde{c} / \tilde{T}_\infty)}{(T + \tilde{c} / \tilde{T}_\infty)} \right] \quad (2-73)$$

where Sutherland's constant is $\tilde{c} = 198.6^\circ \text{R} = 110.4^\circ \text{K}$. The tilde, ($\tilde{}$), superscript denotes a dimensional quantity and the subscript infinity denotes evaluation at freestream conditions. The other quantities are defined as: Reynolds number, $\text{Re}_L = \tilde{\rho}_\infty \tilde{q}_\infty \tilde{L} / \tilde{\mu}_\infty$, Mach number, $M_\infty = \tilde{q}_\infty / \tilde{a}_\infty$, and Prandtl number, $\text{Pr} = \tilde{\mu} \tilde{c}_p / \tilde{k}$. Stoke's hypothesis for bulk viscosity is used, meaning $\lambda + 2\mu / 3 = 0$, and the freestream velocity magnitude is, $\tilde{q}_\infty = [\tilde{u}_\infty^2 + \tilde{v}_\infty^2 + \tilde{w}_\infty^2]^{1/2}$.

The velocity components are given by:

$$\begin{aligned} u &= \tilde{u} / \tilde{a}_\infty & u_\infty &= M_\infty \cos \alpha \cos \beta \\ v &= \tilde{v} / \tilde{a}_\infty & v_\infty &= -M_\infty \sin \beta \\ w &= \tilde{w} / \tilde{a}_\infty & w_\infty &= M_\infty \sin \alpha \cos \beta \end{aligned} \quad (2-74)$$

and the thermodynamic variables are given by:

$$\begin{aligned} \rho &= \tilde{\rho} / \tilde{\rho}_\infty, & \rho_\infty &= 1 \\ p &= \tilde{p} / \tilde{\rho} \tilde{a}_\infty^2, & p_\infty &= 1 / \gamma \\ T &= \tilde{T} / \tilde{T}_\infty = \gamma p / \rho = a^2 & T_\infty &= 1 \end{aligned} \quad (2-75)$$

and,

$$E_t = \tilde{E}_t / \tilde{\rho}_\infty \tilde{a}_\infty^2 \quad E_{t_\infty} = 1 / [\gamma(\gamma - 1)] + M_\infty^2 / 2. \quad (2-76)$$

This completes the nomenclature for one typical example of the application of the Navier-Stokes equations in an actual current computer code. Note that these equations are for a Cartesian coordinate system. We will discuss the necessary extension to general coordinate systems in the Chapter 9, Geometry and Grids: Major Considerations Using Computational Aerodynamics.

2.5 The Gas Dynamics Equation and the Full Potential Equation

For inviscid flow (and even some viscous flow problems) it is useful to combine the equations in a special form known as the gas dynamics equation. In particular, this equation is used to obtain the complete or “full” nonlinear potential flow equation. Many valuable results can be obtained in computational aerodynamics (CA) using the potential flow approximation. When compressibility effects are important, a special form of the governing equation can be obtained. This equation is based on the so-called gas dynamics equation, which we derive here. The gas dynamics equation is valid for any flow assumed to be inviscid. The starting point for the derivation is the Euler equations, the continuity equation and the equation of state.

2.5.1 The Gas Dynamics Equation

We demonstrate the derivation using two-dimensional steady flow. (This is not required. Furthermore, the notation x_i , which is known as index notation, denotes x, y, z for $i = 1, 2, 3$). To start, we make use of a thermodynamic definition to rewrite the pressure term in the momentum equation.

$$\frac{\partial p}{\partial x_i} = \left(\frac{\partial p}{\partial \rho} \right)_s \frac{\partial \rho}{\partial x_i} \quad (2-77)$$

and recall the definition of the speed of sound:

$$a^2 = \left(\frac{\partial p}{\partial \rho} \right)_s \quad (2-78)$$

allowing $\partial p / \partial x_i$ to be written as:

$$\frac{\partial p}{\partial x_i} = a^2 \frac{\partial \rho}{\partial x_i} \quad (2-79)$$

We next write u times the x and v times the y momentum equations:

$$\begin{aligned} u^2 \frac{\partial u}{\partial x} + uv \frac{\partial u}{\partial y} &= -\frac{u}{\rho} \frac{\partial p}{\partial x} = -u \frac{a^2}{\rho} \frac{\partial \rho}{\partial x} \\ vu \frac{\partial v}{\partial x} + v^2 \frac{\partial v}{\partial y} &= -\frac{v}{\rho} \frac{\partial p}{\partial y} = -v \frac{a^2}{\rho} \frac{\partial \rho}{\partial y} \end{aligned} \quad (2-80)$$

and use the continuity equation by expanding it from

$$\frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} = 0 \quad (2-81)$$

to

$$u \frac{\partial \rho}{\partial x} + \rho \frac{\partial u}{\partial x} + v \frac{\partial \rho}{\partial y} + \rho \frac{\partial v}{\partial y} = 0 \quad (2-82)$$

or

$$u \frac{\partial \rho}{\partial x} + v \frac{\partial \rho}{\partial y} = -\rho \frac{\partial u}{\partial x} - \rho \frac{\partial v}{\partial y}. \quad (2-83)$$

Now add the modified x- and y- momentum equations given above:

$$\begin{aligned} u^2 \frac{\partial u}{\partial x} + uv \frac{\partial u}{\partial y} + vu \frac{\partial v}{\partial x} + v^2 \frac{\partial v}{\partial y} &= -u \frac{a^2}{\rho} \frac{\partial \rho}{\partial x} - v \frac{a^2}{\rho} \frac{\partial \rho}{\partial y} \\ &= -\frac{a^2}{\rho} \left(u \frac{\partial \rho}{\partial x} + v \frac{\partial \rho}{\partial y} \right). \end{aligned} \quad (2-84)$$

Substitute into this equation the rewritten continuity equation from above:

$$\begin{aligned} u^2 \frac{\partial u}{\partial x} + uv \frac{\partial u}{\partial y} + vu \frac{\partial v}{\partial x} + v^2 \frac{\partial v}{\partial y} &= -\frac{a^2}{\rho} \left(-\rho \frac{\partial u}{\partial x} - \rho \frac{\partial v}{\partial y} \right) \\ &= a^2 \frac{\partial u}{\partial x} + a^2 \frac{\partial v}{\partial y}. \end{aligned} \quad (2-85)$$

Finally, collecting terms we obtain in two dimensions:

$$(u^2 - a^2) \frac{\partial u}{\partial x} + uv \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) + (v^2 - a^2) \frac{\partial v}{\partial y} = 0 \quad (2-86)$$

or in three dimensions:

$$(u^2 - a^2) \frac{\partial u}{\partial x} + (w^2 - a^2) \frac{\partial w}{\partial z} + (v^2 - a^2) \frac{\partial v}{\partial y} + uv \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) + vw \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) + wu \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) = 0 \quad (2-87)$$

This equation is known as the gas dynamics equation.

2.5.2 Derivation of the Classical Gas Dynamics-Related Energy Equation

The special form of the energy equation that is used to close the system is given by (in 2D):

$$a^2 = a_0^2 - \left(\frac{\gamma - 1}{2}\right)(u^2 + v^2) \quad (2-88)$$

and we need to show exactly how this relation is obtained. Start with the form of the energy equation for inviscid, adiabatic flow:

$$\frac{DH}{Dt} = 0 \quad (2-89)$$

which yields $H = \text{constant}$, where H is (in two dimensions) the total enthalpy, defined by:

$$H = h + \frac{1}{2}(u^2 + v^2). \quad (2-90)$$

Thus we have a purely algebraic statement of the energy equation instead of a partial differential equation. This is an important reduction in complexity.

For a thermally and calorically perfect gas, $h = c_p T$, and $c_p = \text{constant}$. Substituting for the enthalpy, we get

$$c_p T_0 = c_p T + \frac{1}{2}(u^2 + v^2). \quad (2-91)$$

Recalling that $a^2 = \gamma RT$ and $R = c_p - c_v$, with $\gamma = c_p/c_v$, we write

$$a^2 = \frac{c_p}{c_v} (c_p - c_v) T = \left(\frac{c_p - c_v}{c_v} \right) c_p T \quad (2-92)$$

or:

$$c_p T = \left(\frac{c_v}{c_p - c_v} \right) a^2 = \left(\frac{1}{\gamma - 1} \right) a^2 \quad (2-93)$$

and substitute into the total energy equation ($H = \text{constant}$), Eqn. (2-91),

$$\frac{a_0^2}{\gamma - 1} = \frac{a^2}{\gamma - 1} + \frac{1}{2}(u^2 + v^2) \quad (2-94)$$

or:

$$a_0^2 = a^2 + \left(\frac{\gamma - 1}{2} \right) (u^2 + v^2) \quad (2-95)$$

and finally, solving for a (and including the third dimension):

$$a^2 = a_0^2 - \left(\frac{\gamma - 1}{2}\right)(u^2 + v^2 + w^2) \quad (2-96)$$

which is the equation we have been working to find.

2.5.3 Full Potential Equation

The gas dynamics equation is converted to the classical nonlinear potential equation when we make the irrotational flow assumption. The potential flow assumption requires that the flow be irrotational. This is valid for inviscid flow when the onset flow is uniform and there are no shock waves. However, we often continue to assume the flow can be represented approximately by a potential when the Mach number normal to any shock wave is close to one ($M_n < 1.25$, say). Recall that the irrotational flow assumption is stated mathematically as $\text{curl } \mathbf{V} = 0$. When this is true, $\mathbf{V}$ can be defined as the gradient of a scalar quantity, $\mathbf{V} = \nabla \Phi$. Using the common subscript notation to represent partial derivatives, the velocity components are $u = \Phi_x$, $v = \Phi_y$ and $w = \Phi_z$. Using the gas dynamics equation, the non-linear or “full” potential equation is then:

$$(\Phi_x^2 - a^2)\Phi_{xx} + (\Phi_y^2 - a^2)\Phi_{yy} + (\Phi_z^2 - a^2)\Phi_{zz} + 2\Phi_x\Phi_y\Phi_{xy} + 2\Phi_y\Phi_z\Phi_{yz} + 2\Phi_z\Phi_x\Phi_{zx} = 0. \quad (2-97)$$

This is the classic form of the equation. It has been used for many years to obtain physical insight into a wide variety of flows. This is a single partial differential equation. However, it is a nonlinear equation, and as written above, it is not in divergence (or conservation) form.

2.5.4 Equivalent Divergence Form and Energy Equation

The equivalent equation written in conservation form makes use of the continuity equation. This is the form that is used in most computational fluid dynamics codes. Written here in two dimensions it is:

$$\frac{\partial}{\partial x}(\rho\Phi_x) + \frac{\partial}{\partial y}(\rho\Phi_y) = 0. \quad (2-98)$$

The relation between ρ and the potential is given by:

$$\rho = \left[1 - \left(\frac{\gamma - 1}{\gamma + 1}\right)(\Phi_x^2 + \Phi_y^2)\right]^{\frac{1}{\gamma - 1}} \quad (2-99)$$

which is a statement of the energy equation. Note that the full potential equation is still nonlinear when the density varies and ρ must be considered a dependent variable.

2.5.5 Derivation of another form of the Related Energy Equation

It is informative to demonstrate the derivation of the energy equation given above. To get this standard form, understand the specific non-dimensionalization employed with this form:

$$\rho = \frac{\tilde{\rho}}{\rho_0}, \quad \Phi_x = \frac{\tilde{u}}{a^*}, \quad \Phi_y = \frac{\tilde{v}}{a^*}, \quad (2-100)$$

where a^* denotes the sonic value. Start with the previous energy equation and work with dimensional variables for the moment:

$$a^2 = a_0^2 - \left(\frac{\gamma-1}{2}\right)(u^2 + v^2) \quad (2-101)$$

or

$$\frac{a^2}{a_0^2} = 1 - \left(\frac{\gamma-1}{2}\right)\left(\frac{u^2 + v^2}{a_0^2}\right). \quad (2-102)$$

Now, get a relation for a_0 in terms of the eventual nondimensionalizing velocity a^* :

$$a_0^2 = a^2 + \underbrace{\left(\frac{\gamma-1}{2}\right)(u^2 + v^2)}_{=a^2} \quad (2-103)$$

when the velocity is equal to the speed of sound $a = a^*$. Combining terms:

$$a_0^2 = a^{*2} + \left(\frac{\gamma-1}{2}\right)a^{*2} = \left(1 + \frac{\gamma-1}{2}\right)a^{*2} \quad (2-104)$$

or:

$$a_0^2 = \left(\frac{\gamma+1}{2}\right)a^{*2}. \quad (2-105)$$

Replace a_0^2 in the energy relation with a^{*2} in the velocity term (denominator of Eq. 2-102). And in the first term use:

$$\left(\frac{a}{a_0}\right)^2 = \frac{T}{T_0}, \quad (2-106)$$

recalling $a^2 = \gamma RT$, to get:

$$\frac{T}{T_0} = 1 - \frac{\gamma-1}{2} \frac{u^2 + v^2}{\frac{\gamma+1}{2} a^{*2}} \quad (2-107)$$

or

$$\frac{T}{T_0} = 1 - \left(\frac{\gamma - 1}{\gamma + 1} \right) \left[\left(\frac{u}{a^*} \right)^2 + \left(\frac{v}{a^*} \right)^2 \right]. \quad (2-108)$$

Recall for isentropic flow (a consistent assumption if the use of Φ is valid):

$$s = \text{const} \quad (2-109)$$

and

$$\frac{p}{\rho^\gamma} = \text{const} = \frac{p_0}{\rho_0^\gamma}. \quad (2-110)$$

Now, we introduce ($\tilde{}$) to denote dimensional quantities and convert to the desired nondimensional form:

$$\left(\frac{\tilde{p}}{p_0} \right) = \left(\frac{\tilde{\rho}}{\rho_0} \right)^\gamma = \left(\frac{\tilde{T}}{T_0} \right)^{\frac{\gamma}{\gamma-1}} \quad (2-111)$$

or

$$\frac{\tilde{T}}{T_0} = \left(\frac{\tilde{\rho}}{\rho_0} \right)^{\gamma-1}. \quad (2-112)$$

Using Eq. (2-112) we write the energy equation, Eq. (2-108), as:

$$\left(\frac{\tilde{p}}{\rho_0} \right)^{\gamma-1} = 1 - \left(\frac{\gamma - 1}{\gamma + 1} \right) \left[\left(\frac{\tilde{u}}{a^*} \right)^2 + \left(\frac{\tilde{v}}{a^*} \right)^2 \right]. \quad (2-113)$$

Using the nondimensionalizing definition given above, we finally obtain:

$$\rho = [1 - \left(\frac{\gamma - 1}{\gamma + 1} \right) (\Phi_x^2 + \Phi_y^2)]^{\frac{1}{\gamma-1}}. \quad (2-114)$$

This is an energy equation in ρ to use with the divergence form of the full potential equation. It is also an example of how to get an energy equation in a typical nondimensional form used in the literature.

2.6 Special Cases

In this section we present a number of special, simplified forms of the equations described above. These simplified equations are entirely adequate for many of the problems of computational aerodynamics, and until recently were used nearly exclusively. The ability to obtain simpler relations, which provide explicit physical insight into the flowfield process, has played an important role in the development of aerodynamic concepts. One key idea is the notion of small disturbance equations. The assumption is that the flowfield is only slightly disturbed by the body. We expect this assumption to be valid for inviscid flows over streamlined shapes. These ideas are expressed mathematically by small perturbation or asymptotic expansion methods, and are elegantly described in the book by Van Dyke.⁹ The figure at the end of this section summarizes the theoretical path required to obtain these equations.

2.6.1 Small Disturbance Form of the Energy Equation

The expansion of the simple algebraic statement of the energy equation provides an example of a small disturbance analysis. In this case the square of the speed of sound (or equivalently the temperature) is linearly related to the velocity field. Start with the energy equation:

$$a^2 = a_0^2 - \left(\frac{\gamma-1}{2}\right)(u^2 + v^2) \quad (2-115)$$

and

$$a_0^2 = \text{const} = a^2 + \left(\frac{\gamma-1}{2}\right)(u^2 + v^2) = a_\infty^2 + \frac{\gamma-1}{2}U_\infty^2. \quad (2-116)$$

Letting $u = U_\infty + u'$, $v = v'$:

$$a^2 = a_\infty^2 + \frac{\gamma-1}{2}U_\infty^2 - \left(\frac{\gamma-1}{2}\right)[U_\infty^2 + 2U_\infty u' + u'^2 + v'^2] \quad (2-117)$$

and combining terms:

$$a^2 = a_\infty^2 - \left(\frac{\gamma-1}{2}\right)[2U_\infty u' + u'^2 + v'^2]. \quad (2-118)$$

At this point the relation is still exact, but now it is written so that it can easily be simplified. The basic idea will be to take advantage of the assumption:

$$u' < U_\infty, \quad v' < U_\infty \quad (2-119)$$

and thus,

$$\frac{u'}{U_\infty} < 1 \Rightarrow \left(\frac{u'}{U_\infty}\right)^2 \approx 0 \quad (2-120)$$

where the above equation becomes:

$$a^2 = a_\infty^2 - \left(\frac{\gamma - 1}{2} \right) 2U_\infty u' + \underbrace{\mu'^2 + v'^2}_{\substack{\text{neglect as small} \\ \text{henceforth}}} \quad (2-121)$$

This is a linear relation between the disturbance velocity and the speed of sound. It is a heuristic example of the procedures used in a more formal approach known as perturbation theory.

2.6.2 Small Disturbance Expansion of the Full Potential Equation

We now use a similar approach to show how to obtain a small disturbance version of the full potential equation. Again consider the situation where we assume that the disturbance to the freestream is small. Now we examine the full potential equation. First, we rewrite the full potential equation given above (in 2D for simplicity):

$$(\Phi_x^2 - a^2)\Phi_{xx} + 2\Phi_x\Phi_y\Phi_{xy} + (\Phi_y^2 - a^2)\Phi_{yy} = 0. \quad (2-122)$$

Now write the velocity as a difference from the freestream velocity. Introduce a disturbance potential ϕ , defined by:

$$\begin{aligned} \Phi &= U_\infty x + \phi(x, y) \\ \Phi_x &= u = U_\infty + \phi_x \\ \Phi_y &= v = \phi_y \end{aligned} \quad (2-123)$$

where we have introduced a *directional bias*. The x -direction is the direction of the freestream velocity. We will assume that ϕ_x and ϕ_y are small compared to U_∞ . Using the idea of a small disturbance to the freestream, simplified (and even linear) forms of a small disturbance potential equation and an energy equation can be derived.

As an example of the expansion process, consider the first term. Use the definition of the disturbance potential and the simplified energy equation as:

$$\begin{aligned}
 (\Phi_x^2 - a^2) &\cong (U_\infty + \phi_x)^2 - \left\{ a_\infty^2 - \left(\frac{\gamma-1}{2} \right) [2U_\infty u'] \right\} \\
 &\cong U_\infty^2 + 2U_\infty \phi_x + \phi_x^2 - a_\infty^2 + \frac{\gamma-1}{2} 2U_\infty \underbrace{u'}_{=\phi_x} .
 \end{aligned} \tag{2-124}$$

Regroup and drop the square of the disturbance velocity as small:

$$\begin{aligned}
 (\Phi_x^2 - a^2) &\cong U_\infty^2 - a_\infty^2 + 2U_\infty \phi_x + (\gamma-1)U_\infty \phi_x \\
 &\cong U_\infty^2 - a_\infty^2 + \underbrace{[2 + (\gamma-1)]}_{\gamma+1} U_\infty \phi_x \\
 &\cong U_\infty^2 - a_\infty^2 + (\gamma+1)U_\infty \phi_x .
 \end{aligned} \tag{2-125}$$

Dividing by a_∞^2

$$\begin{aligned}
 \left(\frac{\Phi_x^2}{a_\infty^2} - \frac{a^2}{a_\infty^2} \right) &\cong \frac{U_\infty^2}{a_\infty^2} - 1 + (\gamma+1) \frac{U_\infty}{a_\infty} \frac{\phi_x}{a_\infty} \\
 &\cong M_\infty^2 - 1 + (\gamma+1) M_\infty \underbrace{\frac{U_\infty}{a_\infty} \frac{\phi_x}{a_\infty}}_{\frac{U_\infty}{a_\infty} \frac{\phi_x}{U_\infty}} \\
 &\cong (M_\infty^2 - 1) + (\gamma+1) M_\infty^2 \left(\frac{\phi_x}{U_\infty} \right) .
 \end{aligned} \tag{2-126}$$

Rewrite the potential equation, Eq. (2-122) dividing by a_∞^2 . Then replace the coefficient of the first term using Eq. (2-126):

$$\begin{aligned}
 &\underbrace{\left(\frac{\Phi_x^2}{a_\infty^2} - \frac{a^2}{a_\infty^2} \right)}_{\left[(M_\infty^2 - 1) + (\gamma+1) M_\infty^2 \left(\frac{\phi_x}{U_\infty} \right) \right]} \Phi_{xx} + 2 \frac{\Phi_x}{a_\infty} \frac{\Phi_y}{a_\infty} \Phi_{xy} + \left(\frac{\Phi_y^2}{a_\infty^2} - \frac{a^2}{a_\infty^2} \right) \Phi_{yy} = 0 \\
 &\left[(M_\infty^2 - 1) + (\gamma+1) M_\infty^2 \left(\frac{\phi_x}{U_\infty} \right) \right] \Phi_{xx} + 2 \frac{\Phi_x}{a_\infty} \frac{\Phi_y}{a_\infty} \Phi_{xy} + \left(\frac{\Phi_y^2}{a_\infty^2} - \frac{a^2}{a_\infty^2} \right) \Phi_{yy} = 0 .
 \end{aligned} \tag{2-127}$$

Now, by definition

$$\Phi_{xx} = \phi_{xx}, \quad \Phi_{yy} = \phi_{yy}, \quad \Phi_{xy} = \phi_{xy} \quad (2-128)$$

while:

$$\frac{\Phi_x}{a_\infty} = M_\infty \left(1 + \frac{\phi_x}{U_\infty} \right) \quad \frac{\Phi_y}{a_\infty} = M_\infty \frac{\phi_y}{U_\infty} \quad (2-129)$$

and using the same approach demonstrated above we can write:

$$\left(\frac{\Phi_y^2}{a_\infty^2} - \frac{a^2}{a_\infty^2} \right) \cong -1 + (\gamma - 1) M_\infty^2 \left(\frac{\phi_y}{U_\infty} \right). \quad (2-130)$$

Putting these relations all into the potential equation we obtain:

$$\left[M_\infty^2 - 1 + (\gamma + 1) M_\infty^2 \frac{\phi_x}{U_\infty} \right] \phi_{xx} + 2 M_\infty^2 \left(1 + \frac{\phi_x}{U_\infty} \right) \frac{\phi_y}{U_\infty} \phi_{xy} + \left[-1 + (\gamma - 1) M_\infty^2 \frac{\phi_y}{U_\infty} \right] \phi_{yy} = 0 \quad (2-131)$$

where the ϕ_x^2 , ϕ_y^2 terms are neglected in the coefficients. This equation is still nonlinear, but is in a form ready for the further simplifications described below.

2.6.3 Transonic Small Disturbance Equation

Transonic flows contain regions with both subsonic and supersonic velocities. Any equation describing this flow must simulate the correct physics in the two different flow regimes. As we will show below, this makes the problem difficult to solve numerically. Indeed, the numerical solution of transonic flows was one of the primary thrusts of research in CFD over the decades of the '70s and '80s. A small disturbance equation can be derived that captures the essential nonlinearity of transonic flow, which is the rapid streamwise variation of flow disturbances in the x -direction, including normal shock waves. Therefore, in transonic flows:

$$\frac{\partial}{\partial x} > \frac{\partial}{\partial y}. \quad (2-132)$$

The transonic small disturbance equation retains the key term in the convective derivative, $u(\partial u/\partial x)$, which allows the shock to occur in the solution. Retaining this key nonlinear term the small disturbance equation given above becomes:

$$\left[(1 - M_\infty^2) - (\gamma + 1) M_\infty^2 \frac{\phi_x}{U_\infty} \right] \phi_{xx} + \phi_{yy} = 0. \quad (2-133)$$

Note that using the definition of the potential from Eq.(2-123) we can identify the nonlinear term, $u(\partial u/\partial x)$, which appears as the product of the second term in the bracket, $u = \phi_x$, and the ϕ_{xx} term, which is $\partial u/\partial x$.

This is one version of the *transonic small disturbance equation*. It is still nonlinear, and can change mathematical *type* (to be discussed in section 2.8). This means that the sign of the coefficient of ϕ_{xx} can change in the flowfield, depending on the value of the nonlinear term. It is valid for transonic flow, and, as written, it is not in a divergence form. Transonic flows occur for Mach numbers from .6 to 1.2, depending on the degree of flow disturbance. They also occur under other circumstances. At high-lift conditions, the flow around the leading edge may become locally supersonic at freestream Mach numbers as low as .20 or .25. Transonic flow occurs on rotor blades and propellers. At hypersonic speeds the flow between the bow shock and the body will frequently be locally subsonic. These are also transonic flows. The transonic small disturbance equation can be solved on your personal computer.

2.6.4 Prandtl-Glauert Equation

When the flowfield is entirely subsonic or supersonic, all terms involving products of small quantities can be neglected in the small disturbance equation. When this is done we obtain the Prandtl-Glauert Equation:

$$(1 - M_\infty^2) \phi_{xx} + \phi_{yy} = 0. \quad (2-134)$$

This is a linear equation valid for small disturbance flows that are either entirely supersonic or subsonic. For subsonic flows this equation can be transformed to Laplace's Equation, while at supersonic speeds this equation takes the form of a wave equation. The difference is important, as described below in the section on the mathematical type of partial differential equations (PDEs). This equation requires that the onset flow be in the x-direction, an example of the importance that coordinate systems assume when simplifying assumptions are made. Thus, use of simplifying assumptions introduced a directional bias into the resulting approximate equation.

The extension to three dimensions is:

$$(1 - M_\infty^2)\phi_{xx} + \phi_{yy} + \phi_{zz} = 0. \quad (2-135)$$

2.6.5. Incompressible irrotational flow: Laplace's Equation

Assuming that the flow is incompressible, ρ is a constant and can be removed from the modified continuity equation, Eq.(2-97), given above. Alternately, divide the full potential equation by the speed of sound, a , squared, and take the limit as a goes to infinity. Either way, the following equation is obtained:

$$\phi_{xx} + \phi_{yy} = 0. \quad (2-136)$$

This is Laplace's Equation. Frequently people call this equation the potential equation. For that reason the complete potential equation given above is known as the *full potential equation*. Do not confuse the true potential flow equation with Laplace's equation, which requires the assumption of incompressible flow. When the flow is incompressible, this equation is exact when using the inviscid irrotational flow model, and does not require the assumption of small disturbances.

2.6.6 The Boundary Layer Equations

The last special case retains a viscous term, while assuming that the pressure is a known function and independent of the y -coordinate value. These are the Prandtl boundary layer equations that describe the flow immediately adjacent to the body surface. For 2D, steady flow they are:

$$\frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} = 0 \quad (2-137)$$

$$\rho u \frac{\partial u}{\partial x} + \rho v \frac{\partial u}{\partial y} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial y} \left(\mu \frac{\partial u}{\partial y} \right) \quad (2-138)$$

$$0 = -\frac{\partial p}{\partial y}. \quad (2-139)$$

The related energy equation must also be included if compressibility effects are important.

All the equations presented in this section provide physical models of classes of flows that, under the right circumstances, are completely adequate to obtain an accurate representation of the flow. Many, many other approximate flow models have been proposed. Those presented in this section represent by far the majority of methods currently used. In recent times, numerous versions of the Navier-Stokes equations (taken here to include the time-averaged Reynolds equations to be discussed in Chap. 10) have also been used. These equations will be discussed as appropriate in subsequent chapters. Figure 2-10 given below summarizes the connection between the various flowfield models.

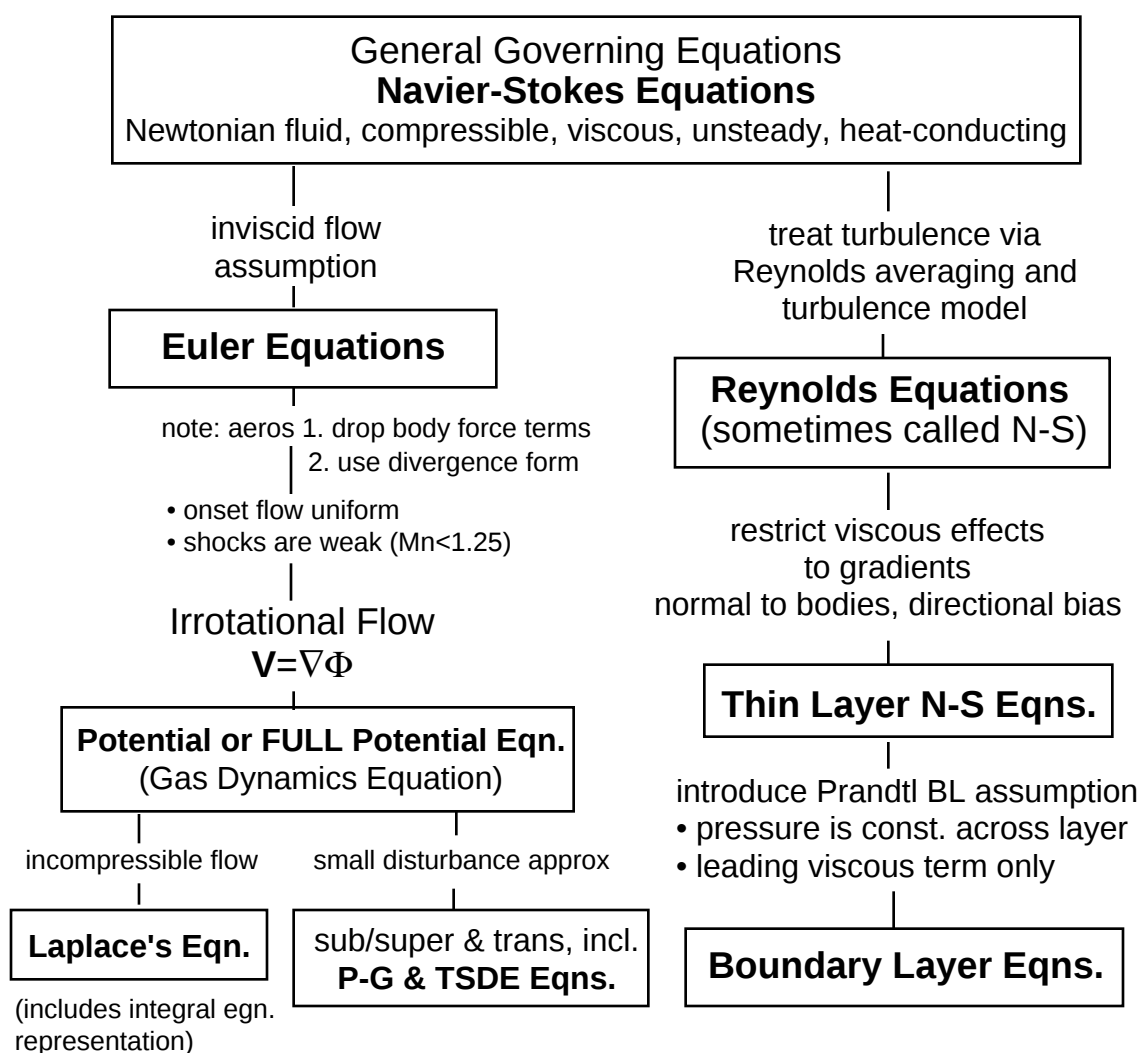


Figure 2-10. Connection between various approximations to the governing equations.

2.7 Examples of Zones of Application

The appropriate version of the governing equation depends on the type of flowfield being investigated. For high Reynolds number attached flow, the pressure can be obtained very accurately without considering viscosity. Recall that the use of a Kutta condition provides a simple way of enforcing key physics associated with viscosity by specifying this feature as a boundary condition on an otherwise inviscid solution. If the onset flow is uniform, and any shocks are weak, $M_n < 1.25$ or 1.3, then the potential flow approximation is valid. If a slight flow separation exists, a special approach using the boundary layer equations can be used interactively with the inviscid solution to obtain a solution. As speed increases, shocks begin to get strong and are curved. Under these circumstances the solution of the complete Euler equations is required.

When significant separation occurs, or you cannot figure out the preferred direction to apply a boundary layer approach, the Navier-Stokes equations are used. Note that many different “levels” of the N-S Equations are in use.

To avoid having many different codes, some people would like to have just one code that does everything. While this is a goal, most applications are better treated using a variety of methods. A step in the right direction is the use of a system that employs a common geometry and grid processing system, and a common output/graphics systems.

2.8 Mathematical Classification or the "Type" of Partial Differential Equations (PDEs)

A key property of any system of PDEs is the “type” of the equations. In mathematics, an equation “type” has a very precise meaning. Essentially, the *type* of the equation determines the domain on which boundary or initial conditions must be specified. The mathematical theory has been developed over a number of years for PDEs, and is given in books on PDEs. Two examples include Sneddon¹⁰ (pages 105-109), and Chester¹¹ (chapter 6). Discussions from the computational fluid dynamics viewpoint are available in Anderson, Tannehill, and Pletcher¹² (chapter 2), Fletcher¹³ (chapter 2), and Hoffman¹⁴ (chapter 1).

To successfully obtain the numerical solution of a PDE you must satisfy the “spirit” of the theory for the type of a PDE. Usually the theory has been developed for model problems, frequently linear. For PDEs describing physical systems, the type will be related to the following categorization:

1. *Equilibrium problems.* Examples include steady state temperature distributions and steady incompressible flow. These are similar to boundary value problems for ordinary differential equations.

2. *Marching or Propagation Problems.* These are transient or transient-like problems. Examples include transient heat conduction and steady supersonic flow. These are similar to initial value problems for ODEs.

The *types* are elliptic, parabolic, and hyperbolic. A linear equation will have a constant type. The nonlinear equations of fluid flow can change type locally depending on the local values of the equation. This “mixed-type” feature had a profound influence on the development of methods for computational aerodynamics. A mismatch between the *type* of the PDE and the prescribed boundary conditions dooms any attempt at numerical solution to failure.

The standard mathematical illustration of *type* uses a second order PDE:

$$A\phi_{xx} + B\phi_{xy} + C\phi_{yy} + D\phi_x + E\phi_y + F\phi + G = 0. \quad (2-140)$$

where A , B , C , D , E , F , and G can be constants or functions of x , y , and ϕ . Depending on the values of A , B , and C , the PDE will be of different type. The specific type of the PDE depends on the characteristics of the PDE. One of the important properties of characteristics is that the second derivative of the dependent variables are allowed, although there can be no discontinuity of the first derivative. The slopes of the characteristics can be found from A , B , and C . From mathematical theory the characteristics are found depending on the sign of determinant:

		<u>Characteristics</u>	<u>Type</u>	
$(B^2 - 4AC)$	> 0	real	hyperbolic	
	$= 0$	real, equal	parabolic	(2-141)
	< 0	imaginary	elliptic	

Hyperbolic: The basic property is a limited domain of dependence. Initial data are required on a curve C , which does not coincide with a characteristic curve. Figure 2-11 illustrates this requirement.

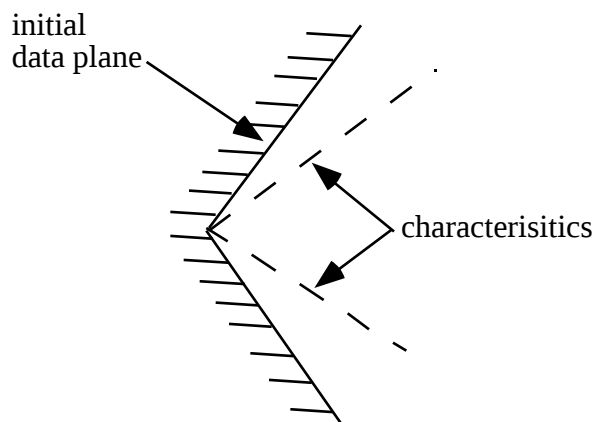


Figure 2-11. Connection between characteristics and initial condition data planes.

Classical linearized supersonic aerodynamic theory is an example of a hyperbolic system.

Parabolic: This is associated with a diffusion process. Data must be specified at an initial plane, and march forward in a time or time-like direction. There is no limited zone of influence equivalent to the hyperbolic case. Data are required on the entire time-like surface. Figure 2-12 illustrates the requirement.

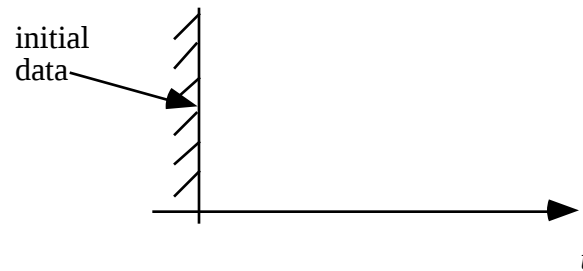


Figure 2-12. Initial data plane for parabolic equation.

In aerodynamics, boundary layers have a parabolic type.

Elliptic: These are equilibrium problems. They require boundary conditions everywhere, as shown in Figure 2-13. Incompressible potential flow is an example of a governing equation of elliptic type.

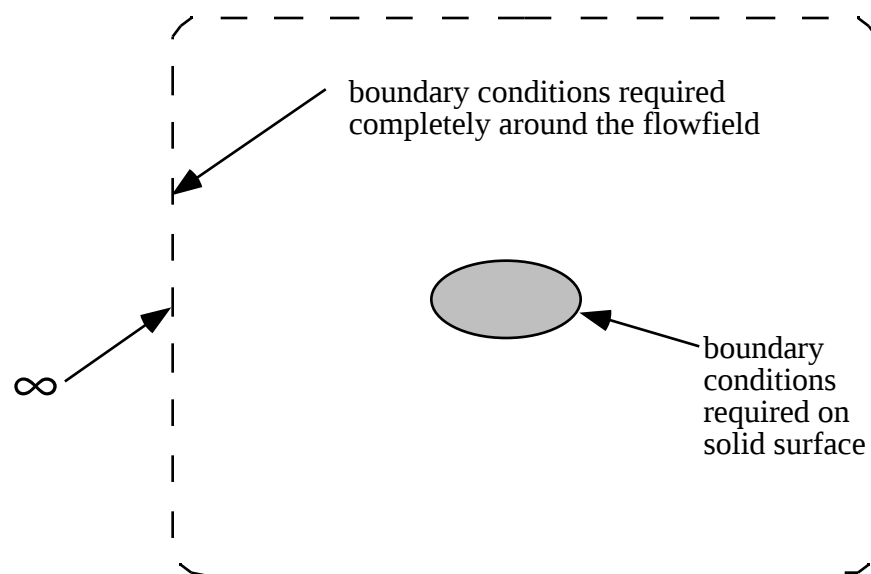


Figure 2-13. Boundary conditions required for elliptic PDEs.

Consider the following examples. For the Prandtl-Glauert equation:

$$(1 - M_\infty^2)\phi_{xx} + \phi_{yy} = 0 \quad (2-142)$$

and:

$$\begin{array}{ll} M_\infty < 1 & \text{elliptic} \\ > 1 & \text{hyperbolic} \end{array} \quad (2-143)$$

For the transonic small disturbance equation:

$$\underbrace{\left[(1 - M_\infty^2) - (\gamma + 1)M_\infty^2 \frac{\phi_x}{U_\infty} \right]}_{\substack{\text{sign depends on the solution} \\ \text{- locally subsonic: elliptic} \\ \text{- locally supersonic: hyperbolic}}} \phi_{xx} + \phi_{yy} = 0 \quad (2-144)$$

This is an equation of mixed type. It is required to treat the physics of transonic flows.

Type plays a key role in computational approaches. The type can be used to advantage. In the case of the Euler equations, the steady state Euler equations are hard to solve. It is standard procedure to consider the unsteady case, which is hyperbolic, and obtain the steady state solution by marching in time until the solution is constant in time.

Alternate approaches are available for systems of first order PDEs. Classification is sometimes difficult to determine. The *type* of an equation is determined with respect to a particular variable. The *type* of equations with respect to time may be completely different than their type with respect to space. The *type* of the equation often helps to define the appropriate solution coordinate system. The different types of the equations given above are responsible for the distinct numerical approaches that are adopted to solve different problems.

2.8.1 Elaboration on Characteristics

This section provides additional details that provide some insight into the reason that the determinant of the coefficients of the second derivative terms define the type of the equation.

Considering:

$$A\phi_{xx} + B\phi_{xy} + C\phi_{yy} + D\phi_x + E\phi_y + F\phi + G = 0 \quad (2-145)$$

- Assume ϕ is a solution describing a curve in space
- These curves “patch” various solutions, known as characteristic curves
- Discontinuity of the second derivative of the dependent variable is allowed, but no discontinuity of the first derivative

The differentials of ϕ_x and ϕ_y which represent changes from x, y to $x + dx, y + dy$ along characteristics are:

$$d\phi_x = \frac{\partial \phi_x}{\partial x} dx + \frac{\partial \phi_x}{\partial y} dy = \phi_{xx} dx + \phi_{xy} dy \quad (2-146)$$

$$d\phi_y = \frac{\partial \phi_y}{\partial x} dx + \frac{\partial \phi_y}{\partial y} dy = \phi_{yx} dx + \phi_{yy} dy. \quad (2-147)$$

Express (2-145) as

$$A\phi_{xx} + B\phi_{xy} + C\phi_{yy} = H \quad (2-148)$$

with:

$$H = -(D\phi_x + E\phi_y + F\phi + G). \quad (2-149)$$

Assume (2-148) is linear. Solve (2-148) with (2-146) and (2-147) for second derivatives of ϕ :

$$\begin{aligned} A\phi_{xx} + B\phi_{xy} + C\phi_{yy} &= H \\ dx\phi_{xx} + dy\phi_{xy} &= d\phi_x \\ dx\phi_{xy} + dy\phi_{yy} &= d\phi_y \end{aligned} \quad (2-150)$$

or

$$\begin{bmatrix} A & B & C \\ dx & dy & 0 \\ 0 & dx & dy \end{bmatrix} \begin{bmatrix} \phi_{xx} \\ \phi_{xy} \\ \phi_{yy} \end{bmatrix} = \begin{bmatrix} H \\ d\phi_x \\ d\phi_y \end{bmatrix} \quad (2-151)$$

and solve for $\phi_{xx}, \phi_{xy}, \phi_{yy}$. Since second derivatives can be discontinuous on the characteristics, the derivatives are indeterminate and the coefficient matrix would be singular:

$$\begin{bmatrix} A & B & C \\ dx & dy & 0 \\ 0 & dx & dy \end{bmatrix} = 0. \quad (2-152)$$

Expanding:

$$A(dy)^2 - Bdx dy + C(dx)^2 = 0 \quad (2-153)$$

and the slopes of the characteristics curves are found by dividing by $(dx)^2$:

$$A\left(\frac{dy}{dx}\right)^2 - B\left(\frac{dy}{dx}\right) + C = 0. \quad (2-154)$$

Solve for dy/dx :

$$\left.\frac{dy}{dx}\right|_{\alpha,\beta} = \frac{B \pm \sqrt{B^2 - 4AC}}{2A} \quad (2-155)$$

and hence the requirement on $\sqrt{B^2 - 4AC}$ to define the type of the PDE as related to the characteristics of the equation. See the references cited above for more details.

2.9 Requirements for a Complete Problem Formulation

When formulating a mathematical representation of a fluid flow problem, you have to consider carefully both the flowfield model equations and the boundary conditions. An evaluation of the mathematical type of the PDEs that are being solved plays a key role in this. Boundary conditions must be properly specified. Either over- or under-specifying boundary conditions will doom your calculation before you start. A proper formulation requires:

- governing equations
- boundary conditions
- coordinate system specification.

All before computing the first number! If this is done, then the mathematical problem being solved is considered to be *well posed*.

2.10 Exercises

1. Convert the unsteady 3D Euler equations from classical non-conservative form to divergence form.
2. Eqn. (2-70) is an unusual form of the equation of state. It is from viewgraphs defining the equations used in **cf13d**. Turn in your derivation of this equation. Is there a typo?
3. Show how Eqn. (2-76) can be obtained.
4. Why is Eqn. (2-97) not in divergence form?
5. Show that point source and point vortex singularities are solutions of Laplace's equation in two dimensions.

Recall that a point source can be expressed as:

$$\phi(x, y) = \frac{q}{4\pi} \ln(x^2 + y^2)$$

and a point vortex is:

$$\phi(x, y) = \frac{\Gamma}{2\pi} \tan^{-1}\left(\frac{y}{x}\right).$$

6. Consider the point source of problem 2. What is the behavior of the velocity as the distance from the source becomes large? What is the potential function for a point source? How does it behave as the distance from the source becomes large. Comment from the standpoint of having to satisfy the “infinity” boundary condition in a program for a potential flow solution.
7. Find the classification type of the following equations:

Laplace: $U_{xx} + U_{yy} = 0.$

Heat Eqn. : $U_y = \sigma U_{xx}, \quad \sigma \text{ real}$

Wave Eqn.: $U_{xx} = c^2 U_{yy}, \quad c \text{ real.}$

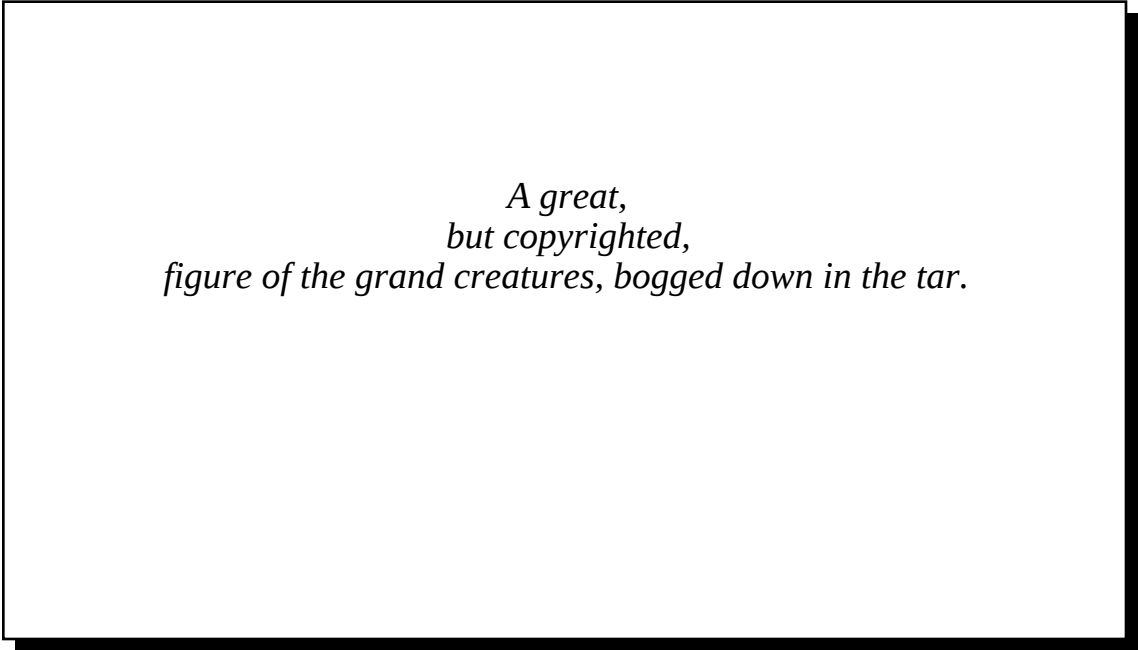
2.11 References

1. Karamcheti, K., *Principles of Ideal-Fluid Aerodynamics*, John Wiley and Son, New York, 1966. (chap 4, 5, 6, and 7)
2. Bertin, J.J., and Smith, M.L., *Aerodynamics for Engineers*, 2nd Ed., Prentice-Hall, Inc., 1989
3. Anderson, J., *Modern Compressible Flow*, 2nd Ed., McGraw-Hill Book Company, New York, 1990.
4. Moran, J., *An Introduction to Theoretical and Computational Aerodynamics*, John Wiley and Sons, New York, 1984.
5. Grossman, B., "Derivation of the Navier-Stokes Equations," Lecture Notes No. 2, AOE 3044, Boundary Layer and Heat Transfer, Virginia Tech, 1992.
6. Owczarek, J.A., *Fundamentals of Gas Dynamics*, International Textbook Co., 1964.
7. Schlichting, H., *Boundary-Layer Theory*, Sixth Ed., McGraw-Hill, New York, 1968.
8. White, F.M., *Viscous Fluid Flow*, McGraw-Hill, New York, 1974.
9. Van Dyke, M., *Perturbation Methods in Fluid Mechanics*, The Parabolic Press, Stanford, 1975.
10. Sneddon, I.N., *Elements of Partial Differential Equations*, McGraw-Hill, New York, 1957.
11. Chester, C.R., *Techniques in Partial Differential Equations*, McGraw-Hill, New York, 1971.
12. Anderson, D.A., Tannehill, J.C., and Pletcher, R.H., *Computational Fluid Mechanics and Heat Transfer*, Hemisphere Publishing, New York, 1984.
13. Fletcher, C.A.J., *Computational Techniques for Fluid Dynamics, Volume I*, Springer-Verlag, Berlin, 1988.
14. Hoffman, K.A., *Computational Fluid Dynamics for Engineers*, Engineering Education System, Austin, 1989.

3. Computers, Codes, and Engineering

3.1 General Comments

Success in using the methods of computational aerodynamics also depends on an ability to use computers effectively. In this chapter we present some guidelines for the effective use of computing systems. Software development and computing systems are called the *tarpits of engineering* by Brooks.¹ He describes the problems of software development through analogy with the ancient tarpits. Figure 3-1, from Brooks' book, shows the prehistoric beasts, completely bogged down in the primordial ooze. Almost all computer software development jobs get bogged down in a similar quagmire. His book of essays on software engineering is required reading in many places. Unfortunately for the beginner, a true appreciation of the essays comes only with experience. Reading Brooks's book may help avoid the stickiest traps. Students should understand that software development and maintenance/support costs completely overshadow the cost of computer hardware.



*A great,
but copyrighted,
figure of the grand creatures, bogged down in the tar.*

Figure 3-1. Mural of the La Brea Tar pits, by C. R. Knight, from Brooks¹.

3 - 2 Applied Computational Aerodynamics

Before providing detailed suggestions for code development and use, a couple of comments based on previous experience with students is in order:

i. Accuracy. Students are told a lot about roundoff error. In making students understand that computers process finite length numbers, the message students get is that computers aren't accu-

A Story There were all sorts of rumors about the poor accuracy of the trigonometric functions in Applesoft, the Apple version of BASIC. So after I got an Apple II computer, and converted an airfoil analysis program that I had developed for a programmable calculator, I was willing to accept 2~3 place accuracy. After all, it was BASIC, and this method relied heavily on trig functions. Telling this story at work, a colleague winked knowingly, shook his head and told me to find the bug. I gave it one more try. The result? I found the bug. In fact, I learned something important: the order of precedence of operations in BASIC (the unary minus in particular) as compared to FORTRAN. It was just luck that the test case agreed as closely as it did. The moral? More often than not, it's not machine roundoff that causes poor accuracy!

rate. Rubbish. Roundoff error has become a favorite excuse (aerodynamicists frequently use unknown Reynolds number effects in a similar fashion). Don't accept three place accuracy. The computer is not a slide rule. More than likely, poor accuracy indicates bugs. At the very least it denotes poor numerical practice, which should be fixed before you find the case where the code goes entirely wrong. The issue here is not whether the theory warrants plotting the output to more than three places, the issue is whether the code is correct. When roundoff error is a problem, it usually arises as the result of taking the difference of two large numbers. In aerodynamics, most algorithms operate accurately using 64-bit arithmetic. This is single precision on

scientific machines (Cray), but is known as double precision on commercial machines (IBM). Double precision must be specified to use 64-bit arithmetic on those computers. Investigation of the accuracy of a particular computer is left to the reader as an exercise.

ii. Disdain for "canned" programs. This may reflect some instructor's attitudes. Students should realize they won't be the authors of most programs they use. At best they will be making modifications and fixing bugs. They may be combing existing codes to develop design systems. Current codes are often the result of many man-years of development work (even hundreds of man-years for some engineering codes). Adopt a positive approach to using other people's codes. This requires learning how to use the code, demonstrating a desire to make it work on your specific problem (this often requires considerable ingenuity), and knowing how to check code accuracy against other results. You must gain confidence in a code before making an engineering decision that may mean millions of dollars, possibly lives and the future of the company (an exaggeration, but not an excessively large one). Because of the importance of maintaining software integrity, many engineers are not even allowed to modify the source codes they use. Frequently they won't even be allowed to see the code.

iii. Time. Trying to use a computer program for the first time takes more time than you expect. Working with computers is a sequential (and intense) process. It's very hard to skip steps. When you know you are going to need to use a program, try it out as soon as you can. Don't

delay. Brooks¹ poses the question (and answer): “How do computer projects get a year behind schedule? One day at a time.” There are almost always unexpected delays. Letting problems slide is a sure recipe for disaster. This is especially true when a student waits until the night before an assignment is due to try out a program for the first time. Usually, difficulties can be easily resolved if you can contact someone, or if you can step back and calmly reassess the situation. That’s hard to do in the middle of the night with a deadline looming. This also holds true for developing codes. Code development is deceptively time consuming. A good rule of thumb states that the last 10% of the code development work takes at least 90% of the time.

iv. *UNIX*. UNIX is the current operating system of engineering. Learn UNIX and the **vi** editor. Without this skill you won’t be an effective engineer. This is the only operating system that is nearly universal. It’s used on all workstations and most advanced computing machines.

v. *It’s a dynamic world*.* Computational aerodynamics codes are always changing. Every new problem seems to require code extensions. One problem with computer science majors working on codes (or directing software projects) is their assumption that codes are “finished”. In aerodynamics, if a code is used it’s never finished. Someone will always need one more modification. Be prepared to change your code. This also means paying attention to defining versions of programs as well as backing up your programs. In addition, scientific computing is in a period of rapid change. After a long period of thinking in terms of sequential, or “scalar,” computations, most aerospace engineers now have access to computers which offer increased performance through advanced computer architectures. To use these architectures effectively requires using algorithms and software designed to take advantage of the specific machine capability. Examples are vector and parallel processing. Engineers graduating today will be using massively parallel computing machines over a significant portion of their careers. Computational aerodynamics requires that you stay abreast of scientific computing developments.

3.2 Introduction to Software Engineering

The process of developing and maintaining computer programs is known as software engineering. This field is developing approaches to code development that are intended to delay getting bogged down in the tar pits described by Brooks. Before proceeding, we need to outline the elements of software engineering and provide an overview of the proper approach to developing a code which will prove useful after the original programmers have gone on to other projects. Our discussion is based on Chapter 13 of the book by Darnell and Margolis² and Chapter 11 of

* The choice of FORTRAN, C or C++ as your programming language is frequently an issue (sometimes an emotional one!). Usually the particular circumstances dictate which language to use. If you know FORTRAN or C, there are many books available to help you learn the other. You can gain proficiency with a few evenings’ study and some practice. Studying code is also valuable, and if the circumstances require you know it, that usually means that lots of code is already available to study. Essentially, FORTRAN is important because many, many existing aerodynamics codes are written in FORTRAN. C is important because there are more graphics and data acquisition software tools written in C. In either case, object oriented programming will be used by engineers in the future, and, if you program, you will continually learn new programming methods. C++ is a little harder to learn if you are used to FORTRAN or C, but may be better for constructing large, complicated, multi-disciplinary systems. Although Java is not yet relevant for scientific computing, it could become important because of its rapid development and cross-platform capability. Always be prepared for change.

the book by Stroustrup.³ The problem is that real software systems are incredibly complex. The “problem analysis, overall program design, documentation, testing, maintenance and management dwarfs the actual writing and debugging of code.”³ Software development is done by people, and relies on common sense and personal commitment. While we list numerous activities below, software development procedures must not be allowed to discourage creativity.

Darnell and Margolis break software engineering into the following elements:

- *Product specification:* The product (program) that is going to be developed must be defined before work starts. Unless the goal of the project is specifically and realistically defined the project will fail. How it looks to the user must be defined before the details of the code required to implement the solution are designed. The user needs to be involved at this point to make sure there is no misunderstanding about exactly what and how information goes in, and exactly what comes out of the program. Vague language must be avoided. Use of “fast” or “easy to use” as specifications will inevitably result in arguments, and possibly lawsuits, when the product is delivered. Nevertheless, the specification will likely be revised as the customer and designers interact during the development. Availability of a new tool results in a change in the process as soon as it’s used. The specification includes an abstract of the problem, the equations to be solved, the input and input interface, the operation of the program as it appears to the user (screen design, subroutine calls and argument lists, *etc.*), output file descriptions, error messages, and plans for future extensions.
- *Software design:* Once the product is defined, the software can be designed. This includes the major divisions of functionality, the major data structures, and the numerical/computational algorithms to be used. Quoting Stroustrup,³ “The most fundamental problem in software development is complexity. There is only one basic way of dealing with complexity: Divide and conquer.” Thus the problem should be split up. But to be effective, the communication of the various pieces requires that the interface between pieces be well designed. The result will be a program with a clean internal structure and clear connections.
- *Project planning and code estimation:* The estimation of code development time is a major problem. Even experienced programmers usually grossly underestimate the time a software job will take. One of the problems is the enormous difference in productivity between programmers. Brooks continues to be the key source of insight in this area.¹ One of the keys to tracking software schedules is to use specific measurable milestones. Typically the schedule can be broken into:
 - 1/3 product specification and scheduling
 - 1/6 coding
 - 1/4 component testing and early system testing
 - 1/4 complete system integration and testing
- *Software tools for software production:* To improve productivity use tools available on your system. These include *lint* programs to double check source code, *profilers* to evaluate where the time is being spent in programs, and tools to examine the function call tree. Most systems have *make* routines⁴ which make sure that the latest versions of routines are being used without compiling the entire code after every change. Learn how to use these tools.

- *Debugging techniques*: Software development suffers from poor productivity. Research is continually being directed toward ways to improve productivity. Development and use of debugging techniques is one area where we can expect continual improvement. Compilers generally include debuggers. Take advantage of the best debuggers available.
- *Testing*: Software validation is a difficult job. “The program that has not been tested does not work.”³ The code must be correct, and it must be usable. The developers need to establish a set of test cases to use during code development. Once the accuracy of the code is established, the usefulness of the code is evaluated by having others use it. This usually involved an alpha and a beta test group. The alpha group is usually part of the organization that developed the code, while the beta group is usually made up of customers for the product. Users that call the developers frequently to complain are often selected to be part of the beta group. It is amazing how many problems these groups can unearth. Although bugs are never completely eliminated, the problems found in the testing process can quickly reduce the initial bugs that are found after the product is released. Special considerations for computational aerodynamics codes will be described below.
- *Performance analysis*: particularly in computational aerodynamics, the time and memory required to solve problems must be defined and evaluated compared to other methods. The question is always going to be asked. You must have an answer, and it will help determine if the new code will be competitive.
- *Documentation*: The methods and the code should be documented separately. The product specification should include most of this information. The user’s manual should strike a balance between being too long, discouraging use, and being too short, so that it doesn’t help the user. Generally the user’s manual and overview documentation should be written by a new user of the code. The developers and long-time users do not bring the viewpoint of a new user to the documentation, and usually do a poor job. Documentation should include sample input and output files for the key cases and options available to the user.
- *Source control and organization*: As the code development effort proceeds the code changes become hard to keep track of, especially when the work is done by a team. Once the initial code development effort is completed, the code will be changed much less frequently, often by people not on the original development team. Without a formal process to track the history of the code changes and to ensure that the proper version is distributed there will be problems. In UNIX there is a system known as sccs/rcs which can be used.⁵ Commercial products are available to help do the so-called version control. Use of version control requires self-discipline that is difficult to do in a student environment. But development of good habits from the beginning will greatly improve the development of a professional approach to software engineering.

Above all, any code should be developed for:

- Readability
- Portability
- Maintainability.

When designing a computer solution to a problem, it is important to make sure that the problem is completely defined before the computer programming begins. Evin Cramer of Boeing

recently described the proper procedure.⁶ Most code projects should adopt a team approach, where the team consists of the core team, an extended team, and the customers. The extended team represents consultants who provide fast answers to questions that arise in the development process. The customer, or user, needs to be involved from the beginning to provide specifications and to make sure that the final product solves the right problem and the interface fits the user expectations.

Tackling the problem, it is important to keep the parts of the problem separate. First define the engineering problem. Next, look for a solution method. Once the solution method is selected, then develop the mathematical definition of the problem precisely. Only now should the code development effort start. In designing the computer code she suggested that modern simulation methods be used, and that simulation (analysis) methods be kept separate from the optimization formulation and strategy. This approach results in a modular and easily upgradeable code.

3.3 Specific Approaches to Code Development

Now we provide methods for code development applicable to computational aerodynamics. Perhaps the most important requirement is to use a disciplined approach. Because programming does not obey a specific natural law, the programmers must establish the process. At the heart of effective programming is self-discipline and personal responsibility. One approach to good programming practice has been developed by Watts Humphrey.⁷ Since most code projects will be done by teams, consider the minimum team effort to be divided between interface design, numerical methods coding and code verification.

Considering specifics, engineers should develop a clean, coherent style for their coding. When engineers write poor code they provide the computer science majors with evidence to support their claim that engineers shouldn't be allowed to touch source codes. That's okay if the computer science people work for you. If they take control of the organization, and engineers depend on them for software support, it isn't. Engineers can also work on codes. It simply requires a good common sense approach and self discipline.

The code segments in Figures 3-2 and 3-3 illustrate both the old fashioned, terrible, coding style frequently found in codes written by engineers, and a modern, good, code style. Consider first Figure 3-2. This atrocious example actually exists in a series of widely used codes.

In this example:

- variables names don't mean anything
- the statement numbers are out of order
- the logic is virtually impossible to follow
- computed **go tos** and **arithmetic ifs** are used almost exclusively*
- there is no white space or structure to the statements

I was assigned to modify this code (this segment is part of over 2000 lines of similar code) as one of my first assignments in an aerodynamics development group after graduating from school.

* **arithmetic ifs** have been declared obsolescent in FORTRAN 90, see Section 3.10.

In contrast, consider Figure 3-3, an example of code in a modern program. Here, the code appears well structured, white space is well used, and the variable names seem to be systematically defined. It's a good example of current code practice. I have also modified this code. Although the program is much longer, the job was much easier.

```

68 IF(1.-UFUT(K)/RK-1.0E-3)69,70,70
69 IF(TFUT(K)/RK**2-1.0E-6)71,70,70
70 IF(K-I(6))72,72,71
      .
      IF(I(9 ))76,76,110
110 DIV=1./(X+XSTEP-ORDIN(Z,A(2),XFUT))
      IF(I(9 )-1)76,76,77
77 DIV=SLOPBL(Z,A(2),I(5),XFUT)/ORDIN(Z,A(2),XFUT)
      IF(I(9 )-2)76,76,1055
1055 DIV=ORDIN(Z,A(2),XFUT)
76 I1=I(6)+1
      IF(I(6).GE.(I8-1)) GO TO 73
      I(6)=I8-1
73 I61=I(6)+1
      .
      D082 J=1,I61
      Y=1+.25*(UFUT(J+1)+UFUT(J))*(WFUT(J+1)+WFUT(J))
82 V(J+1)=(UFUT(J+1)*V(J)-Y*(TFUT(J+1)-TFUT(J))- .5*(WFUT(J+1)+WFUT(J)
1) *(TFUT(J+1)+TFUT(J))*Y*(UFUT(J+1)-UFUT(J))* .5-.25*(UFUT(J)+UFUT
2(J+1))**2*A(1)*(VDLPDX*(1.-Y*AY/A(12)/(Y-1.))+DIV))/UFUT(J)
94 A(5)=TAU0/RK2/(1.+RK5)
      I625=I(6)/4
      D0 1056 J=I625,I6
      IF(UFUT(J)/RK-.995)1057,1056,1056
1057 L=J
      RL=FLOAT(L)
1056 CONTINUE
      .
      TAU0=TAU0+DTW
      IF(TAU0)9411,9411,9412
9411 IQ=1
      GO TO 28
9412 ALPHA=ALPHA+DA
      A162=2.*A(16)
      IF(ABS(DU/UFUT(1))-A(16))1042,1042,1041
1042 IF(ABS(DTW/TAU0)-A(16))1043,1043,1041
1043 IF(ABS(DA/ALPHA)-A162) 1044,1044,1041
1044 WFUT(1)=F7
      TFUTP=TFUT(1)
      TFUT(1)=(TAU0+ALPHA *A(1))/(1+.5*UFUT(1)*WFUT(1))
      IF(TFUT(1).GT.0.0) GO TO 1039
      RMAX=WR*RMAX
      GO TO 34
1039 CONTINUE
      IF(ABS(1.-TFUT(1)/TOR0)-2.*A(16))1045,1046,1046

```

Figure 3-2. An example of a terrible programming style.

```

c
    if (ivisc(3).gt.0) then
        tau    = vmu(j,i)*const/vol(j,i)*sk(j,i,4)**2
        vnorm  = ub(j,i)*sk(j,i,1)+vb(j,i)*sk(j,i,2)+wb(j,i)
        dcx    = dcx+tau*(ub(j,i)-vnorm*sk(j,i,1))
        dcz    = dcz+tau*(wb(j,i)-vnorm*sk(j,i,3))
        dcy    = dcy+tau*(vb(j,i)-vnorm*sk(j,i,2))

        end if

c
    chdl    = chdl+abs(sk(j,i,3))*sk(j,i,4)
    swetl   = swetl+sk(j,i,4)
    cxl     = cxl+dcx
    cyl     = cyl+dcy
    czl     = czl+dcz
50 cml     = cml-dcz*(xa-xmc)+dcx*(za-zmc)
    xas     = xas/float(jte2-jte1)
    yas     = yas/float(jte2-jte1)
    zas     = zas/float(jte2-jte1)
    cds     = cxl*cosa+czl*sina
    cls     = -cxl*sina+czl*cosa
    cms     = cml
    chds    = chdl
    swets   = swetl
    cl      = cl+cls

```

Figure 3-3. An example of good programming style.

Kernighan and Plauger⁸ have written a book containing basic rules for good programming practice. Their book should be read before starting to do serious programming. We repeat some of their rules here:

- Write clearly - don't be too clever
- Choose variable names that won't be confused
- Write first in an easy-to-understand pseudo-language; then translate in- to whatever language you are using
- Modularize. Use subroutines
- Write and test a big program in small pieces
- Make input easy to proofread
- Make sure all variable are initialized before use
- Don't stop at one bug (keep looking!)
- Don't test floating point numbers for equality
- Make it right before you make it fast*
- Don't sacrifice clarity for small gains in efficiency
- Make sure comments and code agree
- Don't just echo the code with comments—make every comment count

* With proper planning and code design, these shouldn't have to be contradictory requirements.

Consider also the following rules from Roache⁹ which are directed toward CFD:

- Start simple
- Debug and test on a coarse mesh first
- Print out “enough” information:
 - some at each step
 - lots sometimes
 - print good diagnostic functionals
- Always check on the finest mesh possible before releasing code (this is part of the “testing at the boundaries” requirement)
- Test convergence to machine accuracy
- Try to check all option combinations in a production program*
- Check convergence/stability over the widest possible range of parameters
- Test accuracy against:
 - exact solutions
 - approximate solutions
 - experimental data
- Avoid unnecessary hardware dependence

Additional comments on style and language peculiarities of FORTRAN are discussed in the book of *Numerical Recipes*.¹⁰ Here we consider only FORTRAN. Although other languages are becoming more popular for engineering computing, most existing code is written in FORTRAN, and knowledge of FORTRAN is required in computational aerodynamics.

Other good practice:

- Avoid system-dependent code.** Any useful code will be put on different systems. The user’s own system will change. Over the long term, system dependencies almost always cause more trouble than the apparent short term gain.
- An exception to the system-dependent rule: Consider using standard math libraries. Computing centers have libraries of mathematical subroutines available for the solution of most standard math problems. These programs are written by professionals, and take advantage of machine-specific advantages of a particular system. They help you avoid numerical accuracy problems. However, *never* use one of these subroutines in your code without first using it in a pilot code on a problem you can use as a check, to make sure you understand how the program is supposed to be used. The text by Kahaner, et al.¹¹, provides examples of this approach, and a disk of useful subroutines.
- Don’t get carried away with the computer science possibilities. Concentrate on the specific development job. Keep it simple. Many very bright engineers have lost the forest for the trees when working on computer codes.

* This is essentially impossible to do with commercial codes, where millions of option combinations may be possible. That’s why code design is so important. However, any code should be tested as much as possible. Certainly, a set of standard test cases must be developed to check code modifications (fixing one bug often results in the addition of another).

** An important exception is code written to take advantage of vectorization and parallel processing. If you are using a computer with these features, the code should be modified to use the machine specific techniques to achieve maximum computing speed. However, the compute-intensive portion of any program is usually a small part of the overall code. That’s the only portion that should be made machine specific.

3 - 10 Applied Computational Aerodynamics

Specific programming standards have been established at VPI for instruction in FORTRAN in ESM 3074. Generally these are good rules. However, considering that new engineers with jobs in computational aerodynamics will mainly work with existing codes, students should be exposed to widely used, although poor, programming practices.

Many of the ESM 3074 guidelines duplicate items given above. Several require comments:

<u>Item</u>	<u>Comment</u>
• No FORTRAN 77 extensions allowed	• OK for learning FORTRAN. But some are standard. Should know NAMELIST.* Workstation manufacturers had to add NAMELIST as one of the first upgrades to their systems. It's a very nice way to handle input, and is widely used in existing aerospace engineering codes.
• ALL variables type declared.	• Although good programming practice, it is highly unusual to see this in most existing FORTRAN programs. Students should understand that programs that do not type-declare variables are not "wrong".
• FORMAT statements to be placed together, just before END statement	• This is a holdover from the days of cards. Sometimes it is inefficient at display terminals. FORMATS are best near the WRITE statements (but not obscuring executable code). Another way is to use another series of statement numbers, <i>e.g.</i> put all FORMATS in a 2XXX series.
• All DO loops end in CONTINUE	• A little harsh, just make code clear. Indent nested loops. But FORTRAN 90 requires this.
• GO TO statement	• Guideline allows. Use of GO TO should be minimized, if not eliminated. Use IF.. THEN.. ELSE for clarity.
• COMMON	COMMON is still widely used. You should know how they work. Especially since many problems occur due to errors in COMMON block use.
• DIMENSION	Still widely used.
• EQUIVALENCE	Still seen in some codes. Should know. Don't use in computational aerodynamics.
• Computed GO TO/Arithmetic IF**	Still seen, know what they are, don't use.
• Hollerith formats**	Should know, no one would ever use again.

* NAMELIST is standard in FORTRAN 90, see Section 3.10.

** Also declared obsolescent in FORTRAN 90, see Section 3.10.

3.4 Debugging Your Own Code

There is art and talent involved in debugging programs. However, experience is also an important ingredient. This is detective work. One of the problems is the difficulty in distinguishing between errors in an analysis and the code implementation. In general:

- To find out what's wrong, you need to know what's right. Check carefully.
- Use modern systems which include debuggers. Learn to use them!
Typically they are part of software tools packages to enhance code development productivity. Consider using a *lint* program to examine your source files. These programs were originally developed for C programs, but now there are usually equivalent packages for FORTRAN (FTNCHEK is available on good UNIX systems).
- As a last resort if you don't have debugging tools, use *lots* of write statements.
- Plot your results! EVERYWHERE.
- Stop and think, be patient and don't panic.
- Don't wait until the whole code is written, test small parts separately.

3.5 Looking at Other People's Codes

When you do have the source code, take time to look at the structure of the program and sketch the flow of the information. Also study the program for style. This is the best way to find ways to improve your own programming style. Try to get a feel for the organization of the computation. Make a chart of SUBROUTINE and COMMON Block structure, as well as external I/O unit use. Figure 3-4 illustrates an example of a program tree that I make to help understand program structure. Clearly this is not a conventional flowchart, it's useful! Table 3-1 provides an example of the related COMMON block map. Both of these provide a basis for finding a bug or modifying a code.

In executing the code (even if you don't have the source), observe carefully where things start to go wrong. Study your input data carefully. Try changing your input data. With complicated analysis work, start simple and build up, *i.e.*, if an entire airplane is going to be analyzed, do the wing, then the fuselage, and then put them together, *etc.* Identify when the results become "strange", and always have a mental model of the expected flowfield and a "back of the envelope" idea of what the answer should be.

3.6 Getting help.

Sometimes it helps to let someone else look at the code. One typical problem is not seeing a misspelled variable name. If you wrote the code, and have been staring at it for an hour, you might not see it. You can frequently overcome this problem by looking at an XREF, and spotting a variable used only once. If you want someone else to look at your code, *bring it all*:

- the *exact* source code that was run
- *all* code and input documentation
- the *exact* input file

3 - 12 Applied Computational Aerodynamics

- *detailed* description of the method of running
- *all* the output, and exact system messages.

Without this information you are wasting someone's time. More often than not, after collecting all the information in preparation for getting help, you'll either find your mistake, or more likely, the next step in the debugging process will become obvious.

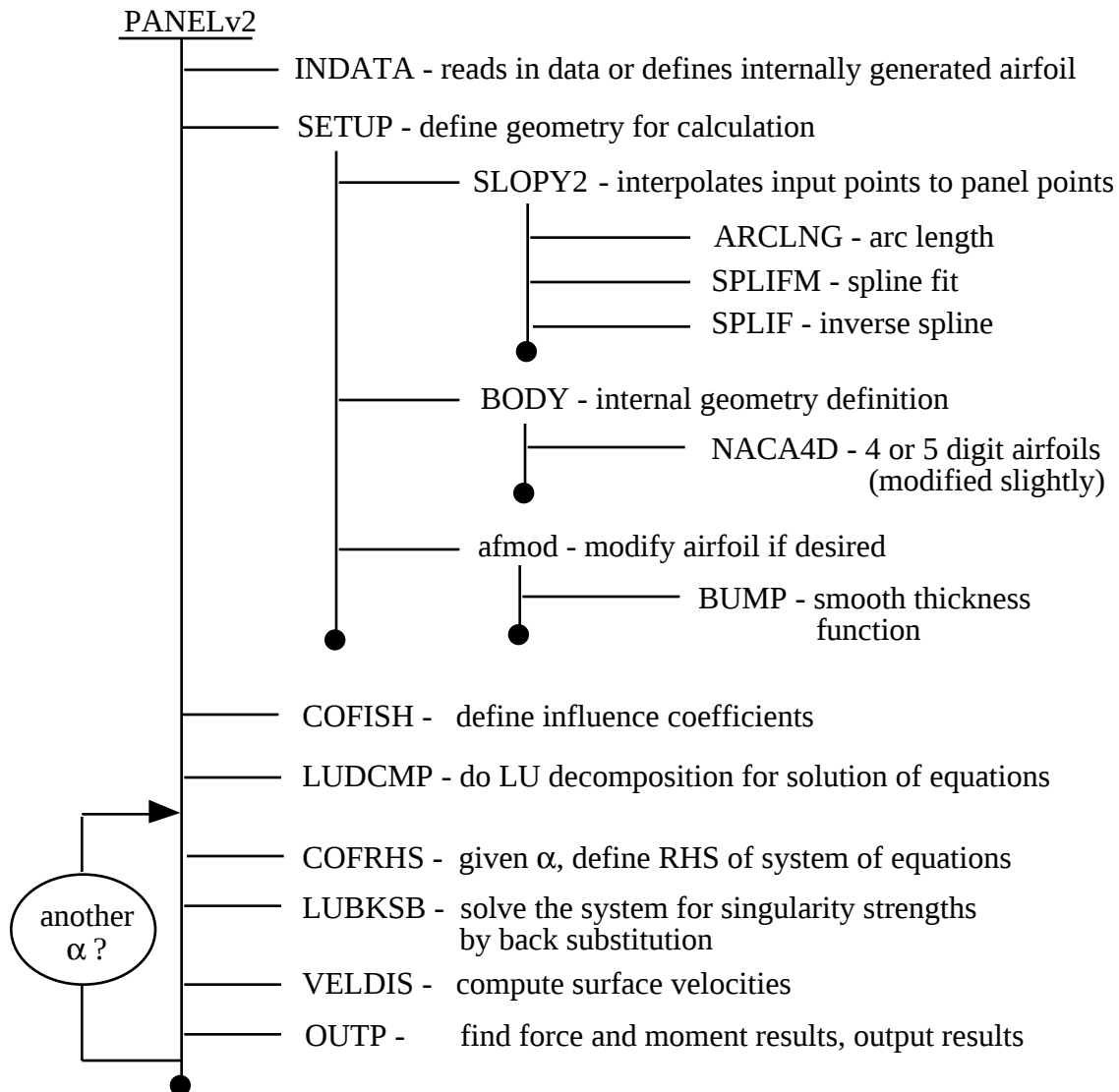


Figure 3-4. Routine tree for a small program to help understand the program structure.

Table 3-1							
Typical COMMON block map (program PANELv2)							
Routine Names	COMMON Block Names						
	BOD	NUM	PAR	COF	ORD	IOU	CPD
PANELv2	✗	✗		✗		✗	
INDATA	✗		✗		✗	✗	
SETUP	✗	✗			✗	✗	
SLOPY2							
ARCLNG							
SPLIFM							
SPLIF							
BODY			✗				
NACA4D			✗				
afmod						✗	
bump							
COFISH	✗	✗		✗			
LUDCMP							
COFRHS	✗	✗		✗			
LUBKSB							
VELDIS	✗	✗		✗		✗	✗
OUTP	✗	✗				✗	✗

3.7 Cost, Time, and Money

A serious effort using computational aerodynamics methods will take a significant investment in your time. Try to make sure that it will payoff. Spend time before starting work assessing whether your approach will produce the results you need. Which method will produce the information in time for it to be useful? There are still many situations where codes can't produce reliable information in time - although in many cases they can.

Next, develop a sense for what the required execution time and resources are to do a calculation. Insight into how long a job takes and how much core storage is required are also keys to effective use of computational aerodynamics. How long will it take to get your results? How much will it cost? Figure 3-5 appeared on my desk one day over a decade ago. Computing dead-

lines are still with us. Consider any special problems. You may need too much of the machine and be put into a special (slow) queue. Trying to do code work at the last minute is always dangerous. Two stories illustrate that Murphy's Law applied to software work. Once, after getting

Don't take getting a new code too lightly. Even when you don't write the code, and think it will be easy to use, before you can use the code you will have to make a surprisingly big investment in time before you have enough confidence in the results to use the code to make engineering decisions.

overtime pay approved and a priority to get fast turnaround, a team of six engineers arrived on Saturday morning, only to discover that the code wouldn't run because a systems programmer took the "clock" routine off the system for the weekend. The disturbance this caused was unbelievable. A contract deadline on an important Air Force contract hung in the balance. We had to get the job done.

This is an example where the computer systems people can wreck the job. A more common occurrence is to have systems programmers move codes to different disk packs. Naturally, the engineer working against a deadline will discover this in the middle of the night, with no way to get help.

Asking a code developer how long his code takes to run may not make much sense without getting the qualifiers. A code developer may quote a time for a grid too crude to use for an application, or without reaching a reliable level of convergence. What's the batting average? Realize that an advanced code may require many submissions on a single case to get everything straightened out. In those cases, the CPU time of the last submission that produced the final result is not meaningful. You need to consider the CPU time (and calendar time) of all the runs leading up to the final run that produced the desired results.

A cartoon of unknown origin:

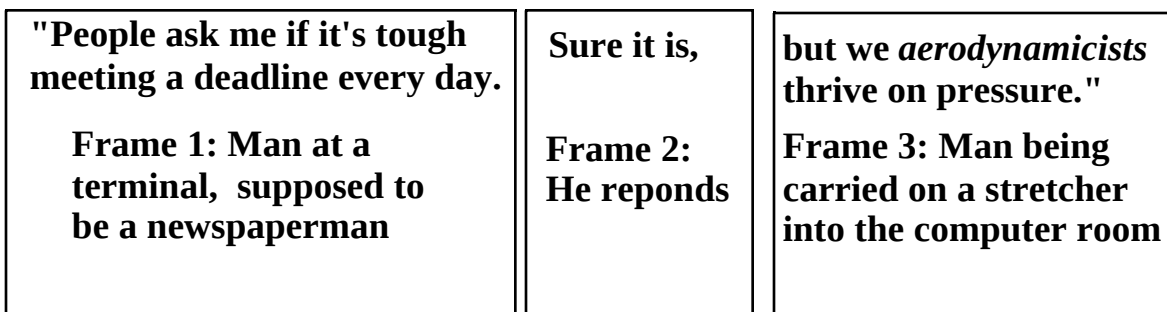


Figure 3-5. Computational aerodynamics always means deadlines.

3.8 Validation/Verification

Always check a new code against test cases for which you know the correct answers before starting to use it for the analysis cases which made you get it. It is amazing how many times engineers, under time pressure, try to skip this step. Trying to use a code on a project without having prior experience with it is a sure recipe for disaster.

The process of establishing credibility in a computer code has come to be known as code verification/validation/certification. Trying to establish code accuracy on a scientific basis is noble goal, but is a very tricky proposition. Attempts to correlate code results with experimental data on high performance designs that push the flowfield to extreme conditions often shows the importance of using codes and experimental together to understand the relative importance of competing flowfield features.

Establishing a scientific basis for “certifying” codes for use has only recently started to receive significant attention.¹²⁻¹⁴ In fact, the semantics of this area are still the subject of discussion, see for example the paper by Roache.¹⁵ In general:

- code verification means solving the specified equation right
- code validation means solving the right equation (modeling the physics correctly)
- code certification means establishing the range of applicability of a validated/verified code

The wide range of ideas about what constitutes code validation is clear in the difference of content in papers addressing the subject. Among numerous papers, the one by Aeschliman, *et al*¹⁶ deserves consideration.

In practice, code users should develop a library of test cases. For a particular code the sensitivity to the key solution control parameters should be well understood. They should also get to know the tricks used in computational aerodynamics. Be wary of code validation cases presented by the code developer. One old trick: a famous transonic test case included data at a span station that showed a double shock. For years the code developers presented results for this case without including this span station of data: the most interesting data on the wing! They got poor agreement at that station, so they ignored the data at that station. Another trick: researchers often compare results with data and not other theories. They may present comparisons of results for Euler or Navier-Stokes solutions without including computations for those same cases from simpler theories. The impression: that they are presenting results for cases that couldn't previously be computed. The truth: often small disturbance, full potential, and full potential/boundary layer methods are able to demonstrate results as accurate as the solution of the more exact equations.*

Sometimes using the complete equations precludes the use of enough grid points or sufficient run time to obtain fully converged results using the Euler or Navier-Stokes equations. Thus a more approximate method with improved resolution may be better. Also, some numerical

* One of my colleagues who develops CFD codes objects to this statement. However, my observations at national meetings with numerous presentations on CFD methods continue to confirm this view.

methods used to treat more exact equations introduced more numerical errors than results obtained using better numerical techniques with more approximate equations. This last problem is disappearing with the development of improved numerical methods.

More discussion is presented in Chapter 14, Using Computational Aerodynamics: Review and Reinforcement. There we will provide details of the issues associated with code validation for computational aerodynamics. Appendix B provides references to cases that can be used to validate calculations.

3.9 Presenting Analysis Results: visualization, time, grids and convergence, etc.

Can you understand your results by looking at a table of numbers on a computer screen? Make maximum use of graphics to examine your results. Try to look at all the results. Computer graphics and computational flow visualization provide powerful means of examining results. But be cautious. Contour plots can provide a good means of assessing sharpness of shocks, presence of wiggles, and for checking that there are no incorrect flow gradients next to farfield boundaries. The original standard computational flow visualization graphics package is PLOT3D,¹⁷ which originated at NASA Ames Research Center. PLOT3D has now been absorbed into a newer program: FAST. Computational flow visualization is part of a rapidly growing area known as *scientific visualization*. This area has recently been reviewed by Edwards.¹⁸

Watch out however, fancy color flowfield plots can be deceiving. Detailed surface pressure and force and moment comparisons should be made with appropriate experimental data and other calculations to assess a code's accuracy. Beware of pretty pictures. They are extremely valuable, but they do not provide the quantitative assessment required for engineering decision making. In a recent article on this subject,¹⁹ we find this statement: "According to Tufte,* the danger in the movieland of computer-generated graphics lies in 'dequantification': the numbers get lost in a fascination for shapes and effects. Scales are dropped, meaningless colors are added...." Honest, effective presentation of results requires skill. This is an important part of computational aerodynamics. Globus and Raible have assembled a satirical look at the misuse of visualization in "13 Ways to Say Nothing with Scientific Visualization."²⁰

To demonstrate the integrity of the calculation you must always present plots of the convergence history and results of grid convergence studies. Even though you may not be able to afford to converge iterative procedures to machine accuracy for all your calculations, you should demonstrate the effect of not doing this by presenting examples of not doing this for cases representative of the current problems you are studying. Examples of convergence with the number of panels, mesh points, and iterative solution convergence are presented throughout the rest of the text.

* Edward R. Tufte has written two fascinating books on graphical presentations: *The Visual Display of Quantitative Information* (1983), and *Envisioning Information* (1990). Both books are published by the Graphics Press, Cheshire, Connecticut. They present a more mature (and honest) view of graphics presentation than is available in most technical publications.

3.10 Modern Computing Developments

Computer architectures for scientific computing are entering a period of rapid change. Traditional scalar processing available on personal computers (and many standard mainframes) is being replaced by vector, parallel, and massively parallel machines. To use these machines effectively, the user must develop methods (and program them) to exploit the specific advantages of these machine architectures. An introduction to these issues for computational fluid dynamics has been given by Rizzi and Enquist,²¹ and in an entire AGARD volume.²² The field is changing quickly. Some of the latest information is contained in a book based on a recent conference edited by Simon.²³ A review of the situation, including descriptions of two current advanced computers, was given in a recent BYTE article.²⁴ Advances in computational aerodynamics are being driven by the High Performance Computing and Communications (HPCC) Project. An overview of this program is described in the paper by Holst, Salas and Claus.²⁵ Here we review a couple of aspects of the computing hardware and languages important to computational aerodynamics.

The key issue in advanced computing is how to increase the computation speed. Several ways of measuring speed are important. This includes the basic processor speed, the size of memory, and the speed of the data transfer through the machine. Although the “raw” computation speed is misleading, it is nevertheless used to quantify the speed of computers. With advanced computers there is a large difference between the peak speed and the maximum sustainable speed obtained in practice. The following table shows the large increases in speed obtained over the last thirty years.

The basic speed is quoted in floating point operations per second, or flops. They are defined as:

<u>flops</u>	<u>or</u>	<u>defined as:</u>
1 million	1×10^6	Megaflop or Mflop
1 billion	1×10^9	Gigaflop or Gflop
1 trillion	1×10^{12}	Teraflop or Tflop

The goal of the HPCC Project is to demonstrate one Teraflop of sustained computing speed.

Typical results obtained by famous machines as initially released have been:

<u>year</u>	<u>machine</u>	<u>speed</u>
1964	CDC 6600	1 Mflop
1968	CDC 7600	4 Mflops
1976	Cray 1	27 Mflops
1983	Cray X-MP	70 Mflops

The current speed record was set recently at Sandia Labs using an Intel Paragon computer. Thus remarkable gains are being made. The history of advances is shown in Fig. 3-6. The top line is the peak performance including advanced approaches (vector and parallel architectures). The bottom curve shows that the scalar, or serial, computing method is starting to reach its limit.

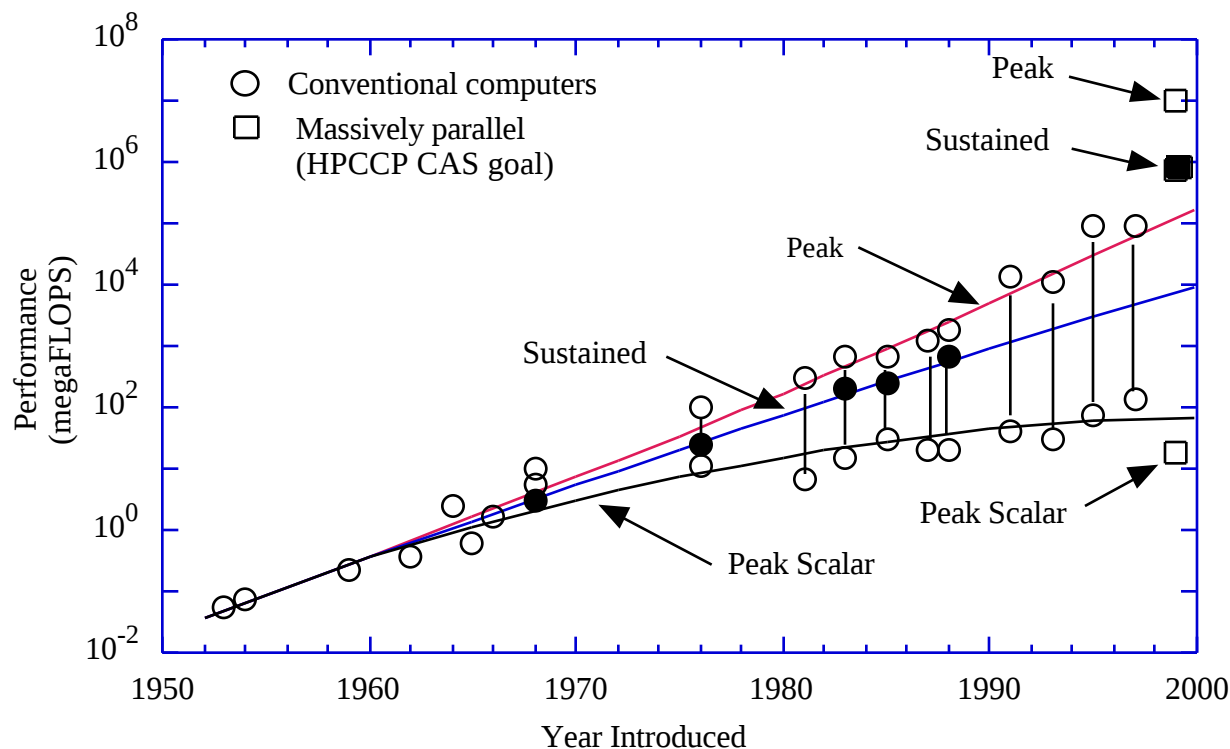


Figure 3-6. The history of computer speed increase and current goals.²⁵

Vector Computing

Essentially, the original sequential computing architecture is being replaced by machines which can perform calculations simultaneously. The first step in this direction was the use of *pipelining*. In performing specific operations, the computer may take several *clock cycles* (the basic measure of time on a computer) to complete the instruction. During part of this time some of the CPU may be sitting idle. The idea of pipelining is to take advantage of the idle cycles to begin the next instruction before the machine has completed the previous instruction. This can produce a speedup in the throughput of the CPU. Typically, this procedure is applied to arrays, where the same operation is repeated, and is implemented in DO loops. This is known as code vectorization. To be effective, the operations must allow for simultaneous calculations. If the calculations are not independent, it may not be possible to *vectorize* the loop.

The VPI IBM 3090 computer (**vtvm1**) has vector capability. As an example of the potential of vector processing, the program **vtest**, presented here in Fig. 3-7, was run with several different compiler options (it is in caps because it was copied from the IBM screen).

The following results were obtained using the **fortvs2** compiler:

<u>compiler command</u>	<u>execution time (hundredths of a second)</u>
fortvs2 vtest	889
fortvs2 vtest (novector	892
fortvs2 vtest (novector opt(2)	252
fortvs2 vtest (novector opt(3)	251
fortvs2 vtest (vector opt(3)	33

```

C
C      VECTOR PROCESSING CHECKS - W.H. MASON, FEB. 1, 1992

      PARAMETER (ILOOP = 9000, JLOOP = 1000)
      DIMENSION A(ILOOP),B(ILOOP),C(ILOOP)

      CALL TIMEON

      DO 5 I = 1,ILOOP
        A(I) = I
        B(I) = 2.*I
5     C(I) = 0.01*I

      DO 20 J = 1,JLOOP
        DO 10 I = 1,ILOOP
10     A(I) = A(I) + B(I)*C(I)
20    CONTINUE

      CALL TIMECK(N)

      WRITE(6,100) N
100   FORMAT(/5X,'TIME = ',I4,3X,'IN HUNDRETHS OF SECONDS'/)

      STOP
      END

```

Figure 3-7. A sample code used to illustrate benefits from vector processing.

These results provide several messages. Clearly, use of the vector compiler option results in a significant reduction in computing time. However, the standard scalar execution optimization option (novector) also makes a large difference in execution time. This simple example illustrates the potential of vectorized computing. Practical cases would not produce this much improvement. This example also illustrates the potential for presenting misleading comparisons. Comparison of unoptimized scalar results to optimized vectorized results overpredict the effects of vectorization by more than a factor of three. Discussions of vectorization can be found in monographs²⁶ and computer manuals.²⁷

Parallel Computers

Another approach to increasing processing speed is to perform calculations on several processors simultaneously. This is known as parallel computing. The recent article by Miel²⁸ provides a good overview. There are two approaches of interest:²⁸ “arrays of processing elements operating in unison with a single program (SIMD or Single Instruction Multiple Data), and arrays of cooperating computers running independently with distinct program memories (MIMD or Multiple Instruction Multiple Data).” Parallel computing is becoming practical, and a number of parallel processor computers are now available. Research is currently being conducted to understand how to do computational aerodynamics easily on these machines. Within the very near future computational aerodynamicists will be using these machines routinely. This will require the development of new computer languages and solution algorithms. A major government initiative, the High Performance Computing and Communications (HPCC) program, is addressing

these problems. Expect rapid progress. Current practical aspects of parallel computing in aerodynamics are discussed in the newsletter of the Numerical Aerodynamic Simulation Program, located at NASA Ames.

In computational aerodynamics two other aspects need discussion. First, the machines can be used in either “coarse” or “fine” grain parallelization modes. The coarse-grained mode is of particular interest in aerodynamic and multidisciplinary design. Here, many solutions are required using the same program with different inputs. This is done to find the sensitivity of the design to various design variables. Thus the same code is run on each different processor or node at the same time. This approach is the easiest way to exploit the capability of parallel computing. Fine grain parallel computing requires that the code be changed to make a single calculation using numerous nodes.

The other aspect of concern in parallel computing is scalability. Here, the issue is whether the speedup obtained using a small number of processors can be extrapolated to cases where a large number of processors are used. Experience shows that the performance achieved with a small numbers of processors, say twenty to thirty, does not scale up linearly when hundreds or thousands of processors are used. One standard computer science rule-of-thumb, Ahmdahl’s Law, says that the speedup decreases to a finite limit, which depends on the fraction of the code where serial computations are required. Figure 3-8²⁵ shows that if even small parts of the code require sequential computation, the speedup using parallel processing will not increase without limit. In the figure R is the fraction of the code requiring serial computation and N is the number of processors used. If none of the code requires serial computation, then $R = 0$, and the linear trend is maintained. Otherwise, a slowdown is inevitable. Some computational scientists are currently trying to demonstrate that for CFD this “law “ is not valid, and the trend can be shown to be approximately $R = 1/N$.

Experience at Virginia Tech using the coarse-grain approach is illustrated in Fig. 3-9.²⁹ Here we show results obtained by a student, after considerable effort, on the Virginia Tech Intel Paragon parallel computer. The results were obtained for some typical aerodynamics programs. The results look good for this low number of processors. One of the bottlenecks in obtaining increasing speedup with increasing numbers of processors is the use of disk IO by some codes written for older machines.

Finally, note that progress is being made making calculations on networks of workstations. Supercomputing is moving from very large, very expensive machines to distributed processing on machines that individually are much smaller and cheaper.

Language evolution

After years of little change, the computing languages are changing also. There are two reasons for this. First, the C language introduced many desirable features. Many of these have been adopted in FORTRAN 90, which also includes as standard many extensions that were so common that most users thought they were part of FORTRAN 77. Extensions that are now standard

include NAMELIST, IMPLICIT NONE and INCLUDE statements. Other new features are: standard calls for date and time routines, new symbols for relational operators, long variable names, and many new intrinsic functions. In addition, several of the worst features of FORTRAN are being declared obsolete, and are likely to be dropped in the future. Free format code is now allowed, so FORTRAN 90 code will not necessarily look like FORTRAN code has in the past. The new capabilities may lead to codes that are 50% shorter than current programs. The second reason for change is the emergence of parallel computers. Although FORTRAN 90 will not address parallel computing explicitly, it provides a basis for compiler builders to develop extensions for use in parallel computing.

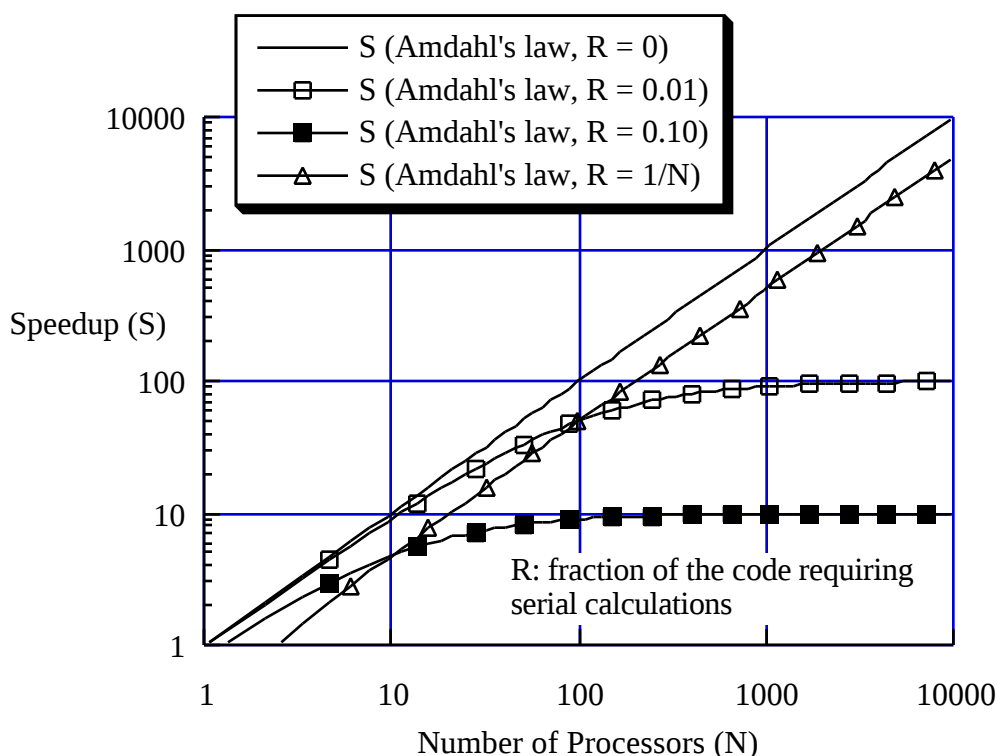


Figure 3-8. Theoretical speed up due to parallelization.²⁵

FORTRAN 90 is much bigger than FORTRAN 77, and some effort will be required to learn the new features. One book to read in making the transition is by Kerrigan.³⁰ In addition, the issues for languages specifically designed for parallel computing are addressed in a recent ICASE Report.³¹ The two flavors of FORTRAN being developed for parallel computing are High Performance FORTRAN (HPF) and FORTRAN-S. A standard FORTRAN for parallel computing will not be available for some time. The HPF research is being conducted at the Center for Research on Parallel Computation at Rice University.*

* A web viewer can use the URL: [gopher://softlib.rice.edu/](http://softlib.rice.edu/) to access reports from Rice. Other reports are available in HTML format via URL: <http://softlib.rice.edu/>. A list of reports is available by sending email to softlib@cs.rice.edu. In the message of the body type send trlist.ps. You will get a postscript file with the list of re-

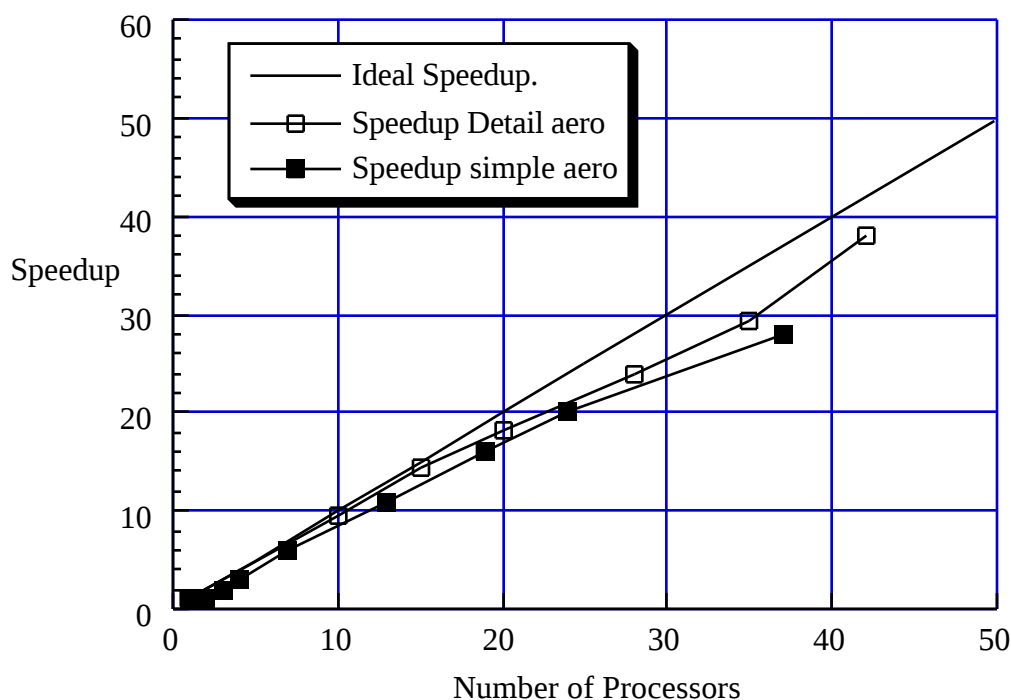


Figure 3-9. Experience at Virginia Tech with course-grained parallel computing.²⁹

To keep abreast of current developments in hardware and language developments, students should use the internet. In particular, the web pages associated with NASA's Numerical Aerodynamic Simulation facility (NAS) provide information on current developments. The NAS activity can be reviewed at

<http://www.nas.nasa.gov/home.html>.*

See the NAS Newsletter in particular. Appendix F also has addresses home pages containing the latest information on computational developments for aerodynamicists.

ports.

* The addresses are subject to change, this is a very dynamic environment. However, with the addresses of the various pages listed in Appendix F, you should be able to locate these pages.

3.11 Exercises

1. Determine the accuracy of your computer/compiler. Find the size of the smallest number that the computer can distinguish from 1, *i.e.*, what is the smallest value of ϵ such that it produces the correct result when testing for $1 + \epsilon > 1$? Recall that to have meaning, the computer and compiler, including the version number, must be included in your summary of results.

Hint: try a program similar to the following:

```

C      compiler precision
C
C      write(6,100)
      epsmch      = 1.0

      do 10 i      = 1,999
        epsmch    = epsmch/2.0
        eps1      = 1.0 + epsmch
        if(eps1 .le. 1.0) go to 20
10     write(6,110) i,epsmch

20    continue

      write(6,120) epsmch

100  format(/5x,'Estimate of computer/compiler accuracy'/
1     /9x,'i',9x,'eps')
110  format(5x,i5,5x,e14.7)
120  format(/5x,'Approx. machine zero is ',e14.7)

      stop
      end

```

2. A practical matter: Do not put TAB characters in your FORTRAN code or a data set to be read in by a FORTRAN code. Some editors do this automatically in the default mode. Some compilers allow TABS in source code, many do not. If you have TABS in your code, this severely limits the portability of the code. In this exercise, find out how your editor treats tabs. Write a program to read in a simple data set in 6F10.5 format. Determine what happens if you use TABS to put data in the correct column. Understand this now as an isolated test case, before using codes described later in this text. This will avoid a lot of late night frustration.

3.12 References

1. Brooks, Frederick P., Jr., *The Mythical Man-Month*, Addison-Wesley, Reading, 1975, an Anniversary Edition with additional chapters, 1995.
2. Darnell, P.A., and Margolis, P.E., *C: A Software Engineering Approach*, 2nd ed., Springer-Verlag, New York, 1991.
3. Stroustrup, B., *The C++ Programming Language*, 2nd edition, Addison-Wesley, Reading, 1991.
4. Oram, Andrew, and Talbott, Steve, *Managing Projects with make*, O'Reilly & Associates, Sebastopol, CA, 1991.
5. Bolinger, Don, and Bronson, Ted, *Applying RCS and SCCS*, O'Reilly & Associates, Sebastopol, CA, 1995.
6. Cramer, E.J., "Airplane Performance Optimization: A Design Case Study," AIAA Paper 95-0465, January 1995.
7. Humphrey, Watts S., *A Discipline for Software Engineering*, Addison-Wesley, Reading, 1995.
8. Kernighan, B.W., and Plauger, P.J., *The Elements of Programming Style*, McGraw-Hill, New York, 1974.
9. Roache, Patrick J., *Computational Fluid Dynamics*, Hermosa Press, 1972.
10. Press, W.H., Teukolsky, S.A., Vetterling, W.T., and Flannery, B.P., *Numerical Recipes in FORTRAN: The Art of Scientific Computing, Second Edition*, Cambridge University Press, Cambridge, 1992 (C version also available).
11. Kahaner, D., Moler, C., and Nash, S., *Numerical Methods and Software*, Prentice-Hall, Englewood Cliffs, 1989.
12. Sacher, P.W., "Technical Evaluation Report on the Fluid Dynamics Panel Symposium on Validation of Computational Fluid Dynamics," AGARD Advisory Report No. 257, May 1989.
13. Melnik, R.E., Siclari, M.J., Barber, T., and Verhoff, A., "A Process for Industry Certification of Physical Simulation of Codes," AIAA Paper 94-2235, June 1994.
14. Melnik, R.E., Siclari, M.J., Marconi, F., Barber, T., and Verhoff, A., "An Overview of a Recent Industry Effort at CFD Code Certification," AIAA Paper 95-2229, June 1995.
15. Roache, P.J., "Verification of Codes and Calculations," AIAA Paper 95-2224, June 1995.
16. Aeschliman, D.F., Oberkampf, W.L., and Blottner, F.G., "A Proposed Methodology for Computational Fluid Dynamics Code Verification, Calibration, and Validation," International Congress on Instrumentation in Aerospace Simulation Facilities (ICIASF), July 18-21, 1995, Wright-Patterson AFB, OH.
17. Walatka, P.P., Buning, P.G., Pierce, L., and Elson, A., "PLOT3D User's Manual," NASA TM 101067, March, 1990.
18. Edwards, David E., "Scientific Visualization: Current Trends and Future Directions," AIAA Paper 92-0068, Jan. 1992.
19. Patton, P., "Up From Flatland," *New York Times Magazine*, January 19, 1992, pg.61.
20. Globus, A., and Raible, E., "13 Ways to Say Nothing with Scientific Visualization," available as a postscript file from the www page of the National Aerodynamic Simulation (NAS) Facility.

21. Rizzi, A., and Enquist, B., "Selected Topics in the Theory and Practice of Computational Fluid Dynamics," *Journal of Computational Physics*, 72, 1-69 (1987).
22. Neves, K.W., in Chap 2, "Hardware Architecture" in *Computational Fluid Dynamics: Algorithms and Supercomputers*, AGARD-AG-311, 1988.
23. Simon, H.D., ed., *Parallel Computational Fluid Dynamics*, The MIT Press, Cambridge, 1992.
24. Sharp, O., "The Grand Challenges," *BYTE*, Feb. 1995, pp. 65-71.
25. Holst, T.L., Salas, M.D., and Claus, R.W., "The NASA Computational Aerosciences Program—Toward Teraflops Computing," AIAA Paper 92-0558, January 1992.
26. Schofield, C.F., *Optimising FORTRAN Programs*, Ellis Horwood, Ltd., Chichester, 1989.
27. ___, "Vectorization with VS FORTRAN and ESSL," Doc. FT16, User Services Department, Virginia Tech Computing Center, June 12, 1990.
28. Miel, G., "Supercomputers and CFD," *Aerospace America*, January 1992, pp. 32-35,51.
29. Giunta, Anthony A. , Narducci, Robert , Burgee,Susan L. , Grossman, Bernard , Haftka, Raphael T., Mason,William H. , and Watson, Layne T. "Variable-Complexity Response Surface Aerodynamic Design of an HSCT Wing," AIAA Paper 95-1886, AIAA 13th Applied Aerodynamics Conference, San Diego, CA, June 21, 1995 , Pages 994-1002 of Proceedings.
30. Kerrigan, James F., *Migrating to Fortran 90*, O'Reilly & Associates, Inc., Sebastopol, 1993. (the example code used in the book is available electronically for free)
31. Bodin, F., Priol, T., Mehrotra, P., and Gannon, D., "Directions in Parallel Programming: HPF, Shared Virtual Memory and Object Parallelism in pC++", ICASE Report No. 94-54, June 1994. (also NASA CR 194943)

4. Incompressible Potential Flow Using Panel Methods

4.1 An Introduction

The incompressible potential flow model provides reliable flowfield predictions over a wide range of conditions. For the potential flow assumption to be valid for aerodynamics calculations the primary requirement is that viscous effects are small in the flowfield, and that the flowfield must be subsonic everywhere. Locally supersonic velocities can occur at surprisingly low freestream Mach numbers. For high-lift airfoils the peak velocities around the leading edge can become supersonic at freestream Mach numbers of $0.20 \sim 0.25$. If the local flow is at such a low speed everywhere that it can be assumed incompressible ($M \leq .4$, say), Laplace's Equation is essentially an exact representation of the inviscid flow. For higher subsonic Mach numbers with small disturbances to the freestream flow, the Prandtl-Glauert (P-G) Equation can be used. The P-G Equation can be converted to Laplace's Equation by a simple transformation.¹ This provides the basis for estimating the initial effects of compressibility on the flowfield, i.e., "linearized" subsonic flow. In both cases, the flowfield can be found by the solution of a single linear partial differential equation. Not only is the mathematical problem much simpler than any of the other equations that can be used to model the flowfield, but since the problem is linear, a large body of mathematical theory is available.

The Prandtl-Glauert Equation can also be used to describe supersonic flows. In that case the mathematical type of the equation is hyperbolic, and will be mentioned briefly in Chapter 12. Recall the important distinction between the two cases:

subsonic flow:	elliptic PDE, each point influences every other point,
supersonic flow:	hyperbolic PDE, discontinuities exist, "zone of influence" solution dependency.

In this chapter we consider incompressible flow only. One of the key features of Laplace's Equation is the property that allows the equation governing the flowfield to be converted from a 3D problem throughout the field to a 2D problem for finding the potential on the surface. The solution is then found using this property by distributing "singularities" of unknown strength over discretized portions of the surface: *panels*. Hence the flowfield solution is found by representing

the surface by a number of panels, and solving a linear set of algebraic equations to determine the unknown strengths of the singularities.* The flexibility and relative economy of the panel methods is so important in practice that the methods continue to be widely used despite the availability of more exact methods (which generally aren't yet capable of treating the range of geometries that the panel method codes can handle). An entry into the panel method literature is available through two recent reviews by Hess,²³ the survey by Erickson,⁴ and the book by Katz and Plotkin.⁵

The general derivation of the integral equation for the potential solution of Laplace's equation is given in Section 4.3. Complete details are presented for one specific approach to solving the integral equation in Section 4.4. For clarity and simplicity of the algebra, the analysis will use the two-dimensional case to illustrate the methods following the analysis given by Moran.⁶ This results in two ironic aspects of the presentation:

- The algebraic forms of the singularities are different between 2D and 3D, due to 3D relief. You can't use the actual formulas we derive in Section 4.4 for 3D problems.
- The power of panel methods arises in three-dimensional applications. Two-dimensional work in computational aerodynamics is usually done in industry using more exact mappings,** not panels.

After the general derivation, a panel method is used to examine the aerodynamics of airfoils. Finally, an example and some distinctive aspects of the 3D problem are presented.

4.2 Some Potential Theory

Potential theory is an extremely well developed (old) and elegant mathematical theory, devoted to the solution of Laplace's Equation:

$$\nabla^2 \phi = 0. \quad (4.1)$$

There are several ways to view the solution of this equation. The one most familiar to aerodynamicists is the notion of "singularities". These are algebraic functions which satisfy Laplace's equation, and can be combined to construct flowfields. Since the equation is linear, superposition of solutions can be used. The most familiar singularities are the point source, doublet and vortex. In classical examples the singularities are located inside the body. Unfortunately, an arbitrary body shape cannot be created using singularities placed inside the body. A more sophisticated approach has to be used to determine the potential flow over arbitrary shapes. Mathematicians have developed this theory. We will draw on a few selected results to help understand the development of panel methods. Initially, we are interested in the specification of the boundary conditions. Consider the situation illustrated Fig. 4-1.

* The singularities are distributed across the panel. They are not specified at a point. However, the boundary conditions usually are satisfied at a specific location.

** These will be mentioned in more detail in Chapter 9.

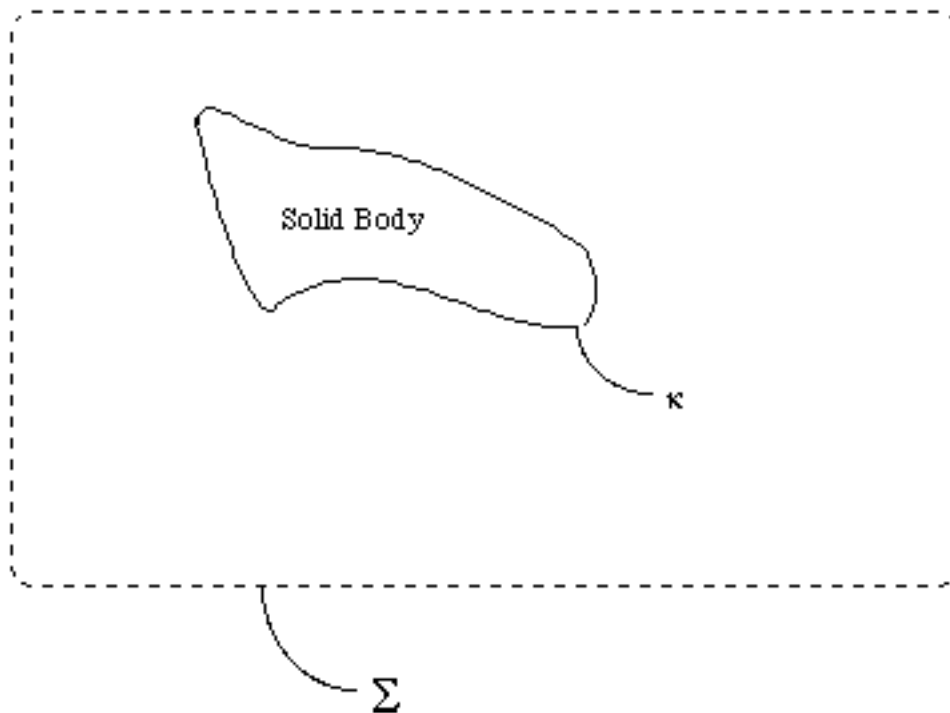


Figure 4-1. Boundaries for flowfield analysis.

The flow pattern is uniquely determined by giving either:

$$f \text{ on } S + \kappa \text{ \{Dirichlet Problem: Design\}} \quad (4-2)$$

or

$$\partial f / \partial n \text{ on } S + \kappa \text{ \{Neuman Problem: Analysis\}}. \quad (4-3)$$

Potential flow theory states that you cannot specify both arbitrarily, but can have a mixed boundary condition, $af + b\partial f / \partial n$ on $S + \kappa$. The Neumann Problem is identified as “analysis” above because it naturally corresponds to the problem where the flow through the surface is specified (usually zero). The Dirichlet Problem is identified as “design” because it tends to correspond to the aerodynamic case where a surface pressure distribution is specified and the body shape corresponding to the pressure distribution is sought. Because of the wide range of problem formulations available in linear theory, some analysis procedures appear to be Dirichlet problems, but Eq. (4-3) must still be used.

Some other key properties of potential flow theory:

- If either f or $\partial f / \partial n$ is zero everywhere on $S + \kappa$ then $f = 0$ at all interior points.
- f cannot have a maximum or minimum at any interior point. Its maximum value can only occur on the surface boundary, and therefore the minimum pressure (and maximum velocity) occurs on the surface.

4.3 Derivation of the Integral Equation for the Potential

We need to obtain the equation for the potential in a form suitable for use in panel method calculations. This section follows the presentation given by Karamcheti⁷ on pages 344-348 and Katz and Plotkin⁵ on pages 52-58. An equivalent analysis is given by Moran⁶ in his Section 8.1. The objective is to obtain an expression for the potential anywhere in the flowfield in terms of values on the surface bounding the flowfield. Starting with the Gauss Divergence Theorem, which relates a volume integral and a surface integral,

$$\iiint_R \text{div} \mathbf{A} dV = \iint_S \mathbf{A} \cdot \mathbf{n} dS \quad (4-4)$$

we follow the classical derivation and consider the interior problem as shown in Fig. 4-2.

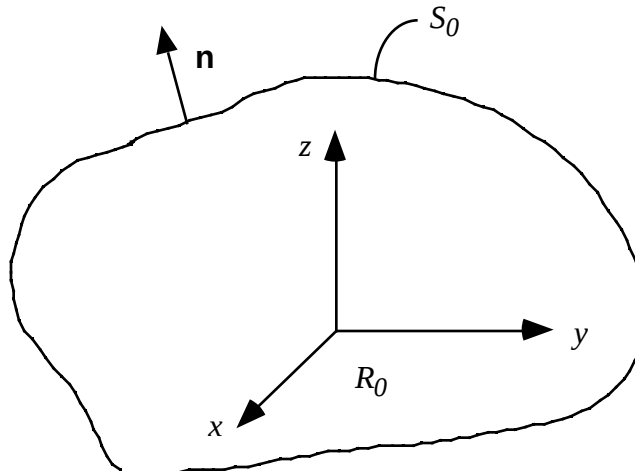


Figure 4-2. Nomenclature for integral equation derivation.

To start the derivation introduce the vector function of two scalars:

$$\mathbf{A} = w \text{grad} c - c \text{grad} w. \quad (4-5)$$

Substitute this function into the Gauss Divergence Theorem, Eq. (4-4), to obtain:

$$\iiint_R \text{div}(w \text{grad} c - c \text{grad} w) dV = \iint_S (w \text{grad} c - c \text{grad} w) \cdot \mathbf{n} dS. \quad (4-6)$$

Now use the vector identity: $\text{div}(\mathbf{F} \phi) = \mathbf{F} \cdot \nabla \phi + \phi \text{div} \mathbf{F}$ to simplify the left hand side of Eq. (4-6).

Recalling that $\text{div} \mathbf{A} = \nabla \cdot \mathbf{A}$, write the integrand of the LHS of Eq. (4-6) as:

$$\begin{aligned} \text{div}(w \text{grad} c - c \text{grad} w) &= \nabla \cdot (w \nabla c - c \nabla w) \\ &= w \nabla^2 c + \nabla w \cdot \nabla c - c \nabla^2 w - \nabla c \cdot \nabla w \\ &= w \nabla^2 c - c \nabla^2 w \end{aligned} \quad (4-7)$$

Substituting the result of Eq. (4-7) for the integrand in the LHS of Eq. (4-6), we obtain:

$$\int_R (\psi^2 c - c \psi^2) dV = \int_S (\psi \text{grad} c - c \text{grad} \psi) \cdot \mathbf{n} dS, \quad (4-8)$$

or equivalently (recalling that $\text{grad} c \cdot \mathbf{n} = \partial c / \partial n$),

$$\int_R (\psi^2 c - c \psi^2) dV = \int_S \psi \frac{\partial c}{\partial n} - c \frac{\partial \psi}{\partial n} dS. \quad (4-9)$$

Either statement is known as Green's theorem of the second form.

Now, define $w = 1/r$ and $c = f$, where f is a harmonic function (a function that satisfies Laplace's equation). The $1/r$ term is a source singularity in three dimensions. This makes our analysis three-dimensional. In two dimensions the form of the source singularity is $\ln r$, and a two-dimensional analysis starts by defining $w = \ln r$. Now rewrite Eq. (4-8) using the definitions of w and c given at the first of this paragraph and switch sides,

$$\int_{S_0} \psi \frac{\partial}{\partial n} \left(f - f \frac{\partial}{\partial n} \frac{1}{r} \right) \cdot \mathbf{n} dS = \int_{R_0} \psi \left(\frac{\partial}{\partial n} \frac{1}{r} - \frac{1}{r} \frac{\partial \psi}{\partial n} \right) dV. \quad (4-10)$$

R_0 is the region enclosed by the surface S_0 . Recognize that on the right hand side the first term, $\frac{\partial}{\partial n} \frac{1}{r}$, is equal to zero by definition so that Eq. (4-10) becomes

$$\int_{S_0} \psi \left(f - f \frac{\partial}{\partial n} \frac{1}{r} \right) \cdot \mathbf{n} dS = - \int_{R_0} \psi \frac{\partial}{\partial n} \frac{1}{r} dV. \quad (4-11)$$

If a point P is external to S_0 , then $\frac{\partial}{\partial n} \frac{1}{r} = 0$ everywhere since $1/r$ is a source, and thus satisfies Laplace's Equation. This leaves the RHS of Eq. (4-11) equal to zero, with the following result:

$$\int_{S_0} \psi \left(f - f \frac{\partial}{\partial n} \frac{1}{r} \right) \cdot \mathbf{n} dS = 0. \quad (4-12)$$

However, we *have* included the origin in our region S_0 as defined above. If P is inside S_0 , then $\frac{\partial}{\partial n} \frac{1}{r} \neq 0$ at $r = 0$. Therefore, we exclude this point by defining a new region which excludes the origin by drawing a sphere of radius ϵ around $r = 0$, and applying Eq. (4-12) to the region between ϵ and S_0 :

$$\underbrace{\oint_{S_0} \frac{\hat{e}_1}{r} f - f \frac{\hat{e}_1}{r} \cdot \hat{n} dS}_{\text{arbitrary region}} - \underbrace{\oint_e \frac{\hat{e}_1}{r} \frac{\partial f}{\partial r} + \frac{f}{r^2} \hat{n} dS}_{\text{sphere}} = 0 \quad (4-13)$$

or:

$$\oint_e \frac{\hat{e}_1}{r} \frac{\partial f}{\partial r} + \frac{f}{r^2} \hat{n} dS = \oint_{S_0} \frac{\hat{e}_1}{r} f - f \frac{\hat{e}_1}{r} \cdot \hat{n} dS. \quad (4-14)$$

Consider the first integral on the left hand side of Eq. (4-14). Let $e \rightarrow 0$, where (as $e \rightarrow 0$) we take f as constant ($\partial f / \partial r = 0$), assuming that f is well-behaved and using the mean value theorem. Then we need to evaluate

$$\oint_e \frac{dS}{r^2}$$

over the surface of the sphere where $e = r$. Recall that for a sphere* the elemental area is

$$dS = r^2 \sin \theta \, d\theta \, d\phi \quad (4-15)$$

where we define the angles in Fig. 4-3. Do not confuse the classical notation for the spherical coordinate angles with the potential function. The spherical coordinate θ will disappear as soon as we evaluate the integral.

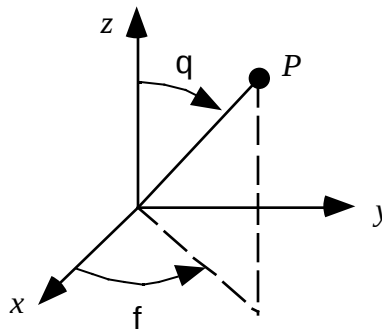


Figure 4-3. Spherical coordinate system nomenclature.

Substituting for dS in the integral above, we get:

$$\oint_e \sin \theta \, d\theta \, d\phi.$$

Integrating from $\theta = 0$ to π , and ϕ from 0 to 2π , we get:

* See Hildebrand, F.B., *Advanced Calculus for Applications*, 2nd Ed., Prentice-Hall, Englewood Cliffs, 1976 for an excellent review of spherical coordinates and vector analysis.

$$\int_0^{f=2p} \int_0^{q=p} \sin q \, dq \, df = 4p. \quad (4-16)$$

The final result for the first integral in Eq. (4-14) is:

$$\oint_{S_0} \frac{\hat{E}_1}{r} \frac{\partial f}{\partial r} + \frac{f}{r^2} dS = 4pf. \quad (4-17)$$

Replacing this integral by its value from Eq. (4-17) in Eq. (4-14), we can write the expression for the potential at any point P as (where the origin can be placed anywhere inside S_0):

$$f(p) = \frac{1}{4p} \oint_{S_0} \frac{\hat{E}_1}{r} f - f \frac{\hat{E}_1}{r^2} \cdot \mathbf{n} dS \quad (4-18)$$

and the value of f at any point P is now known as a function of f and $\partial f / \partial n$ on the boundary.

We used the interior region to allow the origin to be written at point P . This equation can be extended to the solution for f for the region exterior to R_0 . Apply the results to the region between the surface S_B of the body and an arbitrary surface S enclosing S_B and then let S go to infinity. The integrals over S go to f_∞ as S goes to infinity. Thus potential flow theory is used to obtain the important result that the potential at any point P' in the flowfield outside the body can be expressed as:

$$f(p) = f_\infty - \frac{1}{4p} \oint_{S_B} \frac{\hat{E}_1}{r} f - f \frac{\hat{E}_1}{r^2} \cdot \mathbf{n} dS. \quad (4-19)$$

Here the unit normal $\mathbf{n}$ is now considered to be pointing outward and the area can include not only solid surfaces but also wakes. Equation 4-19 can also be written using the dot product of the normal and the gradient as:

$$f(p) = f_\infty - \frac{1}{4p} \oint_{S_B} \frac{\hat{E}_1}{r} \frac{\partial f}{\partial n} - f \frac{\partial}{\partial n} \frac{\hat{E}_1}{r^2} dS. \quad (4-20)$$

The $1/r$ in Eq. (4-19) can be interpreted as a source of strength $\partial f / \partial n$, and the $-(1/r)$ term in Eq. (4-19) as a doublet of strength f . Both of these functions play the role of Green's functions in the mathematical theory. Therefore, we can find the potential as a function of a distribution of sources and doublets over the surface. *The integral in Eq. (4-20) is normally broken up into body and wake pieces.* The wake is generally considered to be infinitely thin. Therefore, only doublets are used to represent the wakes.

Now consider the potential to be given by the superposition of two different known functions, the first and second terms in the integral, Eq. (4-20). These can be taken to be the distribution of the source and doublet strengths, s and m , respectively. Thus Eq (4-20) can be written in the form usually seen in the literature,

$$\phi(p) = \phi_\infty - \frac{1}{4\pi} \oint_{S_B} \frac{\partial \phi}{\partial n} \frac{1}{r} - m \frac{\partial}{\partial n} \frac{1}{r} dS. \quad (4-21)$$

The problem is to find the values of the unknown source and doublet strengths s and m for a specific geometry and given freestream, ϕ_∞ .

What just happened? We replaced the requirement to find the solution over the entire flowfield (a 3D problem) with the problem of finding the solution for the singularity distribution over a surface (a 2D problem). In addition, we now have an integral equation to solve for the unknown surface singularity distributions instead of a partial differential equation. The problem is linear, allowing us to use superposition to construct solutions. We also have the freedom to pick whether to represent the solution as a distribution of sources or doublets distributed over the surface. In practice it's been found best to use a combination of sources and doublets. The theory can be extended to include other singularities.

At one time the change from a 3D to a 2D problem was considered significant. However, the total information content is the same computationally. This shows up as a dense "2D" matrix vs. a sparse "3D" matrix. As methods for sparse matrix solutions evolved, computationally the problems became nearly equivalent. The advantage in using the panel methods arises because there is no need to define a grid throughout the flowfield.

This is the theory that justifies panel methods, i.e., that we can represent the surface by panels with distributions of singularities placed on them. Special precautions must be taken when applying the theory described here. Care should be used to ensure that the region S_B is in fact completely closed. In addition, care must be taken to ensure that the outward normal is properly defined.

Furthermore, in general, the interior problem cannot be ignored. Surface distributions of sources and doublets affect the interior region as well as exterior. In some methods the interior problem is implicitly satisfied. In other methods the interior problem requires explicit attention. The need to consider this subtlety arose when advanced panel methods were developed. The problem is not well posed unless the interior problem is considered, and numerical solutions failed when this aspect of the problem was not addressed. References 4 and 5 provide further discussion.

When the exterior and interior problems are formulated properly the boundary value problem is properly posed. Additional discussions are available in the books by Ashley and Landahl⁸ and Curle and Davis.⁹

We implement the ideas give above by:

- a) approximating the surface by a series of line segments (2D) or panels (3D)
- b) placing distributions of sources and vortices or doublets on each panel.

There are many ways to tackle the problem (and many competing codes). Possible differences in approaches to the implementation include the use of:

- various singularities
- various distributions of the singularity strength over each panel
- panel geometry (panels don't have to be flat).

Recall that superposition allows us to construct the solution by adding separate contributions [Watch out! You have to get all of them. Sometimes this can be a problem]. Thus we write the potential as the sum of several contributions. Figure 4-4 provides an example of a panel representation of an airplane. The wakes are not shown, and a more precise illustration of a panel method representation is given in Section 4.8.

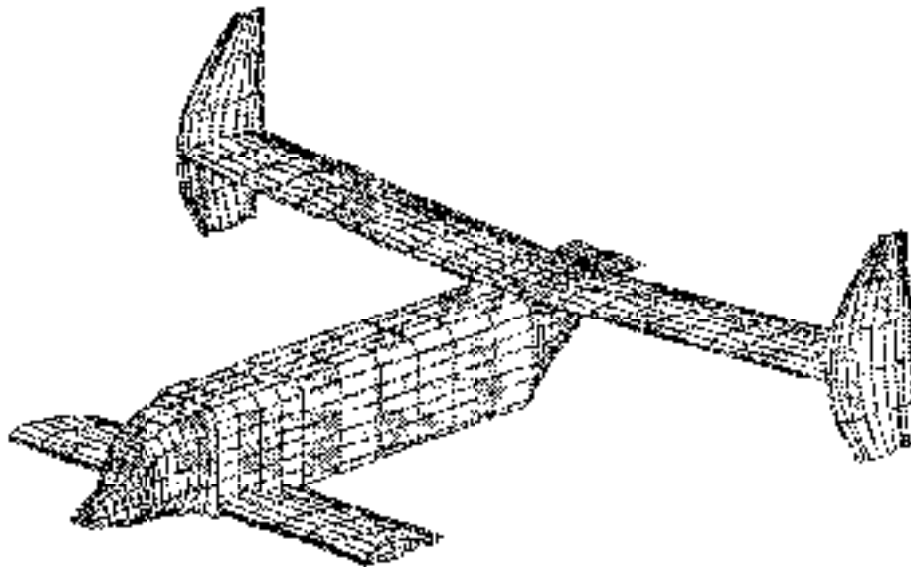


Figure 4-4. Panel model representation of an airplane.
(Joe Mazza, M.S. Thesis, Virginia Tech, 1993).

An example of the implementation of a panel method is carried out in Section 4.4 in two dimensions. To do this, we write down the two-dimensional version of Eq. (4-21). In addition, we use a vortex singularity in place of the doublet singularity (Ref. 4 and 5 provide details on this change). The resulting expression for the potential is:

$$f = \underbrace{f_0}_{\substack{\text{uniform onset flow} \\ = V_\infty x \cos \alpha + V_\infty y \sin \alpha}} + \underbrace{\sum_{j=1}^N \int_{\text{panel } j} \frac{q(s)}{2p} \ln r}_{\substack{q \text{ is the 2D} \\ \text{source strength}}} - \underbrace{\sum_{j=1}^N \int_{\text{panel } j} \frac{g(s)}{2p} q}_{\substack{\text{this is a vortex singularity} \\ \text{of strength } g(s)}} ds \quad (4-22)$$

and $q = \tan^{-1}(y/x)$. Although the equation above shows contributions from various components of the flowfield, the relation is still exact. No small disturbance assumption has been made.

4.4 The Classic Hess and Smith Method

A.M.O. Smith at Douglas Aircraft directed an incredibly productive aerodynamics development group in the late '50s through the early '70s. In this section we describe the implementation of the theory given above that originated in his group.* Our derivation follows Moran's description⁶ of the Hess and Smith method quite closely. The approach is to *i*) break up the surface into straight line segments, *ii*) assume the source strength is constant over each line segment (panel) but has a different value for each panel, and *iii*) the vortex strength is constant and equal over each panel.

Roughly, think of the constant vortices as adding up to the circulation to satisfy the Kutta condition. The sources are required to satisfy flow tangency on the surface (thickness).

Figure 4-5 illustrates the representation of a smooth surface by a series of line segments. The numbering system starts at the lower surface trailing edge and proceeds forward, around the leading edge and aft to the upper surface trailing edge. $N+1$ points define N panels.

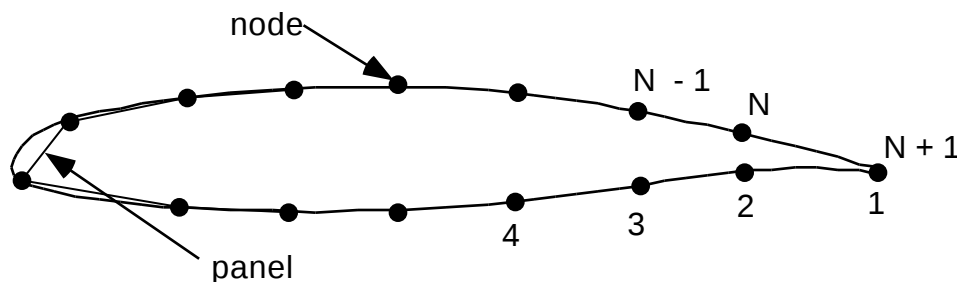


Figure 4-5. Representation of a smooth airfoil with straight line segments.

The potential relation given above in Eq. (4-22) can then be evaluated by breaking the integral up into segments along each panel:

$$f = V_\infty (x \cos \alpha + y \sin \alpha) + \sum_{j=1}^N \int_{\text{panel } j} \frac{q(s)}{2p} \ln r - \frac{g}{2p} q_0 dS \quad (4-23)$$

* In the recent AIAA book, *Applied Computational Aerodynamics*, A.M.O. Smith contributed the first chapter, an account of the initial development of panel methods.

with $q(s)$ taken to be constant on each panel, allowing us to write $q(s) = q_i$, $i = 1, \dots, N$. Here we need to find N values of q_i and one value of g .

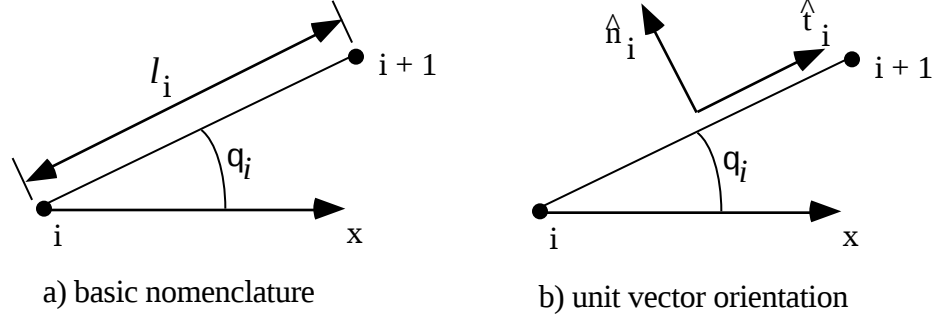


Figure 4-6. Nomenclature for local coordinate systems.

Use Figure 4-6 to define the nomenclature on each panel. Let the i^{th} panel be the one between the i^{th} and $i+1^{\text{th}}$ nodes, and let the i^{th} panel's inclination to the x axis be q . Under these assumptions the sin and cos of q are given by:

$$\sin q_i = \frac{y_{i+1} - y_i}{l_i}, \quad \cos q_i = \frac{x_{i+1} - x_i}{l_i} \quad (4-24)$$

and the normal and tangential unit vectors are:

$$\begin{aligned} \mathbf{n}_i &= -\sin q_i \mathbf{i} + \cos q_i \mathbf{j} \\ \mathbf{t}_i &= \cos q_i \mathbf{i} + \sin q_i \mathbf{j} \end{aligned} \quad (4-25)$$

We will find the unknowns by satisfying the flow tangency condition on each panel at one specific control point (also known as a collocation point) and requiring the solution to satisfy the Kutta condition. The control point will be picked to be at the mid-point of each panel, as shown in Fig. 4-7.

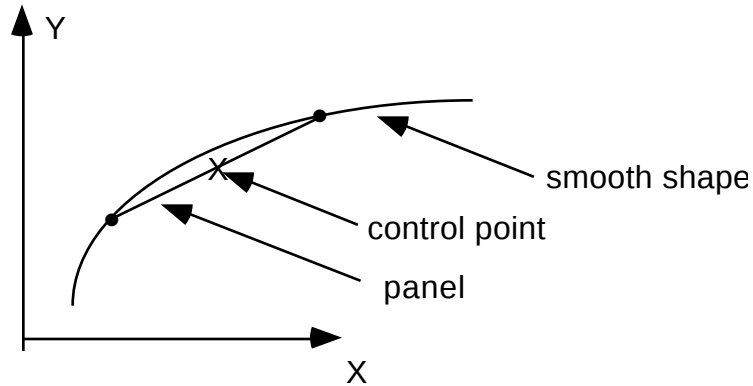


Figure 4-7. Local panel nomenclature.

Thus the coordinates of the midpoint of the control point are given by:

$$\bar{x}_i = \frac{x_i + x_{i+1}}{2}, \quad \bar{y}_i = \frac{y_i + y_{i+1}}{2} \quad (4-26)$$

and the velocity components at the control point $\bar{x}_i, \bar{y}_i$ are $u_i = u(\bar{x}_i, \bar{y}_i)$, $v_i = v(\bar{x}_i, \bar{y}_i)$.

The flow tangency boundary condition is given by $\mathbf{V} \cdot \mathbf{n} = 0$, and is written using the relations given here as:

$$(u_i \mathbf{i} + v_i \mathbf{j}) \cdot (-\sin \alpha_i \mathbf{i} + \cos \alpha_i \mathbf{j}) = 0$$

or

$$-u_i \sin \alpha_i + v_i \cos \alpha_i = 0, \quad \text{for each } i, i = 1, \dots, N. \quad (4-27)$$

The remaining relation is found from the Kutta condition. This condition states that the flow must leave the trailing edge smoothly. Many different numerical approaches have been adopted to satisfy this condition. In practice this implies that at the trailing edge the pressures on the upper and lower surface are equal. Here we satisfy the Kutta condition approximately by equating velocity components tangential to the panels adjacent to the trailing edge on the upper and lower surface. Because of the importance of the Kutta condition in determining the flow, the solution is extremely sensitive to the flow details at the trailing edge. When we make the assumption that the velocities are equal on the top and bottom panels at the trailing edge we need to understand that we must make sure that the last panels on the top and bottom are small and of equal length. Otherwise we have an inconsistent approximation. Accuracy will deteriorate rapidly if the panels are not the same length. We will develop the numerical formula using the nomenclature for the trailing edge shown in Fig. 4-8.

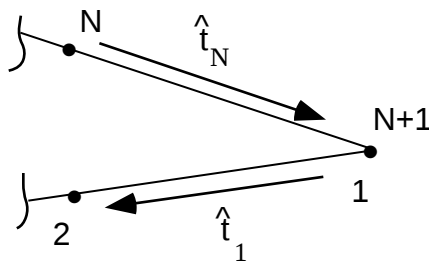


Figure 4-8. Trailing edge panel nomenclature.

Equating the magnitude of the tangential velocities on the upper and lower surface:

$$u_{t_1} = u_{t_N}. \quad (4-28)$$

and taking the difference in direction of the tangential unit vectors into account this is written as

$$\mathbf{V} \cdot \mathbf{t}_1 = -\mathbf{V} \cdot \mathbf{t}_N. \quad (4-29)$$

Carrying out the operation we get the relation:

$$(u_1 \mathbf{i} + v_1 \mathbf{j}) \times (\cos q_1 \mathbf{i} + \sin q_1 \mathbf{j}) = - (u_N \mathbf{i} + v_N \mathbf{j}) \times (\cos q_N \mathbf{i} + \sin q_N \mathbf{j})$$

which is expanded to obtain the final relation:

$$u_1 \cos q_1 + v_1 \sin q_1 = - u_N \cos q_N + v_N \sin q_N \quad (4-30)$$

The expression for the potential in terms of the singularities on each panel and the boundary conditions derived above for the flow tangency and Kutta condition are used to construct a system of linear algebraic equations for the strengths of the sources and the vortex. The steps required are summarized below. Then we will carry out the details of the algebra required in each step.

Steps to determine the solution:

1. Write down the velocities, u_i , v_i , in terms of contributions from all the singularities. This includes q_i , $\mathbf{g}$ from each panel and the influence coefficients which are a function of the geometry only.
2. Find the algebraic equations defining the “influence” coefficients.

To generate the system of algebraic equations:

3. Write down flow tangency conditions in terms of the velocities (N eqn's., $N+1$ unknowns).
4. Write down the Kutta condition equation to get the $N+1$ equation.
5. Solve the resulting linear algebraic system of equations for the q_i , $\mathbf{g}$.
6. Given q_i , $\mathbf{g}$, write down the equations for u_{ti} , the tangential velocity at each panel control point.
7. Determine the pressure distribution from Bernoulli's equation using the tangential velocity on each panel.

We now carry out each step in detail. The algebra gets tedious, but there's no problem in carrying it out. As we carry out the analysis for two dimensions, consider the additional algebra required for the general three dimensional case.

Step 1. Velocities

The velocity components at any point i are given by contributions from the velocities induced by the source and vortex distributions over each panel. The mathematical statement is:

$$\begin{aligned} u_i &= V \cos \alpha + \sum_{j=1}^N q_j u_{sij} + g \sum_{j=1}^N u_{vij} \\ v_i &= V \sin \alpha + \sum_{j=1}^N q_j v_{sij} + g \sum_{j=1}^N v_{vij} \end{aligned} \quad (4-31)$$

where q_j and g are the singularity strengths, and the u_{sij} , v_{sij} , u_{vij} , and v_{vij} are the influence coefficients. As an example, the influence coefficient u_{sij} is the x -component of velocity at x_i due to a unit source distribution over the j^{th} panel.

Step 2. Influence coefficients

To find u_{sij} , v_{sij} , u_{vij} , and v_{vij} we need to work in a local panel coordinate system x^* , y^* which leads to a straightforward means of integrating source and vortex distributions along a straight line segment. This system will be locally aligned with each panel j , and is connected to the global coordinate system as illustrated in Fig. 4-9.

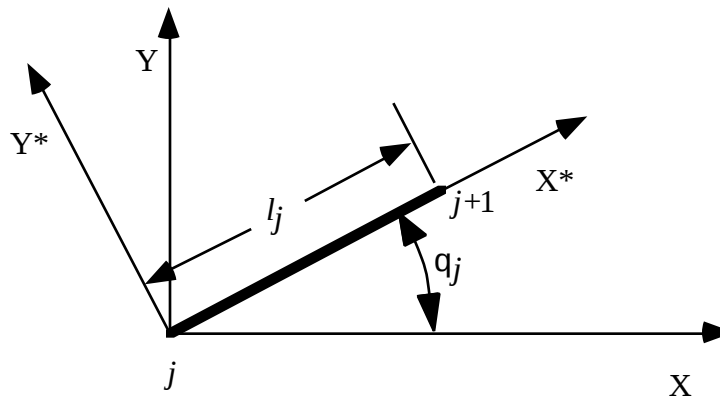


Figure 4-9. Local panel coordinate system and nomenclature.

The influence coefficients determined in the local coordinate system aligned with a particular panel are u^* and v^* , and are transformed back to the global coordinate system by:

$$\begin{aligned} u &= u^* \cos q_j - v^* \sin q_j \\ v &= u^* \sin q_j + v^* \cos q_j \end{aligned} \quad (4-32)$$

We now need to find the velocities induced by the singularity distributions. We consider the source distributions first. The velocity field induced by a source in its natural cylindrical coordinate system is:

$$\mathbf{V} = \frac{Q}{2\pi r} \hat{\mathbf{e}}_r. \quad (4-33)$$

Rewriting in Cartesian coordinates (and noting that the source described in Eq. (4-33) is located at the origin, $r = 0$) we have:

$$u(x,y) = \frac{Q}{2\pi} \frac{x}{x^2 + y^2}, \quad v(x,y) = \frac{Q}{2\pi} \frac{y}{x^2 + y^2}. \quad (4-34)$$

In general, if we locate the sources along the x -axis at a point $x = t$, and integrate over a length l , the velocities induced by the source distributions are obtained from:

$$\begin{aligned} u_s &= \int_{t=0}^{t=l} \frac{q(t)}{2\pi} \frac{x-t}{(x-t)^2 + y^2} dt \\ v_s &= \int_{t=0}^{t=l} \frac{q(t)}{2\pi} \frac{y}{(x-t)^2 + y^2} dt. \end{aligned} \quad (4-35)$$

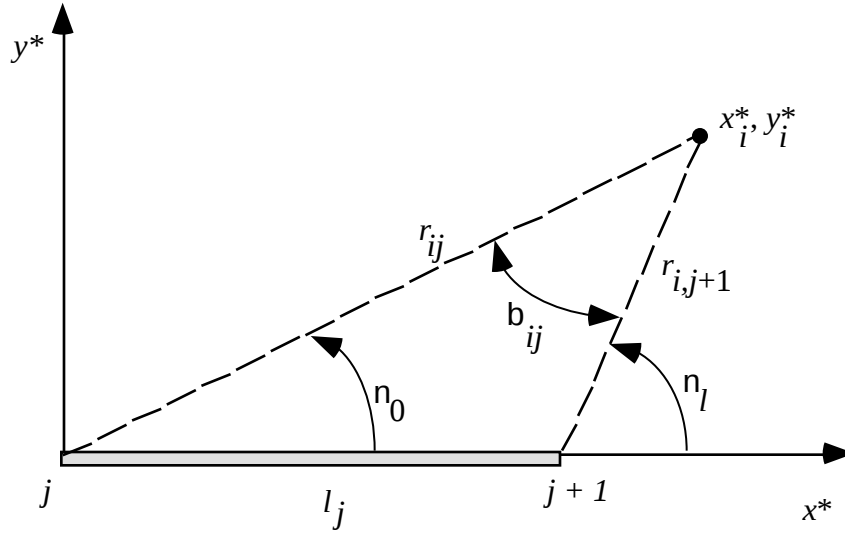
To obtain the influence coefficients, write down this equation in the $(*)$ coordinate system, with $q(t) = 1$ (unit source strength):

$$\begin{aligned} u_{sij}^* &= \frac{1}{2\pi} \int_0^l \frac{x_i^* - t}{(x_i^* - t)^2 + y_i^{*2}} dt \\ v_{sij}^* &= \frac{1}{2\pi} \int_0^l \frac{y_i^*}{(x_i^* - t)^2 + y_i^{*2}} dt. \end{aligned} \quad (4-36)$$

These integrals can be found (from tables) in closed form:

$$\begin{aligned} u_{sij}^* &= -\frac{1}{2\pi} \ln \left[\frac{(x_i^* - t)^2 + y_i^{*2}}{1} \right] \bigg|_{t=0}^{t=l} \\ v_{sij}^* &= \frac{1}{2\pi} \tan^{-1} \left[\frac{y_i^*}{x_i^* - t} \right] \bigg|_{t=0}^{t=l} \end{aligned} \quad (4-37)$$

To interpret these expressions examine Fig. 4-10. The notation adopted and illustrated in the sketch makes it easy to translate the results back to global coordinates.

Figure 4-10. Relations between the point x^*, y^* and a panel.

Note that the formulas for the integrals given in Eq. (4-37) can be interpreted as a radius and an angle. Substituting the limits into the expressions and evaluating results in the final formulas for the influence coefficients due to the sources:

$$u_{sij}^* = -\frac{1}{2p} \ln \frac{\hat{r}_{i,j+1}}{\hat{r}_{ij}} \quad (4-38)$$

$$v_{sij}^* = \frac{n_l - n_0}{2p} = \frac{b_{ij}}{2p}$$

Here r_{ij} is the distance from the j^{th} node to the point i , which is taken to be the control point location of the i^{th} panel. The angle b_{ij} is the angle subtended at the middle of the i^{th} panel by the j^{th} panel.

The case of determining the influence coefficient for a panel's influence on itself requires some special consideration. Consider the influence of the panel source distribution on itself. The source induces normal velocities, and no tangential velocities, Thus, $u_{sii}^* = 0$ and v_{sii}^* depends on the side from which you approach the panel control point. Approaching the panel control point from the outside leads to $b_{ii} = p$, while approaching from inside leads to $b_{ii} = -p$. Since we are working on the exterior problem,

$$b_{ii} = p, \quad (4-39)$$

and to keep the correct sign on b_{ij} , $j \neq i$, use the FORTRAN subroutine ATAN2, which takes into account the correct quadrant of the angle.*

* Review a FORTRAN manual to understand how ATAN2 is used.

Now consider the influence coefficients due to vortices. There is a simple connection between source and vortex flows that allows us to use the previous results obtained for the source distribution directly in the vortex singularity distribution analysis.

The velocity due to a point vortex is usually given as:

$$V = - \frac{G}{2\pi r} \mathbf{e}_q. \quad (4-40)$$

Compared to the source flow, the u , v components simply trade places (with consideration of the direction of the flow to define the proper signs). In Cartesian coordinates the velocity due to a point vortex is:

$$u(x,y) = + \frac{G}{2\pi} \frac{y}{x^2 + y^2}, \quad v(x,y) = - \frac{G}{2\pi} \frac{x}{x^2 + y^2}. \quad (4-41)$$

where the origin (the location of the vortex) is $x = y = 0$.

Using the same analysis used for source singularities for vortex singularities the equivalent vortex distribution results can be obtained. Summing over the panel with a vortex strength of unity we get the formulas for the influence coefficients due to the vortex distribution:

$$\begin{aligned} u_{vij}^* &= + \frac{1}{2\pi} \oint_0^{t_j} \frac{y_i^*}{(x_i^* - t)^2 + y_i^{*2}} dt = \frac{b_{ij}}{2\pi} \\ v_{vij}^* &= - \frac{1}{2\pi} \oint_0^{t_j} \frac{x_i^* - t}{(x_i^* - t)^2 + y_i^{*2}} dt = \frac{1}{2\pi} \ln \frac{\hat{\mathbf{A}}_{i,j+1} \cdot \hat{\mathbf{r}}_{ij}}{\hat{\mathbf{E}}_{ij}} \end{aligned} \quad (4-42)$$

where the definitions and special circumstances described for the source singularities are the same in the current case of distributed vortices.* In this case the vortex distribution induces an axial velocity on itself at the sheet, and no normal velocity.

Step 3. Flow tangency conditions to get N equations.

Our goal is to obtain a system of equations of the form:

$$\hat{\mathbf{A}}_{ij} A_{ij} q_j + A_{i,N+1} g = b_i \quad i = 1, \dots, N \quad (4-43)$$

which are solved for the unknown source and vortex strengths.

Recall the flow tangency condition was found to be:

$$-u_i \sin q_i + v_i \cos q_i = 0, \quad \text{for each } i, \quad i = 1, \dots, N \quad (4-44)$$

* Note that Moran's Equation (4-88) has a sign error typo. The correct sign is used in Eq. (4-42) above.

where the velocities are given by:

$$\begin{aligned} u_i &= V \cdot \cos \alpha + \sum_{j=1}^N \hat{A}_{ij} q_j u_{s_{ij}} + g \sum_{j=1}^N \hat{A}_{ij} u_{v_{ij}} \\ v_i &= V \cdot \sin \alpha + \sum_{j=1}^N \hat{A}_{ij} q_j v_{s_{ij}} + g \sum_{j=1}^N \hat{A}_{ij} v_{v_{ij}} \end{aligned} \quad (4-45)$$

Substituting into Eq. (4-45), the flow tangency equations, Eq. (4-44), above:

$$\frac{\hat{E}}{\hat{A}_{iE}} V \cdot \cos \alpha - \sum_{j=1}^N \hat{A}_{ij} q_j u_{s_{ij}} - g \sum_{j=1}^N \hat{A}_{ij} u_{v_{ij}} \hat{\sim} \sin q_i + \frac{\hat{E}}{\hat{A}_{iE}} V \cdot \sin \alpha + \sum_{j=1}^N \hat{A}_{ij} q_j v_{s_{ij}} + g \sum_{j=1}^N \hat{A}_{ij} v_{v_{ij}} \hat{\sim} \cos q_i = 0 \quad (4-46)$$

which is rewritten into:

$$\begin{aligned} & \left[-V \cdot \sin q_i \cos \alpha + V \cdot \cos q_i \sin \alpha \right] - \sin q_i \sum_{j=1}^N \hat{A}_{ij} q_j u_{s_{ij}} + \cos q_i \sum_{j=1}^N \hat{A}_{ij} q_j v_{s_{ij}} \\ & - g \sin q_i \sum_{j=1}^N \hat{A}_{ij} u_{v_{ij}} + g \cos q_i \sum_{j=1}^N \hat{A}_{ij} v_{v_{ij}} = 0 \end{aligned}$$

or

$$\begin{aligned} & \underbrace{V \cdot (\cos q_i \sin \alpha - \sin q_i \cos \alpha)}_{-b_i} + \sum_{j=1}^N \underbrace{\hat{A}_{ij} (\cos q_i v_{s_{ij}} - \sin q_i u_{s_{ij}})}_{A_{ij}} q_j \\ & + \underbrace{\frac{\hat{E}}{\hat{A}_{iE}} \cos q_i \sum_{j=1}^N \hat{A}_{ij} v_{v_{ij}} - \sin q_i \sum_{j=1}^N \hat{A}_{ij} u_{v_{ij}} \hat{\sim} g}_{A_{i,N+1}} = 0 \end{aligned} \quad (4-47)$$

Now get the formulas for A_{ij} and $A_{i,N+1}$ by replacing the formulas for $u_{s_{ij}}, v_{s_{ij}}, u_{v_{ij}}, v_{v_{ij}}$ with the $()^*$ values, where:

$$\begin{aligned} u &= u^* \cos q_j - v^* \sin q_j \\ v &= u^* \sin q_j + v^* \cos q_j \end{aligned} \quad (4-48)$$

and we substitute into Eq. (4-47) for the values in A_{ij} and $A_{i,N+1}$ above.

Start with:

$$\begin{aligned}
A_{ij} &= \cos q_i v_{sij} - \sin q_i u_{sij} \\
&= \cos q_i \left(u_{sij}^* \sin q_j + v_{sij}^* \cos q_j \right) - \sin q_i \left(u_{sij}^* \cos q_j - v_{sij}^* \sin q_j \right) \\
&= \left(\cos q_i \sin q_j - \sin q_i \cos q_j \right) u_{sij}^* + \left(\cos q_i \cos q_j - \sin q_i \sin q_j \right) v_{sij}^*
\end{aligned} \tag{4-49}$$

and we use trigonometric identities to combine terms into a more compact form. Operating on the first term in parenthesis:

$$\begin{aligned}
\cos q_i \sin q_j &= \frac{1}{2} \sin(q_i + q_j) + \frac{1}{2} \sin(-\{q_i - q_j\}) \\
&= \frac{1}{2} \sin(q_i + q_j) - \frac{1}{2} \sin(q_i - q_j)
\end{aligned} \tag{4-50}$$

and

$$\sin q_i \sin q_j = \frac{1}{2} \sin(q_i + q_j) + \frac{1}{2} \sin(q_i - q_j) \tag{4-51}$$

results in:

$$\left(\cos q_i \sin q_j - \sin q_i \cos q_j \right) = 0 - \sin(q_i - q_j). \tag{4-52}$$

Moving to the second term in parentheses above:

$$\begin{aligned}
\cos q_i \cos q_j &= \frac{1}{2} \cos(q_i + q_j) + \frac{1}{2} \cos(q_i - q_j) \\
\sin q_i \sin q_j &= \frac{1}{2} \cos(q_i - q_j) - \frac{1}{2} \cos(q_i + q_j)
\end{aligned} \tag{4-53}$$

and

$$\begin{aligned}
\cos q_i \cos q_j + \sin q_i \sin q_j &= \frac{1}{2} \cos(q_i + q_j) + \frac{1}{2} \cos(q_i - q_j) + \frac{1}{2} \cos(q_i - q_j) - \frac{1}{2} \cos(q_i + q_j) \\
&= \cos(q_i - q_j)
\end{aligned} \tag{4-54}$$

so that the expression for A_{ij} can be written as:

$$A_{ij} = -\sin(q_i - q_j) u_{sij}^* + \cos(q_i - q_j) v_{sij}^* \tag{4-55}$$

and using the definitions of

$$A_{ij} = \frac{1}{2p} \sin(q_i - q_j) \ln \frac{\hat{E}_{r_i, j+1}}{\hat{E}_{r_i, j}} + \frac{1}{2p} \cos(q_i - q_j) b_{ij}. \tag{4-56}$$

Now look at the expression for b_i identified in (4-47):

$$b_i = V \cdot (\cos q_i \sin \alpha - \sin q_i \cos \alpha) \tag{4-57}$$

where in the same fashion used above:

$$\begin{aligned}\cos q_i \sin a &= \frac{1}{2} \sin(q_i + a) - \frac{1}{2} \sin(q_i - a) \\ \sin q_i \cos a &= \frac{1}{2} \sin(q_i + a) + \frac{1}{2} \sin(q_i - a)\end{aligned}\quad (4-58)$$

and

$$\cos q_i \sin a - \sin q_i \cos a = -\sin(q_i - a) \quad (4-59)$$

so that we get:

$$b_i = V_\infty \sin(q_i - a). \quad (4-60)$$

Finally, work with the $A_{i,N+1}$ term:

$$\begin{aligned}A_{i,N+1} &= \hat{\mathbf{A}} \cos q_i \hat{\mathbf{A}}^N v_{vij} - \sin q_i \hat{\mathbf{A}}^N u_{vij} \\ &= \cos q_i \hat{\mathbf{A}}^N \left(u_{vij}^* \sin q_j + v_{vij}^* \cos q_j \right) - \sin q_i \hat{\mathbf{A}}^N \left(u_{vij}^* \cos q_j - v_{vij}^* \sin q_j \right) \\ &= \hat{\mathbf{A}}^N \left(\cos q_i \sin q_j u_{vij}^* + \cos q_i \cos q_j v_{vij}^* - \sin q_i \cos q_j u_{vij}^* + \sin q_i \sin q_j v_{vij}^* \right) \\ &= \hat{\mathbf{A}}^N \left[\underbrace{(\cos q_i \cos q_j + \sin q_i \sin q_j)}_a v_{vij}^* + \underbrace{(\cos q_i \sin q_j - \sin q_i \cos q_j)}_b u_{vij}^* \right]\end{aligned}\quad (4-61)$$

and a and b can be simplified to:

$$\begin{aligned}a &= \cos(q_i - q_j) \\ b &= -\sin(q_i - q_j)\end{aligned}\quad (4-62)$$

Substituting for a and b in the above equation:

$$A_{i,N+1} = \hat{\mathbf{A}}^N \left\{ \cos(q_i - q_j) v_{vij}^* - \sin(q_i - q_j) u_{vij}^* \right\} \quad (4-63)$$

and using the definition of $\hat{\mathbf{A}}$ we arrive at the final result:

$$A_{i,N+1} = \frac{1}{2p} \hat{\mathbf{A}}^N \left[\cos(q_i - q_j) \ln \frac{\hat{\mathbf{r}}_{i,j+1}}{\hat{\mathbf{r}}_{i,j}} - \sin(q_i - q_j) b_{ij} \right] \quad (4-64)$$

To sum up (repeating the results found above), the equations for the A_{ij} , $A_{i,N+1}$, and b_i are given by (4-56), (4-64), and (4-60):

$$A_{ij} = \frac{1}{2p} \sin(q_i - q_j) \ln \frac{\hat{E}_{r_{i,j+1}}}{\hat{E}_{r_{i,j}}} + \frac{1}{2p} \cos(q_i - q_j) b_{ij}$$

$$A_{i,N+1} = \frac{1}{2p} \sum_{j=1}^N \hat{A}_{ij} \cos(q_i - q_j) \ln \frac{\hat{E}_{r_{i,j+1}}}{\hat{E}_{r_{i,j}}} - \sin(q_i - q_j) b_{ij}$$

$$b_i = V_\infty \sin(q_i - \alpha)$$

Step 4. Kutta Condition to get equation N+1

To complete the system of $N+1$ equations, we use the Kutta condition, which we previously defined as:

$$u_1 \cos q_1 + v_1 \sin q_1 = -u_N \cos q_N - v_N \sin q_N \quad (4-66)$$

and substitute into this expression the formulas for the velocities due to the freestream and singularities given in equation (4-31). In this case they are written as:

$$\begin{aligned} u_1 &= V_\infty \cos \alpha + \sum_{j=1}^N \hat{A}_{1j} q_j u_{s_{1j}} + g \sum_{j=1}^N \hat{A}_{1j} u_{v_{1j}} \\ v_1 &= V_\infty \sin \alpha + \sum_{j=1}^N \hat{A}_{1j} q_j v_{s_{1j}} + g \sum_{j=1}^N \hat{A}_{1j} v_{v_{1j}} \\ u_N &= V_\infty \cos \alpha + \sum_{j=1}^N \hat{A}_{Nj} q_j u_{s_{Nj}} + g \sum_{j=1}^N \hat{A}_{Nj} u_{v_{Nj}} \\ v_N &= V_\infty \sin \alpha + \sum_{j=1}^N \hat{A}_{Nj} q_j v_{s_{Nj}} + g \sum_{j=1}^N \hat{A}_{Nj} v_{v_{Nj}} \end{aligned} \quad (4-67)$$

Substituting into the Kutta condition equation we obtain:

$$\begin{aligned}
& \frac{\hat{E}}{\hat{A}} V. \cos \alpha + \sum_{j=1}^N \hat{A} q_j u_{s1j} + g \sum_{j=1}^N \hat{A} u_{v1j} \tilde{\sim} \cos q_1 \\
& + \frac{\hat{E}}{\hat{A}} V. \sin \alpha + \sum_{j=1}^N \hat{A} q_j v_{s1j} + g \sum_{j=1}^N \hat{A} v_{v1j} \tilde{\sim} \sin q_1 \\
& + \frac{\hat{E}}{\hat{A}} V. \cos \alpha + \sum_{j=1}^N \hat{A} q_j u_{sNj} + g \sum_{j=1}^N \hat{A} u_{vNj} \tilde{\sim} \cos q_N \\
& + \frac{\hat{E}}{\hat{A}} V. \sin \alpha + \sum_{j=1}^N \hat{A} q_j v_{sNj} + g \sum_{j=1}^N \hat{A} v_{vNj} \tilde{\sim} \sin q_N = 0
\end{aligned} \tag{4-68}$$

and our goal will be to manipulate this expression into the form:

$$\sum_{j=1}^N \hat{A} A_{N+1,j} q_j + A_{N+1,N+1} g = b_{N+1} \tag{4-69}$$

which is the $N + 1^{st}$ equation which completes the system for the $N + 1$ unknowns.

Start by regrouping terms in the above equation to write it in the form:

$$\begin{aligned}
& \sum_{j=1}^N \hat{A} \left(u_{s1j} \cos q_1 + v_{s1j} \sin q_1 + u_{sNj} \cos q_N + v_{sNj} \sin q_N \right) q_j \\
& + \underbrace{\sum_{j=1}^N \hat{A} \left(u_{v1j} \cos q_1 + v_{v1j} \sin q_1 + u_{vNj} \cos q_N + v_{vNj} \sin q_N \right)}_{A_{N+1,N+1}} g \\
& = - \underbrace{\left(V. \cos \alpha \cos q_1 + V. \sin \alpha \sin q_1 + V. \cos \alpha \cos q_N + V. \sin \alpha \sin q_N \right)}_{b_{N+1}}
\end{aligned} \tag{4-70}$$

Obtain the final expression for b_{N+1} first:

$$b_{N+1} = - V. \left(\underbrace{\cos \alpha \cos q_1 + \sin \alpha \sin q_1}_{\cos(\alpha - q_1)} + \underbrace{\cos \alpha \cos q_N + \sin \alpha \sin q_N}_{\cos(\alpha - q_N)} \right) \tag{4-71}$$

and using the trigonometric identities to obtain the expression for b_{N+1} :

$$b_{N+1} = - V. \cos(q_1 - \alpha) - V. \cos(q_N - \alpha) \tag{4-72}$$

where we made use of $\cos(-A) = \cos A$.

Now work with $A_{N+1,j}$:

$$A_{N+1,j} = u_{s1j} \cos q_1 + v_{s1j} \sin q_1 + u_{sNj} \cos q_N + v_{sNj} \sin q_N \quad (4-73)$$

and replace the influence coefficients with their related (*) values:

$$\begin{aligned} u_{s1j} &= u_{s1j}^* \cos q_j - v_{s1j}^* \sin q_j \\ v_{s1j} &= u_{s1j}^* \sin q_j + v_{s1j}^* \cos q_j \\ u_{sNj} &= u_{sNj}^* \cos q_j - v_{sNj}^* \sin q_j \\ v_{sNj} &= u_{sNj}^* \sin q_j + v_{sNj}^* \cos q_j \end{aligned} \quad (4-74)$$

so that we can write:

$$\begin{aligned} A_{N+1,j} &= \left(u_{s1j}^* \cos q_j - v_{s1j}^* \sin q_j \right) \cos q_1 \\ &\quad + \left(u_{s1j}^* \sin q_j + v_{s1j}^* \cos q_j \right) \sin q_1 \\ &\quad + \left(u_{sNj}^* \cos q_j - v_{sNj}^* \sin q_j \right) \cos q_N \\ &\quad + \left(u_{sNj}^* \sin q_j + v_{sNj}^* \cos q_j \right) \sin q_N \end{aligned} \quad (4-75)$$

or:

$$\begin{aligned} A_{N+1,j} &= \left(\cos q_j \cos q_1 + \sin q_j \sin q_1 \right) u_{s1j}^* \\ &\quad + \left(\cos q_j \cos q_N + \sin q_j \sin q_N \right) u_{sNj}^* \\ &\quad + \left(\cos q_j \sin q_1 - \sin q_j \cos q_1 \right) v_{s1j}^* \\ &\quad + \left(\cos q_j \sin q_N - \sin q_j \cos q_N \right) v_{sNj}^* \end{aligned} \quad (4-76)$$

Use the following trig relations to simplify the equation:

$$\begin{aligned} \cos q_j \cos q_1 + \sin q_j \sin q_1 &= \cos(q_j - q_1) \\ \cos q_j \cos q_N + \sin q_j \sin q_N &= \cos(q_j - q_N) \\ \cos q_j \sin q_1 - \sin q_j \cos q_1 &= -\sin(q_j - q_1) \\ \cos q_j \sin q_N - \sin q_j \cos q_N &= -\sin(q_j - q_N) \end{aligned} \quad (4-77)$$

and substitute into Eq. (4-76) to obtain:

$$A_{N+1,j} = \cos(q_j - q_1) u_{s1j}^* + \cos(q_j - q_N) u_{sNj}^* - \sin(q_j - q_1) v_{s1j}^* - \sin(q_j - q_N) v_{sNj}^* \quad (4-78)$$

Use the definition of the influence coefficients:

$$\begin{aligned} u_{s1j}^* &= -\frac{1}{2p} \ln \frac{\hat{E}_{r_{1,j+1}}}{\hat{E}_{r_{1,j}}} & u_{sNj}^* &= -\frac{1}{2p} \ln \frac{\hat{E}_{r_{N,j+1}}}{\hat{E}_{r_{N,j}}} \\ v_{s1j}^* &= \frac{b_{1,j}}{2p} & v_{sNj}^* &= \frac{b_{N,j}}{2p} \end{aligned} \quad (4-79)$$

to write the equation for $A_{N+1,j}$:

$$\begin{aligned} A_{N+1,j} &= -\frac{\cos(q_j - q_1)}{2p} \ln \frac{\hat{E}_{r_{1,j+1}}}{\hat{E}_{r_{1,j}}} - \frac{\cos(q_j - q_N)}{2p} \ln \frac{\hat{E}_{r_{N,j+1}}}{\hat{E}_{r_{N,j}}} \\ &\quad - \frac{\sin(q_j - q_1)}{2p} b_{1,j} - \frac{\sin(q_j - q_N)}{2p} b_{N,j} \end{aligned} \quad (4-80)$$

Finally, use symmetry and odd/even relations to write down the final form:

$$A_{N+1,j} = \frac{1}{2p} \left[\sin(q_1 - q_j) b_{1,j} + \sin(q_N - q_j) b_{N,j} - \cos(q_1 - q_j) \ln \frac{\hat{E}_{r_{1,j+1}}}{\hat{E}_{r_{1,j}}} - \cos(q_N - q_j) \ln \frac{\hat{E}_{r_{N,j+1}}}{\hat{E}_{r_{N,j}}} \right] \quad (4-81)$$

Now work with $A_{N+1,N+1}$:

$$A_{N+1,N+1} = \sum_{j=1}^N \left(u_{v1j} \cos q_1 + v_{v1j} \sin q_1 + u_{vNj} \cos q_N + v_{vNj} \sin q_N \right) \quad (4-82)$$

where we substitute in for the $()^*$ coordinate system, Eq. (4-32), and obtain:

$$A_{N+1,N+1} = \sum_{j=1}^N \left[\left(u_{v1j}^* \cos q_j - v_{v1j}^* \sin q_j \right) \cos q_1 + \left(u_{v1j}^* \sin q_j + v_{v1j}^* \cos q_j \right) \sin q_1 \right. \\ \left. + \left(u_{vNj}^* \cos q_j - v_{vNj}^* \sin q_j \right) \cos q_N + \left(u_{vNj}^* \sin q_j + v_{vNj}^* \cos q_j \right) \sin q_N \right] \quad (4-83)$$

or:

$$A_{N+1,N+1} = \sum_{j=1}^N \left[\frac{\cos q_j \cos q_1 + \sin q_j \sin q_1}{\cos(q_j - q_1)} u_{v1j}^* + \frac{\cos q_j \sin q_1 - \sin q_j \cos q_1}{-\sin(q_j - q_1)} v_{v1j}^* \right] + \sum_{j=1}^N \left[\frac{\cos q_j \cos q_N + \sin q_j \sin q_N}{\cos(q_j - q_N)} u_{vNj}^* + \frac{\cos q_j \sin q_N - \sin q_j \cos q_N}{-\sin(q_j - q_N)} v_{vNj}^* \right] \quad (4-84)$$

which is:

$$A_{N+1,N+1} = \sum_{j=1}^N \left[\cos(q_j - q_1) u_{v1j}^* - \sin(q_j - q_1) v_{v1j}^* \right] + \sum_{j=1}^N \left[\cos(q_j - q_N) u_{vNj}^* - \sin(q_j - q_N) v_{vNj}^* \right]$$

and using odd/even trig relations we get the form given by Moran⁶:

$$A_{N+1,N+1} = \sum_{j=1}^N \left[\sin(q_1 - q_j) v_{v1j}^* + \sin(q_N - q_j) v_{vNj}^* \right] + \sum_{j=1}^N \left[\cos(q_1 - q_j) u_{v1j}^* + \cos(q_N - q_j) u_{vNj}^* \right] \quad (4-86)$$

We now substitute the formulas derived above for the influence coefficients given in Eq. (4-42). The final equation is:

$$A_{N+1,N+1} = \frac{1}{2p} \sum_{j=1}^N \left[\sin(q_1 - q_j) \ln \frac{\hat{r}_{1,j+1}}{\hat{r}_{1,j}} + \sin(q_N - q_j) \ln \frac{\hat{r}_{N,j+1}}{\hat{r}_{N,j}} \right] + \sum_{j=1}^N \left[\cos(q_1 - q_j) b_{1,j} + \cos(q_N - q_j) b_{N,j} \right] \quad (4-86)$$

After substituting in the values of the velocities in terms of the singularity strengths, and performing some algebraic manipulation, a form of the coefficients suitable for computations is obtained.

The final equations associated with the Kutta condition are:

$$A_{N+1,j} = \frac{1}{2p} \left[\sin(q_1 - q_j) b_{1,j} + \sin(q_N - q_j) b_{N,j} \right] - \sum_{i=1}^N \left[\cos(q_1 - q_j) \ln \frac{\hat{r}_{1,j+1}}{\hat{r}_{1,i}} - \cos(q_N - q_j) \ln \frac{\hat{r}_{N,j+1}}{\hat{r}_{N,i}} \right] \quad (4-81)$$

$$A_{N+1,N+1} = \frac{1}{2p} \sum_{j=1}^N \left[\sin(q_1 - q_j) \ln \frac{\hat{r}_{1,j+1}}{\hat{r}_{1,j}} + \sin(q_N - q_j) \ln \frac{\hat{r}_{N,j+1}}{\hat{r}_{N,j}} \right] + \sum_{j=1}^N \left[\cos(q_1 - q_j) b_{1,j} + \cos(q_N - q_j) b_{N,j} \right] \quad (4-86)$$

$$b_{N+1} = -V_\infty \cos(q_1 - \alpha) - V_\infty \cos(q_N - \alpha). \quad (4-72)$$

Step 5. Solve the system for q_i , g .

The coefficients derived above provide the required coefficients to solve a system of linear algebraic equations for the $N+1$ unknowns, q_i , $i = 1, \dots, N$ and g given by (4-43) and (4-69):

$$\begin{aligned} \hat{\mathbf{A}}_{ij} q_j + A_{i,N+1} g &= b_i \quad i = 1, \dots, N \\ \hat{\mathbf{A}}_{N+1,j} q_j + A_{N+1,N+1} g &= b_{N+1} \end{aligned} \quad (4-87)$$

This is easily done using any number of computer subroutines.

Step 6. Given q_i and g , write down the equations for the tangential velocity at each panel control point.

At each control point, ($v_n = 0$), find u_t , the tangential velocity starting with:

$$\begin{aligned} u_{t_i} &= u_i \cos \alpha_i + v_i \sin \alpha_i \\ &= \hat{\mathbf{A}}_{iE} \cdot \cos \alpha + \sum_{j=1}^N \hat{\mathbf{A}}_{s_{ij}} q_j + g \sum_{j=1}^N \hat{\mathbf{A}}_{v_{ij}} \tilde{\cos} \alpha_i \\ &\quad + \hat{\mathbf{A}}_{iE} \cdot \sin \alpha + \sum_{j=1}^N \hat{\mathbf{A}}_{s_{ij}} q_j + g \sum_{j=1}^N \hat{\mathbf{A}}_{v_{ij}} \tilde{\sin} \alpha_i \end{aligned} \quad (4-88)$$

Using the (*) values of the influence coefficients,

$$\begin{aligned} u_{t_i} &= \hat{\mathbf{A}}_{iE} \cdot \cos \alpha + \sum_{j=1}^N \left(u_{s_{ij}}^* \cos \alpha_j - v_{s_{ij}}^* \sin \alpha_j \right) q_j + g \sum_{j=1}^N \left(u_{v_{ij}}^* \cos \alpha_j - v_{v_{ij}}^* \sin \alpha_j \right) \tilde{\cos} \alpha_i \\ &\quad + \hat{\mathbf{A}}_{iE} \cdot \sin \alpha + \sum_{j=1}^N \left(u_{s_{ij}}^* \sin \alpha_j + v_{s_{ij}}^* \cos \alpha_j \right) q_j + g \sum_{j=1}^N \left(u_{v_{ij}}^* \sin \alpha_j + v_{v_{ij}}^* \cos \alpha_j \right) \tilde{\sin} \alpha_i \end{aligned} \quad (4-89)$$

or:

$$\begin{aligned}
u_{t_i} &= V \cdot \cos a \cos q_i + V \cdot \sin a \sin q_i \\
&+ \hat{\mathbf{A}} \sum_{j=1}^N \left\{ u_{s_{ij}}^* \cos q_j \cos q_i - v_{s_{ij}}^* \sin q_j \cos q_i + u_{s_{ij}}^* \sin q_j \sin q_i + v_{s_{ij}}^* \cos q_j \sin q_i \right\} q_j \cdot \\
&+ g \hat{\mathbf{A}} \sum_{j=1}^N \left\{ u_{v_{ij}}^* \cos q_j \cos q_i - v_{v_{ij}}^* \sin q_j \cos q_i + u_{v_{ij}}^* \sin q_j \sin q_i + v_{v_{ij}}^* \cos q_j \sin q_i \right\}
\end{aligned} \tag{4-90}$$

Collecting terms:

$$\begin{aligned}
u_{t_i} &= \underbrace{(\cos a \cos q_i + \sin a \sin q_i)}_{\cos(a - q_i)} V \cdot \\
&+ \hat{\mathbf{A}} \sum_{j=1}^N \underbrace{(\cos q_j \cos q_i + \sin q_j \sin q_i)}_{\cos(q_j - q_i)} u_{s_{ij}}^* + \underbrace{(\cos q_j \sin q_i - \sin q_j \cos q_i)}_{-\sin(q_j - q_i)} v_{s_{ij}}^* q_j \\
&+ g \hat{\mathbf{A}} \sum_{j=1}^N \underbrace{(\cos q_j \cos q_i + \sin q_j \sin q_i)}_{\cos(q_j - q_i)} u_{v_{ij}}^* + \underbrace{(\cos q_j \sin q_i - \sin q_j \cos q_i)}_{-\sin(q_j - q_i)} v_{v_{ij}}^*
\end{aligned} \tag{4-91}$$

which becomes:

$$\begin{aligned}
u_{t_i} &= \cos(a - q_i) V \cdot + \hat{\mathbf{A}} \sum_{j=1}^N \left\{ \cos(q_j - q_i) u_{s_{ij}}^* - \sin(q_j - q_i) v_{s_{ij}}^* \right\} q_j \\
&+ g \hat{\mathbf{A}} \sum_{j=1}^N \left\{ \cos(q_j - q_i) u_{v_{ij}}^* - \sin(q_j - q_i) v_{v_{ij}}^* \right\}
\end{aligned} \tag{4-92}$$

Using the definitions of the ()* influence coefficients, and some trigonometric identities, we obtain the final result:

$$\begin{aligned}
u_{t_i} &= \cos(q_i - a) V \cdot + \hat{\mathbf{A}} \sum_{j=1}^N \frac{q_j}{2p} \left[\sin(q_i - q_j) b_{ij} - \cos(q_i - q_j) \ln \frac{\hat{r}_{i,j+1}}{\hat{r}_{i,j}} \right] \\
&+ \frac{g}{2p} \hat{\mathbf{A}} \sum_{j=1}^N \left[\sin(q_i - q_j) \ln \frac{\hat{r}_{i,j+1}}{\hat{r}_{i,j}} + \cos(q_i - q_j) b_{ij} \right]
\end{aligned} \tag{4-93}$$

Step 7. Finally, the surface pressure coefficient can be found from:

$$C_{P_i} = 1 - \frac{\hat{u}_{t_i}^2}{\hat{V}_\infty^2} \quad (4-94)$$

using u_i from Eq. (4-93).

This completes our derivation of one panel method scheme in two dimensions. Imagine the difficulty in performing the algebra required to extend this approach to three dimensions! That's why we've used a two-dimensional example.

4.5 Program PANEL

Program **PANEL** is an exact implementation of the analysis given in Section 4.4, and is essentially the program given by Moran.⁶ Other panel method programs are available in the textbooks by Houghton and Carpenter,¹⁰ and Kuethe and Chow.¹¹ Moran's program includes a subroutine to generate the ordinates for the NACA 4-digit and 5-digit airfoils (see Appendix A for a description of these airfoil sections). The main drawback is the requirement for a trailing edge thickness that's exactly zero. To accommodate this restriction, the ordinates generated internally have been altered slightly from the official ordinates. The extension of the program to handle arbitrary airfoils is an exercise. The freestream velocity in **PANEL** is assumed to be unity, since the inviscid solution in coefficient form is independent of scale.

PANEL's node points are distributed employing the widely used cosine spacing function. The equation for this spacing is given by defining the points on the thickness distribution to be placed at:

$$\frac{x_i}{c} = \frac{1}{2} \left[1 - \cos \left(\frac{(i-1)\pi}{N-1} \right) \right] \quad i = 1, \dots, N. \quad (4-95)$$

These locations are then altered when camber is added (see Eqns. (A-1) and (A-2) in App. A). This approach is used to provide a smoothly varying distribution of panel node points which concentrate points around the leading and trailing edges.

An example of the accuracy of program **PANEL** is given in Fig. 4-11, where the results from **PANEL** for the NACA 4412 airfoil are compared with results obtained from an exact conformal mapping of the airfoil (comments on the mapping methods are given in Chapter 9 on Geometry and Grids. Conformal transformations can also be used to generate meshes of points for use in field methods). The agreement is nearly perfect.

Numerical studies need to be conducted to determine how many panels are required to obtain accurate results. Both forces and moments and pressure distributions should be examined.

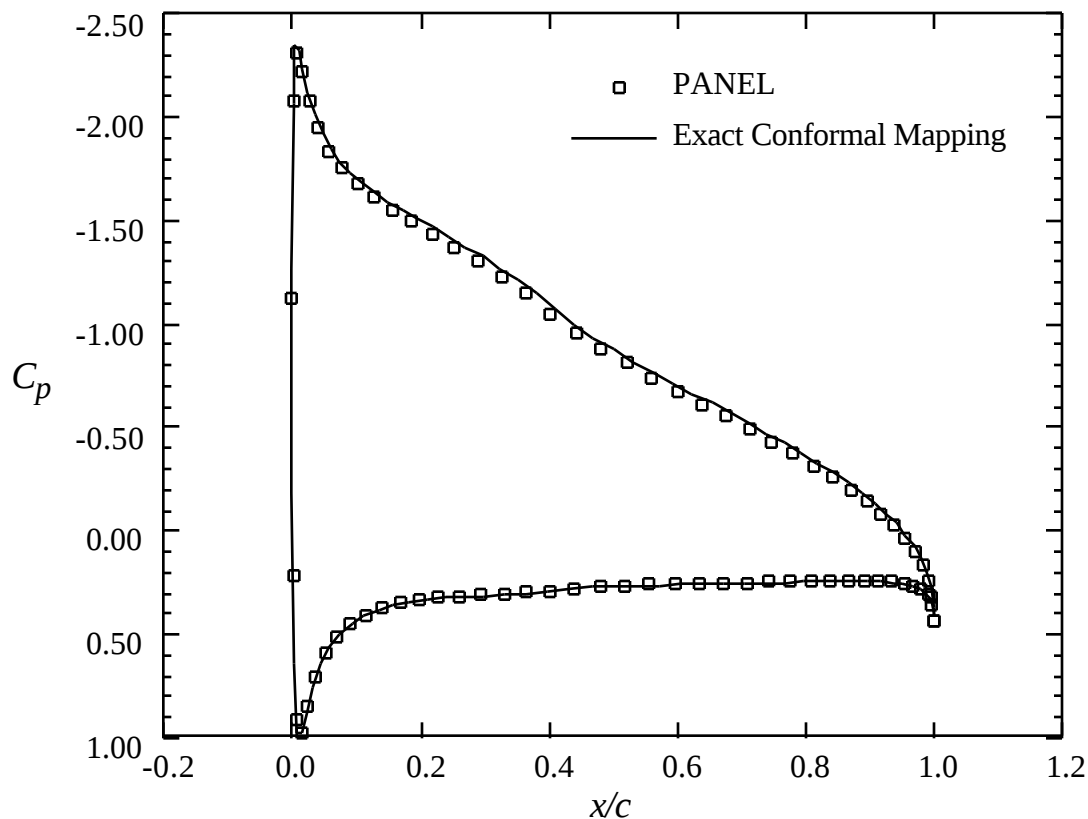


Figure 4-11. Comparison of results from program **PANEL** with an essentially exact mapping solution for the NACA 4412 airfoil at 6° angle-of-attack.

You can select the number of panels used to represent the surface. How many should you use? Most computational programs provide the user with freedom to decide how detailed (expensive - in dollars or time) the calculations should be. One of the first things the user should do is evaluate how detailed the calculation should be to obtain the level of accuracy desired. In the **PANEL** code your control is through the number of panels used.

We check the sensitivity of the solution to the number of panels by comparing force and moment results and pressure distributions with increasing numbers of panels. This is done using two different methods. Figures 4-12 and 4-13 present the change of drag and lift, respectively, using the first method. For **PANEL**, which uses an inviscid incompressible flowfield model, the drag should be exactly zero. The drag coefficient found by integrating the pressures over the airfoil is an indication of the error in the numerical scheme. The drag obtained using a surface (or “nearfield”) pressure integration is a numerically sensitive calculation, and is a strict test of the method. The figures show the drag going to zero, and the lift becoming constant as the number of

panels increase. In this style of presentation it is hard to see exactly how quickly the solution is converging to a fixed value.

The results given in Figures 4-12 and 4-13 indicate that 60-80 panels (30 upper, 30 lower for example) should be enough panels. Note that the lift is presented in an extremely expanded scale. Drag also uses an expanded scale. Because drag is typically a small number, it is frequently described in drag counts, where 1 drag count is a C_D of 0.0001.

To estimate the limit for an infinitely large number of panels the results can be plotted as a function of the reciprocal of the number of panels. Thus the limit result occurs as $1/n$ goes to zero. Figures 4-14, 4-15, and 4-16 present the results in this manner for the case given above, and with the pitching moment included for examination in the analysis.

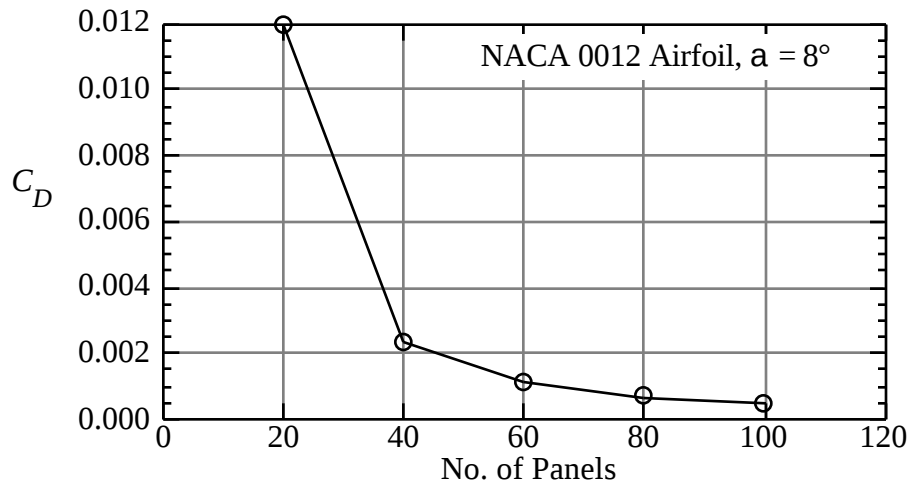


Figure 4-12. Change of drag with number of panels.

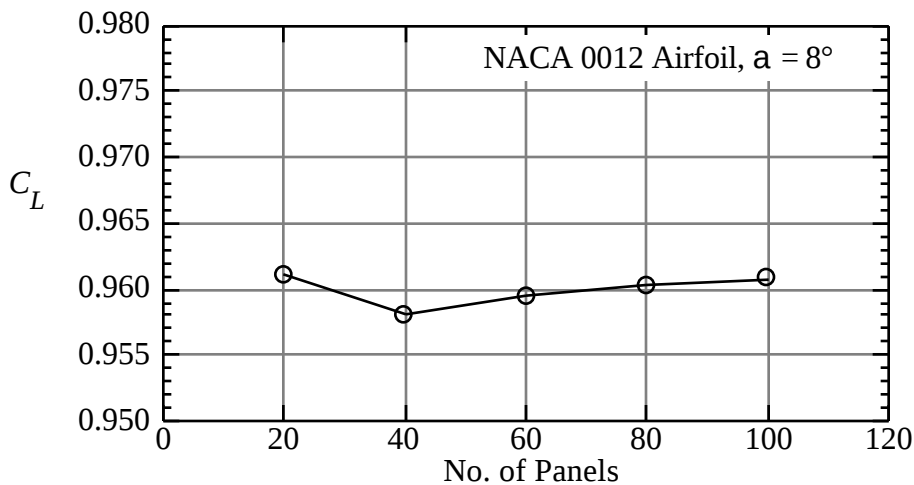


Figure 4-13. Change of lift with number of panels.

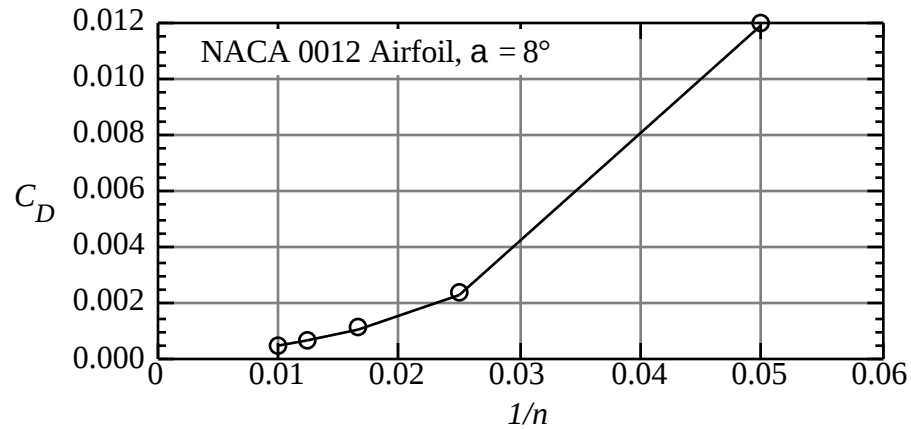


Figure 4-14. Change of drag with the inverse of the number of panels.

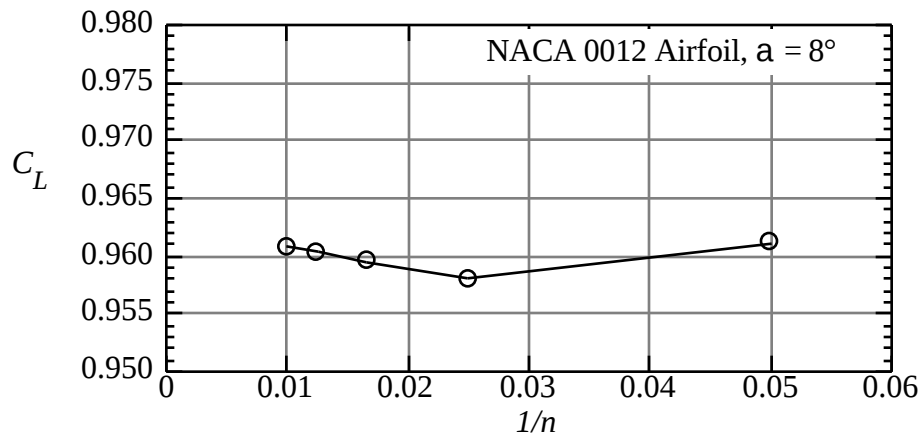


Figure 4-15. Change of lift with the inverse of the number of panels.

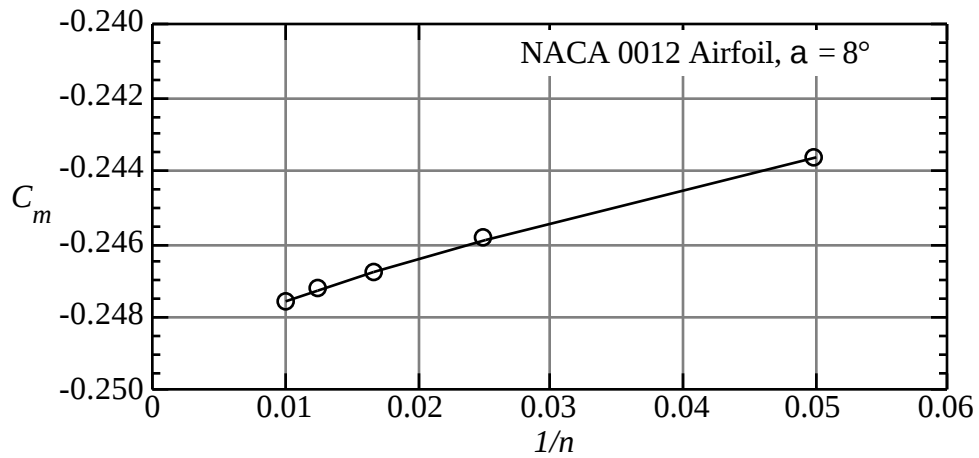


Figure 4-16. Change of pitching moment with the inverse of the number of panels.

The results given in Figures 4-14 through 4-16 show that the program **PANEL** produces results that are relatively insensitive to the number of panels once fifty or sixty panels are used, and by extrapolating to $1/n = 0$ an estimate of the limiting value can be obtained.

In addition to forces and moments, the sensitivity of the pressure distributions to changes in panel density should also be investigated. Pressure distributions are shown in Figures 4-17, 4-18, and 4-19. The case for 20 panels is given in Figure 4-17. Although the character of the pressure distribution is emerging, it's clear that more panels are required to define the details of the pressure distribution. The stagnation pressure region on the lower surface of the leading edge is not yet distinct. The expansion peak and trailing edge recovery pressure are also not resolved clearly. Figure 4-18 contains a comparison between 20 and 60 panel cases. In this case it appears that the pressure distribution is well defined with 60 panels. This is confirmed in Figure 4-19, which demonstrates that it is almost impossible to identify the differences between the 60 and 100 panel cases. This type of study should (and in fact *must*) be conducted when using computational aerodynamics methods.

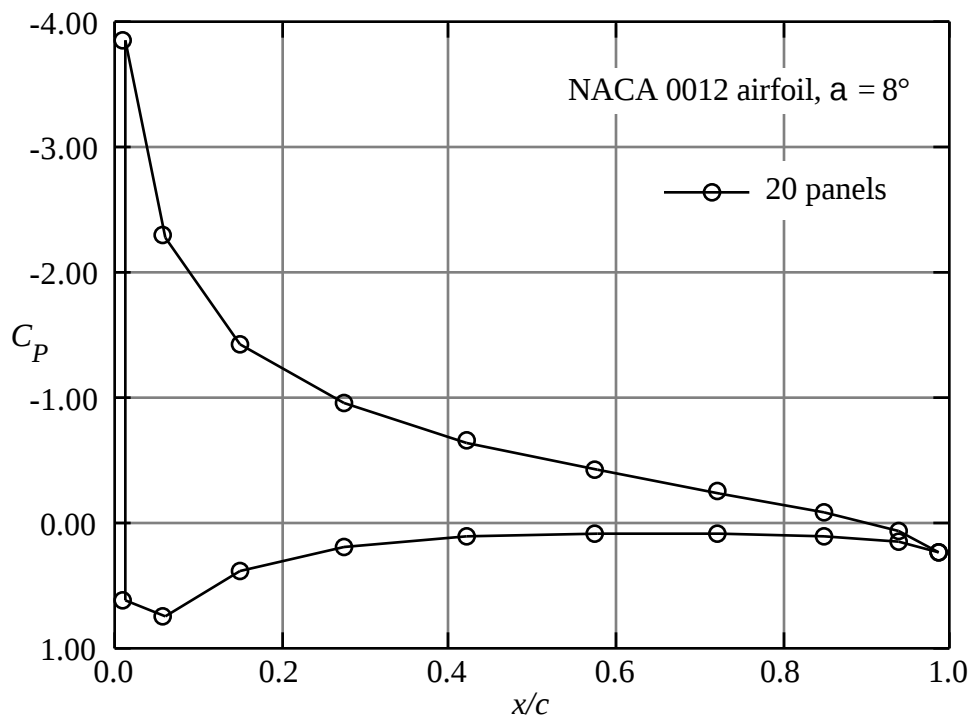


Figure 4-17. Pressure distribution from program PANEL, 20 panels.

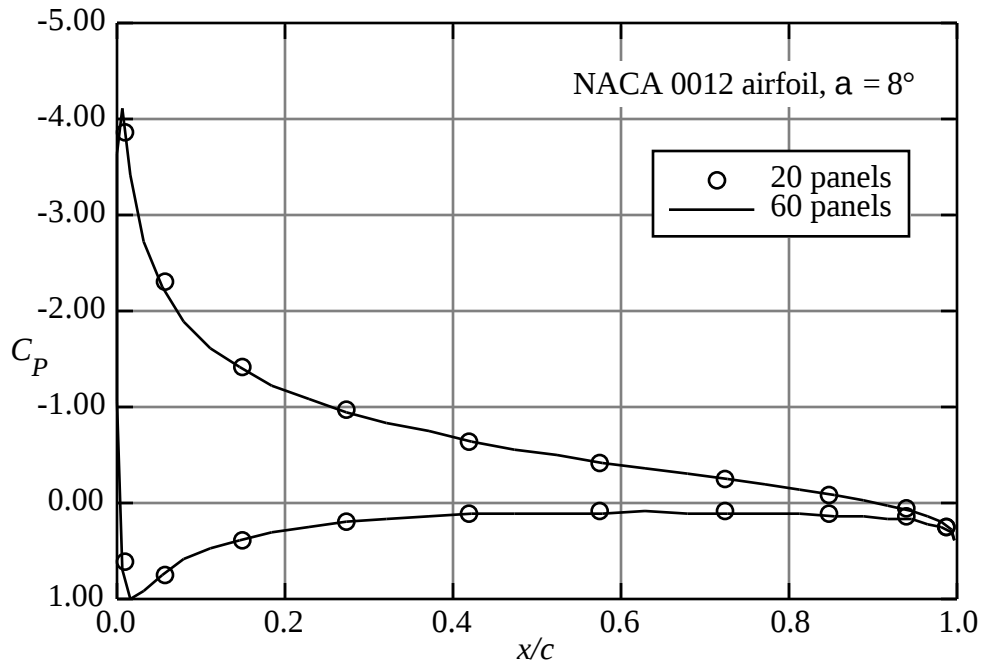


Figure 4-18. Pressure distribution from program PANEL, comparing results using 20 and 60 panels.

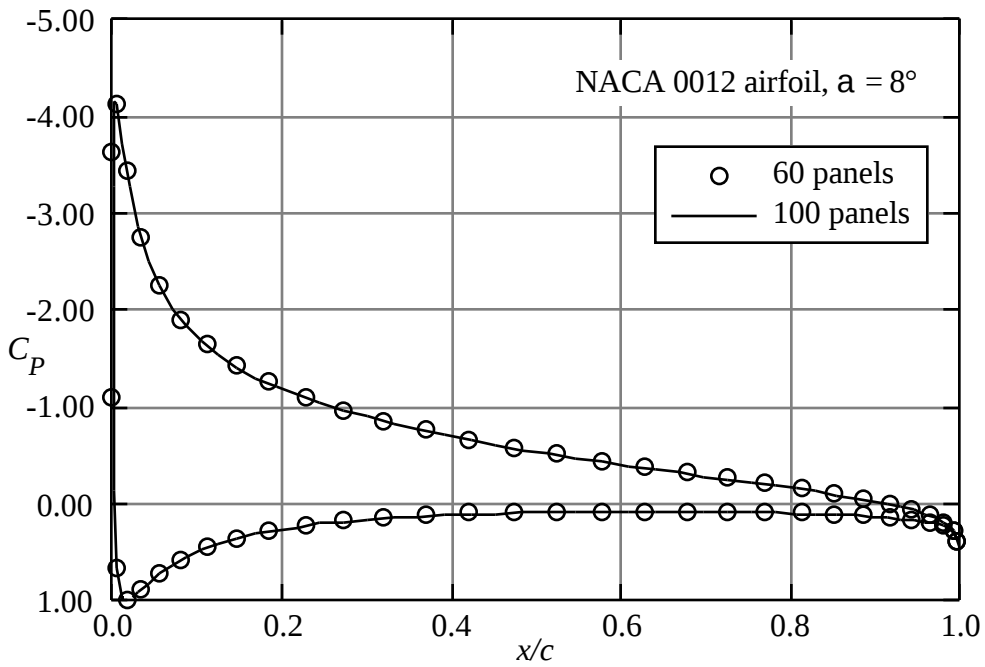


Figure 4-19. Pressure distribution from program PANEL, comparing results using 60 and 100 panels.

Having examined the convergence of the mathematical solution, we investigate the agreement with experimental data. Figure 4-20 compares the lift coefficients from the inviscid solutions obtained from **PANEL** with experimental data from Abbott and von Doenhof.¹² Agreement is good at low angles of attack, where the flow is fully attached. The agreement deteriorates as the angle of attack increases, and viscous effects start to show up as a reduction in lift with increasing angle of attack, until, finally, the airfoil stalls. The inviscid solutions from **PANEL** cannot capture this part of the physics. The difference in the airfoil behavior at stall between the cambered and uncambered airfoil will be discussed further in Chapter 10. Essentially, the differences arise due to different flow separation locations on the different airfoils. The cambered airfoil separates at the trailing edge first. Stall occurs gradually as the separation point moves forward on the airfoil with increasing incidence. The uncambered airfoil stalls due to a sudden separation at the leading edge. An examination of the difference in pressure distributions to be discussed next can be studied to see why this might be the case.

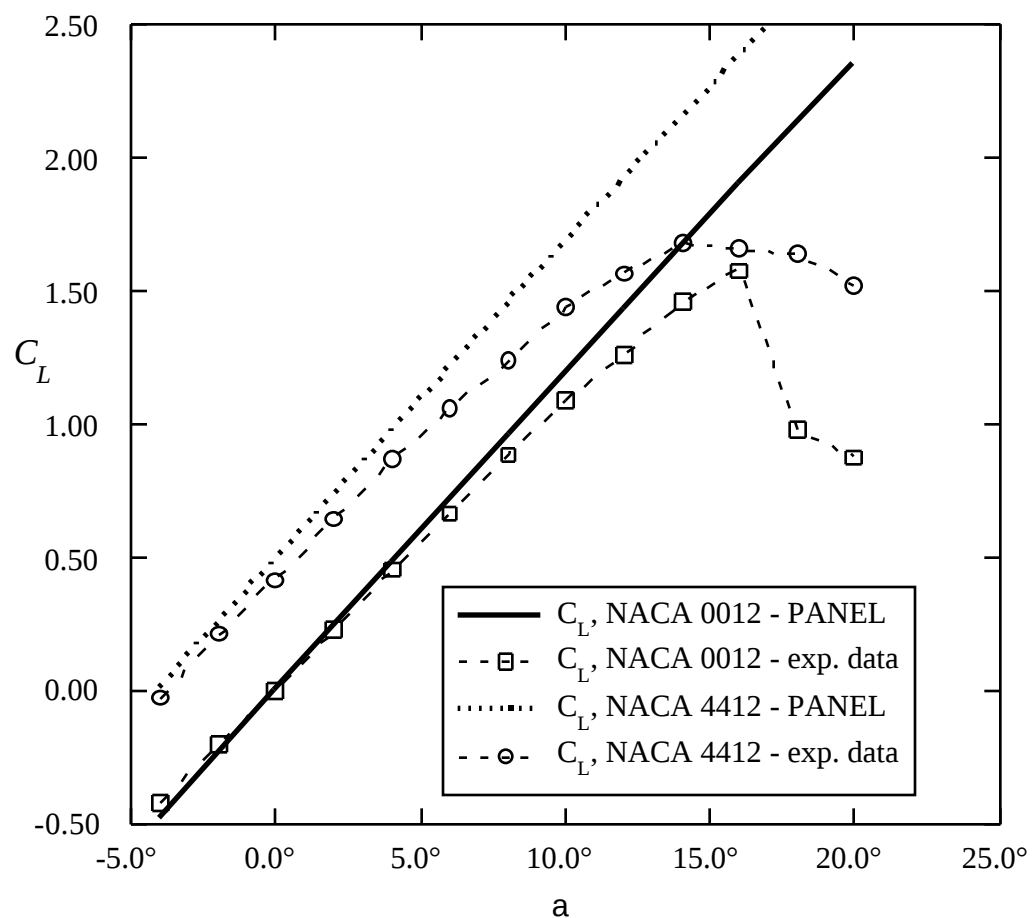


Figure 4-20. Comparison of PANEL lift predictions with experimental data, (Ref. 12).

The pitching moment characteristics are also important. Figure 4-21 provides a comparison of the **PANEL** pitching moment predictions (about the quarter chord point) with experimental data. In this case the calculations indicate that the computed location of the aerodynamic center, $dC_m / dC_L = 0$, is not exactly at the quarter chord, although the experimental data is very close to this value. The uncambered NACA 0012 data shows nearly zero pitching moment until flow separation starts to occur. The cambered airfoil shows a significant pitching moment, and a trend due to viscous effects that is exactly opposite the computed prediction.

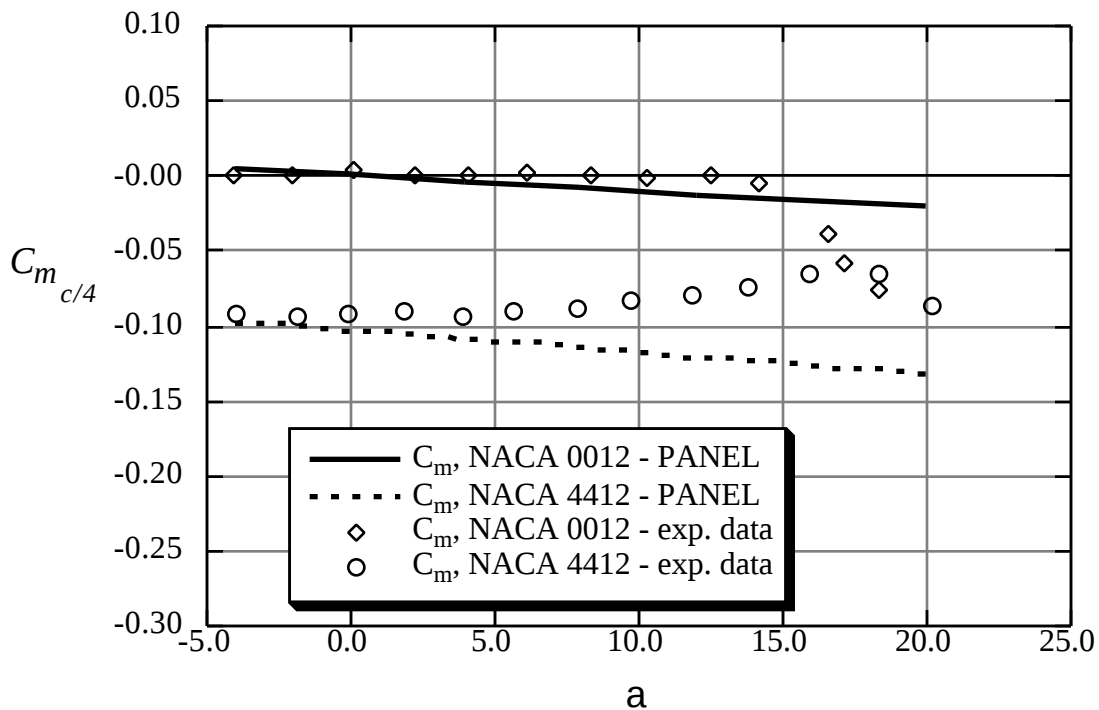


Figure 4-21. Comparison of PANEL moment predictions with experimental data, (Ref. 12).

We do not compare the drag prediction from **PANEL** with experimental data. In two-dimensional incompressible inviscid flow the drag is zero. In the actual case, drag arises from skin friction effects, further additional form drag due to the small change of pressure on the body due to the boundary layer (which primarily prevents full pressure recovery at the trailing edge), and drag due to increasing viscous effects with increasing angle of attack. A well designed airfoil will have a drag value very nearly equal to the skin friction and nearly invariant with incidence until the maximum lift coefficient is approached.

In addition to the force and moment comparisons, we need to compare the pressure distributions predicted with **PANEL** to experimental data. Figure 4-22 provides one example. The NACA 4412 experimental pressure distribution is compared with **PANEL** predictions. In general

the agreement is very good. The primary area of disagreement is at the trailing edge. Here viscous effects act to prevent the recovery of the experimental pressure to the levels predicted by the inviscid solution. The disagreement on the lower surface is surprising, and suggests that the angle of attack from the experiment is not precise.

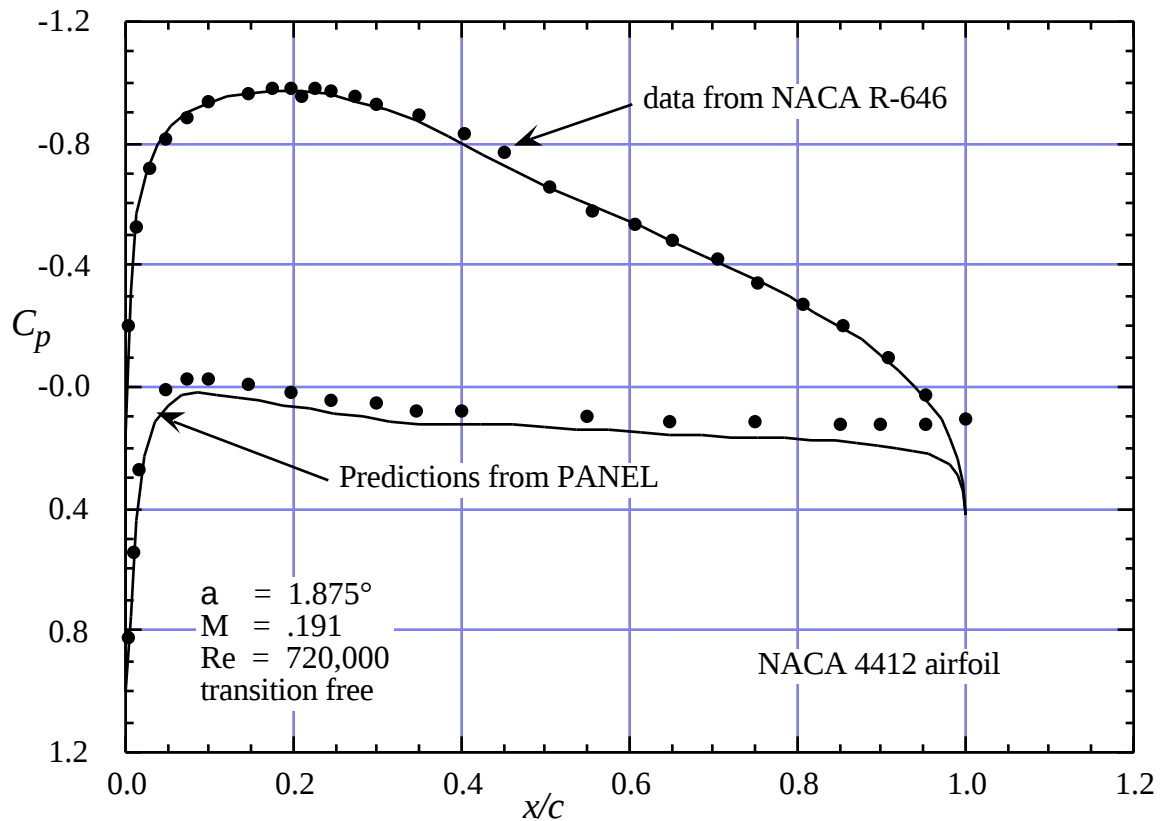


Figure 4-22. Comparison of pressure distribution from PANEL with data.

Panel methods often have trouble with accuracy at the trailing edge of airfoils with cusped trailing edges, so that the included angle at the trailing edge is zero. Figure 4-23 shows the predictions of program **PANEL** compared with an exact mapping solution (**FLO36** run at low Mach number, see Chap. 11) for two cases. Figure 4-23a is for a case with a small trailing edge angle: the NACA 651-012, while Fig. 4-23b is for the more standard 6A version of the airfoil. The corresponding airfoil shapes are shown Fig. 4-24.

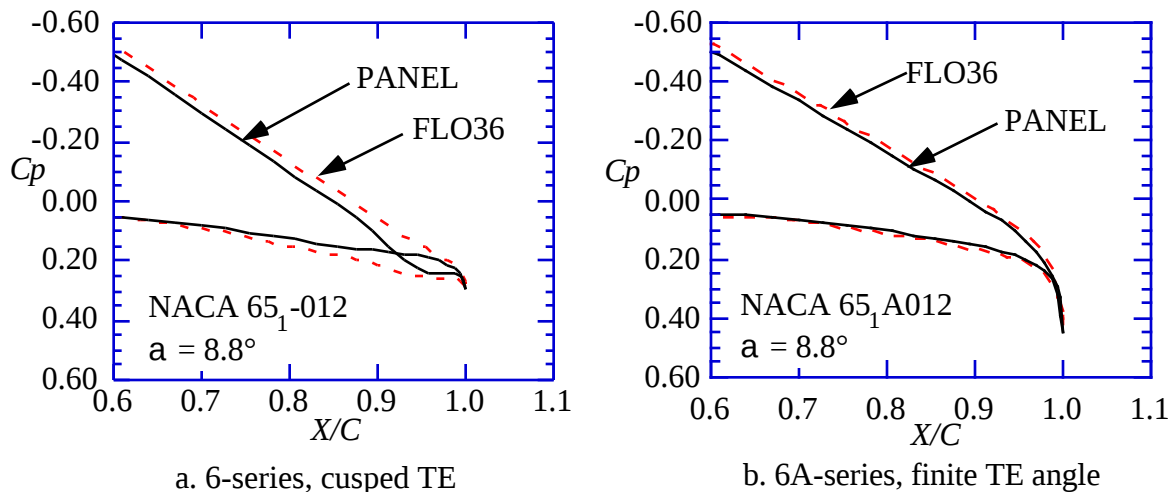
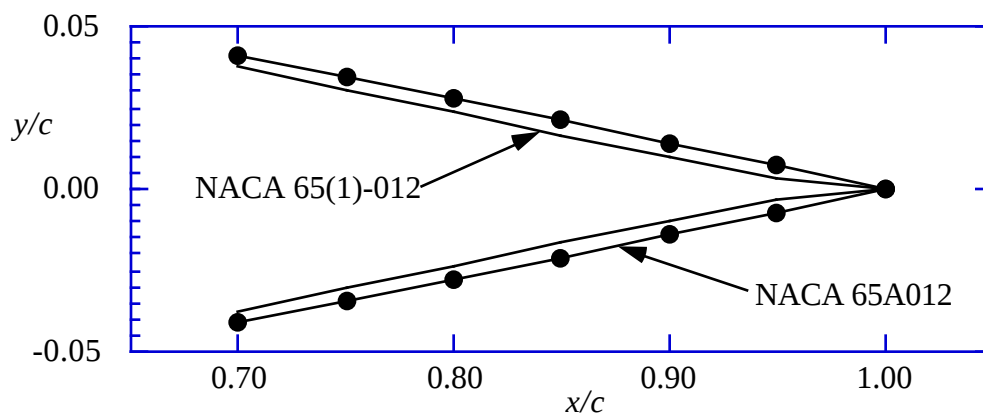
Figure 23. **PANEL** Performance near the airfoil trailing edge

Figure 4-24. Comparison at the trailing edge of 6- and 6A-series airfoil geometries.

This case demonstrates a situation where this particular panel method is not accurate. Is this a practical consideration? Yes and no. The 6-series airfoils were theoretically derived by specifying a pressure distribution and determining the required shape. The small trailing edge angles (less than half those of the 4-digit series), cusped shape, and the unobtainable zero thickness specified at the trailing edge resulted in objections from the aircraft industry. These airfoils were very difficult to use on operational aircraft. Subsequently, the 6A-series airfoils were introduced to remedy the problem. These airfoils had larger trailing edge angles (approximately the same as the 4-digit series), and were made up of nearly straight (or flat) surfaces over the last 20% of the airfoil. Most applications of 6-series airfoils today actually use the modified 6A-series thickness distribution. This is an area where the user should check the performance of a particular panel method.

4.6 Subsonic Airfoil Aerodynamics

Using **PANEL** we now have a means of easily examining the pressure distributions, and forces and moments, for different airfoil shapes. In this section we present a discussion of airfoil characteristics using an inviscid analysis. All the illustrative examples were computed using program **PANEL**. We illustrate key areas to examine when studying airfoil pressure distributions using the NACA 0012 airfoil at 4° angle of attack as typical in Fig. 4-25.

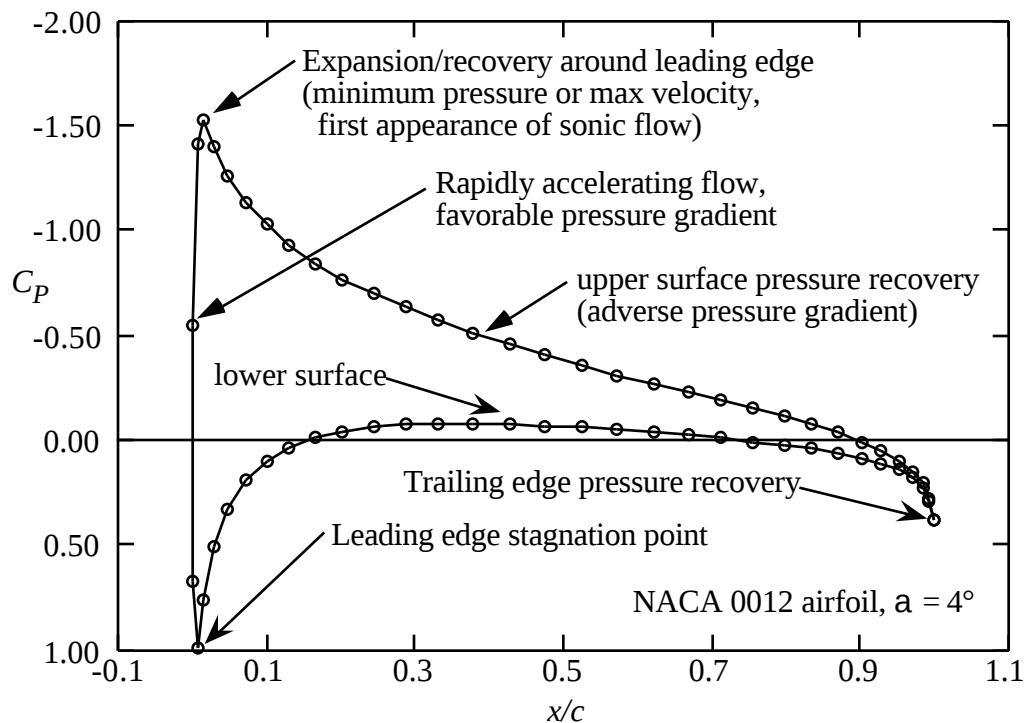


Figure 4-25. Key areas of interest when examining airfoil pressure distributions.

Remember that we are making an incompressible, inviscid analysis when we are using program **PANEL**. Thus, in this section we examine the basic characteristics of airfoils from that point of view. We will examine viscous and compressibility effects in subsequent chapters, when we have the tools to conduct numerical experiments. However, the best way to understand airfoil characteristics from an engineering standpoint is to examine the inviscid properties, and then consider changes in properties due to the effects of viscosity. Controlling the pressure distribution through selection of the geometry, the aerodynamicist controls, or suppresses, adverse viscous effects. The mental concept of the flow best starts as a flowfield driven by the pressure distribution that would exist if there were no viscous effects. The airfoil characteristics then change by the

“relieving” effects of viscosity, where flow separation or boundary layer thickening reduces the degree of pressure recovery which would occur otherwise. For efficient airfoils the viscous effects should be small at normal operating conditions.

4.6.1 Overview of Airfoil Characteristics: Good and Bad

In this section we illustrate the connection between the airfoil geometry and the airfoil pressure distribution. We identify and discuss ways to control the inviscid pressure distribution by changing the airfoil geometry. An aerodynamicist controls viscous effects by controlling the pressure distribution. Further discussion and examples providing insight into aerodynamic design are available in the excellent recent book by Jones.¹³ A terrific book that captures much of the experience of the original designers of the NACA airfoils was written by aeronautical pioneer E.P. Warner.¹⁴

Drag: We discussed the requirement that drag should be zero* for this *two-dimensional* inviscid incompressible irrotational prediction method when we studied the accuracy of the method in the previous section. At this point we infer possible drag and adverse viscous effects by examining the effects of airfoil geometry and angle of attack on the pressure distribution.

Lift: Thin airfoil theory predicts that the lift curve slope should be 2π , and thick airfoil theory says that it should be slightly greater than 2π , with 2π being the limit for zero thickness. You can easily determine how close program **PANEL** comes to this value. These tests should give you confidence that the code is operating correctly. The other key parameter is α_{ZL} , the angle at which the airfoil produces zero lift (a related value is C_{L0} , the value of C_L at $\alpha = 0$).

Moment: Thin airfoil theory predicts that subsonic airfoils have their aerodynamic centers at the quarter chord for attached flow. The value of C_{m0} depends on the camber. We have seen in Fig. 4-21 that the computed aerodynamic center is not precisely located at the quarter chord. However, the slope of the moment curve in Fig. 4-21 corresponds to an aerodynamic center location of $x/c = 0.2597$, which is reasonably close to 0.2500.

Multi-element airfoils are also an important class of airfoils. However, their performance is so closely connect to the effects of viscosity that the discussion of those airfoils is deferred until Chapter 10, Viscous Flows in Aerodynamics.

* Three-dimensional panel methods can estimate the induced drag.

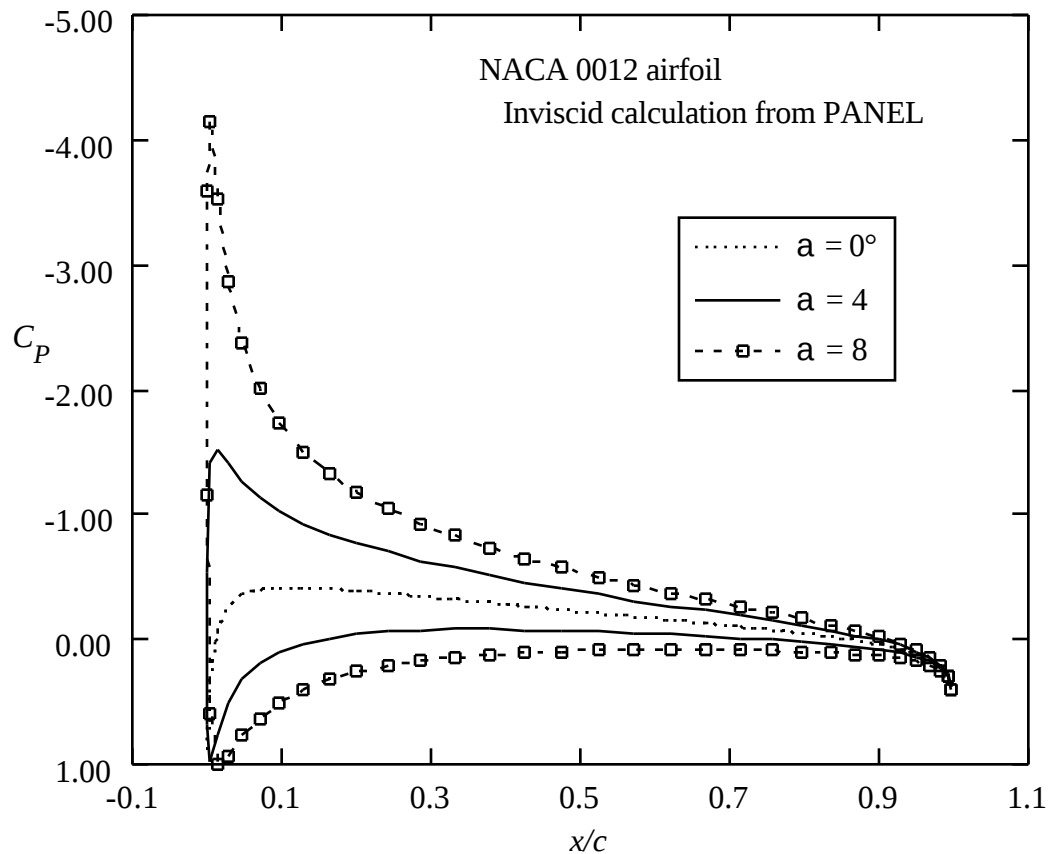


Figure 4-26. Effect of angle of attack on the pressure distribution.

The starting place for understanding airfoil characteristics is an examination of the angle of attack effects on an uncambered airfoil. Figure 4-26 presents this effect for the NACA 0012 airfoil. Here we see the progression from the symmetric zero angle of attack result. The $\alpha = 0^\circ$ case produces a mild expansion around the leading edge followed by a monotonic recovery to the trailing edge pressure. As the angle of attack increases the pressure begins to expand rapidly around the leading edge, reaching a very low pressure, and resulting in an increasingly steep pressure recovery at the leading edge.

The next effect of interest is thickness. Figure 4-27 presents airfoil shapes for NACA 4 digit sections of 6, 12, and 18 percent thick. The associated basic pressure distributions at zero angle of attack are shown in Fig. 4-28. Clearly the thicker airfoil produces a larger disturbance, and hence a lower minimum pressure. However, the 18 percent thick airfoil produces a milder expansion around the leading edge and a recompression extending further upstream than the thinner airfoils, especially at the trailing edge.

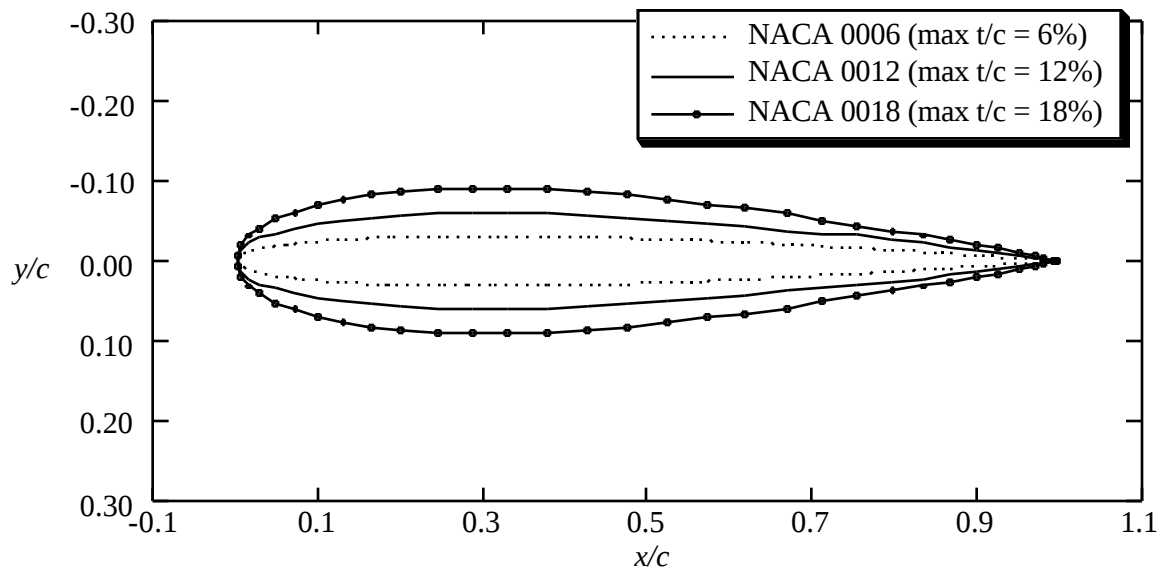


Figure 4-27. Comparison of NACA 4-digit airfoils of 6, 12, and 18% thicknesses.

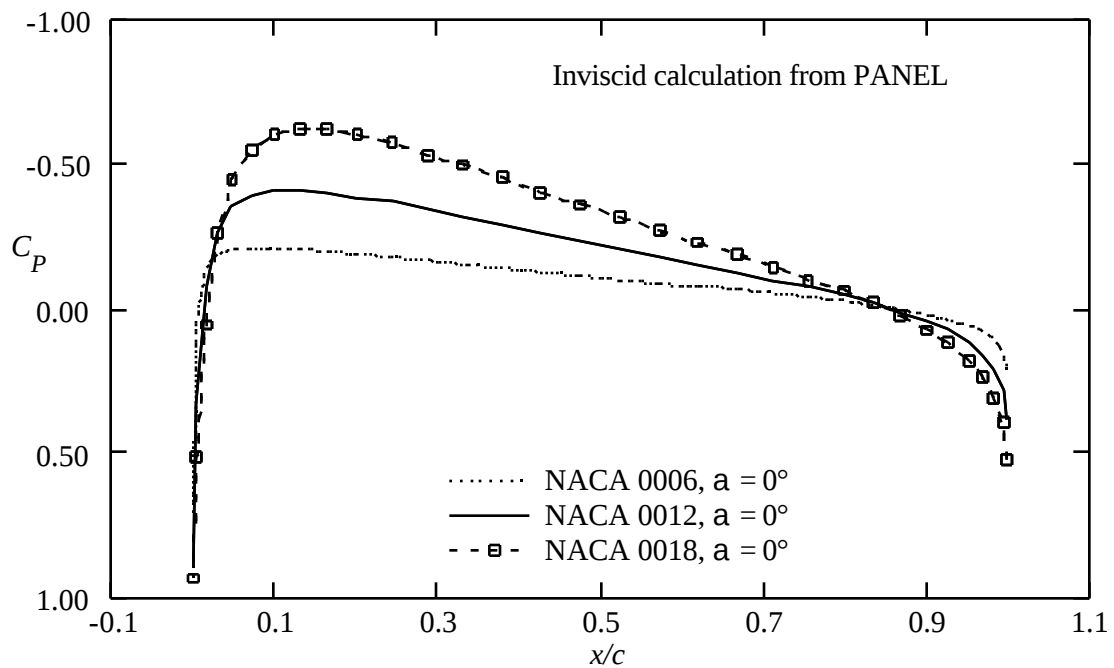


Figure 4-28. Effect of airfoil thickness on the pressure distribution at zero lift.

The effect of thickness in softening the expansion and recompression around the leading edge is even more evident at an angle of attack. Figure 4-29 shows this effect at a lift coefficient of .48. The thinnest airfoil shows a dramatic expansion/recompression due to the location of the stagnation point below the leading edge point, requiring a large expansion around the leading edge which has a very small radius of curvature. The thicker airfoil results in a significantly milder expansion and subsequent recompression.

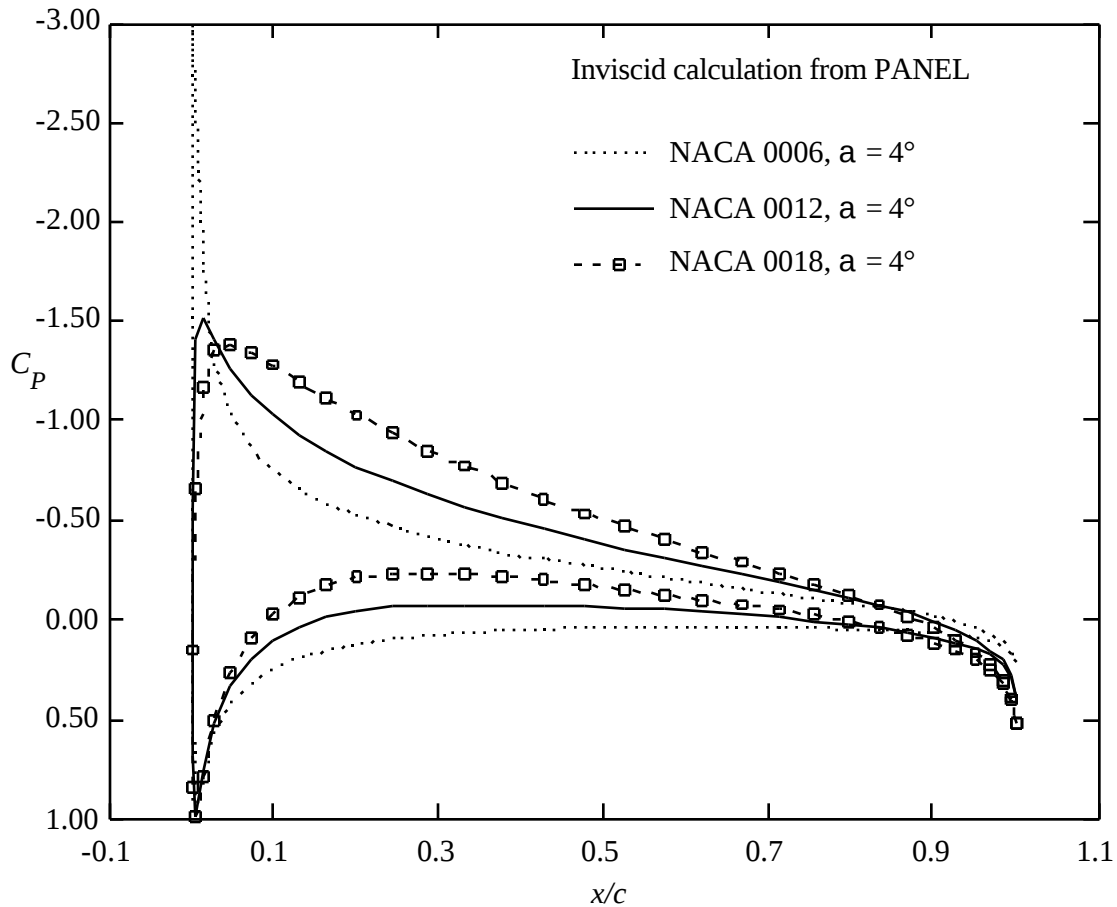


Figure 4-29. Effect of airfoil thickness on the pressure distribution at $C_L = 0.48$.

The next effect to examine is camber. Figure 4-30 compares the shapes of the NACA 0012 and 4412 airfoils. The pressure distributions on the cambered airfoil for two different angles of attack are shown in Figure 4-31. Note the role of camber in obtaining lift without producing a leading edge expansion followed by a rapid recompression immediately behind the expansion. This reduces the possibility of leading edge separation.

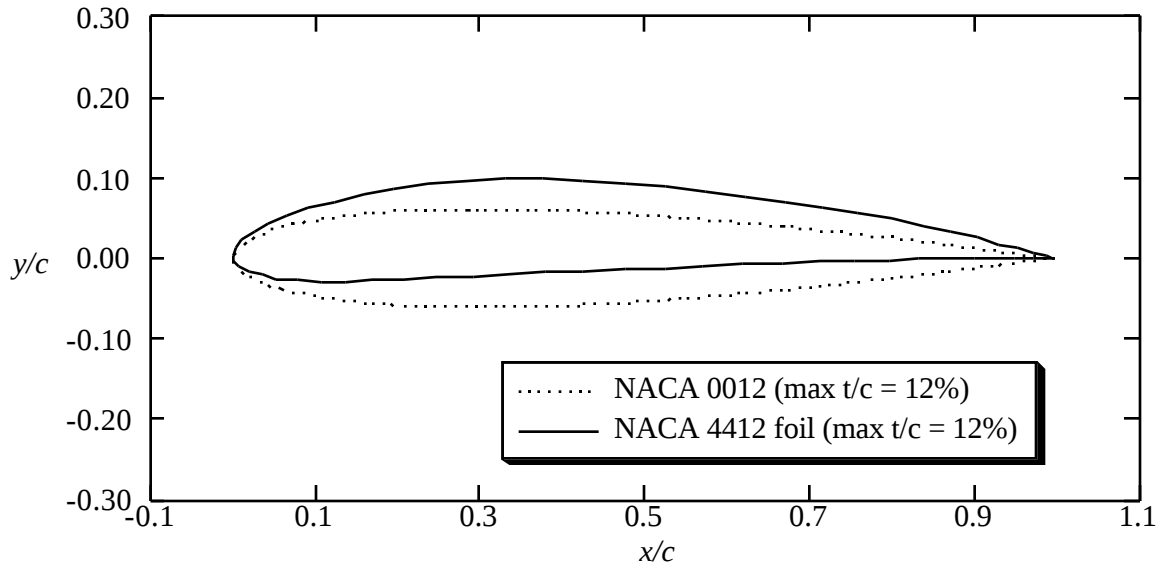


Figure 4-30. Comparison of uncambered and cambered NACA 4-digit airfoils.

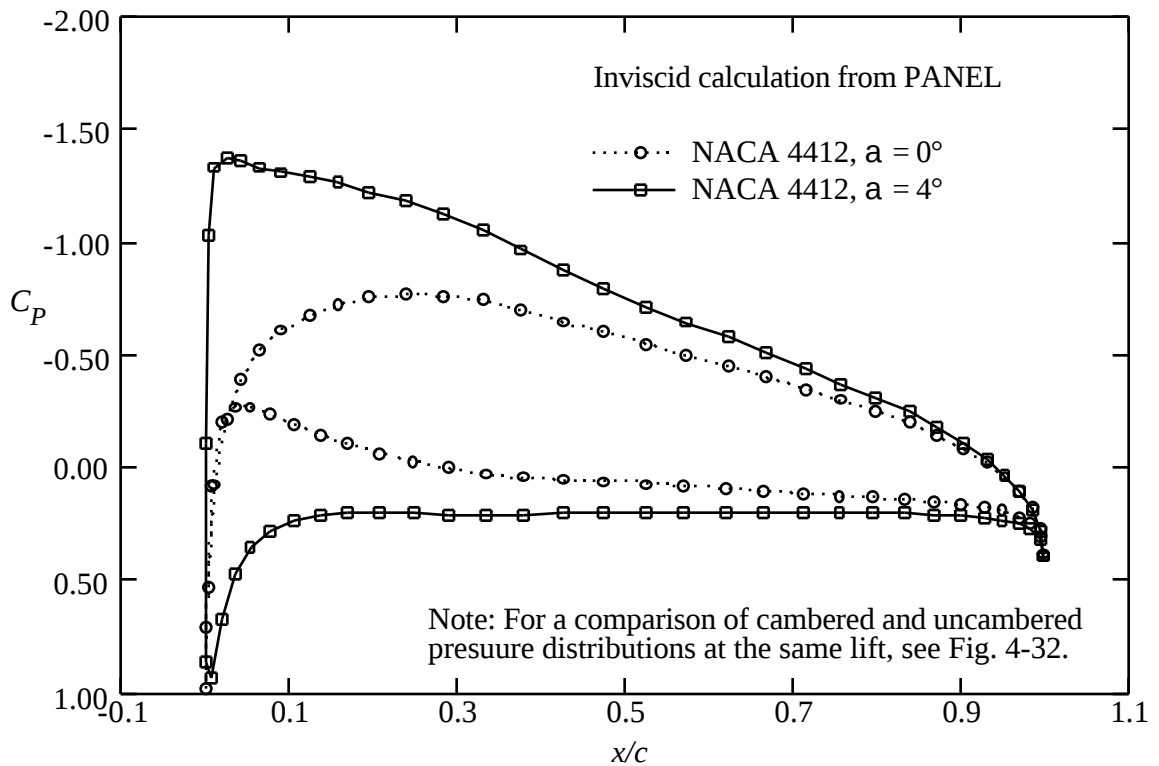


Figure 4-31. Effect of angle of attack on cambered airfoil pressure distributions at low lift.

A comparison of the NACA 0012 and NACA 4412 airfoil pressure distributions at the same lift coefficient is presented for several values of lift in Figures 4-32, 4-33 and 4-34. As the lift increases, the camber effects start to be dominated by the angle of attack effects, and the dramatic effects of camber are diminished until at a lift coefficient of 1.43 the pressure distributions start to look similar.

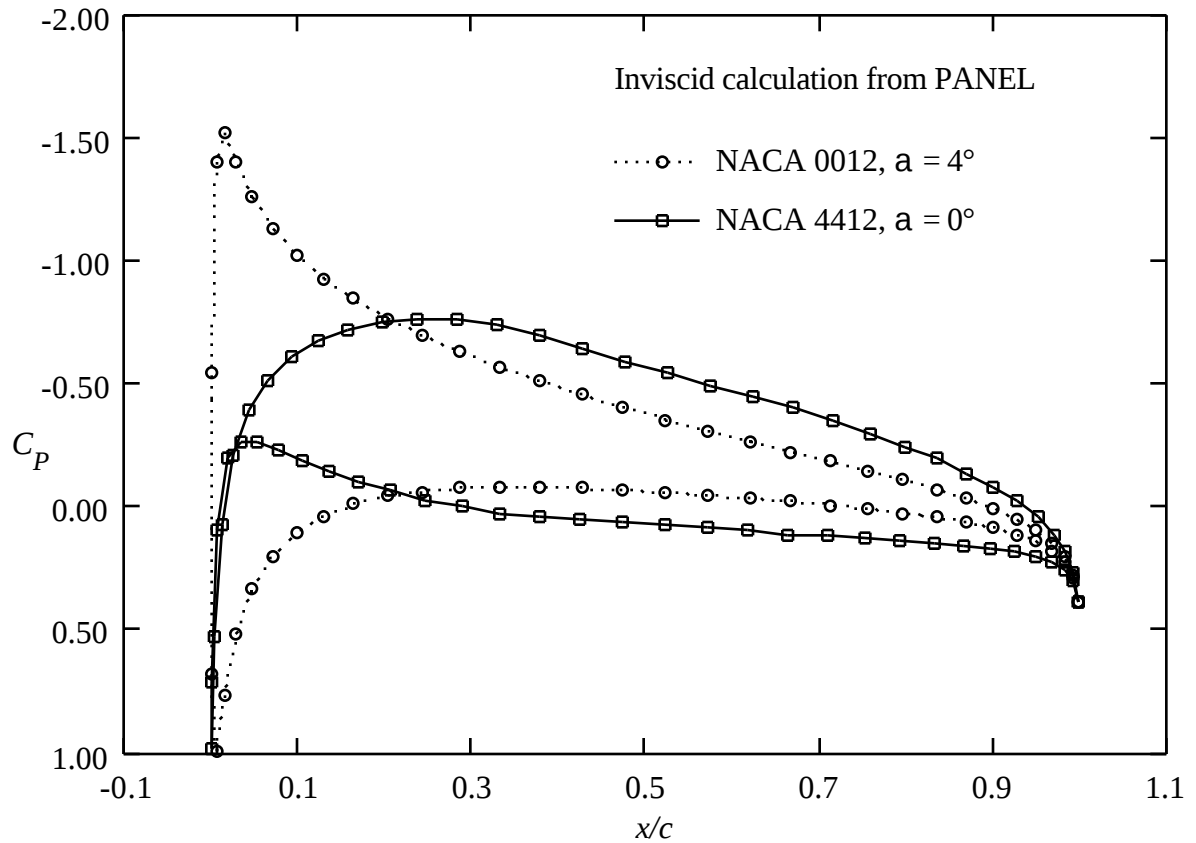


Figure 4-32. Camber effects on airfoil pressure distributions at $C_L = 0.48$.

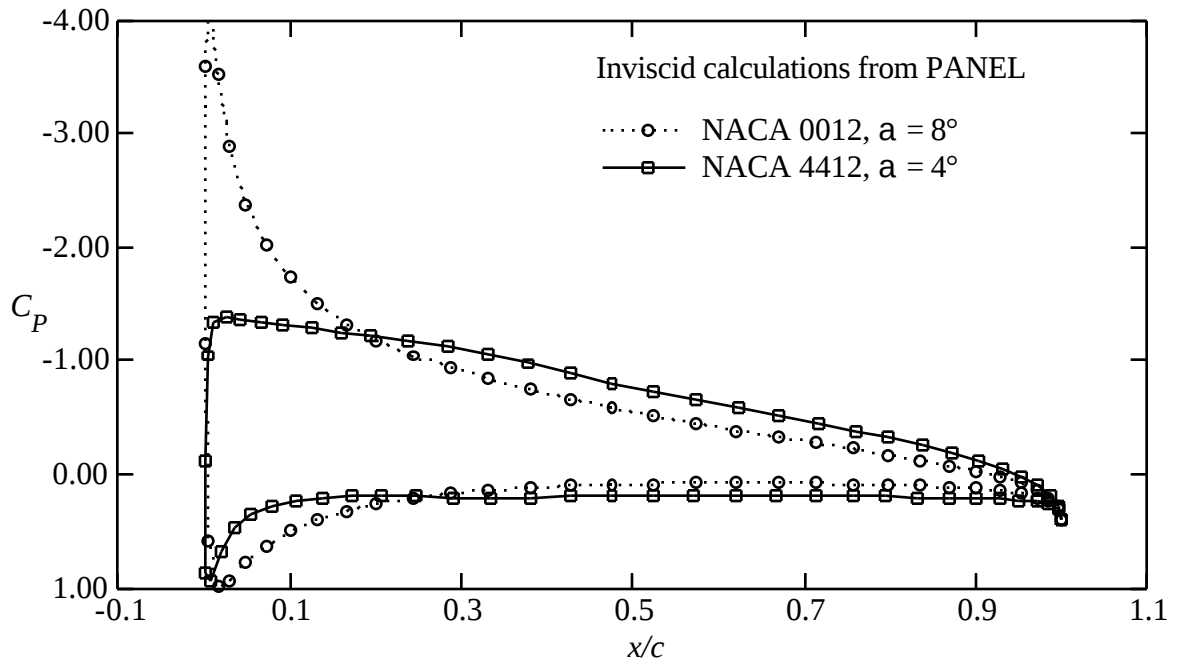


Figure 4-33. Camber effects airfoil pressure distributions at $C_L = 0.96$.

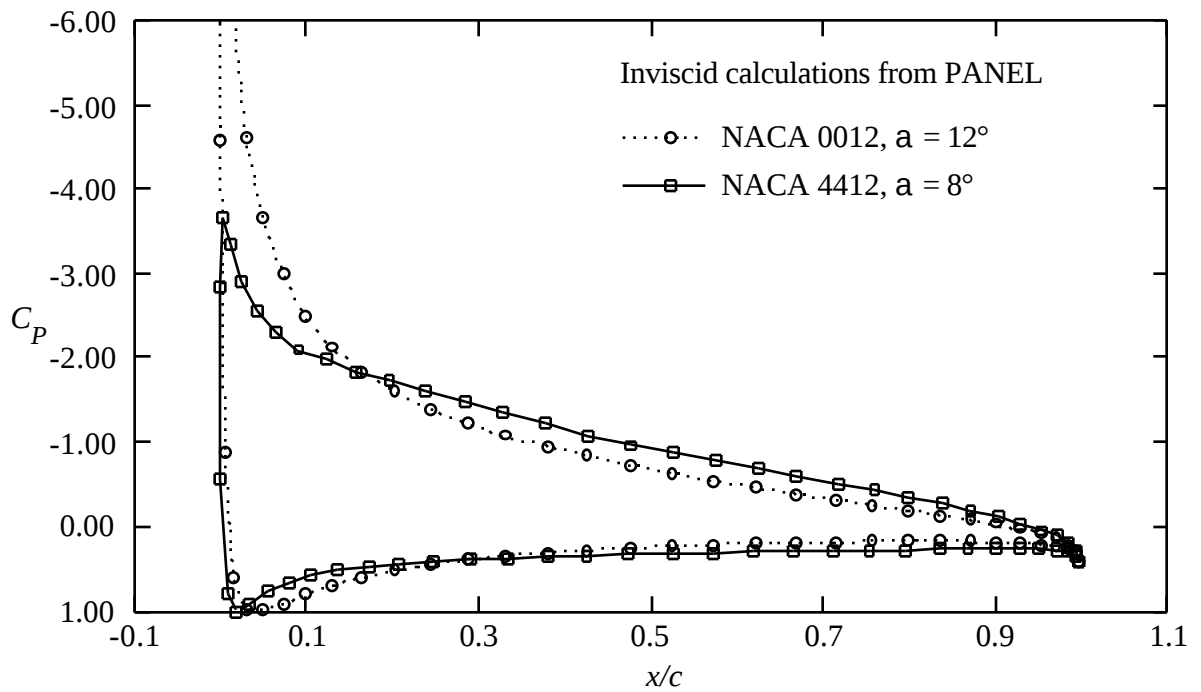


Figure 4-34. Camber effects airfoil pressure distributions at $C_L = 1.43$.

Finally, we examine the effect of extreme aft camber, which was part of the design strategy of Whitcomb when the so-called NASA supercritical airfoils were developed. This effect can be simulated using the NACA 6712 airfoil, as shown in Figure 4-35. The resulting pressure distribution is given in Figure 4-36. Note that the aft camber “opens up” the pressure distribution near the trailing edge. Two adverse properties of this type of pressure distribution are the large zero lift pitching moment and the delayed and then rapid pressure recovery on the upper surface. This type of pressure recovery is a very poor way to try to achieve a significant pressure recovery because the boundary layer will separate early. Whitcomb’s design work primarily improved the pressure recovery curve.

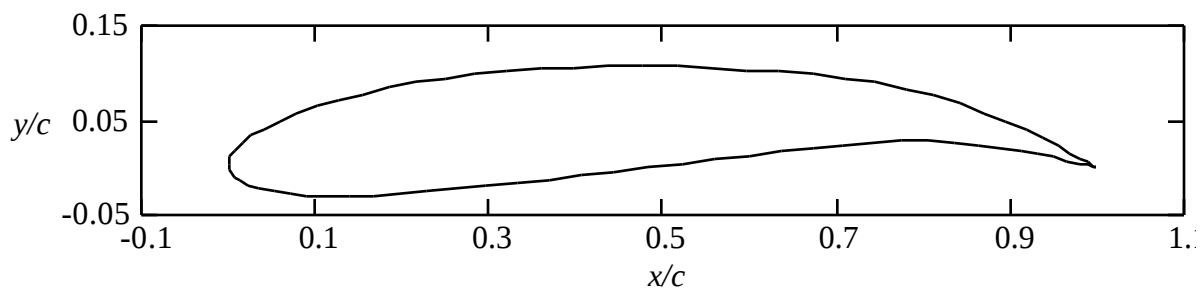


Figure 4-35. Highly aft cambered NACA airfoil, an NACA 6712.

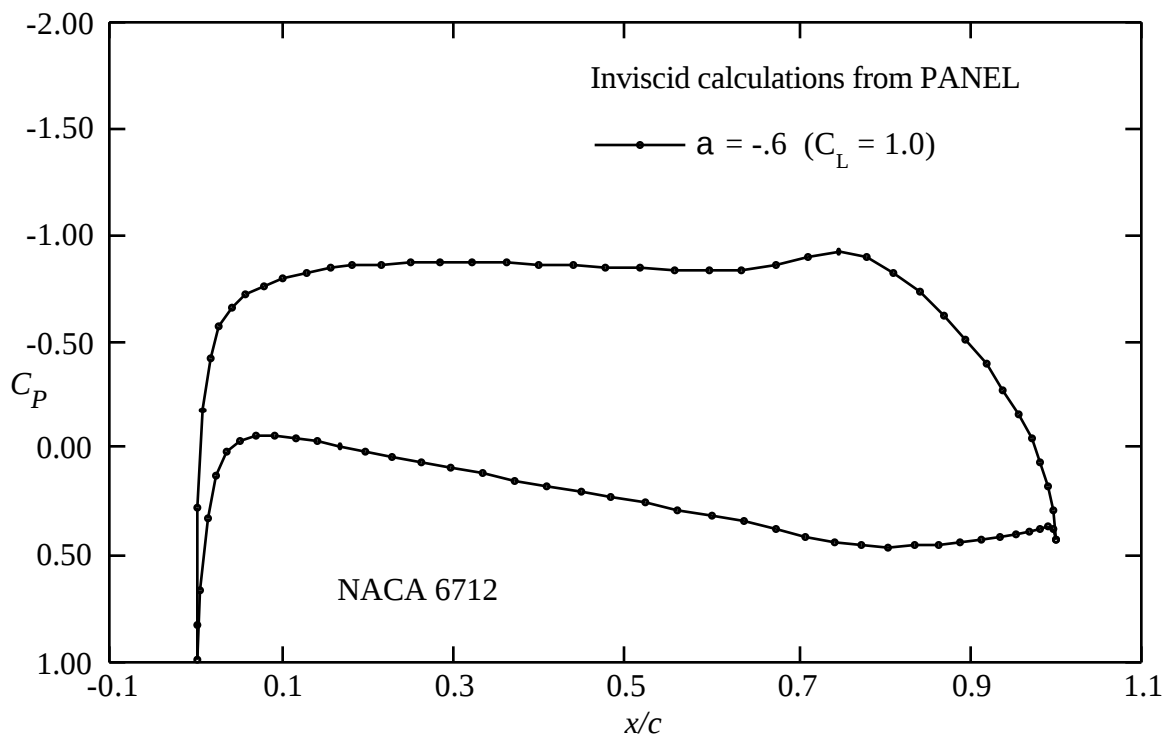


Figure 4-36. Example of the use of aft camber to “open up” the pressure distribution near the trailing edge.

The airfoils used to demonstrate geometry effects on pressure distributions above use parametric geometry definition formulas developed in the 1930s. More modern airfoils are available to the aerodynamicist. Unfortunately, to obtain improved performance, the designs were developed without the use of simple geometric definitions, and are available only as tables of coordinates. One modern airfoil that extends some of the previous shapes to obtain a high performance airfoil is the GA(W)-1 airfoil.¹⁵ This 17% thick airfoil designed by NASA's Richard Whitcomb provides better maximum lift and stall characteristics. Figure 4-37 shows the airfoil shape, and Fig. 4-38 shows the pressure distribution.

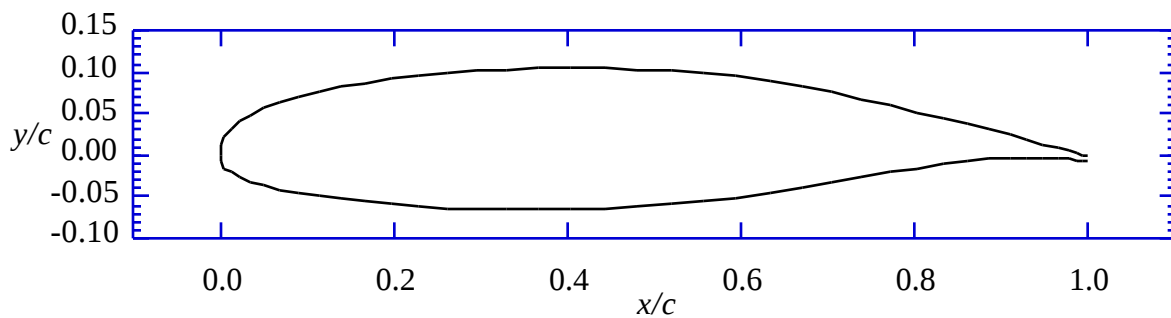


Figure 4-37. GA(W)-1 airfoil, also known as NASA LS(1)-0417.

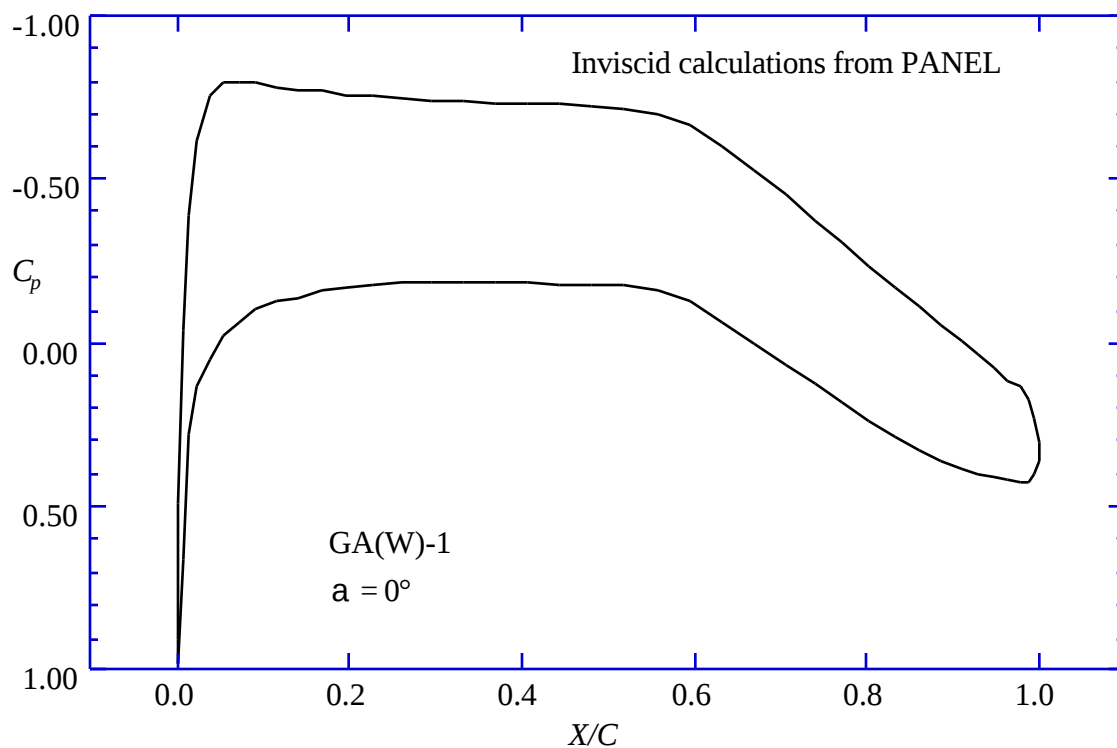


Figure 4-38. Pressure distribution at zero angle of attack of the GA(W)-1.

Notice that in this case the upper surface pressure distribution reaches a constant pressure plateau, and then has a moderate pressure recovery. Aft camber is used to obtain lift on the lower surface and “open up” the airfoil pressure distribution near the trailing edge in a manner suggested previously in Fig. 4-36. The area of aft camber on the lower surface is known as the “cove” region. If the camber is too extreme here the adverse pressure gradient will be too steep, and the flow will separate on the lower surface before it separates on the upper surface. Also, this type of pressure distribution has a significantly higher C_{m0} than conventional airfoil sections.

4.6.2 Geometry and Design

Effects of Shape Changes on Pressure Distributions: So far the examples have demonstrated global effects of camber and thickness. To develop an understanding of the typical effects of adding local modifications to the airfoil surface, Exercise 5 provides a framework for the reader to carry out an investigation analogous to the one for which results were presented in Section 4.6.1. It is also worthwhile to investigate the very powerful effects that small deflections of the trailing edge can produce. This reveals the power of the Kutta condition, and alerts the aerodynamicist to the basis for the importance of viscous effects at the trailing edge.

This approach is extremely educational when implemented in an interactive computer program, where the aerodynamicist can make shape changes with a mouse and see the effect on the pressure distribution immediately. An outstanding code that does this has been created by Ilan Kroo.¹⁶ It is called **PANDA**, originally was for the Macintosh, but now is available for a PC.

Shape for a specified pressure distribution: There is another way that aerodynamicists view the design problem. The local modification approach described above is useful to make minor changes in airfoil pressure distributions. Often the aerodynamic designer wants to find the geometric shape corresponding to a prescribed pressure distribution from scratch. This problem is known as the inverse problem. This problem is more difficult than the analysis problem. It is possible to prescribe a pressure distribution for which no geometry exists. Even if the geometry exists, it may not be acceptable from a structural standpoint. For two-dimensional incompressible flow it is possible to obtain conditions on the surface velocity distribution that ensure that a closed airfoil shape exists. Excellent discussions of this problem have been given by Volpe¹⁷ and Sloof.¹⁸ A two-dimensional panel method has been developed by Bristow.¹⁹ Numerical optimization can also be used to find the shape corresponding to a prescribed pressure distribution.²⁰

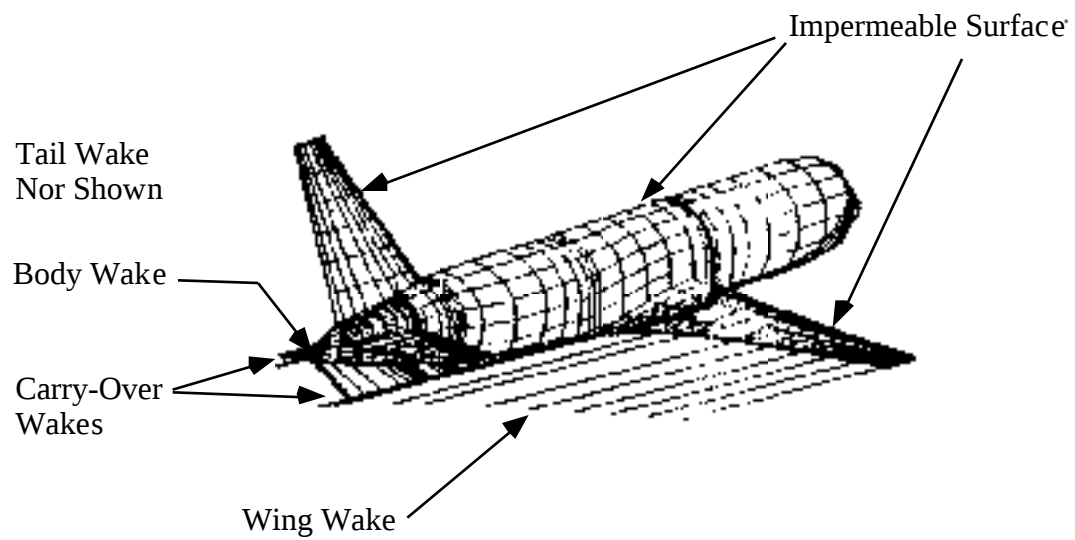
4.7 Issues in the Problem formulation for 3D flow over aircraft

The extension of panel methods to three dimensions leads to fundamental questions regarding the proper specification of the potential flow problem for flow over an aircraft. Examples include the proper treatment of wing tips and the treatment of the wake and fuselage aft of the wing. Hess²¹ provides an excellent discussion of the problems. In particular, the Kutta condition has to be reconsidered in three-dimensional flow. There are several aspects to consider. When solving the flow over a complete aircraft the aerodynamicist has to decide how to model the flow streaming off the fuselage or tip tank. The Kutta condition applies to distinct edges, and the inviscid model is not as precise. Many different approaches have been followed. Carmichael and Erickson²² also provide good insight into the requirements for a proper panel method formulation. Similarly, references 4 and 5 provide good overviews.

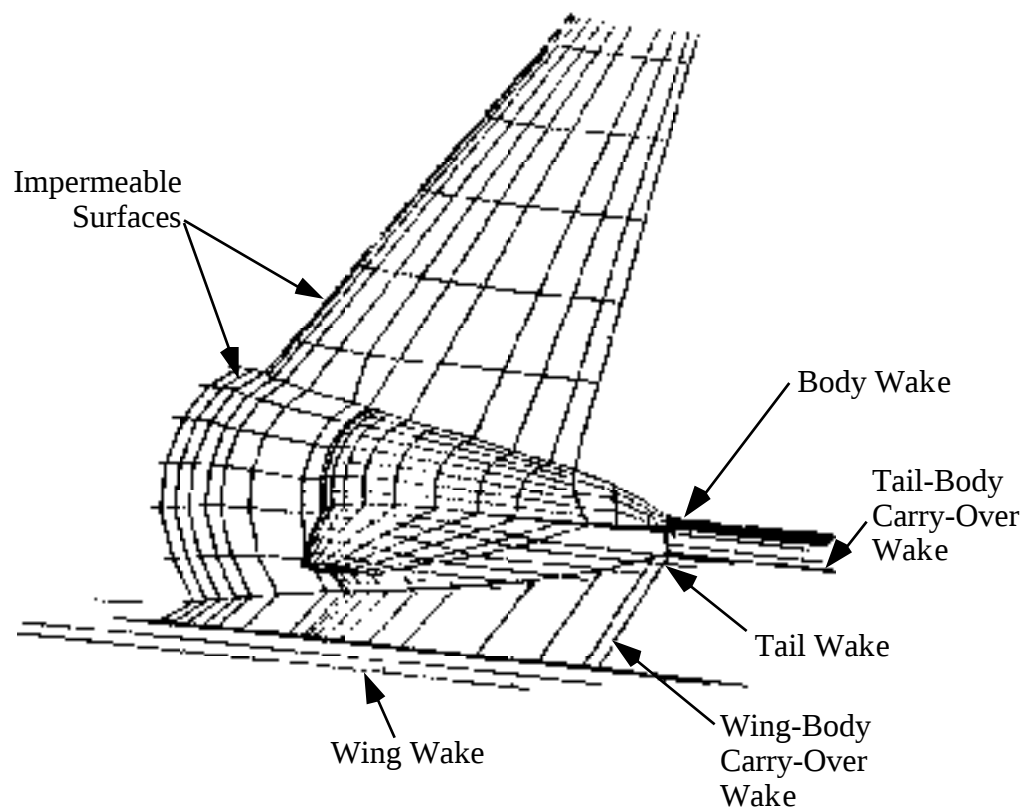
Aerodynamics panel methods generally use quadrilateral panels to define the surface. Since three points determine a plane, the quadrilateral may not necessarily define a consistent flat surface. In practice, the methods actually divide panels into triangular elements to determine an estimate of the outward normal. It is important that edges fit so that there is no leakage in the panel model representation of the surface.

Other practical considerations also require fastidious attention to detail. These include making sure that the outward surface normal is oriented in the proper direction, that all surfaces are properly enclosed, and that wakes are properly specified. In some methods wakes are handled automatically. In other methods the wakes must be precisely specified by the user. This provides complete control over the simulation, but means that the user must understand precisely what the problem statement should be. Figure 4-39 shows an example of a panel model including the details of the wakes. For high lift cases and wakes from one surface streaming near another, wake deflection must be computed as part of the solution. Figure 4-39 comes from a one week “short” course that was given to prospective users of an advanced panel method known as PAN AIR.²³ Clearly, to ensure that the problem is properly specified, and to examine the entire flowfield in detail, a complete graphics capability is required.

There is one other significant difference. Induced drag occurs even in inviscid, irrotational incompressible flow. However, its calculation by integration of pressures over the surface requires extreme accuracy, as we saw above for the two-dimension examples. The use of a farfield momentum approach is much more accurate, and is described in Chap. 5, Drag, An Introduction.



a) wing-body-tail configuration panel scheme with wakes



b) details of the wake model required

Figure 4-39. Example of a panel model containing wake model details.
(from a viewgraph presented at a PAN AIR user's short course, Ref. 23)

4.8 Example applications of panel methods

Many examples of panel methods have been presented. Figure 4-40 shows an example of the use of a panel model to evaluate the effect of the space shuttle on the Boeing 747. This is a classic example. Other uses include the simulation of wind tunnel walls, support interference, and ground effects. Panel methods are also used in ocean engineering. Recent America's Cup designs have been dependent on panel methods for hull and keel design. The effects of the free surface can be treated using panel methods.

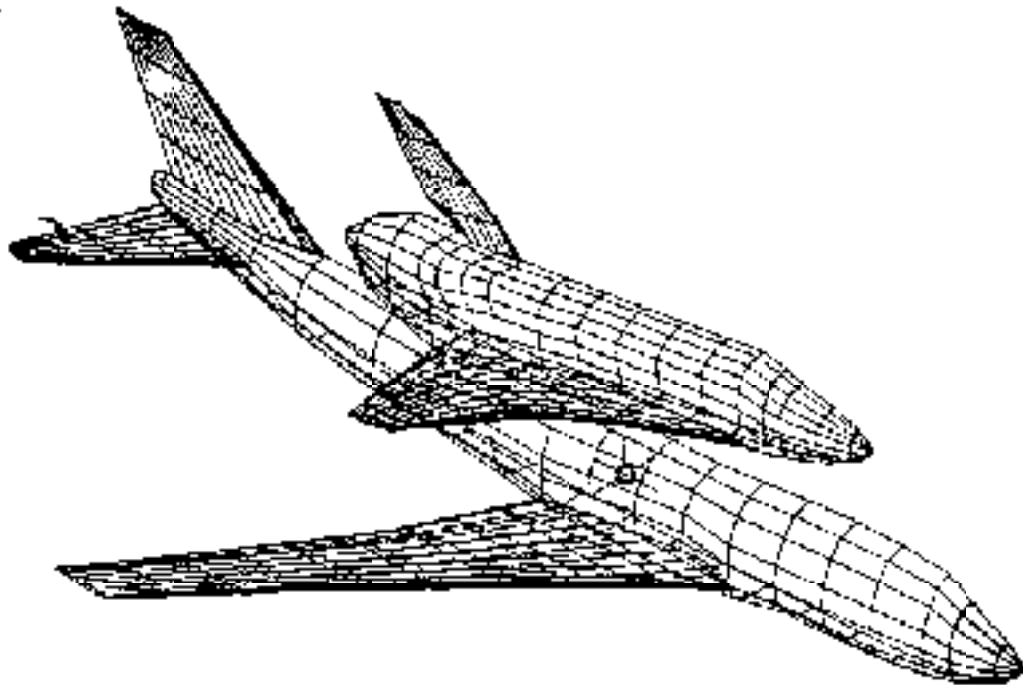


Figure 4-40. The space shuttle mounted on a Boeing 747.

One example has been selected to present in some detail. It is an excellent illustration of how a panel method is used in design, and provides a realistic example of the typical agreement that can be expected between a panel method and experimental data in a demanding real application. The work was done by Ed Tinoco and co-workers at Boeing.²⁴ Figure 4-41 shows the modifications required to modify a Boeing 737-200 to the 737-300 configuration. The panel method was used to investigate the design of a new high lift system. They used PAN AIR, which is a Boeing developed advanced panel method.²⁵ Figure 4-42 shows the panel method representation of the airplane.

Figure 4-41. The Boeing 737-300 relative to the model 737-200 (Ref.24).

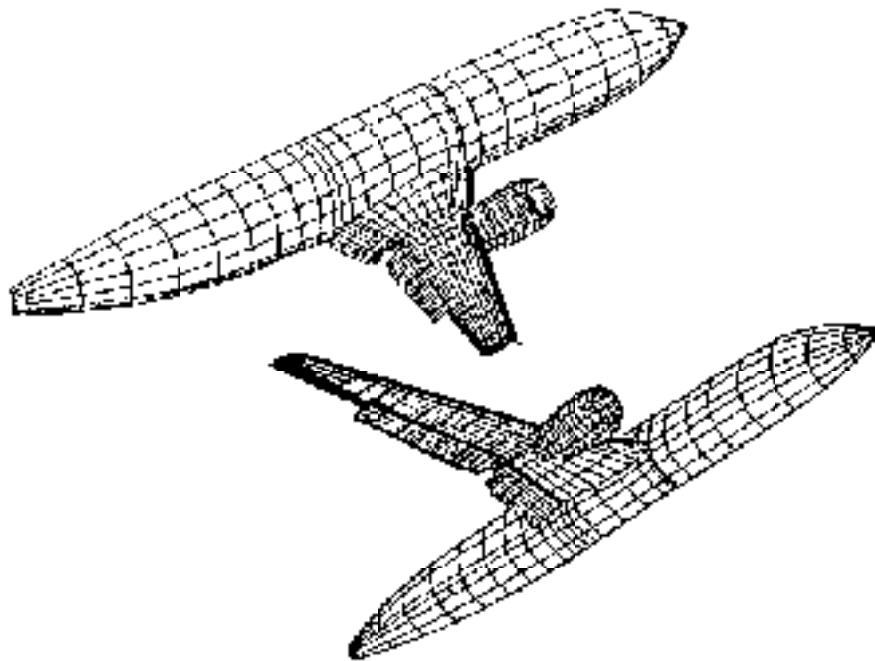


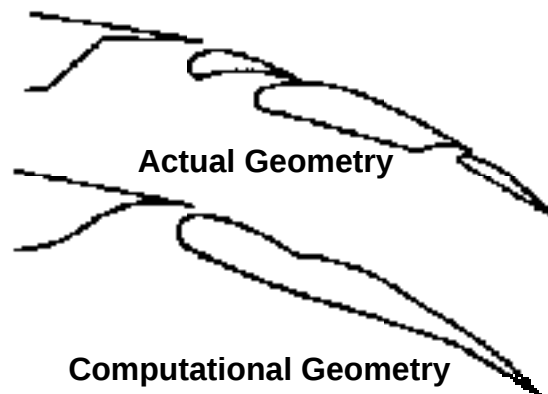
Figure 4-42. The panel representation of the 737-300 with 15° flap deflection (Ref. 4).

An understanding of the wing flowfield for two different takeoff flap settings was desired. The cases are “flaps 15”, the normal takeoff setting, and “flaps 1”, the high altitude, hot day setting. The work was conducted in concert with the flight test program to provide insight into the flight test results by providing complete flowfield details not available from the flight test. The computational models used 1750 panels for flaps 1 and 2900 panels for flaps 15. The modeling used to simulate this flowfield illustrates typical idealizations employed when applying panels methods to actual aircraft. Although typical, it is one of the most geometrically complicated examples ever published.

Figure 4-43 shows the wing leading edge and nacelle. The inboard Krueger flap was actually modeled as a doublet of zero thickness. The position was adjusted slightly to allow the doublet sheet to provide a simple matching of the trailing edge of the Krueger and the leading edge of the wing. These types of slight adjustments to keep panel schemes relatively simple are commonly used. The outboard leading and trailing edge flap geometries were also modified for use in this inviscid simulation. Figure 4-44 a) shows the actual and computational flaps 1 geometry. In this case the airfoil was modeled as a single element airfoil. The flaps 15 trailing edge comparison between the actual and computational geometry is shown in Fig. 4-44 b). The triple slotted flap was modeled as a single element flap. At this setting the gap between the forward vane and main flap is closed, and the gap between the main and aft flap is very small.

Figure 4-43. Inboard wing leading edge and nacelle details (Ref. 24).

a) Comparison of actual and computational wing geometry for the flaps 1 case (Ref. 24).



b) Actual and computational trailing edge geometry for the flaps 15 case (Ref. 4).

Figure 4-44. Examples of computational modeling for a real application.

Several three-dimensional modeling considerations also required attention. In the flaps 1 case shown in Fig. 4-45, spanwise discontinuities included the end of the outboard leading edge slat and trailing edge discontinuities at the back of the nacelle installation (called the thrust gate) between the inboard and outboard flaps. At the outboard leading edge the edges of the slat and wing were paneled to prevent leakage. A 0.1 inch gap was left between these surfaces. At the trailing edge discontinuity a wake was included to model a continuous trailing edge from which a trailing vortex sheet could be shed.

Figure 4-45. Spanwise discontinuity details requiring modeling for flaps 1 case (Ref. 24).

Similar considerations are required for the flaps 15. Here, special care was taken to make sure that the configuration was closed, and contained no holes in the surface at the ends of the flap segments.

Another consideration is the nacelle model. This requires the specification of the inlet flow at the engine face, a model of the strut wake, and both the outer bypass air plume and the primary wake from the inner hot gas jet. Figure 4-46 provides the details.

Figure 4-46 Nacelle model (Ref. 24).

Complete details of the model are contained in Ref. 24. With the model complete, the solution was obtained. The spanwise distribution of airfoil section lift coefficients is presented in Figure 4-47. The first part of the figure shows the results for the flaps 1 case, and the second part of the figure presents the flaps 15 case. In both cases the jig shape and flight shape including aeroelastic deformation are included. This is another consideration in making a proper aerodynamic simulation. In both cases the shape including the deformation under load shows much better agreement with flight and wind tunnel data. Notice the loss of lift on the wing at the nacelle station, and the decrease in lift outboard of the trailing edge flap location.

a) flaps 1 case

b) flaps 15 case

Figure 4-47. Spanwise distribution of lift coefficient on the Boeing 737-300 (Ref.24).

Figure 4-48 presents the change in section lift coefficient with angle of attack at several span stations. The agreement between PAN AIR and flight test is better for the flaps 1 case. Viscous effects are becoming important for the flaps 15 case.

a) flaps 1 case

b) flaps 15 case

Figure 4-48. Comparison of section lift coefficient change with angle of attack(Ref.24)

Figure 4-49 completes this example by presenting the comparison of pressure distributions for the two cases at four spanwise stations. The flaps 1 case agreement is generally good. Calculations are presented for both the actual angle of attack, and the angle of attack which matches the lift coefficient. Matching lift coefficient instead of angle of attack is a common practice in computational aerodynamics. Considering the simplifications made to the geometry and the absence of the simulation of viscous effects the agreement is very good. The flaps 15 case starts to show the problems that arise from these simplifications. This is a good example of the use of a panel method. It illustrates almost all of the considerations that must be addressed in actual applications.

a) flaps 1 case

b) flaps 15 case

Figure 4-49. Comparison of pressure distributions between flight and computations for the 737-300, solid line is PAN AIR at flight lift, dashed line is PAN AIR at flight angle of attack (Ref. 24).

4.9 Using Panel Methods

4.9.1 Common sense rules for panels

- Vary the size of panels smoothly
- Concentrate panels where the flowfield and/or geometry is changing rapidly
- Don't spend more money and time (i.e., numbers of panels) than required

Panel placement and variation of panel size affect the quality of the solution. However, extreme sensitivity of the solution to the panel layout is an indication of an improperly posed problem. If this happens, the user should investigate the problem thoroughly.

Panel methods are an aid to the aerodynamicist. You must use the results as a guide to help you develop your own judgement. (An issue: lawyers often get involved because you frequently sign an agreement that the code developer is not liable for problems that stem from the use of the code; the same disclaimer you see with every PC programs).

Remember that the panel method solution is an approximation of the real life problem; an idealized representation of the flowfield. An understanding of aerodynamics that provides an intuitive expectation of the types of results that may be obtained, and an appreciation of how to relate your idealization to the real flow is required to get the most from the methods. This insight requires experience and study.

4.9.2 What a Panel Method Can't Do

1. Panel methods are inviscid solutions. You will not capture viscous effects except via user "modeling" by changing the geometry.
2. Solutions are invalid as soon as the flow develops local supersonic zones [i.e., $C_p < C_{p_{crit}}$]. For two-dimensional isentropic flow, the exact value of C_p for critical flow is:

$$C_{p_{crit}} = -\frac{2}{\gamma M_\infty^2} \left[1 - \frac{1}{\frac{\gamma+1}{2} M_\infty^2} \right]^{\frac{\gamma}{\gamma-1}} \quad (4-96)$$

4.10 Advanced panel methods: What is a "Higher Order" Panel Method?

So-called "higher-order" panel methods use singularity distributions that are not constant on the panel, and may also use panels which are non-planar. Higher order methods were found to be crucial in obtaining accurate solutions for the Prandtl-Glauert Equation at supersonic speeds. At supersonic speeds, the Prandtl-Glauert equation is actually a wave equation (hyperbolic), and

requires much more accurate numerical solution than the subsonic case in order to avoid pronounced errors in the solution (Magnus and Epton²⁵). However, subsonic higher order panel methods, although not as important as the supersonic flow case, have been studied in some detail. In theory, good results can be obtained using far fewer panels with higher order methods. In practice the need to resolve geometric details often leads to the need to use small panels anyway, and all the advantages of higher order panelling are not necessarily obtained. Nevertheless, since a higher order panel method may also be a new program taking advantage of many years of experience, the higher order code may still be a good candidate for use.

4.11 Today's standard programs: a brief survey

Panel methods are widely used in the aircraft industry, and have been for a long time. Comparisons between codes have been made, the most recent comparison being by Margason, et al.²⁶ In general, all the new professionally-developed codes work well. The selection of a specific code will likely be based on non-technical considerations. In recent times, several codes have emerged as the primary ones. The newest is known as PMARC,²⁷ for Panel Method Ames Research Center. These codes have received the most development effort. We provide a brief description of the codes a new aerodynamicist will most likely encounter. Specific references are provided in Tables 4-1 through 4-3.

PAN AIR - Boeing-developed code, funded by a variety of government agencies, and available through COSMIC (a lease arrangement, about \$7000 last time I looked, and export controlled).

This code provides total flexibility, i.e., it's really an integral equation solver and not an aerodynamicist's tool *per se*. It uses higher order panels, and is both subsonic and supersonic. It's relatively expensive and difficult to run (a PAN AIR user would take months to train, and it would probably become his primary job).

To effectively use the code good pre- and post- processing systems must be available. Although Boeing has these systems in place, they were internally developed and are not available outside the company.

VSAERO - AMI developed (Analytical Mechanics Inc., Frank Dvorak and Brian Maskew). It uses low order panels and is subsonic only. It also handles general geometries, and includes options to treat viscous effects and vortex flows. The original NASA version is available through COSMIC. However, the code has been much further developed by AMI and is for sale by this company. The price for the current code is about \$100K, and they also have a plotting package (OMNILOT, about \$20K) available for purchase. This code also requires considerable user training. Support from AMI is about \$10-\$15K per year, and site licensing is not available (as of 1990). You pay a

large fee for each machine on which you install VSAERO. The business of licensing codes from developers is an important consideration in computational aerodynamics in the '90s.

The public domain version of this code was obtained by several groups that worked on the design of the America's Cup Yacht competitors in the mid-eighties. The code was used for hull and keel design. One of the modifications that was made was the addition of the free surface representing the air-water interface (recall that the free surface problem means that the surface displacement is unknown, and the boundary condition is that a constant pressure exists at the interface).

QUADPAN - Lockheed-developed, and possibly developed at some government labs. Not widely used by industry outside of Lockheed. This is probably because of availability.

Versions of the "Hess Code" - further developments of the team at Douglas now led by Hess. Naturally, Douglas uses this code exclusively. Douglas developed numerous versions under various government contracts, and it seems to be available mainly at Navy facilities.

Woodward: An old panel method that is sometimes encountered is the code known as the "Woodward" or "Woodward-Carmichael" code. Woodward was a pioneer panel method developer, and the most likely Woodward code a new aerodynamicist might encounter is a version of USSAERO, which was developed under NASA contract. Woodward's first methods were developed while he was at Boeing, and were supported by NASA Ames, primarily for the US SST program (which was an important national effort in the sixties). Subsequently, Woodward went into business and continued to develop codes. USSAERO treats both supersonic and subsonic flow, and a version which incorporates design options "Woodward 1-2" is available at VPI.

PMARC -This is the newest panel method code, and was developed at NASA Ames to provide an extremely flexible method to simulate a wide range of very general geometries. An example is the simulation of high lift systems and jet exhausts for VSTOL aircraft. The code is a lower order panel method, and can simulate steady as well as unsteady flow. The wake position can be obtained as part of the solution. It is being used for underwater applications as well as for aircraft. This code is also available at VPI.

The history of panel methods is illustrated in the tables. Table 4-1 summarizes some of the key early methods that were developed. W12SC3 is included because it was a valuable combination of two early codes, providing significant design capability, particularly at supersonic speeds. Table 4-2 reviews the extremely active era of the development of advanced methods. Finally, Table 4-3 provides details on the current production codes likely to be used on current aerodynamic design and analysis projects.

Table 4 - 1
Comparison of Some Major Panel Method Programs: Early Codes

<i>Originator and Method Name</i>	<i>Year</i>	<i>Panel Geometry</i>	<i>Source Type</i>	<i>Doublet Type</i>	<i>Boundary Conditions</i>	<i>Restrictions</i>	<i>Comments</i>
Hess and Smith ¹ (Douglas)	1962	flat	constant	none	specification of normal flow	non-lifting wings and bodies only	
Rubbert ² (vortex lattice)	1964	flat	none	constant	normal flow	planar wings only	
Woodward ³ (Woodward I)	1967	flat	constant	linear	normal flow	wings must be planar	mainly supersonic, includes design & optimization
Rubbert and Saaris ⁴ (Boeing A-230)	1968	flat	constant	constant	normal flow	nearly constant panel density	
Hess I ⁵	1972	flat	constant	linear	normal flow	wings and bodies only	
USSAERO ⁶ (Woodward II)	1973	flat					subsonic and supersonic, analysis only
W12SC3 ⁷ (Grumman)	1983	flat			mixed design and analysis		combines Woodward 1 & 2 features

- 1 Hess, J.L., and Smith, A.M.O., "Calculation of Nonlifting Potential Flow About Arbitrary Three-Dimensional Bodies," Douglas Report ES40622, Douglas Aircraft Company, 1962.
- 2 Rubbert, P.E., "Theoretical Characteristics of Arbitrary Wings by a Nonplanar Vortex Lattice Method," Boeing Report D6-9244, The Boeing Company, 1964.
- 3 Woodward, F.A., Tinoco, E.N., and Larsen, J.W., "Analysis and Design of Supersonic Wing-Body Combinations, Including Flow Properties in the Near Field," Part I - Theory and Application, NASA CR-73106, 1967.
- 4 Rubbert, P.E., and Saaris, G.R., "A General Three-Dimensional Potential Flow Method Applied to V/STOL Aerodynamics," SAE Paper No. 680304, 1968.
- 5 Hess, J.L., "Calculation of Potential Flow About Arbitrary 3-D Lifting Bodies," Douglas Report MDC-J5679-01, October 1972.
- 6 Woodward, F.A., "An Improved Method for the Aerodynamic Analysis of Wing-Body-Tail Configurations in Subsonic and Supersonic Flow," NASA CR-2228, Parts I and II, 1973.
- 7 Mason, W.H., and Rosen, B.S., "The COREL and W12SC3 Computer Programs for Supersonic Wing Design and Analysis," NASA CR 3676, 1983 (contributions by A. Cenko and J. Malone acknowledged).
from Magnus and Epton, NASA CR 3251, April 1980 (with extensions)

Table 4 - 2
Comparison of Some Major Panel Method Programs: Advanced Methods

<i>Originator and Method Name</i>	<i>Year</i>	<i>Panel Geometry</i>	<i>Source Type</i>	<i>Doublet Type</i>	<i>Boundary Conditions</i>	<i>Restrictions</i>	<i>Comments</i>
Roberts and Rundle ¹	1973	paraboloidal	quadratic	quadratic	normal flow		numerical integration, very expensive
Mercer, Weber and Lesford ²	1973	flat	none	smooth, cubic/quadratic	normal flow in least squares sense	planar wings	subsonic/supersonic, cubic spanwise, quadratic chordwise
Morino and Kuo ³ (SOUSSA)	1974	continuous, hyperboloidal	constant	constant	potential	no thin configurations	unsteady
Johnson and Rubbert ⁴	1975	paraboloidal	linear	quadratic	normal flow		
Ehlers and Rubbert ⁵ (Mach line paneling)	1976	flat	linear	continuous quadratic	normal flow	planar wings, special paneling	supersonic flow
Ehlers et al. ⁶ (PAN AIR "pilot code")	1977	continuous piecewise flat	linear	continuous quadratic	arbitrary in $f, -f$		subsonic and supersonic

- ¹ Roberts, A., and Rundle, K., "Computation of First Order Compressible Flow About Wing-Body Configurations," AERO MA No. 20, British Aircraft Corporation, February, 1973.
- ² Mercer, J.E., Weber, J.A., and Lesford, E.P., "Aerodynamic Influence Coefficient Method Using Singularity Splines," NASA CR-2423, May 1974.
- ³ Morino, L., and Kuo, C-C, "Subsonic Potential Aerodynamics for Complex Configurations: A General Theory," *AIAA Journal*, Vol. 12, No. 2, pp 191-197, February, 1974.
- ⁴ Johnson, F.T., and Rubbert, P.E., "Advanced Panel-Type Influence Coefficient Methods Applied to Subsonic Flow," AIAA Paper No. 75-50, January 1975.
- ⁵ Ehlers, F.E., and Rubbert, P.E., "A Mach Line Panel Method for Computing the Linearized Supersonic Flow," NASA CR-152126, 1979.
- ⁶ Ehlers, F.E., Epton, M.A., Johnson, F.T., Magnus, A.E., and Rubbert, P.E., "A Higher Order Panel Method for Linearized Flow," NASA CR-3062, 1979.

from Magnus and Epton, NASA CR 3251, April 1980 (with extensions)

Table 4-3
Comparison of Some Major Panel Method Programs: Production Codes

<i>Originator and Method Name</i>	<i>Year</i>	<i>Panel Geometry</i>	<i>Source Type</i>	<i>Doublet Type</i>	<i>Boundary Conditions</i>	<i>Restrictions</i>	<i>Comments</i>
MCAIR ¹ (McDonnell)	1980	flat	constant	quadratic			design option
PAN AIR ² (Boeing)	1980	continuous piecewise flat	continuous linear	continuous quadratic	arbitrary in f , $-f$		subsonic and supersonic
Hess II ³ (Douglas)	1981	parabolic	linear	quadratic	normal flow		
VSAERO ⁴ (AMI)	1981	flat	constant	constant	exterior and interior normal flow		subsonic
QUADPAN ⁵ (Lockheed)	1981	flat	constant	constant			
PMARC ⁶ (NASA Ames)	1988	flat	constant	constant			unsteady, wake rollup

- ¹ Bristow, D.R., "Development of Panel Methods for Subsonic Analysis and Design," NASA CR 3234, 1980.
- ² Magnus, A.E., and Epton, M.A., "PAN AIR - A Computer Program for Predicting Subsonic or Supersonic Linear Potential Flows About Arbitrary Configurations Using a Higher Order Panel Method," Volume I - Theory Document (Version 1.0), NASA CR 3251, 1980.
- ³ Hess, J.L., and Friedman, D.M., "An Improved Higher Order Panel Method for Three Dimensional Lifting Flow," Douglas Aircraft Co. Report No. NADC-79277-60, 1981.
- ⁴ Maskew, B., "Prediction of Subsonic Aerodynamic Characteristics: A Case for Lower Order Panel Methods," AIAA Paper No. 81-0252, 1981.
- ⁵ Coopersmith, R.M., Youngren, H.H., and Bouchard, E.E., "Quadrilateral Element Panel Method (QUADPAN)," Lockheed-California LR 29671, 1981.
- ⁶ Ashby, D.L., Dudley, M.R., and Iguchi, S.K., "Development and Validation of an Advanced Low-Order Panel Method," NASA TM 101024, 1988 (also TM 102851, 1990).

from Magnus and Epton, NASA CR 3251, April 1980 (with extensions)

4.12 Exercises

1. Program **PANEL**.

- a) Obtain a copy of program **PANEL** and the sample case.
- b) Convert **PANEL** to run on your PC.
- c) Run the sample case: NACA 4412, 20 pts. upper, 20 pts. lower, $\alpha = 4^\circ$, and verify against sample case.
- d) Document
 - i) compile time required on your PC (cite computer and compiler used)
 - ii) the execution time for the sample case
 - iii) the accuracy relative to the sample case.
 - iv) the exact modifications required to make the code work on your computer

2. Start work on program **PANEL**

- a) Save a reference copy of the working code!
- b) Check convergence with panels (NLOWER+NUPPER must be less than 100 currently). How many panels do you need to get results independent of the number of panels? What happens to the computer time as the number of panels increases?
- c) Check the coordinates generated by the airfoil routine vs. exact (consider using the NACA 0012, see App. A for geometry definition), including examination of the coordinates at the trailing edge. This is best done by making a table of exact and computed values at selected values of x/c . What did you find out?
- d) Locate the source strengths, and sum the source strengths $\times$ panel lengths to get the total source strength. Does it sum to zero? Should it?
- e) Where is the moment reference center in this code?

Submit an assessment of your findings.

3. Modify program **PANEL**:

You need a version of **PANEL** that will allow you to compute the pressure distribution on arbitrary airfoils. This exercise will give you this capability. Modify the code to interpolate input airfoil points to the program defined surface points, x/c . The resulting code should:

- a) accept arbitrary airfoil input data
- b) echo all the input data on the output
- c) generate an output file for post processing
(both for plotting and as the input to a boundary layer code)
- d) output C_m about the airfoil quarter chord point.

Hint: Don't alter the panel distribution. The paneling scheme should be independent of the input distribution of airfoil coordinates. This produces a much more general and accurate program. This problem is usually solved by finding both the x/c and y/c

values as functions of the airfoil arc length, starting at the lower surface trailing edge. A spline fit is usually used to interpolate the values along the arc length.

Check your modified code. Run the airfoil you ran previously with internal coordinate generation. This time use an input file with the same coordinates as external inputs. Submit a description of your work, and assess your results.

4. Assess the accuracy of incompressible potential flow theory. Run your modified **PANEL** code using the airfoil you selected in the exercise in Chap. 1. (What happens if your airfoil has a trailing edge with finite thickness? What do you do now?)

- compare the computed pressure distribution with the experimental data
- compare the computed force and moment results with the data (over a range of angles of attack)

Turn in a CONCISE report describing the results of your work. Include a plot showing the pressure distribution comparison, and a plot(s) showing comparison with forces and moments. What do you conclude about the accuracy of this method?

5. Airfoil design using program **PANEL**

Take your reference airfoil:

- a) add thickness on the bottom (mid chord)- what happens?
- b) shave some thickness off the bottom (mid chord) - ?
- c) add thickness on the top (mid chord)- what happens?
- d) deflect the trailing edge down a couple of degrees (how sensitive is the airfoil to changes at the TE?)

Hint: use smooth d's to the reference foil employing analytic formulas.

Turn in a CONCISE report comparing the effects on the pressure distribution due to the above modifications.

6. How good is thin airfoil theory? Compare the thin airfoil DC_p for a flat plate with program **PANEL**.

Recall thin airfoil theory for an uncambered flat plate:

$$DC_p = 4a \sqrt{\frac{(1 - x/c)}{x/c}}.$$

- a) pick an NACA 0012 airfoil at $\alpha = 2^\circ$ and 12° and run **PANEL**.
- b) plot DC_p/a as a function of x/c .
- c) how many panels do you need to get a converged solution from **PANEL**?
- d) what conclusions do you reach?

4.13 References

- ¹ Anderson, John P., Jr., *Modern Compressible Flow*, 2nd Ed., McGraw-Hill, New York, 1990, pp. 258-269.
- ² Hess, J. L., "Panel Methods in Computational Fluid Dynamics," *Annual Review of Fluid Mechanics*, Vol. 22, 1990, pp. 255-274.
- ³ Hess, J.L., "Linear Potential Schemes," *Applied Computational Aerodynamics*, P.A. Henne, ed., AIAA, Washington, 1990. pp.21-36.
- ⁴ Erickson, L.L., "Panel Methods—An Introduction," NASA TP-2995, Dec. 1990.
- ⁵ Katz, J., and Plotkin, A., *Low-Speed Aerodynamics From Wing Theory to Panel Methods*, McGraw-Hill, Inc., New York, 1991.
- ⁶ Moran, J. *An Introduction to Theoretical and Computational Aerodynamics*, John Wiley & Sons, New York, 1984. pp. 103-112, 118-123, 260-287.
- ⁷ Karamcheti, K., *Principles of Ideal-Fluid Aerodynamics*, John Wiley & Sons, New York, 1966.
- ⁸ Ashley, H, and Landahl, M., *Aerodynamics of Wings and Bodies*, Addison-Wesley, Reading, 1965 (republished in paperback by Dover Publishing).
- ⁹ Curle, N., and Davis, H.J., *Modern Fluid Dynamics*, Volume 1: Incompressible Flow, Van Nostrand, London, 1968.
- ¹⁰ Houghton, E.L., and Carpenter, P.W., *Aerodynamics for Engineering Students*, 4th Ed., Halsted Press, New York, 1993, pp. 257-265, 203-211.
- ¹¹ Kuethe, A.M., and Chow, C-Y., *Foundations of Aerodynamics*, 4th Ed., John Wiley, New York, 1986, pp. 128-137.
- ¹² Abbott, I.H., and von Doenhoff, A.E., *Theory of Wing Sections*, Dover, New York, 1959.
- ¹³ Jones, R.T., *Wing Theory*, Princeton University Press, Princeton, 1990.
- ¹⁴ Warner, E.P., *Airplane Design: Performance*, McGraw-Hill, New York, 1936.
- ¹⁵ McGhee, Robert J., and Beasley, William D., "Low Speed Aerodynamic Characteristics of a 17-Percent-Thick Airfoil Section Designed for General Aviation Applications," NASA TN D-7428, 1973.
- ¹⁶ Kroo, Ilan, "Aerodynamic Analyses for Design and Education," AIAA Paper 92-2664, June 1992.
- ¹⁷ Volpe, G., "Inverse Airfoil Design: A Classical Approach Updated for Transonic Applications," in *Applied Computational Aerodynamics*, ed. by P.A. Henne, AIAA Progress in Astronautics and Aeronautics, Vol. 125, AIAA, New York, 1990, pp. 191-220.
- ¹⁸ Labrujere, Th. E., and Sloof, J.W., "Computational Methods for the Aerodynamic Design of Aircraft Components, Ann. Rev. of Fluid Mech., 1993, Vol. 25, pp.183-214.
- ¹⁹ Bristow, D.R., "A New Surface Singularity Method for Multi-Element Airfoil Analysis and Design," AIAA Paper 76-20, Jan. 1976.
- ²⁰ Aidala, P.V., Davis, W.H., Jr., and Mason, W.H., "Smart Aerodynamic Optimization," AIAA Paper 83-1863, July 1983.

²¹ Hess, J. L. "The Problem of Three-Dimensional Lifting Potential Flow and Its Solution by Means of Surface Singularity Distributions", *Computer Methods in Applied Mechanics and Engineering* 4 (1974) pp. 283-319.

²² Carmichael, R.L., and Erickson, L.L., "PAN AIR - A Higher Order Panel Method for Predicting Subsonic or Supersonic Linear Potential Flows About Arbitrary Configurations," AIAA Paper No. 81-1255, June 1981.

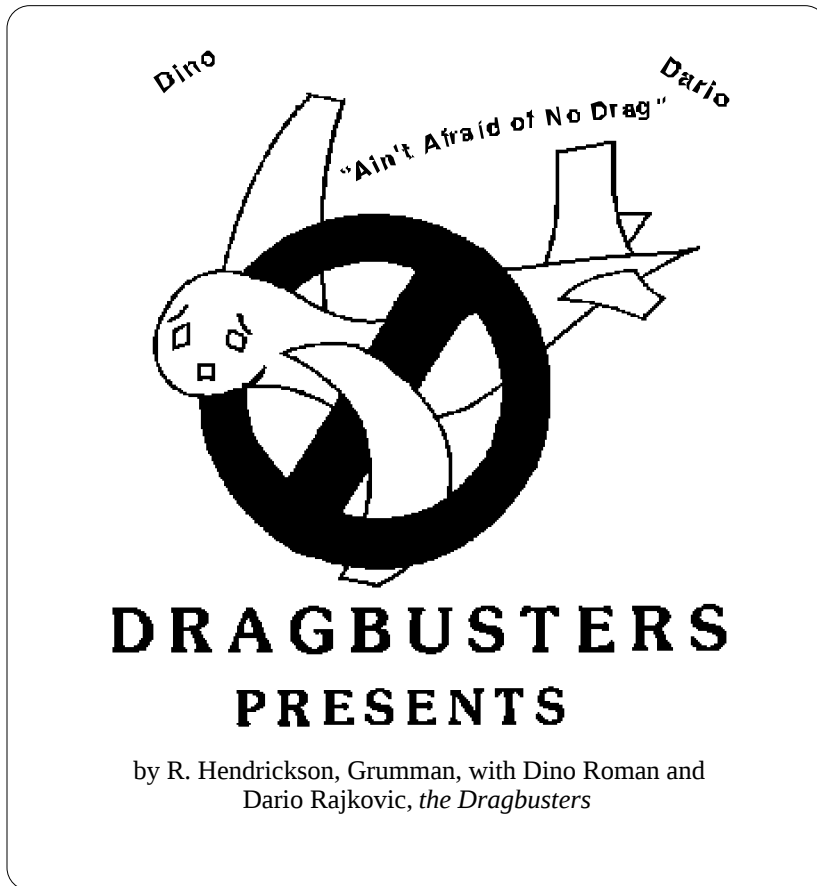
²³ PAN AIR User's Class Short Course Presentation Material, 1981.

²⁴ Tinoco, E.N., Ball, D.N., and Rice, F.A. II, "PAN AIR Analysis of a Transport High-Lift Configuration, *Journal of Aircraft*, Vol. 24, No. 3, March 1987, pp. 181-188.

²⁵ Magnus, A.E., and Epton, M.A., "PAN AIR - A Computer Program for Predicting Subsonic or Supersonic Linear Potential Flows About Arbitrary Configurations Using a Higher Order Panel Method," Volume I - Theory Document (Version 1.0), NASA CR 3251, April 1980.

²⁶ Margason, R.J., Kjelgaard, S.O., Sellers, W.L., Moriis, C.E.K., Jr., Walkley, K.B., and Shields, E.W., "Subsonic Panel Methods - A Comparison of Several Production Codes," AIAA Paper 85-0280, Jan. 1985.

²⁷ Ashby, D.L., Dudley, M.R., Iguchi, S.K., Browne, L., and Katz, J., "Potential Flow Theory and Operation Guide for the Panel Code PMARC," NASA TM 102851, Jan. 1991.



5. Drag: An Introduction

5.1 The Importance of Drag

The subject of drag didn't arise in our use of panel methods to examine the inviscid flowfield around *airfoils* in the last chapter: the theoretical drag was always zero! Before proceeding further in any study of computational aerodynamics the issue of drag must be addressed. There are many sources of drag. In three-dimensional flow, and in two dimensions when compressibility becomes important, drag occurs even when the flow is assumed inviscid. Before discussing the aerodynamics of lifting systems, the fundamental aspects of aerodynamic drag will be examined.

Drag is at the heart of aerodynamic design. The subject is fascinatingly complex. All aerodynamicists secretly hope for negative drag. The subject is tricky and continues to be controversial. It's also terribly important. Even seemingly minor changes in drag can be critical. On the Concorde, a one count drag increase ($\Delta C_D = .0001$) requires two passengers, out of the 90 ~ 100 passenger capacity, be taken off the North Atlantic run.¹ In design studies a drag decrease is equated to the decrease in aircraft weight required to carry a specified payload the required distance. One advanced fighter study² found the drag sensitivity in supersonic cruise was 90 lb/ct and 48 lb/ct for subsonic/transonic cruise. At the transonic maneuver design point the sensitivity was 16 lb/ct (drag is very high here). In comparison, the growth factor was 4.1 lb of takeoff gross weight for

every 1 lb of fixed weight added. For one executive business jet the range sensitivity is 17 miles/drag count. Advanced supersonic transports now being studied have range sensitivities of about 100 miles/drag count. When new aircraft are sold, the sales contract stipulates numerous performance guarantees. One of the most important is range. The aircraft company guarantees a specified range before the aircraft is built and tested. The penalty for failure to meet the range guarantee is severe. Conservative drag projections aren't allowed—the competition is so intense that in the design stage the aerodynamicist will be pressured to make optimistic estimates. In one briefing in the early '80s, an aerodynamicist for a major airframer said that his company was willing to invest \$750,000 for each count of drag reduction. Under these conditions the importance of designing for low drag, and the ability to estimate drag, can hardly be overstated.

The economic viability and future survival of an aircraft manufacturer depends on minimizing aerodynamic drag (together with the other design key technologies of structures, propulsion, and control) while maintaining good handling qualities to ensure flight safety and ride comfort. New designs that employ advanced computational aerodynamics methods are needed to achieve vehicles with less drag than current aircraft. The most recent generation of designs (Boeing 767, 777, Airbus A340, *etc.*) already take advantage of computational aerodynamics, advanced experimental methods, and years of experience. Future advances in aerodynamic performance present tough challenges requiring both innovative concepts and the very best methodology possible.

Initial drag estimates can dictate the selection of a specific configuration concept in comparison with other concepts early in the design phase. The drag projections have a huge effect on the projected configuration size and cost, and thus on the decision to proceed with the design.

There are two other key considerations in discussing drag. First, drag cannot yet be predicted accurately with high confidence levels³ (especially for unusual configuration concepts) without extensive testing, and secondly, no one is exactly sure what the ultimate possible drag level really is that can be achieved for a practical configuration. To this extent, aerodynamic designers are the dreamers of the engineering profession.

Because of its importance, AGARD has held numerous conferences devoted to drag and its reduction. In addition to the study of computational capability cited above, AGARD publications include CP-124,⁴ CP-264,⁵ R-723⁶ and R-786⁷. These reports provide a wealth of information.

An AIAA Progress Series book has also been devoted primarily to drag.⁸ Chapters discuss the history of drag prediction, typical methods currently used to predict drag, and the intricacies of drag prediction for complete configurations. The most complete compilation of drag information available is due to Hoerner.⁹ In this chapter we introduce the key concepts required to use computational aerodynamics to evaluate drag. Additional discussion is included in the chapters on viscous effects, transonic, and supersonic aerodynamics.

5.2 Some Different Ways to View Drag - Nomenclature and Concepts

In discussing drag, the numerous viewpoints that people use to think about drag can create confusion. Here we illustrate the problem by defining drag from several viewpoints. This provides an opportunity to discuss various basic drag concepts.

1. *Simple Integration*: Consider the distribution of forces over the surface. This includes a pressure force and a shear stress force due to the presence of viscosity. This approach is known as a nearfield drag calculation. An accurate integration will result in an accurate estimate of the drag. However, two problems exist:

- i) This integration requires extreme precision (remember that program PANEL did not predict exactly zero drag).
- ii) The results are difficult to interpret for aerodynamic analysis. Exactly where is the drag coming from? Why does it exist, and how do you reduce it?

Thus in most cases a simple integration over the surface is not satisfactory for use in aerodynamic design. Codes have only recently begun to be fairly reliable for nearfield drag estimation, and then only for certain specific types of problems. The best success has been achieved for airfoils, and even there the situation still isn't perfect (see Chapters 10 and 11).

2. *Fluid Mechanics*: This viewpoint emphasizes the drag resulting from various fluid mechanics phenomena. This approach is important in conceiving a means to reduce drag. It also provides a means of computing drag contributions in a systematic manner. Thinking in terms of components from different physical effects, a typical drag breakdown would be:

- friction drag
- form drag
- induced drag
- wave drag.

Each of these terms will be defined below. Figure 5-1 illustrates possible ways to find the total drag. It is based on a figure in Torenbeek's book.¹⁰ He also has a good discussion of drag and its estimation. Clearly, the subject can be confusing.

3. *Aerodynamics*: This approach combines the fluid mechanics viewpoint with more practical considerations. From the aerodynamic design aspect it proves useful to think in terms of contributions from a variety of aircraft features. This includes effects due to the requirement to trim the aircraft, and interactions between the aerodynamics of the vehicle and both propulsion induced flow effects and structural deformation effects. Within this context, several other considerations are identified. The basic contributions from each component must be included. This leads to a drag analysis based on typical configuration features, as shown below:

- individual component contributions to drag
- base drag
- inlet drag with spillage
- boattail drag
- camber drag
- trim drag
- thrust-drag bookkeeping
- aeroelastic effects on drag

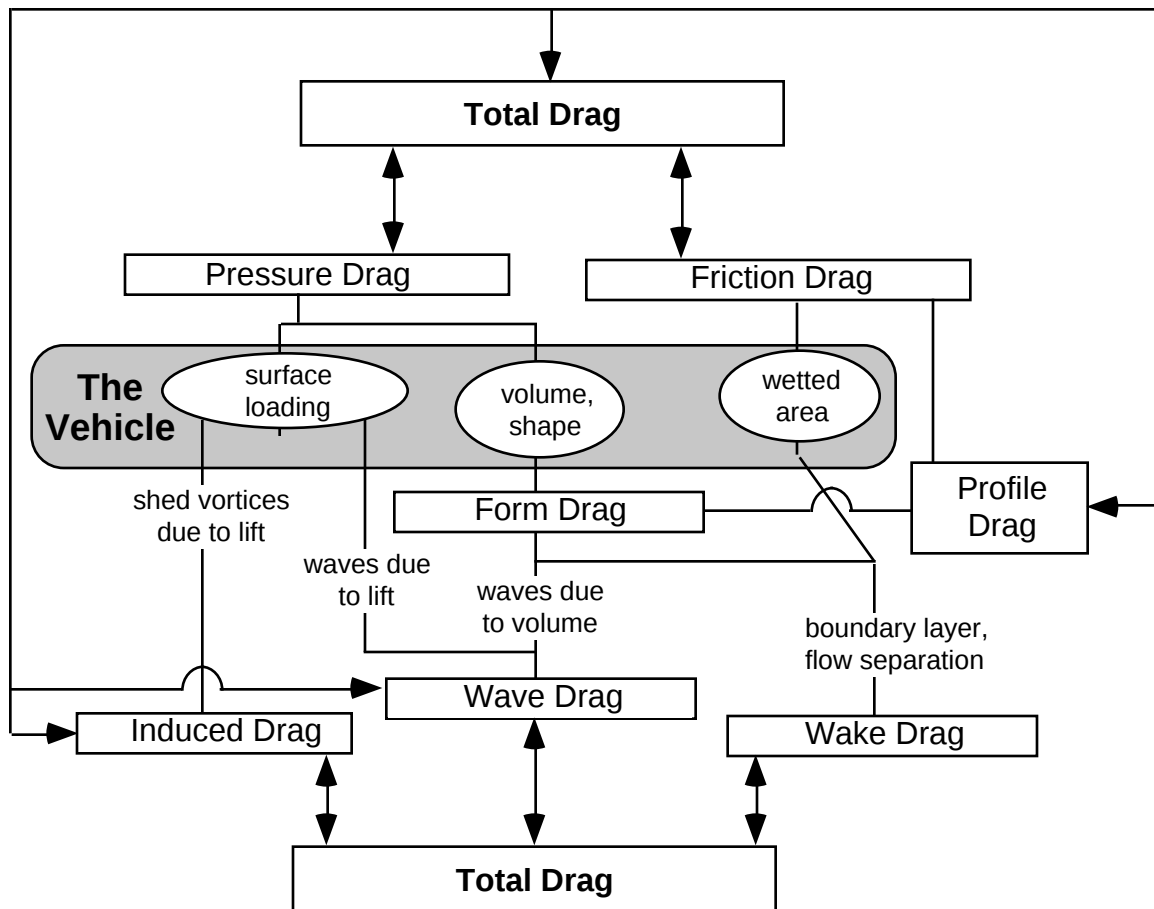


Fig. 5-1. Drag breakdown possibilities (internal flow neglected).

4. *Performance:* To calculate the performance of an airplane it is natural to define drag as the sum of the drag at zero lift and the drag due to lift. This is the approach that leads to the typical drag polar equation:

$$C_D = C_{D_0} + \frac{C_L^2}{\pi A R E} . \quad (5-1)$$

Here each term is a function of Mach number, Reynolds number (in practice this is given to the performance group in terms of Mach number and altitude), and the particular geometric configu-

ration (flap deflection, wing sweep, *etc.*). The drag is not precisely a quadratic function of the lift, and the value of the Oswald efficiency factor, E , in Eq.(5-1) is defined as a function of the lift coefficient and Mach number: $E = E(C_L, M)$. The drag also depends on the throttle setting, but that effect is usually included in the thrust table, as discussed below. There is another drag polar approximation that is seen often. This approximation is more commonly used by aerodynamic designers trying to understand wing performance. It is used to take into account the effect of wing camber and twist, which causes the drag polar to be displaced “upward”, becoming asymmetrical about the $C_L = 0$ axis. It is given as:

$$C_D = C_{D_0} + \Delta C_{D_m} + K(C_L - C_{L_m})^2 \quad (5-2)$$

In taking into account the effect of camber and twist on shifting the polar, the ΔC_{D_m} term represents a penalty associated with using twist and camber to achieve good performance at the design lift coefficient. This equation is for a fixed geometry. Figure 5-2 shows how this looks (ΔC_{D_m} is exaggerated for emphasis). The value of K defines the shape of the polar. C_{D_0} represents the minimum drag of the configuration without camber and twist. The values of ΔC_{D_m} and C_{L_m} are functions of the design lift coefficient. Sometimes novice aerodynamicists fail to include ΔC_{D_m} properly and obtain incorrect values of E when evaluating published drag polars. This type of polar shape will be discussed in more detail later in this chapter. Advanced design concepts such as the X-29 minimize this penalty by defining a device schedule to maximize performance across a broad range of lift coefficients.

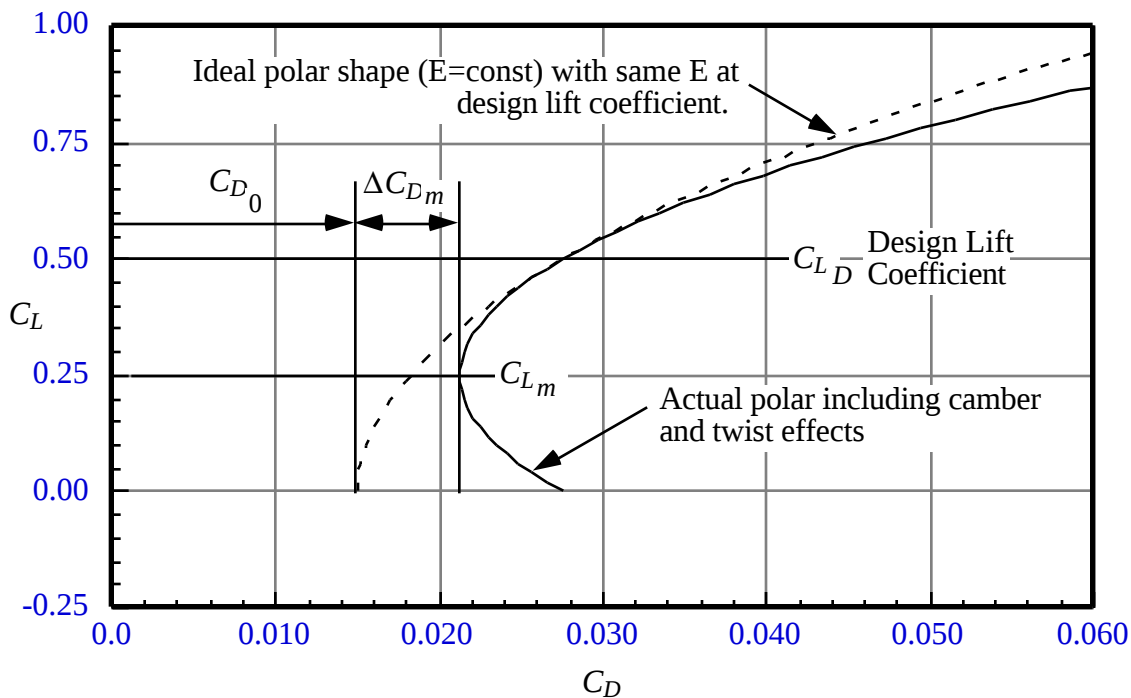


Figure 5-2. Drag polar

As mentioned above, basic drag nomenclature is frequently more confused than it needs to be, and sometimes the nomenclature gets in the way of technical discussions. The chart in Fig. 5-3 provides a basic classification of drag for overview purposes. The aerodynamic configuration-specific approach to drag is not covered in fluid mechanics oriented aerodynamics texts, but is described in aircraft design books. Two other good references are the recent books by Whitford¹¹ and Huenecke.¹² An approach to the evaluation of drag performance, including the efficiency achieved on actual aircraft, was presented by Haines.¹³

We need to define several of these concepts in more detail. The most important overview of aerodynamic drag for design has been given by Küchemann,¹⁴ and should be studied for a complete understanding of drag concepts.

A fluid mechanics refinement: transonic wave drag.

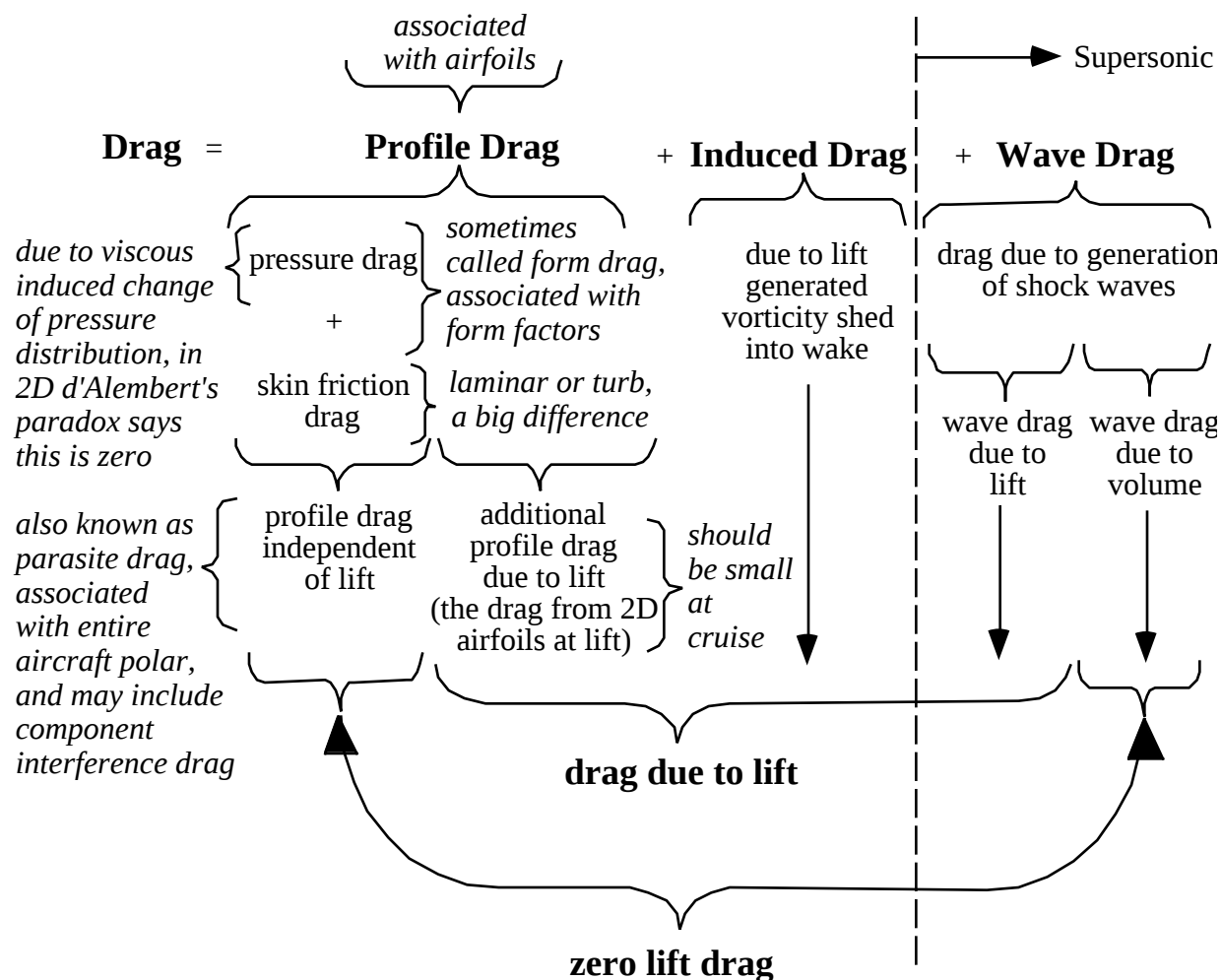
The broadbrush picture of drag presented in Fig. 5.3 suggests that wave drag appears suddenly at supersonic speeds. A more refined examination shows that wave drag arises at subsonic speeds when the flow accelerates locally to supersonic speeds, and then returns to subsonic speed through a shock wave. This leads to the presence of wave drag at subsonic (actually, by definition, transonic) freestream speeds. This initial drag increase, known as *drag rise*, is followed by a rapid increase in drag, and is an important consideration in the design of wings and airfoils. The Mach number at which the rapid drag increase occurs is known as the *drag divergence Mach number*, M_{DD} . The increase in drag occurs directly because of the wave drag associated with the presence of shock waves. However, the drag also increases because the boundary layer thickness increases due to the sudden pressure rise on the surface due to the shock wave, which leads to increased profile drag. Lynch¹⁵ has estimated that at drag divergence the additional transonic drag is approximately evenly divided between the explicit shock drag and the shock induced additional profile drag. Several definitions of the drag rise Mach number are commonly used. The specific definition is usually not important because at drag divergence the drag rises very rapidly and the definitions all result in similar values of M_{DD} .

One standard definition of M_{DD} is the Mach number where

$$\left. \frac{dC_D}{dM} \right|_{C_L = \text{const.}} = 0.1. \quad (5-3)$$

Another definition of drag rise is the Mach number at which

$$\Delta C_D = .0020 \text{ from the subsonic value.} \quad (5-4)$$



Note: A straight surface pressure integration makes it very difficult to separate contributors to the total drag - and this is important in aerodynamic design.

Figure 5-3. A Broadbrush categorization of drag.

Commercial transports fly at or close to M_{DD} , and the drag divergence Mach number is a key part of the performance guarantee. Figure 5.4 (data from Shevell¹⁶) illustrates this refinement to Fig. 5-3, together with the definitions associated with the drag rise. The figure also illustrates a common characteristic, “drag creep,” which occurs with many transonic designs.

An aerodynamics/flight mechanics refinement: trim drag.

A drag not directly related directly to pure fluid mechanics arises from the need to trim the vehicle ($C_m = 0$ about the center of gravity) for steady flight. This requirement can lead to control surface deflections that increase (or decrease) the drag. It can be especially important for supersonic aircraft because of the shift in the aerodynamic center location with Mach number. Other cases with significant trim drag may include configurations with variable wing sweep and the use of airfoils with large values of the zero lift pitching moment about their aerodynamic center. Trim drag details are presented in Section 5.10.

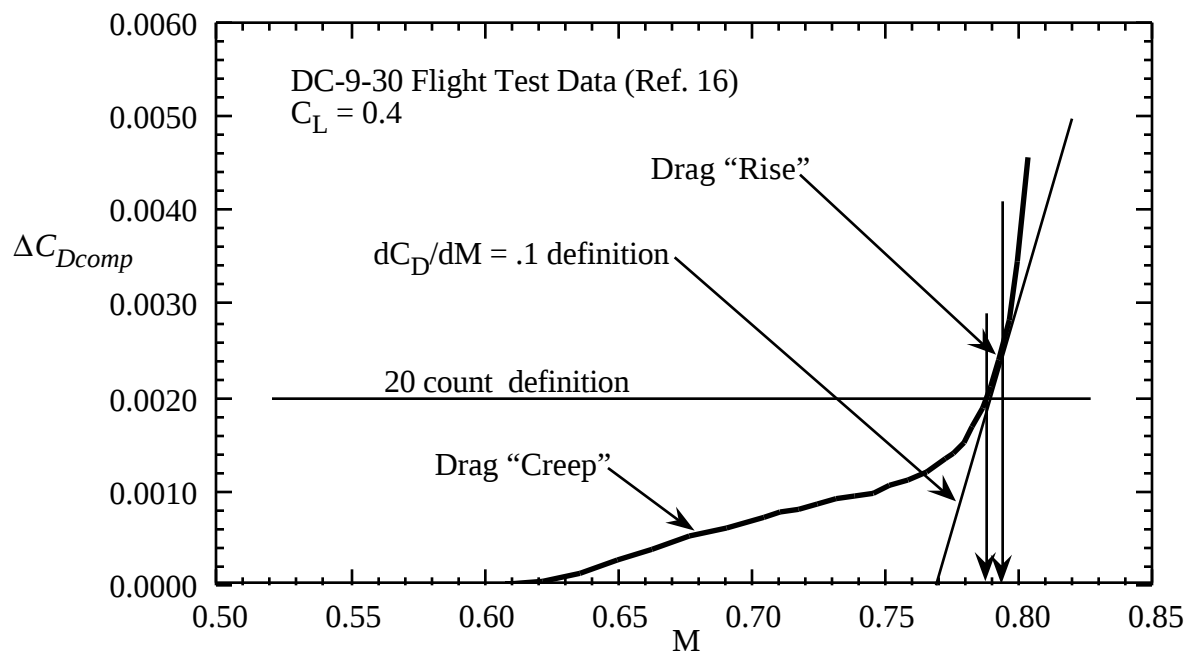


Figure 5.4 Details of wave drag increases at transonic speeds.

A practical aspect of aero-propulsion integration: thrust-drag bookkeeping

To determine aircraft performance, the key value is actually not drag, but the balance between thrust and drag. The drag of the airframe is affected by the operation of the propulsion system, and care must be taken to understand and define these interactions. The amount of air used by the engine defines the size of the streamtube entering the inlet. If all the air in front of the inlet does not enter the inlet, a *spillage drag* will result. Similarly, the boattail drag over the external portion of the nozzle will depend on the nozzle setting in the case of engines with afterburners, and the pressure of the nozzle flow. The definition of a system to properly account for aero-propulsion interactions on the specification of thrust minus drag values is known as thrust-drag bookkeeping. Since thrust is usually provided by the propulsion group, and drag is provided by the aerodynamics group, significant errors in the estimation of aircraft performance have occurred when the necessary coordination and adjustments were not made. The details of this procedure are described in the article by Rooney.¹⁷

Generally, the aerodynamics group provides the performance group with a reference drag polar, and all thrust dependent corrections to the drag polar are accounted for by making adjustments to the thrust values. This is done because it is natural to establish a performance calculation procedure using this approach. The precise details are not important as long as everyone involved in the performance prediction agrees to a specific approach. Usually this requires a specific document defining thrust-drag bookkeeping for each aircraft project.

*Aerodynamic-structural interaction: **aeroelastic effects on drag***

This issue is not strictly a drag consideration, but can make a contribution to the drag if it is not addressed. Aircraft structures deform due to air loads. If the design is centered around a single design point, the aerodynamic shape at the design point can be defined, and the structural analysts will adjust for structural deformation, specifying a “jig shape” that will produce the desired aerodynamic shape at the design point. This is harder to do if there are multiple design points. Deformation of wind tunnel models should also be considered when estimating drag.

5.3 Farfield Drag Analysis

We can estimate the drag on a body most accurately when our predictions methods are not exact by considering the overall momentum balance on a control volume surface well away from the body—a farfield calculation. This is much less sensitive to the detailed calculations of surface pressure and integration of the pressures over the surface to obtain the drag.

The farfield analysis makes use of the momentum theorem. References containing good derivations are by Ashley and Landahl,¹⁸ sections 1.6, 6.6, 7.3 and 9.2, and Heaslet and Lomax,¹⁹ pages 221-229.

For a surface S , which encloses the volume containing an aerodynamic body, the force can be determined by balancing the momentum across S :

$$\mathbf{F} = -\oint_S (p - p_\infty) d\mathbf{S} - \oint_S \rho \mathbf{q} [(\mathbf{V}_\infty + \mathbf{q}) \cdot d\mathbf{S}] \quad (5-5)$$

where $\mathbf{q}$ is the disturbance velocity vector,

$$\mathbf{V} = \mathbf{V}_\infty + \mathbf{q}. \quad (5-6)$$

Define a control volume for use in Eq.(5-5) as shown in Fig. 5-5.

Consider flows far enough away from the body such that linearized flow relations are valid; and use the small disturbance relations:

$$\rho \cong \rho_\infty \left(1 - M_\infty \frac{u}{U_\infty} \right) \quad (5-7)$$

and

$$(p - p_\infty) \cong -\left[U_\infty u + \frac{1}{2}(u^2 + v^2 + w^2) \right] + \frac{1}{2} \rho_\infty M_\infty^2 u^2. \quad (5-8)$$

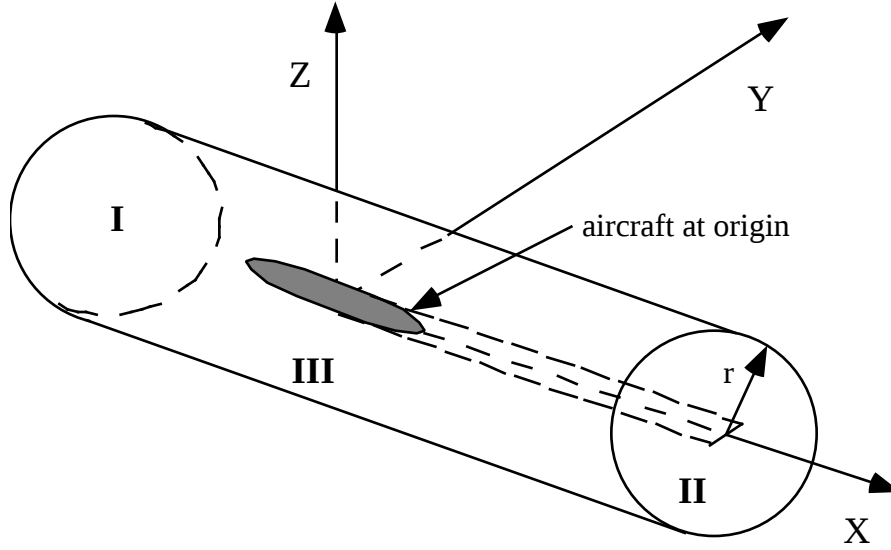


Figure 5-5. Control volume for farfield drag evaluation.

Now, consider the drag component of Eq. (5-5), making use of Eq. (5-7) and Eq. (5-8):

$$D = \frac{1}{2} \rho_{\infty} \iint_{I+II} [(M_{\infty}^2 - 1)u^2 + v^2 + w^2] dy dz - \rho_{\infty} \iint_{III} uv_r r d\theta dx \quad (5-9)$$

and v_r is the radial component, $v_r^2 = v^2 + w^2$, where $r^2 = x^2 + y^2$.

Considering the control volume shown in Fig. 5-5, place **I** and **II** far upstream and downstream and make r large. Then, the integral over **I** is zero as $x \rightarrow -\infty$. The integral over **II** as $x \rightarrow \infty$, corresponds to the so-called Trefftz Plane. The integral over **III** is the wave drag integral, which is zero for subsonic flow, and when any embedded shock waves do not reach **III**.

*Consider the integral over **III***

This is the farfield wave drag integral. This integral corresponds to the last term on the right hand side of Eq. (5-9), and can be written as:

$$D_w = \lim_{r \rightarrow \infty} \left(-\rho_{\infty} r \int_0^{2\pi} d\theta \int_{-\infty}^{+\infty} uv_r dx \right) \quad (5-10)$$

If $u, v_r \rightarrow 0$ as $r \rightarrow \infty$ then $D_w = 0$. Thus, when the flow is subsonic there is no wave drag, as we already know. However, if the flow is supersonic, and shock waves are generated, the inte-

gral is not zero. This integral can be calculated for any numerical solution. In this analysis we assume that the flow is governed by the Prandtl-Glauert equation:

$$(1 - M_\infty^2)\phi_{xx} + \phi_{yy} + \phi_{zz} = 0, \quad (5-11)$$

which implies small disturbance flow. This is valid if the vehicle is highly streamlined, as any supersonic vehicle must be. However, since far from the disturbance this equation will model flows from any vehicle, this is not a significant restriction.

To obtain an expression for ϕ that can be used to calculate the farfield integral, assume that the body can be represented by a distribution of sources on the x -axis (the aircraft looks very “slender” from far away). To illustrate the analysis, assume that the body is axisymmetric. Recall that there are different forms for the subsonic and supersonic source:

$$\begin{aligned} \phi &= -\frac{1}{4\pi} \frac{1}{\sqrt{x^2 + \beta^2 z^2}}, & \phi &= -\frac{1}{2\pi} \frac{1}{\sqrt{x^2 - \beta^2 z^2}} \\ &\text{subsonic source} & &\text{supersonic source} \\ &\downarrow & &\downarrow \\ &\phi \rightarrow 0 \text{ as } r \rightarrow \infty & &\phi \rightarrow 0 \text{ as } r \rightarrow \infty \text{ except} \\ & & &\text{as } r \rightarrow \frac{x}{\beta} \end{aligned} \quad (5-12)$$

This means that the integral will have a contribution along the Mach wave independent of how far away the outer control volume is taken. Figure 5-6 illustrates this effect. The resulting force is exactly what is expected—the shock wave contribution to drag: the wave drag.

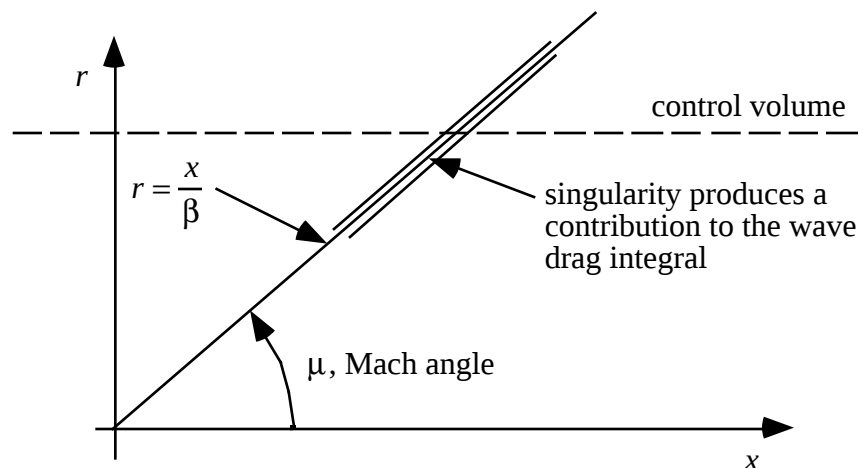


Figure 5-6. Behavior of disturbances along Mach lines in the farfield.

The farfield behavior of the source singularity given in Eq.(5-12) can be used to obtain an expression for the farfield integral in terms of geometric properties of the aircraft. A complete analysis is given in Ashley and Landahl,¹⁸ and Liepman and Roshko.²⁰ The key connection is the assumption relating the supersonic source strength and aircraft geometry. The approximate boundary conditions on the surface equate the change of cross-sectional area to the supersonic source strength: $\sigma(x) = S'(x)$. One required assumption is that the cross-sectional area distribution, $S(x)$, satisfies $S'(0) = S'(l) = 0$. After some algebra the desired relation is obtained:

$$\bar{D}(\theta)_w = -\frac{\rho_\infty U_\infty^2}{4\pi} \int_0^l \int_0^l S''(x_1) S''(x_2) \ln|x_1 - x_2| dx_1 dx_2. \quad (5-13)$$

This is the wave drag integral. The standard method for evaluation of this integral is available in a program known as the “Harris Wave Drag” program.²¹ That program determines the cross-sectional area distribution of the aircraft and then evaluates the integral numerically. Note that as given above, the Mach number doesn’t appear explicitly. A refined analysis¹⁸ for bodies that aren’t extremely slender extends this approach by taking slices, or *Mach cuts*, of the area through the body at the Mach angle. This is how the Mach number dependence enters the analysis. Finally, for non-axisymmetric bodies the area associated with the Mach cuts changes for each angle around the circumferential integral for the cylindrical integration over Region III in Fig. 5-5. Thus the area distribution must be computed for each angle. The total wave drag is then found from

$$D_w = \frac{1}{2\pi} \int_0^{2\pi} \bar{D}_w(\theta) d\theta. \quad (5-14)$$

Examples of the results obtained using this computational method are given in Section 5.7, a discussion of the area rule.

Consider the integral over II

This is the first integral in Eq. (5-9), the induced drag integral:

$$D_i = \frac{1}{2} \rho_\infty \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left[(M_\infty^2 - 1) u^2 + v^2 + w^2 \right] dy dz, \quad (5-15)$$

Note that many supersonic aerodynamicists call this the vortex drag, D_v , since it is associated with the trailing vortex system. However, it is in fact the induced drag. The term vortex drag is

confusing in view of the current use of the term “vortex” to denote effects associated with other vortex flow effects (described in Chapter 6). Far downstream, $u \rightarrow 0$, and we are left with the v and w components of velocity induced by the trailing vortex system. The trailing vortex sheet can be thought of as legs of a horseshoe vortex. Thus the integral becomes:

$$D_l = \frac{1}{2} \rho_\infty \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (v^2 + w^2) dy dz, \quad (5-16)$$

which relates the drag to the kinetic energy of the trailing vortex system.

Now, the flow is governed downstream by the Prandtl-Glauert equation (even if the flow at the vehicle has large disturbances, the perturbations decay downstream):

$$(1 - M_\infty^2) \phi_{xx} + \phi_{yy} + \phi_{zz} = 0 \quad (5-11)$$

and as $x \rightarrow \infty$, $u = 0$, and $u_x = \phi_{xx} = 0$. As a result, the governing equation for the disturbance velocities is Laplace’s Equation for the crossflow velocity:

$$\phi_{yy} + \phi_{zz} = 0. \quad (5-17)$$

An interesting result arises here. The induced drag is explicitly independent of Mach number effects. The analysis is valid for subsonic, transonic and supersonic flows. The Mach number only enters the problem in an indirect manner through the boundary conditions, as we will see.

We now use Green’s Theorem, as discussed previously, to convert the area integral, Eq. (5-16), to a contour integral. Applying the theorem to the drag integral we obtain:

$$\iint_{II} (v^2 + w^2) dS = - \int_c \phi \frac{\partial \phi}{\partial n} dc. \quad (5-18)$$

This is a general relation which converts the integral over the entire cross plane into an integral over the contour. It applies to multiple lifting surfaces. To illustrate the application of the integral to the determination of the induced drag, we consider the special case of a planar lifting surface. Here the contour integral is taken over the surface shown in Fig. 5-7, where the trace of the trailing vortices shed from the wing are contained in the slit from $-b/2$ to $b/2$.

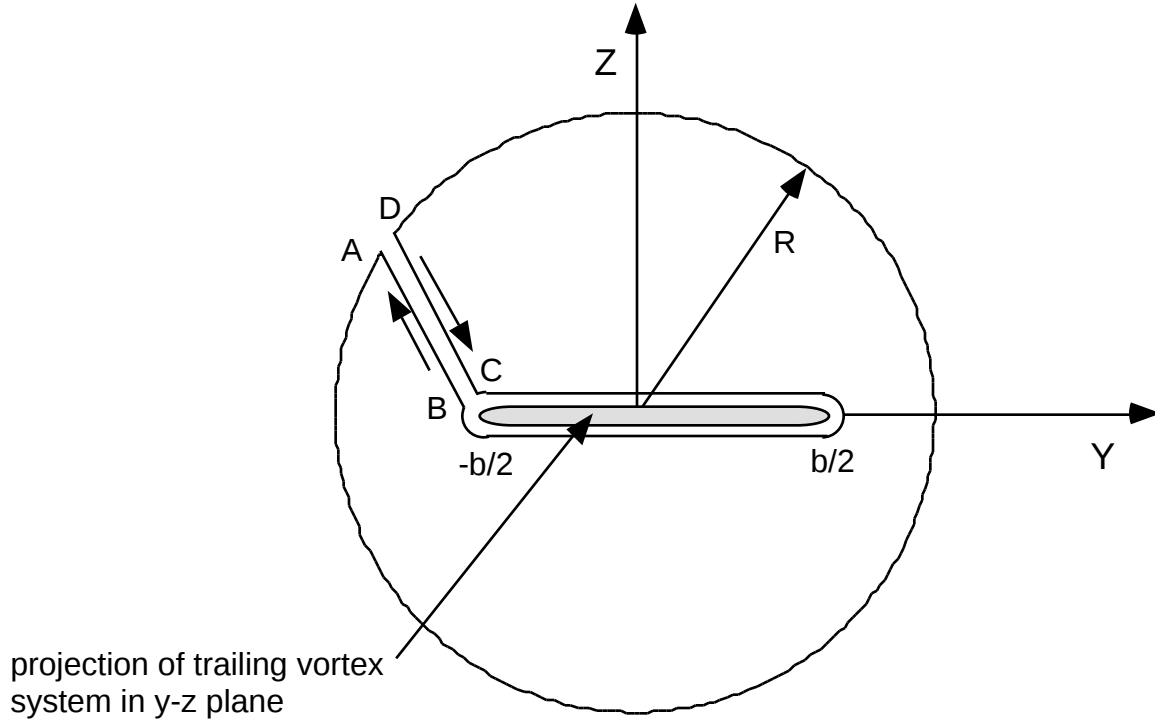


Figure 5-7. Contour integral path for induced drag analysis in the Trefftz plane.

In this *Trefftz plane*, the integral vanishes around the outside contour as $R \rightarrow \infty$ and the integrals along AB and CD cancel. Thus, the only contribution comes from the slit containing the trace of vorticity shed from the wing. The value of ϕ is equal and opposite above and below the vortex sheet, and on the sheet $\partial\phi/\partial n = w$, the downwash velocity.

Thus the integral for a single flat lifting surface can be rewritten as:

$$D_i = -\frac{1}{2} \rho_\infty \int_{-b/2}^{b/2} (\Delta\phi)_{x=\infty} w_{x=\infty} dy \quad (5-19)$$

and w is the velocity induced by the trailing vortex system. The jump in the potential on the slit at infinity can be related to the jump in potential at the trailing edge. To see this, first consider the jump in the potential at the trailing edge. Recall that the circulation is given by the contour integral:

$$\Gamma = \oint \mathbf{V} \cdot d\mathbf{s}. \quad (5-20)$$

For an airfoil we illustrate the concept by considering a small disturbance based argument. However, the results hold regardless of the small disturbance based illustration. Consider the airfoil given in Fig. 5-8.

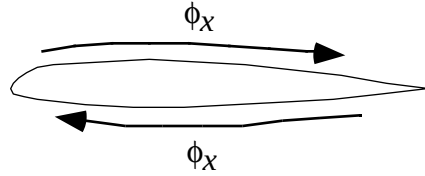


Figure 5-8. Integration path around an airfoil.

The dominant velocity is in the x -direction, $u = \phi_x$, and the integral, Eq. (5-20), around the airfoil can be seen to be essentially:

$$\begin{aligned}
 \Gamma &= \int_{TE_{lower}}^{LE} \phi_x dx + \int_{LE}^{TE_{upper}} \phi_x dx \\
 &= \phi|_{TE_{lower}}^{LE} + \phi|_{LE}^{TE_{upper}} \\
 &= \phi_{LE} - \phi_{TE_{lower}} + \phi_{TE_{upper}} - \phi_{LE} \\
 &= \phi_{TE_{upper}} - \phi_{TE_{lower}} \\
 &= \Delta\phi_{TE}
 \end{aligned} \tag{5-21}$$

The value of the potential jump at infinity can be found by realizing that the circulation is created by the wing, and any increase in the contour of integration will produce the same result.

Therefore,

$$\Delta\phi_{x=\infty} = \Delta\phi_{TE} = \Gamma(y) \tag{5-22}$$

Next, the induced velocity is found from the distribution of vorticity in the trailing vortex sheet. Considering the slit to be a sheet of vorticity, we can find the velocity induced by a distribution of vorticity from the following integral, which is a specialized case of the relation given in Chap.4, Eq.(4-42):

$$w_{x=\infty}(y) = \frac{1}{2\pi} \int_{-b/2}^{b/2} \frac{\gamma(\eta)}{y-\eta} d\eta \tag{5-23}$$

To complete the derivation we have to connect the distribution of vorticity in the trailing vortex sheet to the circulation on the wing. To do this consider the sketch of the circulation distribution given in Fig. 5-9.

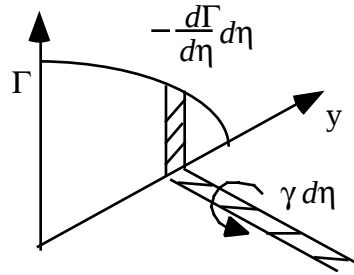


Figure 5-9. Relation between circulation change on the wing and vorticity in the wake.

As the circulation on the wing, Γ , changes across the span, circulation is conserved by shedding an amount equal to the local change into the wake. Thus the trailing vorticity strength is related to the change in circulation on the wing by

$$\gamma(\eta) = - d\Gamma/dy. \quad (5-24)$$

Substituting this into Eq. (5-23), we obtain:

$$w_{x=\infty} = -\frac{1}{2\pi} \int_{-b/2}^{b/2} \frac{d\Gamma/dy}{y-\eta} d\eta. \quad (5-25)$$

Substituting Eq. (5-22) and (5-25) into Eq.(5-19) and integrating by parts using the conditions that $\Gamma(-b/2) = \Gamma(b/2) = 0$ (which simply states that the load distribution drops to zero at the tip), we get:

$$D_i = -\frac{\rho_\infty}{4\pi} \int_{-b/2}^{b/2} \int_{-b/2}^{b/2} \frac{d\Gamma(y_1)}{dy} \frac{d\Gamma(y_2)}{dy} \ln|y_1 - y_2| dy_1 dy_2. \quad (5-26)$$

Note that this is the same form as the wave drag integral, where the area distribution is the key contributor to the wave drag, but here the spanload distribution is responsible for the induced drag. Because of the double integral we can get the total drag, but we have lost the ability to get detailed distributions of the induced drag on the body (or in the case of wave drag, its distribution on the surface). This is the price we pay to use the farfield analysis.

Finally, this result shows that the induced drag is a function of the Γ distribution (spanload) alone. Mach number effects enter only in so far as they affect the circulation distribution on the wing.

5.4 Induced Drag

Although the inviscid flow over a two-dimensional airfoil produces no drag, as we've just seen in Chapter 4, this is not true in three dimensions. The three-dimensional flowfield over a lifting surface (for which a horseshoe vortex system is a very good conceptual model) does result in a drag force, even if the flow is inviscid. This is due to the effective change in the angle of attack along the wing induced by the trailing vortex system. This induced change of angle results in a local inclination of the force vector relative to the freestream, and produces an induced drag. It is one part of the total drag due to lift, and is typically written as:

$$C_{D_i} = \frac{C_L^2}{\pi A R e}. \quad (5-27)$$

The small “e” in this equation is known as the span e . As we will show below, the induced drag is only a function of the spanload. Additional losses due to the fuselage and viscous effects are included when a capital E , known as Oswald's E , is used in this expression. Note that although this notation is the most prevalent in use in the US aircraft industry, other notations are frequently employed, and care must be taken when reading the literature to make sure that you understand the notation used.

When designing and evaluating wings, the question becomes: what is “e”, and how large can we make it? The “conventional wisdom” is that for a planar surface, $e_{max} = 1$, and for a non-planar surface or a combination of lifting surfaces, $e_{max} > 1$, where the aspect ratio, AR , is based on the projected span of the wing with the largest span.* However, studies searching for higher e 's abound. The quest of the aerodynamicist is to find a fundamental way to increase aerodynamic efficiency. In the '70s, increased aerodynamic efficiency, e , was sought by exploiting non-planar surface concepts such as winglets and canard configurations. Indeed, these concepts are now commonly employed on new configurations. In the '80s, a great deal of attention was devoted to the use of advanced wing tip shapes on nominally planar configurations. It is not clear however that the advanced wingtips result in theoretical e 's above unity. However, in practice these improved tip shapes help clean up the flowfield at the wing tip, reducing viscous effects and resulting in a reduction in drag.

To establish a technical basis for understanding the drag due to lift of wings, singly and in combination, three concepts must be discussed: farfield drag (the Trefftz plane), Munk's Stagger Theorem for design of multiple lifting surfaces, and, to understand additional drag above the induced drag due to “e,” it is appropriate to introduce the concept of leading edge suction. Here we will discuss the induced drag. Subsequent sections address Munk's Stagger Theorem (Section 5.6) and leading edge suction (Section 5.9)

* However, e is not too much bigger than unity for practical configurations.

In the last section we derived the expression for the drag due to the trailing vortex system. The far downstream location of this face of the control volume is known as the Trefftz plane. Here we explain the physical basis of the idea of the Trefftz plane following Ashley and Landahl¹⁸ almost verbatim. An alternate and valuable procedure has been described by Sears.²²

The Trefftz Plane

The idea:

1. Far downstream the motion produced by the trailing vortices becomes 2D in the y - z plane (no induced velocity in the x -direction).
2. For a wing moving at a speed U_∞ through the fluid at rest, the amount of mechanical work $D_i U_\infty$ is done on the fluid per unit time. Since the fluid is nondissipative (potential flow), it can store energy in kinetic form only. Therefore, the work $D_i U_\infty$ must show up as the value of kinetic energy contained in a length U_∞ of the distant wake.

and:

3. The vortices in the trailing vortex system far downstream can be used to find the induced drag.

The Trefftz Plane is a y - z plane far downstream, so that all motion is in the crossflow plane (y - z), and no velocity is induced in the x -direction, $u = U_\infty$. For a single planar lifting surface, the expression for drag was found to be:

$$D_i = -\frac{\rho_\infty}{4\pi} \int_{-b/2}^{b/2} \int_{-b/2}^{b/2} \frac{d\Gamma(y_1)}{dy} \frac{d\Gamma(y_2)}{dy} \ln|y_1 - y_2| dy_1 dy_2 \quad (5-26)$$

The usual means of evaluating the induced drag integral is to represent Γ as a Fourier Series,

$$\Gamma = U_\infty b \sum_{n=1}^{\infty} A_n \sin n\theta \quad (5-28)$$

The unknown values of the A_n 's are found from a Fourier series analysis, where $\Gamma(y)$ is known from an analysis of the configuration. Panel or vortex lattice methods can be used to find $\Gamma(y)$. Vortex lattice methods are described next in Chapter 6. Integration of the drag integral with this form of Γ results in:

$$D_i = \frac{\pi \rho_\infty U_\infty^2 b^2}{8} \sum_{n=1}^{\infty} n A_n^2 \quad (5-29)$$

and

$$L = \frac{\pi}{4} \rho_\infty U_\infty^2 b^2 A_1 \quad (5-30)$$

which are the classical results frequently derived using lifting line theory. Note that the lift depends on the first term of the series, whereas all of the components contribute to the drag. Putting the expressions for lift and drag into coefficient form, and then replacing the A_1 term in the drag integral by its definition in terms of the lift coefficient leads to the classical result:

$$C_{D_i} = \frac{C_L^2}{\pi A R e} \quad (5-31)$$

where:

$$e = \frac{1}{1 + \sum_{n=2}^{\infty} n \left(\frac{A_n}{A_1} \right)^2} \quad (5-32)$$

These expressions show that $e_{\max} = 1$ for a planar lifting surface. However, if the slit representing the trailing vortex system is not a simple flat surface, and C_{D_i} is based on the projected span, a nonplanar or multiple lifting surface system can result in values of $e > 1$. In particular, biplane theory addresses the multiple lifting surface case, see Thwaites²³ for a detailed discussion. If the wing is twisted, and the shape of the spanload changes as the lift changes, then e is not a constant, independent of the lift coefficient.

It is important to understand that the induced drag contribution to the drag due to lift assumes that the airfoil sections in the wing are operating perfectly, as if in a two-dimensional potential flow that has been reoriented relative to the freestream velocity at the angle associated with the effects of the trailing vortex system. Wings can be designed to operate very close to these conditions.

We conclude from this discussion:

1. Regardless of the wing planform(s), induced drag is a function of circulation distribution alone, independent of Mach number except in the manner which Mach number influences the circulation distribution (a minor effect in subsonic/transonic flow).
2. Given Γ , “ e ” can be determined by finding the A_n ’s of the Fourier series for the simple planar wing case. Other methods are required for nonplanar systems.
3. Extra drag due to the *airfoil’s* inability to create lift ideally must be added *over and above* the induced drag (our analysis here assumes that the airfoils operate perfectly in a two-dimensional sense; there is no drag due to lift in two-dimensional flow).

5.5 Program LIDRAG

For single planar surfaces, a simple Fourier analysis of the spanload to determine the “ e ” using a Fast Fourier Transform is available from the code **LIDRAG**. The user’s manual is given in Appendix D.3. Numerous other methods could be used. For reference, note that the “ e ” for an elliptic spanload is 1.0, and the “ e ” for a triangular spanload is 0.728. **LIDRAG** was written by Dave Ives, and is employed in numerous aerodynamics codes.²⁴

5.6 Multiple Lifting Surfaces and Munk's Stagger Theorem

An important result in the consideration of multiple lifting surfaces is Munk’s Stagger Theorem.²⁵ It states that the *total* induced drag of a multi-surface system does not change when the elements of the system are translated parallel to the direction of the flow, as illustrated in the sketch shown in Fig. 5-10, *provided* that the circulation distributions on the elements are left unchanged. This theorem is proven in the text by Milne-Thompson.²⁵ Thus the drag depends only on the projection of the system in the cross-plane. This means that given the circulation distributions, the Trefftz plane analysis can be used to find the induced drag. This is consistent with the analysis given for the Trefftz plane above, and reinforces the concept of using the farfield analysis to determine the induced drag. Naturally, to maintain the circulation distribution of the elements when they are repositioned their geometric incidence and twist have to be changed.

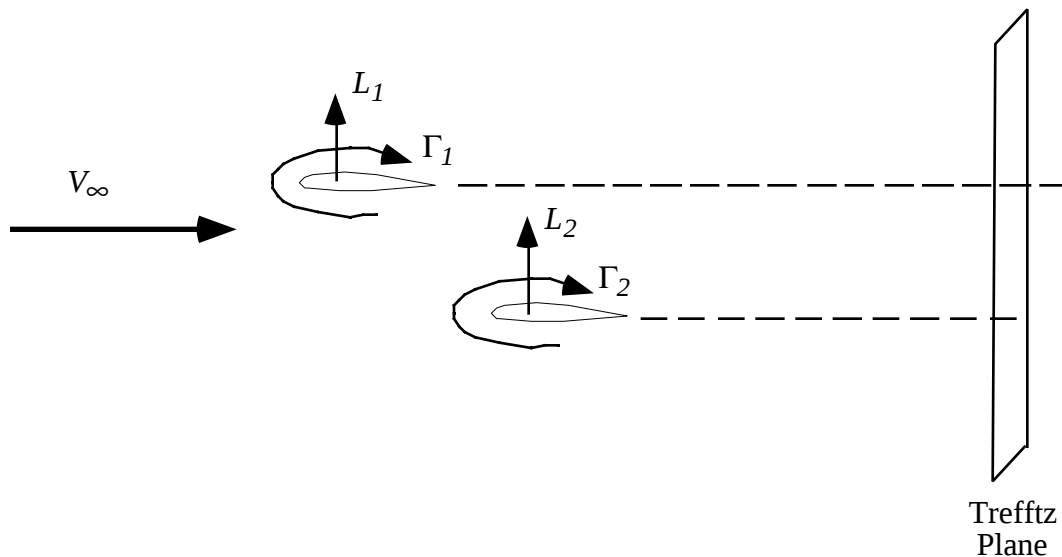


Fig. 5-10. Example of Munk’s Stagger Theorem, where the fore and aft positions of multiple lifting surfaces do not affect drag as long as the circulation distribution remains fixed.

When the lifting system is not limited to a single lifting component, **LIDRAG** cannot be used to find the span e . However, two limiting cases can be considered. If the lifting elements are in the same plane, then the sum of the spanloads should be elliptic for minimum drag. If the elements are vertically separated by a large distance, then each component individually should have an elliptic spanload to obtain minimum induced drag.

When the system is composed of two lifting surfaces, or a lifting surface with dihedral breaks, including winglets, then a code by John Lamar²⁶ is available to analyze the induced drag. As originally developed, this code finds the minimum induced drag and the required spanloads for a prescribed lift and pitching moment constraint. It is known as **LAMDES**, and the user's manual is given in Appendix D.4. This program is much more elaborate than **LIDRAG**. For subsonic flow the program will also estimate the camber and twist of the lifting surfaces required to achieve the minimum drag spanload. I extended this code to incorporate, approximately, the effects of viscosity and find the system e for a user supplied spanload distribution.²⁷

5.7 Zero Lift Drag Friction and Form Drag Estimation

Although not formally part of computational aerodynamics, estimates of skin friction based on classical flat plate skin friction formulas can be used to provide initial estimates of the friction and form drag portion of the zero lift drag. These are required for aerodynamic design studies using the rest of the methods described here. These simple formulas are used in conceptual design in place of detailed boundary layer calculations, and provide good initial estimates until more detailed calculations using the boundary layer methods described in Chapter 10 are made. They are included here because they appear to have been omitted from current basic aerodynamics text books.* An excellent examination of the methods and accuracy of the approach described here was given by Paterson, MacWilkinson and Blackerby of Lockheed.²⁸

For a highly streamlined, aerodynamically clean shape the zero lift drag (friction and form drag at subsonic speeds where there are no shock waves) should be mostly due to these contributions, and can be estimated using skin friction formulas. However, Table 5-1, for a typical military attack airplane, shows that on this airplane only about two-thirds of the zero lift drag is associated with skin friction and form drag. This illustrates the serious performance penalties associated with seemingly small details. R.T. Jones²⁹ has presented a striking figure, included here as Fig. 5-11, comparing the drag on a modern airfoil to that of a single wire. It's hard to believe, and demonstrates the importance of streamlining. An accurate drag estimate requires that these details be included in the estimates.

* Expanded details including compressibility effects and mixed laminar-turbulent skin friction estimates are given in App. D.5, FRICTION.

Fig. 5-11. A wire and airfoil with the same drag!²⁹

Until recently, aerodynamicists assumed the flow was completely turbulent. However, as a result of work at NASA over the last decade and a half, some configurations can now take advantage of at least some laminar flow, with its significant reduction in friction drag. Advanced airfoils can have as much as 30 to 40% laminar flow.

As an example of this approach, consider a typical turbulent flow skin friction formula (for one side of a “flat plate” surface only):

$$C_F = \frac{1.455}{[\log \text{Re}_c]^{2.58}} \quad (5-33)$$

where “log” means log to the base 10. Note also that the capital C_F denotes an integrated value. Formulas for the local skin friction coefficient customarily use a small f subscript.

Numerous form factors are available to help account for effects due to thickness and additional trailing edge pressure drag. Hoerner⁹ and Covert⁸ provide summaries. For planar surfaces, one form factor is,

$$FF = 1 + 1.8 \left(\frac{t}{c} \right) + 50 \left(\frac{t}{c} \right)^4 \quad (5-34)$$

where t/c is the maximum thickness to chord ratio. For bodies, the form factor would be:

$$FF = 1 + 1.5 \left(\frac{d}{l} \right)^{1.5} + 7 \left(\frac{d}{l} \right)^3 \quad (5-35)$$

where d/l is the diameter to length ratio. The skin friction coefficient estimate is then converted to aircraft coefficient form through:

$$C_{D0} \cong C_F \frac{S_{wet}}{S_{ref}} FF \quad (5-36)$$

Here S_{wet} is the total area scrubbed by the flow, and S_{ref} is the reference area used in the definition of the force coefficients. For a thin wing the reference area is usually the planform area and the wetted area is approximately twice the planform area (including the upper and lower surface of the wing).

Program **FRICITION** automates this procedure using slightly improved formulas for the skin friction that include compressibility effects. The program computes the skin friction and form drag over each component, including laminar and turbulent flow. The user can input either the Mach and Reynolds numbers or the Mach number and altitude. The use of this program is described in Appendix D.5. This analysis assumes that the aircraft is highly streamlined. For many aircraft this is not the case. As discussed above, Table 5-1 provides an example of the significantly increased drag that results when developing an aircraft for operational use.

Comment: On a tour of the final assembly lines of the Boeing 747 and 777 on February 29, 1996, I observed that the 777 was much, much smoother aerodynamically than the 747. Clearly, a lot of the advanced performance of the 777 is due to old-fashioned attention to detail. The aerodynamicists have apparently finally convinced the manufacturing engineers of the importance of aerodynamic cleanliness. Think about this the next time you compare a Cessna 182 to the modern homebuilts, as exemplified by the Lancairs and Glassairs.

More details are presented in Chapter 10, Viscous Flows in Aerodynamics. Viscous effects due to lift and shock-wave boundary layer interaction are also discussed in Chapter 11, Transonic Aerodynamics.

Table 5-1
Example of zero lift drag buildup on a “dirty” military airplane.

Low Speed Minimum Parasite Drag Breakdown						
$M < .65, C_L = 0.0$						
Component	S_{wet}	S_π	C_{Df}	$C_{D\pi}$	ΔC_D	% Total
1 Wing						22.1%
a) not affected by slats	262.		.00308		.00308	
b) not affected by slats	150.		.00280		.00162	
2. Horizontal Tail	84.4		.0033		.00108	5.1%
3. Vertical Tail	117.		.00385		.00173	8.1%
4. Fuselage (including inlets)	434.		.00306		.00512	24.0%
5. Enclosure		2.3		.122	.00108	5.1%
6. Appendages						33.1%
a) Upper avionics bay					.00069	
b) Drag-chute fairing					.00012	
c) Landing gear fairings					.00042	
d) Aero 7A Rack-Pylon @ CL					.00058	
e) Arresting hook					.00058	
f) Inflight-Fueling Probe					.00092	
g) Wing-Vortex Generators					.00115	
h) Boundary Layer Diverter					.00042	
i) Boundary-Layer Splitter Plate					.00004	
j) Inlet Vortex Fences					.00023	
k) Landing Spoilers					.00012	
l) ECM Antenna and Chaff Dispensers					.00038	
m) Pitot tube					.00004	
n) Angle-of-Attack Indicator					.00004	
o) Rudder Damper					.00023	
p) Aileron Damper					.00023	
q) Barrier Detents					.00008	
r) Anti-Collision Lights					.00008	
s) Radar altimeter					.00015	
t) Fuel Dump and Vent					.00023	
u) Airblast Rain Removal					.00008	
v) Catapult Holdback					.00027	
7. Inlets and Exits						
a) Powerplant (vents, etc.)					.00027	1.6%
b) Air Conditioning					.00008	
8. Miscellaneous					.00020	.9%
Total Zero lift drag coefficient (based on $S_{ref} = 260 \text{ ft}^2$)					.0213	100.0%
Note: based on a total wetted area of 1119 ft^2 , $C_D = .00495$						

5.8 Supersonic Wave Drag: The Farfield Wave Drag Integral and the Area Rule

The farfield analysis also showed us that for supersonic flight there is a wave drag. Not surprisingly, the supersonic wave drag has played a key role in the aerodynamic design of supersonic aircraft. The equations are repeated here as:

$$\bar{D}_w(\theta) = -\frac{\rho_\infty U_\infty^2}{4\pi} \int_0^l \int_0^l S''(x_1) S''(x_2) \ln|x_1 - x_2| dx_1 dx_2 \quad (5-12)$$

and

$$D_w = \frac{1}{2\pi} \int_0^{2\pi} \bar{D}_w(\theta) d\theta \quad (5-13)$$

where the $S(x)$ values represent the area from an oblique (Mach angle) cut to find the cross section area of the aircraft at a specific theta.

The importance of the distribution of the cross-sectional area is clear in the integral. To minimize the integral the area change should be very smooth. Thus, the shaping of the design geometry plays a major role in the value of the integral. In any case, low drag is achieved by minimizing the maximum cross-sectional area of the design. The key parameter is the fineness ratio, which is the length divided by the maximum diameter. Increasing the fineness ratio decreases the wave drag. A number of minimum drag bodies of revolution have been derived using Eq. (5-12). The geometric details of these shapes are given in Appendix A.

The principle that aerodynamicists use to achieve low values of wave drag is known as the *area rule*. Proposed by Richard Whitcomb* at the NACA's Langley Field, the area rule states that the air displaced by the body should develop in a smooth fashion as it moves around and along the body, with no sudden discontinuities. Thus the total aircraft area distribution should form a smooth progression. In particular, when the wing becomes part of the cross-sectional area, the adjacent fuselage area should be reduced to make the total area distribution smooth. This results in the distinctive area ruled, or "coke bottle," fuselage shape.

Whitcomb's evidence for the validity of this rule was obtained experimentally (the computer had not yet become practical design tool). Figure 5-12 shows the key result obtained by Whitcomb.³⁰ The increase in drag with increasing transonic Mach number is almost identical for a wing-body combination and a body of revolution with the same cross sectional area distribution. The wing-body combination has significantly higher subsonic drag because of the increased surface area compared to the body alone case. All the cases Whitcomb presented weren't as dramatic, but similar trends were found for a number of shapes. Whitcomb's original idea addressed

* He won the Collier trophy for this work.

transonic speeds, and the normal area distribution (the area in the plane perpendicular to the flow) was made smooth to obtain low drag. At supersonic speeds the problem is more complicated. Instead of using the normal area distribution, the supersonic area rule requires that the area on the so-called Mach cuts that correspond to the area distribution along the Mach angle for each theta angle (Eq. 5-13) be smooth.

Figure 5-12. Whitcomb's proof of the area rule.³⁰

The most famous application of the area rule occurred on the F-102 aircraft program.* This airplane was supposed to be supersonic in level flight. When it first flew, the prototype YF-102 was unable to break the sound barrier and fly supersonically. The nose was lengthened approximately five feet and area was added (with the plane already completed it was impossible to remove area) to the fuselage via faired bulges—or “bustles”—at the wing trailing edge-fuselage intersection. The bulges were faired beyond the engine exhaust nozzle to improve the fineness ratio and area distribution. After these modifications, the prototype YF-102 was capable of penetrating deeper into the transonic region. However, it was still not capable of exceeding Mach 1.0 in level flight. A complete redesign was necessary. It had to be done to continue the contract.

* Portions of this section were contributed by Nathan Kirschbaum.

One hundred and seventeen working days later(!), a new, completely redesigned F-102 was ready to fly. The fuselage fineness ratio and area distribution had been increased and refined. The fuselage mid-section cross-sectional area had been reduced (cinched-up, wasp waisted, or coke-bottled) as much as structure and component integration would permit. It was lengthened 11 feet 3 inches, with most of the increased length added ahead of the wing. The cockpit canopy was reduced in cross-section with a near triangular cross-section and headed by a flat plate, highly swept “V” windshield. The cockpit and the side-mounted engine inlets were moved forward to reduce their sudden area build-up, or impact on the fuselage area. The aft fuselage bustles were retained to avoid the rapid collapse of the cross-sectional area at the delta wing trailing edge. So reconfigured, the airplane was able to fly at low supersonic speeds ($M = 1.2$). Figure 5-13 shows the original prototype and the reconfigured F-102A as produced for service use.³¹ The resulting change in drag from the YF-102 to the F-102A was about twenty-five counts, and is shown in Fig. 5-14 (from the original Convair plot). Although the change might not appear dramatic, the reduction in wave drag was sufficient to allow the plane to fly faster than the speed of sound. Notice also that the use of conical camber (discussed later), introduced to improve the lift and drag due to lift characteristics of the delta wing, added a significant penalty (camber drag) to the minimum drag.

Subsequently, the configuration was completely redesigned incorporating a more refined, integrated area rule. Further slimmed down by a reduced weapon bay capacity and shortened and repositioned engine air intake ducts, and powered by a fifty percent more powerful engine, it was capable of routine Mach 2+ speeds. The designation was then changed to F-106A. This design is also shown in Fig. 5-13. The volume of the increased area of the vertical tail on the F-106A, required to counteract the loss of tail surface effectiveness at the increased operational Mach number, replaced the aft side “bustles” on the F-102.

As an historical note, the Grumman F-11F (F-11) was the first aircraft designed “from scratch” using the area rule. The result is clearly evident as shown in Fig. 5-15a.³² Another design employing the area rule in an effective manner was the Northrop F-5A/B (and the T-38 derivative), as shown in Fig. 5-15b.³² This design had essentially unswept wings. Even the wing tip fuel tanks were area ruled, although the inboard localized area reduction could be arguably assigned to Küchemann interface contour theorems.¹⁴

When considering the area rule, remember that this is only one part of successful airplane design.³³ Moreover, extreme area ruling for a specific Mach number may significantly degrade the performance of the design at other Mach numbers.

Figure 5-13 Convair YF-102, F-102A, F-106A configuration evolution.³¹

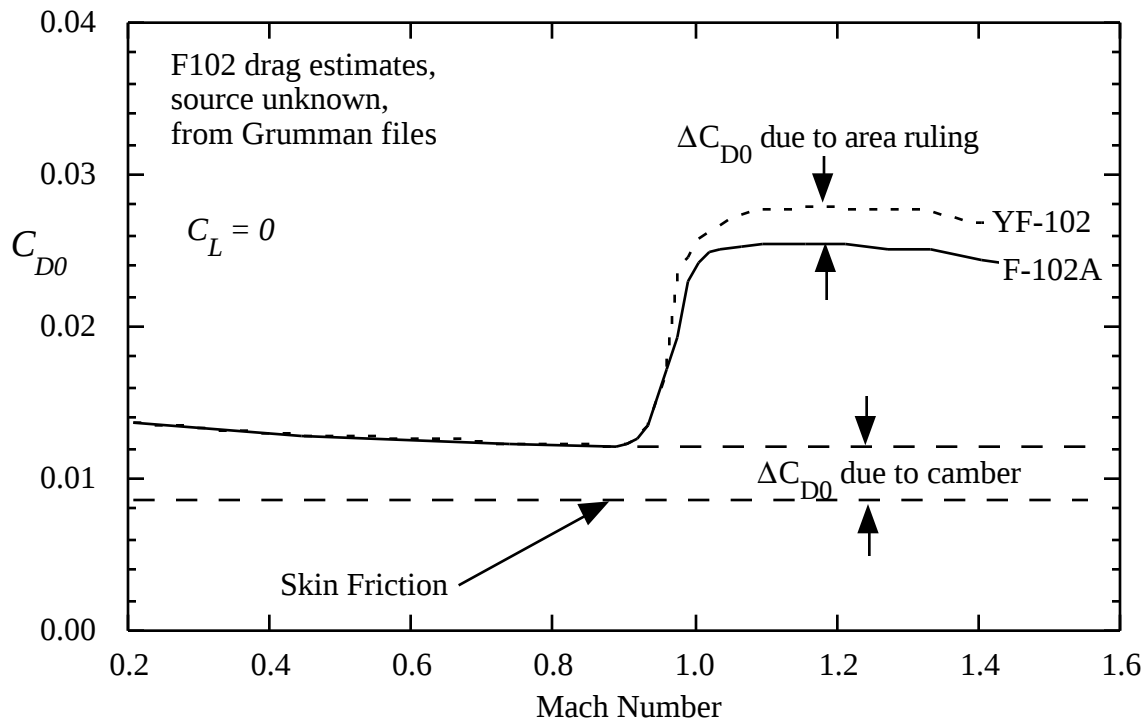


Figure 5-14. Zero lift drag for the YF-102 and F-102A airplanes.

To estimate the wave drag, a theoretical analysis of the integral is available.* Note that the integrand is proportional to the second derivative of the area distribution, so that even without an analysis it is clear that the lowest drag occurs when the distribution is made as smooth as possible. Eminton³⁴ devised the standard method for the numerical evaluation of the integral in Eqn. 5-12. The difficulty in evaluating the integral is that the result depends on the second derivative of the area distribution. This distribution is made up of contributions from numerous components, and it is not known with great precision. Polynomials or other interpolation schemes used to perform the quadrature may amplify any imprecision in the data, and produce unreasonably high drag predictions. Ms. Eminton used a Fourier series for the distribution of the gradient of the area. The coefficients are then found by solving an optimization problem that determines the coefficients that will produce the curve passing through the known values of the area having the least drag. In this sense the method is also a design method. By specifying a small number of control stations (say, from a designer's configuration layout) with a specified area distribution, the method will provide the complete distribution of area required for minimum drag and satisfying the imposed control station constraints.

* Note: advanced CFD calculation methods don't require the aerodynamicist to look at the problem using the area rule diagram. Those approaches don't provide the insight for design available through the area rule diagram.

a) Grumman F-11F

b) Northrop F-5A/B

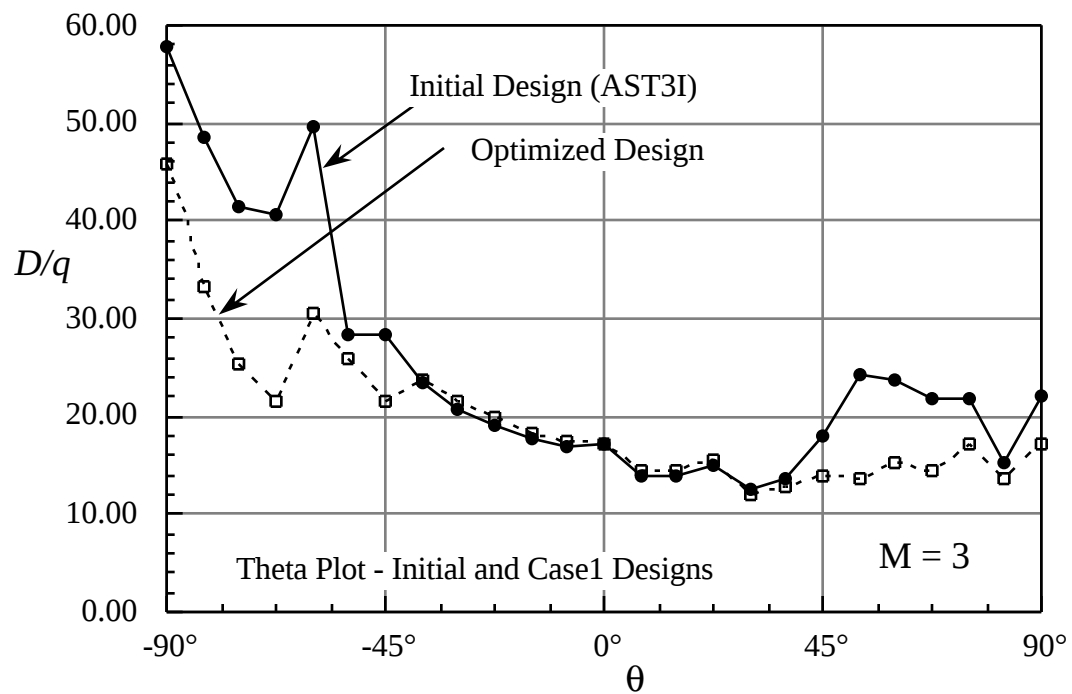
Figure 15. Other aircraft designs with evident area ruling.

The practical implementation of this scheme is available in the so-called Harris wave drag program.²¹ Figure 5-16 illustrates the procedure. At each “roll angle” θ a number of x -cuts are made to use in evaluating the integral. Typically, 50 to 100 x -cuts are made for each of from 24 to 36 θ values. Note that in making these calculations the inlet capture area is removed from the area distribution.

As discussed above, area ruling plays an important role in supersonic cruise vehicle design. Figure 5-17 presents the results of an analysis of a current high speed civil transport (HSCT) concept.³⁵ Figure 5-17a shows the highly blended configuration. Figure 5-17b shows the variation in drag as the integral is computed for various “theta cuts.” This curve also contains the results of a combined structural-aerodynamic study to improve this design using systematic advanced design methodology.³⁶ Note that the drag is presented in terms of D/q . This is a traditional approach, and eliminates any false impressions produced when configurations with differing reference areas are compared. Figure 5-17c shows the normal area distribution. Here the nacelles are seen to make a large impact on the area distribution. However, the area distribution of interest is for $M = 3.0$. Figures 5-17d and e present the area distributions for the theta 0° and 90° cases. Here the area distribution is seen to be much smoother. This is especially true for the theta 0° case. The theta 90° case still shows the problem of integrating the propulsion system into the configuration to obtain a smooth area distribution. Comparing the area distributions presented in Figures 5-17d and 17e with the change in drag at these two different roll angles provides some insight into the importance of shaping to produce a smooth area distribution.

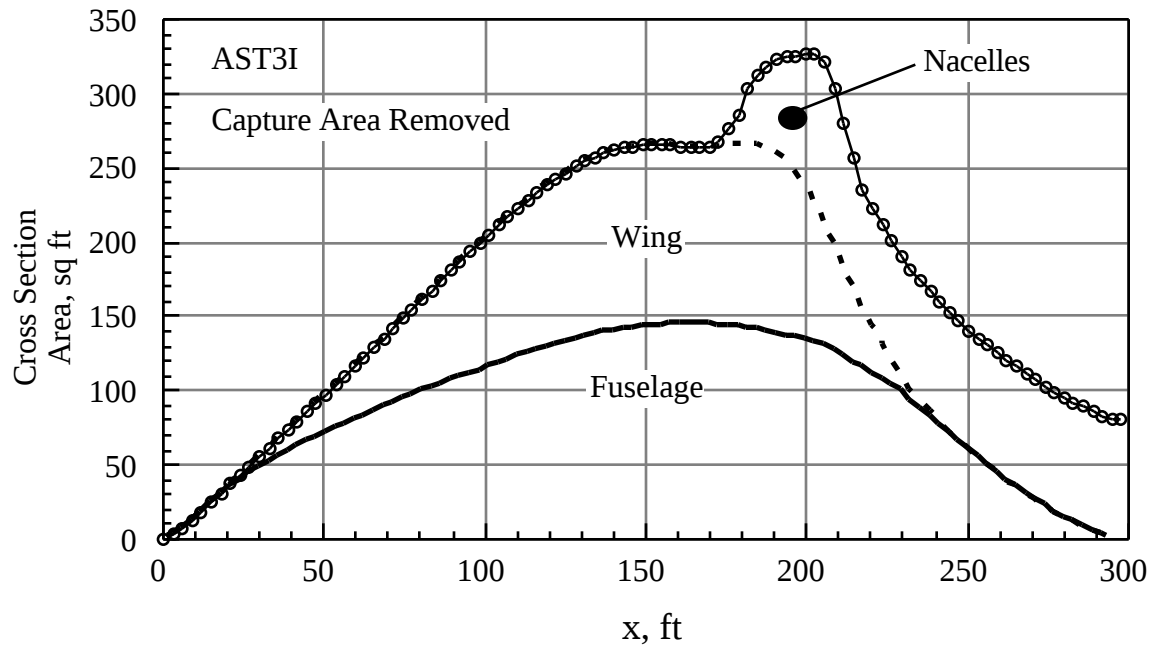
Figure 5-16 Evaluation of the wave drag integral.²¹

a) basic concept three view

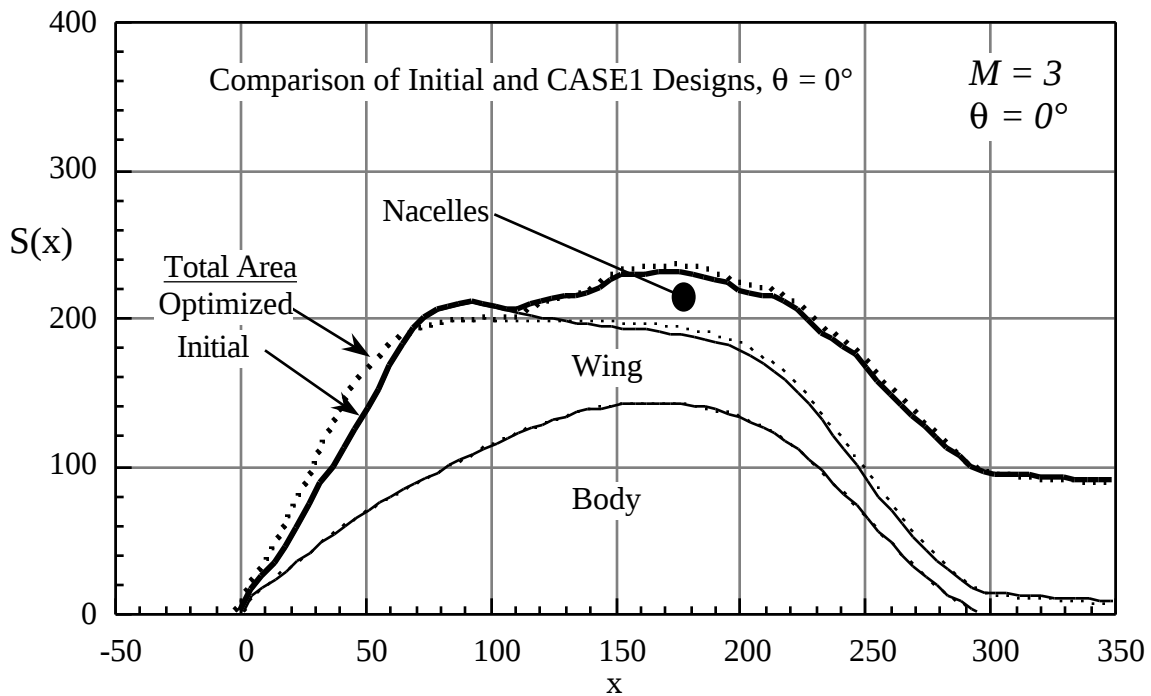


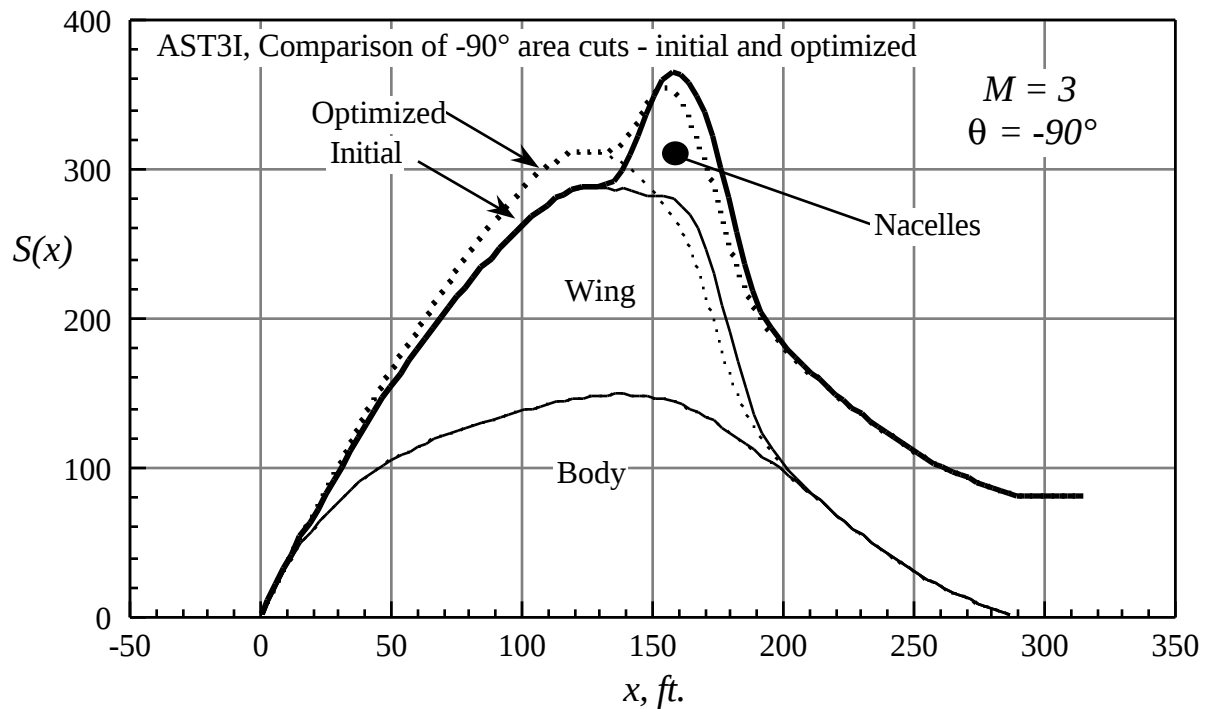
b) distribution of the drag for each circumferential cut.

Figure 5-17 The AST3I,³⁵ an advanced concept for a Mach 3 High Speed Civil Transport.



c) normal area distribution (capture area removed)

d) Mach 3 area distribution, $\theta = 0^\circ$ Figure 5-17 The AST3I,³⁵ an advanced Mach 3 High Speed Civil Transport (cont'd).

e) Mach 3 area distribution, $\theta = 90^\circ$ Figure 5-17 The AST3I,³⁵ an advanced Mach 3 High Speed Civil Transport (concluded).

Area diagrams for typical current fighters are not nearly so streamlined. Figure 5-18 shows the area distribution for the F-16.³⁷ The original area distribution is seen in Fig. 5-18a, and the result of refinements in Fig 5-18b. The F-16 was not designed primarily for supersonic flight, and it has a low fineness ratio and consequently a relatively high wave drag. Small aircraft are much more difficult to lay out to ensure a smooth distribution of area. Note in Fig. 5-18 that the canopy is placed to help “fill in” the area diagram. Figure 5-18b shows the revisions made to improve the contour forward and aft of the maximum cross-sectional area to fill in the shape and also add fuel volume. Note that this curve has no scale. Manufacturers are sensitive about this information.

There is also a wave drag due to lift (see Ashley and Landahl¹⁸). However, almost all area ruling at supersonic speeds primarily emphasizes the volumetric wave drag.

a) Original cross-sectional area

b) Refined area distribution

Figure 5-18 The YF-16 area rule diagram.³⁷

5.9 The Leading Edge Suction Concept

Aerodynamicists often evaluate the performance of configurations in term of so-called leading edge suction. The concept can be explained by considering the inviscid flow over the proverbial zero thickness flat plate at angle of attack in an incompressible inviscid flow, as shown in Fig. 5-19.

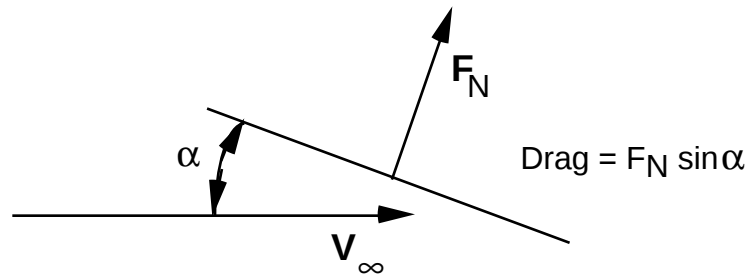


Figure 5-19 Basic relations between forces for an infinitely thin plate.

What is the drag? According to theory, it must be zero. In the sketch we see that the force acts in a direction perpendicular to the plate, and this clearly leads to a force component in the drag direction. What's the explanation of the paradox? Consider the following sketch of the front portion of an airfoil section in Fig. 5-20.

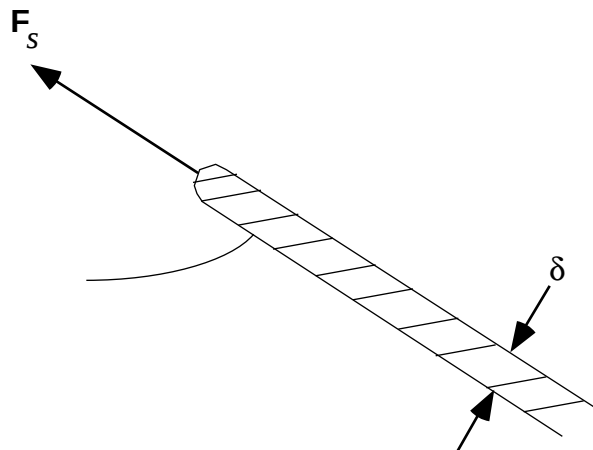


Figure 5-20. Details of the flow near the leading edge of a thin plate.

There is a low pressure over the front edge face due to the expansion of the flow around the leading edge. The expansion becomes stronger as the thickness decreases, so that the force on the front face of the plate due to the product of the pressure and plate thickness is:

$$F_s = \lim_{\delta \rightarrow 0} (\delta \cdot C_{p_s} q_\infty) = \text{finite} \quad (5-37)$$

and the value of the limit is just such that the drag is zero. Thus the correct model of the flow over the flat plate is actually modified from the sketch given above to include an edge force, as shown in Fig. 5-21.

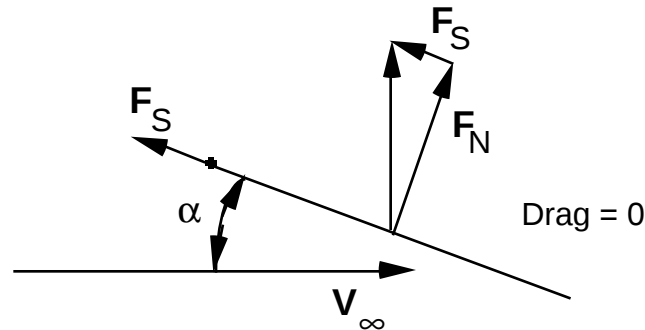


Figure 5-21. Corrected flow model to satisfy inviscid flow theory.

Of course, a very thin flat plate will realize almost none of the suction force, and hence will have a drag component. However, an airfoil section (even a fairly thin one) with a smooth round nose may in fact achieve nearly all of the suction force, at least at small angles of attack. If the airfoil section in the wing does not achieve the full suction performance, the resulting drag must be added to the induced drag.

The drag due to lift is thus broken up into induced drag and additional profile drag. As described previously, the induced drag is a function of the wing spanload only, and is independent of the details of the particular airfoil used in the wing. The additional profile drag is associated with the airfoil used in the wing. At low lift coefficients this drag should be small, only becoming important as flow separation starts to develop on the airfoil section. The additional profile drag becomes large as wing stall is approached.

Wing performance is evaluated based on the ability to obtain a high value of the lift to drag ratio, (L/D) , relative to the maximum possible for that planform, and the ability to achieve a high maximum lift coefficient. Essentially, the wing is designed to allow the airfoil to achieve its full performance. Recalling that a two-dimensional airfoil under the assumption of inviscid subsonic flow has no drag due to lift, the maximum performance should occur by adding the induced drag, assuming an elliptic spanload, to the zero lift drag. This is known as the 100% suction polar, since the airfoil section has no additional profile drag due to lift, and is thus achieving 100% of the leading edge suction required to eliminate the drag force in a two-dimensional flow. This lift is

$$C_{DL100\%} = C_L^2 / \pi AR. \quad (5-38)$$

At the other extreme, the worst case occurs when the airfoil fails to produce any efficient lift, such that the only force is normal to the surface and there is no edge or suction* force (0% leading edge suction). In this case the entire lifting force on the wing is the normal force, and the polar can be determined by resolving that force into lift and drag components. The equation for the 0% suction drag can be expressed in a variety of forms, starting with

$$C_{DL0\%} = C_L \tan(\alpha - \alpha_0) \quad (5-39)$$

where α_0 is the zero lift angle of attack. We also use the linear aerodynamic relation:

$$C_L = C_{L\alpha} (\alpha - \alpha_0) \quad (5-40)$$

which can be solved for the angle of attack:

$$(\alpha - \alpha_0) = \frac{C_L}{C_{L\alpha}} \quad (5-41)$$

Finally, substitute Eqn. (5-41) into Eqn. (5-39) for the angle of attack as follows:

$$\begin{aligned} C_{DL0\%} &= C_L \tan(\alpha - \alpha_0) \cong C_L (\alpha - \alpha_0) \\ &\cong C_L \frac{C_L}{C_{L\alpha}} \end{aligned}$$

or

$$C_{DL0\%} \cong \frac{C_L^2}{C_{L\alpha}} \quad (5-42)$$

This equation for the 0% suction polar shows why this polar is often referred to as the “ $1/C_{L\alpha}$ ” polar by aerodynamicists. Using this approach, effective wing performance is quoted in terms of the fraction of suction achieved, based on the difference between the actual drag and the 100% and 0% suction values as shown in Figure 5-22. This figure illustrates how wings typically perform. The wing will approach the 100% level at low lift coefficients, and then as flow separation starts to develop, the performance deteriorates. Eventually, the wing may have a drag substantially higher than the 0% suction value that was said above to be the worst case.

* On a swept wing the suction force is normal to the leading edge. The component of the leading edge suction force in the streamwise direction is called the leading edge thrust.

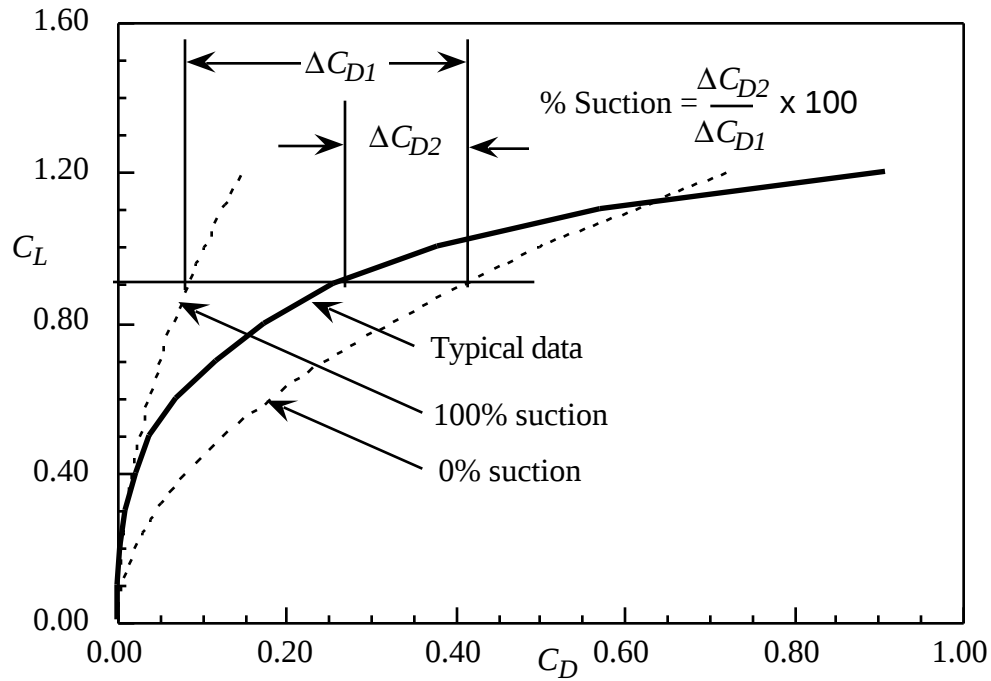


Figure 5-22 Definition of percent leading edge suction performance.

The value of E for this level of performance can be found by equating Eq.(5-42) to the standard form:

$$C_{DL} = \frac{C_L^2}{\pi A R E} \quad (5-43)$$

which leads to:

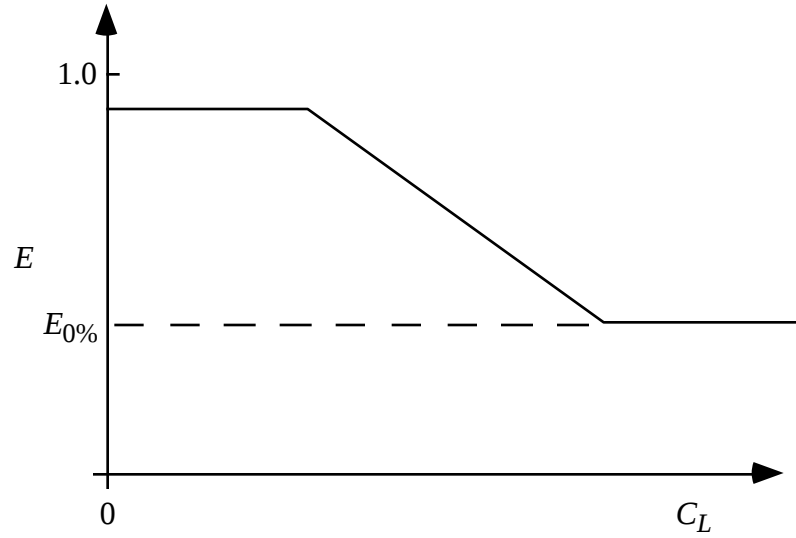
$$E_{0\%} = \frac{C_{L\alpha}}{\pi A R}. \quad (5-44)$$

Typically, the value of E varies with the lift coefficient. By plotting experimental data, typical variations can be obtained for various classes of wings. Figure 5-23 shows the typical variation. This relation was shown in general by McKinney and Dollyhigh.³⁸

Alternately, in supersonic flow, the drag due to lift relation is frequently written as

$$C_{DL} = K C_L^2 \quad (5-45)$$

for uncambered airfoils. For cambered and twisted wings the polar is shifted, and the minimum drag occurs at a C_L other than zero, as shown previously in Fig. 5-2, and described by Eq. (5-2). In practice we expect the wing to achieve a performance level between the $K_{100\%}$ and $K_{0\%}$ limits. This approach is described in detail by Raymer.³⁹

Figure 5-23. Typical variation of E with lift coefficient.

In considering the shift of the polar, a few comments are required. First, the wing performance cannot exceed the optimum value, which for subsonic flow over a single planar lifting surface is $E = 1$. Especially for wings in supersonic flow it is hard to get 100% of the leading edge suction. In that case the approach is to camber the wing to make the drag performance of a wing with less than 100% suction attain the 100% suction level at a specified value of lift, C_{L_D} . Using the polar definition

$$C_D = \Delta C_{D_m} + K_{xx\%} (C_L - C_{L_m})^2 \quad (5-46)$$

where the value of K corresponds to the performance of the wing in terms of leading edge suction (LES), we find the values of ΔC_{D_m} and C_{L_m} in terms of the design lift, C_{L_d} . To do this equate the polar to the 100% suction value at the design lift. This polar must also be tangent to the 100% polar at this point so that the polar will not predict better performance than the optimum at other values of the lift. Using as an example a 0% leading edge suction wing:

$$C_D(100\%LES) = C_D(0\%LES) \Big|_{C_L=C_{L_d}} \quad (5-47)$$

$$\frac{dC_D(100\%LES)}{dC_L} = \frac{dC_D(0\%LES)}{dC_L} \Big|_{C_L=C_{L_d}} \quad (5-48)$$

and the unknown values of ΔC_{D_m} and C_{L_m} are:

$$C_{L_m} = \left(1 - \frac{K_{100\%}}{K_{0\%}} \right) C_{L_d} \quad (5-49)$$

and

$$\Delta C_{D_m} = K_{100\%} C_{L_d}^2 - K_{0\%} (C_{L_d} - C_{L_m})^2. \quad (5-50)$$

In any experimental evaluation of wing performance both the 100% and 0% polars should be constructed, and used to establish bounds on the experimental polar. Thus a typical drag polar would include the 100% and 0% suction polars as well as the predicted or measured performance to establish a basis for evaluating a wing's efficiency. Figure 5-24 presents the actual performance of an unswept rectangular wing at subsonic speed. Here the performance is very close to the lower drag limit until the wing stalls.

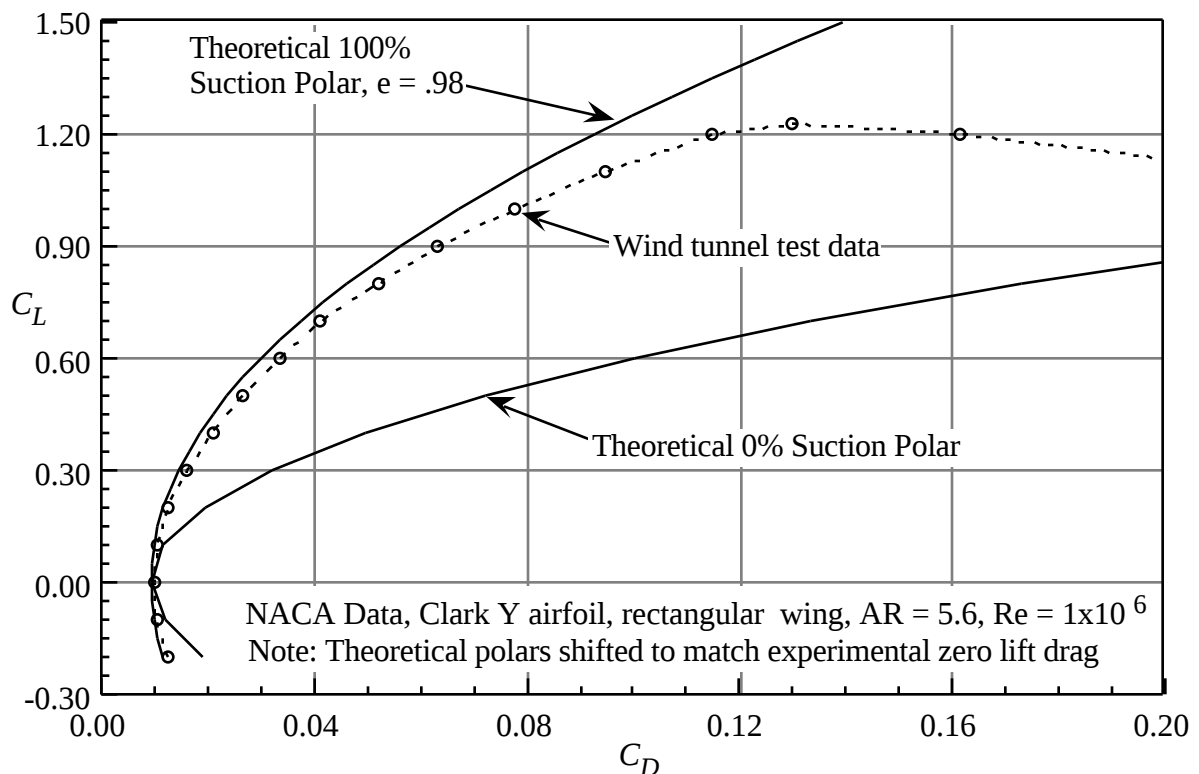


Figure 5-24 Drag performance of simple unswept wing with a Clark-Y airfoil.

It is difficult to identify the initial flow breakdown using the drag polar. Often you can identify flow breakdown more clearly by plotting the axial force as a function of normal force. In this plot the axial force should initially decrease, as described above. When the airfoil section starts to lose leading edge suction the data displays a sharp “break.” Figure 5-25 illustrates this approach to the examination of wing efficiency.

For configurations with very poor aerodynamic efficiency, the 0% suction force provides a good estimate of the vehicle drag. However, 0% suction levels are so inefficient that for most designs this level of performance would be unacceptable and not competitive.

To make estimates of the performance of real configurations, which operate between the two limits, Harry Carlson^{40,41} at NASA Langley established the notion of “attainable” leading edge suction. Based on an extensive analysis of 2D airfoil data, Carlson established an empirical correlation which is used to estimate the fraction of the full suction that should be attained for the specified airfoil, planform and flight condition. Carlson’s concepts are based on linear theory.

Nonlinear effects can be important, and can be exploited. Although the linear theory based concepts described here provide a valuable way of looking at wing designs, nonlinear effects can provide a means of improving performance. Considering nonlinear effects, interactions between thickness and lifting effects can be exploited.⁴²

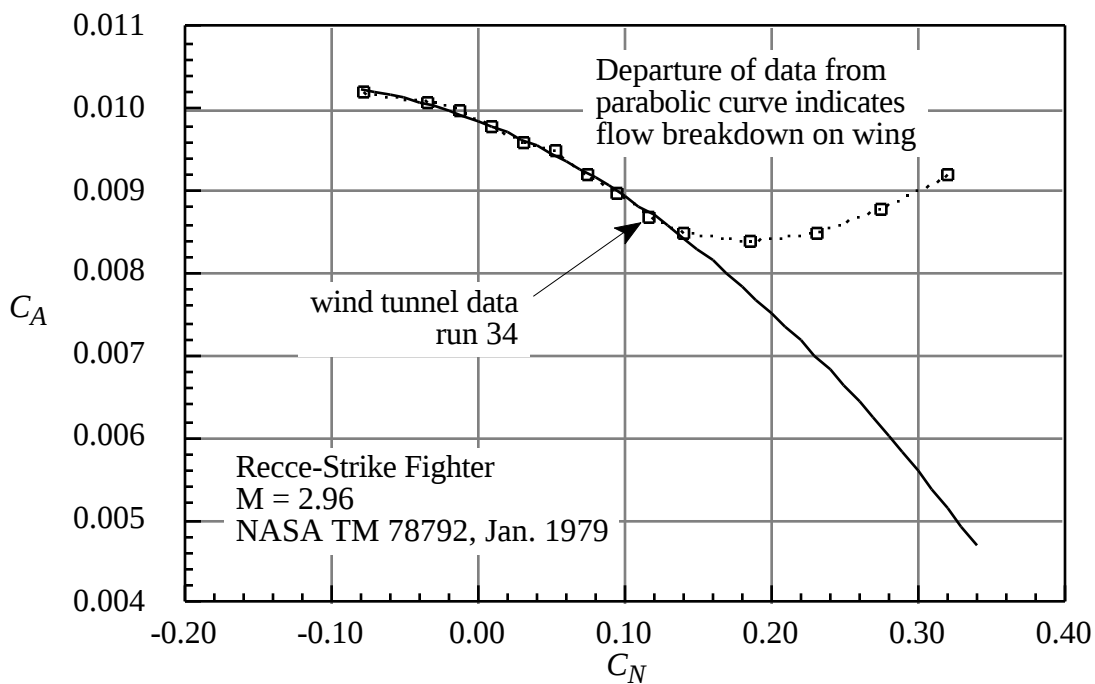
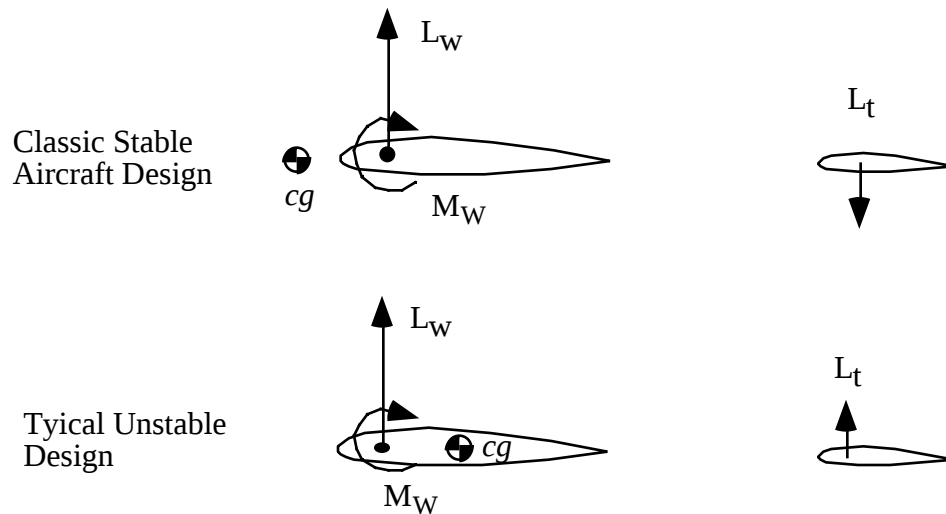


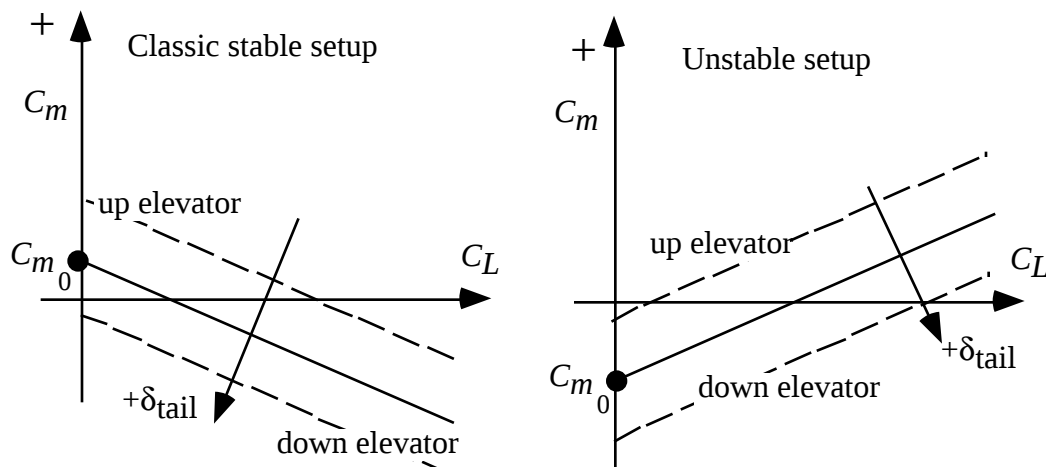
Figure 5-25. Axial force analysis of wing performance.

5.10 Trim Drag

For equilibrium flight the airplane must be trimmed. The forces must be such that the moments about the center of gravity in all axes are zero. To achieve this condition the controls are usually deflected to generate the required trimming moments. Figure 5-26 shows a schematic of the requirement. Two typical situations are shown in Fig. 5-26a. In one case the center of gravity is ahead of the wing center of pressure, the aircraft is stable, and a download on the tail is required to balance the lift of the wing. In the other case the center of gravity is behind the wing center of pressure, the airplane is unstable, and an upload on the tail is required to balance the lift of the wing. Other situations are possible, but these two illustrate the key idea.



a) surface lift requirements for trim



b) idealizations of design requirements

Figure 5-26. Examination of the configuration setup required for trim.

Part *b* of Fig. 5-26 illustrates the difference between stable and unstable configurations. For a stable airplane the basic C_{m0} is typically positive, while for an unstable aircraft the basic C_{m0} is negative. In each case, a control has to be deflected over a range of settings to maintain trim over a range of lift coefficients (unless the configuration is neutrally stable). On modern aircraft the control could be the deflection of the thrust by thrust vectoring.

Control surface deflections change the drag from the reference undeflected value. This difference in drag could be termed a "trim drag." There are many definitions of trim drag. Definitions differ because it is difficult to be precise in defining trim drag. Some definitions contain only the

drag due to the lift of the trimming surface. Some analyses allow for a negative trim drag. However, for a given flight condition the total lift must be fixed, and any change in lift on the trimming surface requires a change in lift, and hence drag, on the primary surface. On a well designed aircraft the trim drag should be small. Canard concepts are often considered advantageous because both the canard and wing supply positive lift to trim, whereas for traditional aft-tail configurations the tail load is negative and the wing must operate at a higher lift to compensate. However, for modern aft-tail designs the tail load is near zero, resulting in little trim drag.

Trim drag has always been an important consideration in airplane design. However, trim drag became especially important with the development of stability and control augmentation systems that allowed the designer much greater freedom in the choice of a center of gravity location. Natural static stability was no longer required. The static stability condition had frequently made it difficult to obtain minimum trim drag. This meant that trim drag could become a key criteria for the placement of the center of gravity in a configuration (this is part of the motivation for so-called control configured vehicle, CCV, concepts).

Trim drag is especially important for several specific classes of aircraft. Supersonic aircraft demand special consideration because of the aerodynamic center shift from subsonic to supersonic flight. To control trim drag as well as stability, fuel is transferred fore and aft between subsonic and supersonic flight to achieve proper balance on supersonic cruise aircraft. Variable sweep wing aircraft also have aerodynamic center locations that vary with wing sweep, potentially leading to high values of trim drag. Finally, maneuvering aircraft can suffer high trim drag at high lift coefficients, severely limiting sustained turn performance. This was especially true of the first generation of supersonic capable fighters. Examples of the contribution of trim drag to the total drag are shown in Figure 5-27, taken from Nicolai.⁴³

A more useful approach to the trim drag analysis is to consider the value of “trimmed drag”. In this approach it is difficult to define a specific trim drag value. The best way to assess the trim penalty is to define the difference between the minimum drag attainable for the system and the minimum trimmed drag for a specified center of gravity position. This provides the designer with a measure of the drag penalty being paid for a particular center of gravity location. This approach also demonstrates directly the connection between static margin and minimum trimmed drag. Different configuration concepts lead to different values of static margin to obtain minimum trimmed lift. In general, for aft swept wings aft tail configurations, the minimum trimmed drag occurs at a slightly unstable center of gravity (5-10%?), canard configurations have minimum trim drag at slightly more unstable conditions (15%?), and forward swept wing canard configurations must be even more unstable to achieve minimum trimmed drag (the X-29 is about 30-35% unstable). Many studies of these fundamental properties of various configuration concepts have been made. See the study by Landfield and Rajkovic⁴⁴ and the references contained therein for more information.

a) C-141 at $M = 0.75$

b) F-111A

Figure 5-27. Contributions to drag for two configurations.⁴³

Several key papers examining trim drag from a nearfield point of view have been written. They are by McKinney and Dollyhigh,³⁸ Lutze,⁴⁵ and Sachs.⁴⁶ In the nearfield, extreme care must be taken to include the downwash incidences and induced angles of attack correctly. Alternatively, an analysis can be made in the Trefftz plane. Lamar²⁶ developed a code for finding the minimum trimmed induced drag for two surfaces, and this was extended to include (approximately) the effects of profile drag by Mason.²⁷ Note that a farfield analysis which combines the minimization of induced drag and wave drag due to lift has been presented by Tulinius and Margason.⁴⁷ A more general approach to treat multiple surfaces was given by Kuhlman.⁴⁸ More recently, three surface configuration have been introduced, and the three surface minimum trim drag problem has been solved by Goodrich, Sliwa, and Lallman⁴⁹ using a nearfield approach.

An example of the possible dramatic effects of cg location on trimmed drag is presented in Figure 5-28.²⁷ All the results contained in the figure are for the minimum trimmed drag at different values of a specified cg location. These results were obtained during early forward-swept wing configuration studies, and illustrate why an aft cg position and resulting highly unstable configuration are required to obtain the full benefits of a forward swept wing configuration simi-

lar to the X-29.* The very high drag values reflect a transonic maneuver condition. Trim drag should be much smaller for the cruise condition (certainly less than 2-4%). As shown here, modern technology should allow the aircraft to fly with no trim drag. The difference between the minimum trimmed drag at $\Delta x_{cg} = -40$ and any other cg could be considered the trim drag. The figure contains both induced and profile drag contributions to the total trimmed drag. As the cg moves forward (x positive in this nomenclature, $x = 0$ corresponds to neutral stability), the additional load on the canard leads to a rapidly increasing value of the minimum trimmed drag. Because of the increasing load on the canard, the canard airfoil section becomes important. Near the cg for minimum drag the canard airfoil is not important because the canard is lightly loaded. This figure shows why canard configurations are most efficient when used with unstable configuration concepts. Stable canard configurations are not necessarily the most efficient aerodynamically. Because of the high loads on a stable canard configuration, the canard airfoil section is very carefully selected. Specifically, it is usually highly cambered to achieve the high lift coefficients, and has a small leading edge radius so that it will stall before the main wing.

Figure 5-28 Minimum trimmed drag throughout a range of balance locations, including the effect of canard airfoil section.²⁷

* Note that the hydraulic power used to activate the canard to achieve apparent stability is obtained from the engine, reducing thrust and resulting in increased fuel flow. Thus, in essence, some trim drag benefits are gained at the expense of increased fuel usage to control the unstable vehicle.

5.11 Current Issues for Drag Calculation Using Computational Aerodynamics

Because of the quest for reduced drag, and the difficulty in computing and measuring small changes in drag, numerous disputes have arisen in aerodynamics. Confusion introduced by non-standard drag nomenclature also contributes to these spirited debates. One recent issue was the so-called sheared wing-tip drag reduction controversy. Here it was speculated that wing tip shaping could lead to span e 's greater than one for planar planforms.⁵⁰ This conclusion arose based on both computations and wind tunnel tests. Refined computational investigations,⁵¹ illustrating the need to study computational solution convergence carefully as shown elsewhere in this text, resulted in the conclusion that e 's greater than one were not actually computed. However, it is clear that wing tip planform shaping can lead to improved aerodynamic efficiency.

Another area currently attracting attention is the search for more fundamental understanding of drag. These theories differ significantly from the accepted approach to drag. One key example is due to Yates.⁵²

In addition to the efforts to reduce drag due to lift by tip shaping, use of winglets, tip sails, and canard configurations among others, significant efforts are being made to reduce skin friction drag. They include efforts to obtain laminar flow through passive means (NLF), suction, or a combination known as hybrid natural laminar flow control. Turbulent friction reduction techniques are also being developed. Riblets are perhaps the most well known means. A recent AIAA book reviews this area.⁵³

5.12 Exercises

1. Derive equations 5-7 and 5-8.
2. Derive equations 5-31 and 5-32.
3. Get a copy of **LIDRAG** and check it against the the known values of the span e for an elliptic and a triangular load distribution. How do your results compare?
4. Use experimental results and show quantitatively that Fig. 5-11 is correct.
5. Get a copy of **FRICION**. Repeat the check case in the user's manual. Then examine the skin friction values in Table 5-1. Are they reasonable?
6. Find the shape of the Karman ogive, the body with minimum wave drag for a given length and base area.

5.13 References

1. Strang, W.J., and McKinlay, R.M., "Concorde in Service," *Aeronautical Journal*, February, 1979.
2. Bavitz, et al., "Configuration Development of Advanced Fighters," Vol. 3, Concept Definition and Validation, AFWAL-TR-3142, November 1980.
3. Sloof, J.W., ed., "Technical Status Review on Drag Prediction and Analysis from Computational Fluid Dynamics: State of the Art," AGARD-AR-256, May 1988.
4. Aerodynamic Drag, AGARD CP-124, 1973.
5. Aircraft excrescence Drag. AGARD 1981.
6. Aircraft Drag Prediction, AGARD-R-723, 1985.
7. Drag Reduction Techniques, AGARD ????
8. Covert, E.E., ed., *Thrust and Drag: Its Prediction and Verification*, AIAA, New York, 1985.
9. Hoerner, S., *Fluid Dynamic Drag*, originally published by the author, and available from Hoerner Fluid Dynamics, 7528 Stauton Place, NW, Albuquerque, NM 87120. (505) 898-0533.
10. Torenbeek, E., *Synthesis of Subsonic Airplane Design*, Delft University Press, Delft, 1982.
11. Whitford, R., *Design for Air Combat*, Jane's London, 1987.
12. Huenecke, K., *Modern Combat Aircraft Design*, Naval Institute Press, Annapolis, 1987.
13. Haines, A.B., "Subsonic Aircraft Drag: An Appreciation of the Standards," *The Aeronautical Journal of the Royal Aeronautical Society*, Mar. 1968, pp. 253-266.
14. Küchemann, D., *The Aerodynamic Design of Aircraft*, Pergamon Press, Oxford, 1978.
15. Lynch, F.T., "Commercial Transports—Aerodynamic Design for Cruise Performance Efficiency," *Transonic Aerodynamics*, Nixon, D., ed., Progress in Astronautics and Aeronautics, Vol. 181, AIAA New York, 1982.
16. Shevell, "Aerodynamic Bugs: Can CFD Spray Them Away?" AIAA Paper 85-4067, Oct. 1985.
17. Rooney, E.C., "Thrust-Drag Accounting Methodology," in *Thrust and Drag: Its Prediction and Verification*, Covert, E.E., ed., AIAA, New York, 1985.
18. Ashley, H., and Landahl, M., *Aerodynamics of Wings and Bodies*, Addison-Wesley, Reading, 1965, sections 1.6, 6.6, 7.3 and 9.2,
19. Heaslet, M.A., and Lomax, H., "Supersonic and Transonic Small Perturbation Theory," in *General Theory of High Speed Aerodynamics*, W.R. Sears, ed., Princeton University Press, 1954, sec. D,14: pp 221-229.
20. Liepman, H.W., and Roshko, A., *Elements of Gasdynamics*, John Wiley, New York, 1957, pp. 235-239.
21. Harris, R.V., "An Analysis and Correlation of Aircraft Wave Drag," NASA TM X-947, 1964.

5.13 References

22. Sears, W.R., "On Calculation of Induced Drag and Conditions Downstream of a Lifting Wing," *Journal of Aircraft*, Vol. 11, No. 3, March 1974, pp.191-192.
23. Thwaites, B., ed., *Incompressible Aerodynamics*, Oxford, 1960, pp. 454-457, 523-527.
24. Mason, W.H., Mackenzie, D., Stern, M., Ballhaus, W.F., and Frick, J., , "An Automated Procedure for Computing the Three Dimensional Transonic Flow over Wing-Body Combinations, Including Viscous Effects," AFFDL-TR-77-122, Feb. 1978.
25. Milne-Thomson, *Theoretical Aerodynamics*, Dover, New York, pp 218-219
26. Lamar, J.E., "A Vortex Lattice Method for the Mean Camber Shapes of Trimmed Non-Coplanar Planforms with Minimum Vortex Drag," NASA TN D-8090, June, 1976.
27. Mason, W.H., "Wing-Canard Aerodynamics at Transonic Speeds—Fundamental Considerations on Minimum Drag Spanloads," AIAA Paper 82-0097, Jan. 1982.
28. Paterson, J.H., MacWilkenson, D.G., and Blackerby, W.T., "A Survey of Drag Prediction Techniques Applicable to Subsonic and Transonic Aircraft Design," in *Aerodynamic Drag*, AGARD CP 124, April 1973.
29. Jones, R.T. *Wing Theory*, Princeton University Press, Princeton, 1991.
30. Whitcomb, R.T., "A Study of the Zero Lift Drag-Rise Characteristics of Wing-Body Combinations near the Speed of Sound," NACA Report 1273, 1956.
31. Mendenhall, C., *Delta Wings—Convair's High Speed Planes of the Fifties and Sixties*, Motorbooks International, Osceola, WI, 1983.
32. Anon., *Aireviews jet Fighters of the World Part 2*, Kantosha Co. Ltd., Tokyo, 1965 (in Japanese).
33. Stuart, William G., "Northrop F-5 Case Study in Aircraft Design," AIAA Professional Study Series, Sept. 1978.
34. Eminton, E., and Lord W.T., "Note on the Numerical Evaluation of the Wave Drag of Smooth Slender Bodies Using Optimum Area Distributions for Minimum Wave Drag," *Journal of the Royal Aeronautical Society*, Vol. 60, Jan. 1956, pp. 61-63.
35. Robins, A.W., Dollyhigh, S.M., Beissner, F.L., Jr., Geiselhart, K., Martin, G.L., Shields, E.W., Swanson, E.E., Coen, P.G., Morris, S.J., Jr., "Concept Development of a Mach 3.0 High-Speed Civil Transport," NASA TM 4058
36. Hutchison, M., Unger, E., Mason, W.H., Grossman, B., and Haftka, R., "Variable-Complexity Aerodynamic Optimization of an HSCT Wing Using Structural Wing-Weight Equations," AIAA Paper 92-0212, Jan. 1992.
37. Buckner, J.K., Benepe, D.B., Hill, D.W., "Aerodynamic Design Evolution of the F-16," AIAA Paper 74-935, 1974.
38. McKinney, L.W., and Dollyhigh, S.M., "Some Trim Drag Considerations for Maneuvering Aircraft," *Journal of Aircraft*, Vol. 8, No. 8, Aug. 1971, pp.623-629.
39. Raymer, D.P., *Aircraft Design: A Conceptual Approach*, AIAA, Washington, 1989.
40. Carlson, H.W., Mack, R.J., and Barger, R.L., "Estimation of Attainable Leading Edge Thrust for Wings at Subsonic and Supersonic Speeds," NASA TP 1500, 1979.

5.13 References

41. Mann, M.J., and Carlson, H.W., "An Assessment of Current Methods for Drag-Due-To-Lift Minimization at Supersonic Speeds," AIAA Paper 91-3302, Sept. 1991.
42. Mason, W.H., and DaForno, G., "Opportunities for Supersonic Performance Gains Through Non-Linear Aerodynamics," AIAA Paper 79-1527, July 1979.
43. Nicolai, L.M., *Fundamentals of Aircraft Design*, METS, Xenia, 1975.
44. Landfield, J.P., and Rajkovic, D., "Canard/Tail Comparison for an Advanced Variable-Sweep-Wing Fighter," AIAA Paper 84-2401, October 1984.
45. Lutze, F.H., "Trimmed Drag Considerations," *Journal of Aircraft*, Vol. 14, No. 6, June 1977, pp. 544-546.
46. Sachs, G., "Minimum Trimmed Drag and Optimum c.g. Position," *Journal of Aircraft*, Vol. 15, No. 8, Aug. 1978, pp. 456-459.
47. Tulinius, J.R., and Margason, R.J., "Aircraft Aerodynamic Design and Evaluation Methods," AIAA Paper 76-15, Jan. 1976.
48. Kuhlman, J.M., "Higher Order Farfield Drag Minimization for a Subcritical Wing Design," *Journal of Aircraft*, Vol. 17, No. 9, Sept. 1980, pp. 648-655.
49. Goodrich, K.H., Sliwa, S.M., and Lallman, F.J., "A Closed-Form Trim Solution Yielding Minimum Trim Drag for Airplanes with Multiple Longitudinal Control Effectors," NASA TP 2907, May 1989.
50. VanDam, C.P., "Induced-Drag Characteristics of Crescent-Moon-Shaped Wings," *Journal of Aircraft*, Vol. 24, No. 2, Feb. 1987, pp. 115-119.
51. Smith, S.C., and Kroo, I.M., "A Closer Look at the Induced Drag of Crescent-Shaped Wings," AIAA Paper 90-3063, Aug. 1990.
52. Yates, J.E., "Prediction of Drag and Lift: A Viscous/Thermodynamic Approach," A.R.A.P. Report No. 651, Aug. 1990 (presented at an AIAA Short Course, Portland, Oregon, Aug. 23-24, 1990).
53. Bushnell, D.M., and Hefner, J.N., ed., *Viscous Drag Reduction in Boundary Layers*, AIAA Progress in Astronautics and Aeronautics Series, Vol. 123, AIAA, Washington, 1990.

6. Aerodynamics of 3D Lifting Surfaces through Vortex Lattice Methods

6.1 An Introduction

There is a method that is similar to panel methods but very easy to use and capable of providing remarkable insight into wing aerodynamics and component interaction. It is the vortex lattice method (**vlm**), and was among the earliest methods utilizing computers to actually assist aerodynamicists in estimating aircraft aerodynamics. Vortex lattice methods are based on solutions to Laplace's Equation, and are subject to the same basic theoretical restrictions that apply to panel methods.

As a comparison, vortex lattice methods are:

Similar to Panel methods:

- singularities are placed on a surface
- the non-penetration condition is satisfied at a number of control points
- a system of linear algebraic equations is solved to determine singularity strengths

Different from Panel methods:

- Oriented toward lifting effects, and classical formulations ignore thickness
- Boundary conditions (BCs) are applied on a mean surface, not the actual surface (not an exact solution of Laplace's equation over a body, but embodies some additional approximations, *i.e.*, together with the first item, we find ΔC_p , not C_{pupper} and C_{plower})
- Singularities are not distributed over the entire surface
- Oriented toward combinations of thin lifting surfaces (recall Panel methods had no limitations on thickness).

Vortex lattice methods were first formulated in the late '30s, and the method was first called "Vortex Lattice" in 1943 by Faulkner. The concept is extremely simple, but because of its purely numerical approach (*i.e.*, no answers are available at all without finding the numerical solution of a matrix too large for routine hand calculation) practical applications awaited sufficient development of computers—the early '60s saw widespread adoption of the method. A workshop was devoted to these methods at NASA in the mid '70s.¹ A nearly universal standard for vortex lattice predictions was established by a code developed at NASA Langley (the various versions were available prior to the report dates):

Margason & Lamar ²	1st Langley report	NASA TN D-6142	1971
Lamar & Gloss ³	2nd " "	NASA TN D-7921	1975
Lamar & Herbert ^{4,5}	3rd " "	NASA TM 83303	1982

Each new version had considerably more capability than the previous version. The “final” development in this series is designated VLM4.997. The original codes could handle two lifting surfaces, while VLM4.997 could handle four. Many, many other people have written vortex lattice method codes, some possibly even better than the code described in the NASA reports. But the NASA code’s general availability, versatility, and reliability resulted in its becoming a de-facto standard.

Some of the most noteworthy variations on the basic method have been developed by Lan⁶ (Quasi-Vortex Lattice Method), Hough⁷, DeJarnette⁸ and Frink⁹. Mook¹⁰ and co-workers at Virginia Tech have developed vortex lattice class methods that treat flowfields that contain leading edge vortex type separation (see Section 6.12) and also handle general unsteady motions. The recent book by Katz and Plotkin¹¹ contains another variation. At Virginia Tech, Jacob Kay wrote a code using the method of Katz and Plotkin to estimate stability derivatives, which is available from the department web page.¹²

To understand the method, a number of basic concepts must be reviewed. Then we describe one implementation of the **vlm** method, and use it to obtain insights into wing and wing-canard aerodynamics. Naturally, the method is based on the idea of a vortex singularity as the solution of Laplace’s equation. A good description of the basic theory for vortices in inviscid flow and thin wing analysis is contained in Karamcheti,¹³ pp. 494-496, 499-500, and 518-534. A good description of the vortex lattice method is given by Bertin and Smith.¹⁴ After the discussion of wing and wing-canard aerodynamics, an example of a vortex lattice method used in a design mode is presented, where the camber line required to produce a specified loading is found. The chapter concludes with a few examples of the extension of vortex lattice methods to treat situations with more complicated flowfields than the method was originally intended to treat.

6.2 Boundary conditions on the mean surface and the pressure relation

An important difference between vortex lattice methods and panel methods is the method in which the boundary conditions are handled. Typically, the vortex lattice method uses an approximate boundary condition treatment. This boundary condition can also be used in other circumstances to good advantage. This is a good “trick” applied aerodynamicists should know and understand. In general, this approach results in the so-called “thin airfoil boundary condition,” and arises by linearizing and transferring the boundary condition from the actual surface to a flat mean “reference” surface, which is typically a constant coordinate surface. Consistent with the boundary condition simplification, a simplified relation between the pressure and velocity is also possible. The simplification in the boundary condition and pressure-velocity relation provides a basis for treating the problem as a superposition of the lift and thickness contributions to the aerodynamic results. Karamcheti¹³ provides an excellent discussion of this approach.

To understand the thin airfoil theory boundary condition treatment, we provide an example in two dimensions. Recall (from Eqn. 2-54) that the exact surface boundary condition for steady inviscid flow is:

$$\mathbf{V} \cdot \mathbf{n} = 0 \quad (6-1)$$

on $F(x,y) = 0 = y - f(x)$. The unit normal vector is $\mathbf{n} = -F(x,y) / |-F(x,y)|$ and the velocity field is defined using the notation defined in Fig. 6-1.

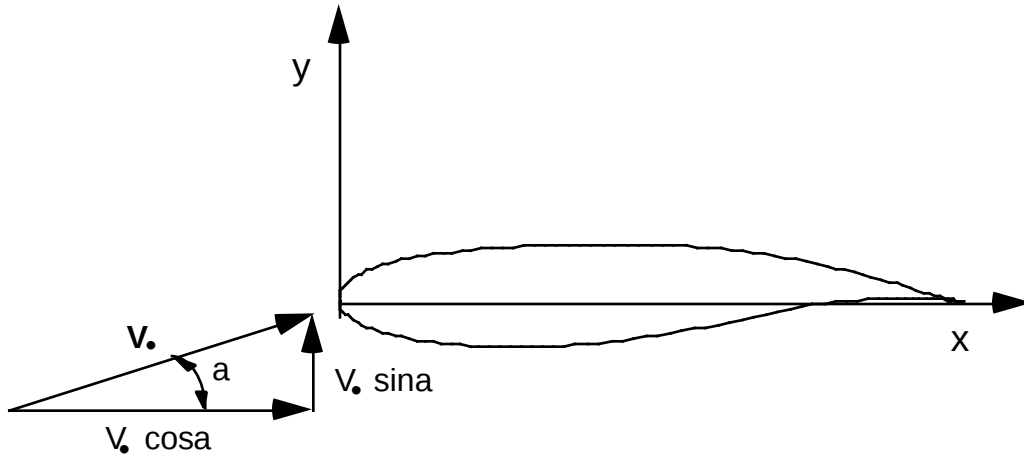


Figure 6-1. Basic coordinate system for boundary condition analysis.

Define the velocity components of $\mathbf{V}$ as:

$$\mathbf{V} = \mathbf{V}_\infty + \underbrace{\mathbf{q}(x,y)}_{\text{a disturbance velocity}} \quad (6-2)$$

where $\mathbf{q}$ is a disturbance velocity with components u and v . If we assume irrotational flow, then these components are described in terms of a scalar potential function, $u = -f_x$ and $v = -f_y$. The total velocity $\mathbf{V}$ then becomes in terms of velocity components:

$$\begin{aligned} u_{\text{TOT}} &= V_\infty \cos \alpha + u(x,y) \\ v_{\text{TOT}} &= V_\infty \sin \alpha + v(x,y) \end{aligned} \quad (6-3)$$

and we can write out the boundary condition as:

$$\mathbf{V} \cdot \mathbf{n} = (u_{\text{TOT}} \mathbf{i} + v_{\text{TOT}} \mathbf{j}) \cdot \left(\frac{\partial F}{\partial x} \mathbf{i} + \frac{\partial F}{\partial y} \mathbf{j} \right) = 0 \quad (6-4)$$

or

$$\left[V_\infty \cos \alpha + u(x,y) \right] \frac{\partial F}{\partial x} + \left[V_\infty \sin \alpha + v(x,y) \right] \frac{\partial F}{\partial y} = 0 \quad (6-5)$$

on $F(x,y) = 0$, and recalling the relationship between F and f given below Eqn. (6-1):

$$\begin{aligned}\frac{\partial F}{\partial x} &= \frac{\partial}{\partial x}\{y - f(x)\} = -\frac{df(x)}{dx} \\ \frac{\partial F}{\partial y} &= \frac{\partial}{\partial y}\{y - f(x)\} = 1\end{aligned}\quad (6-7)$$

Substituting for F in Eq.(6-5) we have:

$$(V_\infty \cos \alpha + u) \frac{df}{dx} + (V_\infty \sin \alpha + v) = 0 \quad (6-8)$$

which, solving for v , is:

$$v = (V_\infty \cos \alpha + u) \frac{df}{dx} - V_\infty \sin \alpha \quad (6-9)$$

on $y = f(x)$. Note that v is defined in terms of the unknown u . Thus Eq. (6-9) is a nonlinear boundary condition and further analysis is needed to obtain a useful relation.*

6.2.1 Linearized form of the boundary condition

The relation given above by Eq.(6-9) is exact. It has been derived as the starting point for the derivation of useful relations when the body (which is assumed to be a thin surface at a small angle of attack) induces disturbances to the freestream velocity that are small in comparison to the freestream velocity. Thus we assume: $u \ll V_\infty$, $v \ll V_\infty$, and $\partial F/\partial x < \partial F/\partial y$. Note that this introduces a bias in the coordinate system to simplify the analysis, a typical consequence of introducing simplifying assumptions. Consistent with this assumption, the components of the freestream velocity are:

$$\begin{aligned}V_\infty \cos \alpha &\approx V_\infty \\ V_\infty \sin \alpha &\approx V_\infty \alpha\end{aligned}\quad (6-10)$$

and the expression for v in Eq.(6-9) becomes:

$$v = (V_\infty + u) \frac{df}{dx} - V_\infty \alpha \quad (6-11)$$

Dividing by V_∞ ,

$$\frac{v}{V_\infty} = \frac{u}{V_\infty} + \frac{df}{dx} - \alpha \quad (6-12)$$

the linearized boundary condition is obtained by neglecting u/V_∞ compared with unity (consistent with the previous approximations). With this assumption, the linearized boundary condition becomes:

$$\frac{v}{V_\infty} = \frac{df}{dx} - \alpha \quad \text{on } y = f(x). \quad (6-13)$$

* Observe that even when the flowfield model is defined by a linear partial differential equation, an assumption which we have not yet made, the boundary condition can make the problem nonlinear.

This form of the boundary condition is not valid if the flow disturbance is large compared to the freestream velocity {for aerodynamically streamlined shapes this is usually valid everywhere except at the leading edge of the airfoil, where a stagnation point exists ($u = -V_\infty$) and the slope is infinite ($df/dx = \infty$)}. In practice, a local violation of this assumption leads to a local error. Thus, if the details of the flow at the leading edge are not important to the analysis, which surprisingly is often the case, the linearized boundary condition can be used.

6.2.2 Transfer of the boundary condition

Although Eqn. (6-13) is linear, it's hard to apply because it is not applied on a coordinate line.* We now use a further approximation of this relation to get the useful form of the linearized boundary condition. Using a Taylor series expansion of the v component of velocity about the coordinate axis we obtain the v velocity on the surface:

$$v(x, y = f(x)) = v(x, 0) + f(x) \left. \frac{\partial v}{\partial y} \right|_{y=0} + \dots \quad (6-14)$$

For the thin surfaces under consideration, $f(x)$ is small, and because the disturbances are assumed small, $\partial v / \partial y$ is also small. For example, assume that v and $\partial v / \partial y$ are the same size, equal to 0.1, and df/dy is also about 0.1. The relation between v on the airfoil surface and the axis is:

$$v(x, y = f(x)) = (.1) + (.1)(.1) = .1 + \underbrace{.01}_{\text{neglect}} \quad (6-15)$$

Neglecting the second term, we assume:

$$v(x, f(x)) \approx v(x, 0). \quad (6-16)$$

We now apply both the upper and lower surface boundary conditions on the axis $y = 0$, and distinguish between the upper and lower surface shapes by using:

$$\begin{aligned} f &= f_u && \text{on the upper surface} \\ f &= f_l && \text{on the lower surface} \end{aligned} \quad (6-17)$$

Using Eq. (6-17), we write the upper and lower surface boundary conditions as:

$$\left. \frac{v(x, 0^+)}{V_\infty} \right|_{up} = \frac{df_u}{dx} - a, \quad \left. \frac{v(x, 0^-)}{V_\infty} \right|_{low} = \frac{df_l}{dx} - a. \quad (6-18)$$

* The simplification introduced by applying boundary conditions on constant coordinate surfaces justifies the use of rather elaborate transformations, which will be discussed in more detail in Chap. 9, Geometry and Aerodynamics.

These are the linearized and transferred boundary conditions. Frequently, these boundary conditions result in a surprisingly good approximation to the flowfield, even in transonic and supersonic flow.

6.2.3 Decomposition of boundary conditions to camber/thickness/alpha

Further simplification and insight can be gained by considering the airfoils in terms of the combination of thickness and camber, a natural point of view. We thus write the upper and lower surface shapes in terms of camber, f_c , and thickness, f_t , as:

$$\begin{aligned} f_u &= f_c + f_t \\ f_l &= f_c - f_t \end{aligned} \quad (6-19)$$

and the general problem is then divided into the sum of three parts as shown in Fig. 6-2.

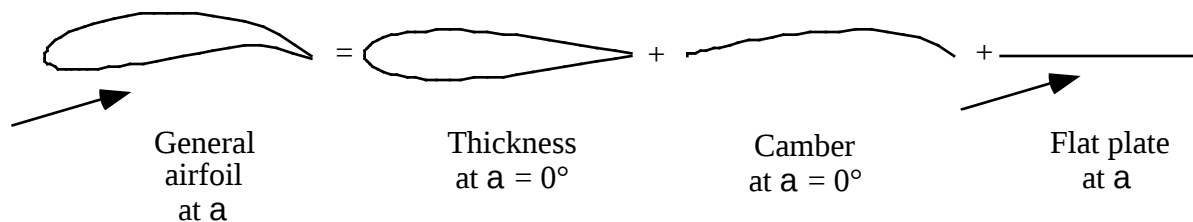


Figure 6-2. Decomposition of a general shape at incidence.

The decomposition of the problem is somewhat arbitrary. Camber could also be considered to include angle of attack effects using the boundary condition relations given above, the sign is the same for both the upper and lower surface. The aerodynamicist must keep track of details for a particular problem. *To proceed further, we make use of the basic vortex lattice method assumption: the flowfield is governed by a linear partial differential equation (Laplace's equation).* Superposition allows us to solve the problem in pieces and add up the contributions from the various parts of the problem. This results in the final form of the thin airfoil theory boundary conditions:

$$\begin{aligned} \left. \frac{v(x, 0^+)}{V_\infty} \right|_{up} &= \frac{df_c}{dx} + \frac{df_t}{dx} - \alpha \\ \left. \frac{v(x, 0^-)}{V_\infty} \right|_{low} &= \frac{df_c}{dx} - \frac{df_t}{dx} - \alpha \end{aligned} \quad (6-20)$$

The problem can be solved for the various contributions and the contributions are added together to obtain the complete solution. If thickness is neglected the boundary conditions are the same for the upper and lower surface.

6.2.4 Thin airfoil theory pressure relation

Consistent with the linearization of the boundary conditions, a useful relation between the pressure and velocity can also be obtained. For incompressible flows, the exact relation between pressure and velocity is:

$$C_p = 1 - \frac{\hat{V}^2}{V_\infty^2} \quad (6-21)$$

and we expand the velocity considering disturbances to the freestream velocity using the approximations discussed above:

$$V^2 = (V_\infty \cos \alpha + u)^2 + (V_\infty \sin \alpha + v)^2$$

Expanding:

$$V^2 = V_\infty^2 + 2V_\infty u + u^2 + (V_\infty \alpha)^2 + 2V_\infty \alpha v + v^2 \quad (6-22)$$

and dividing by V_∞^2 we get:

$$\frac{V^2}{V_\infty^2} = 1 + 2\frac{u}{V_\infty} + \frac{u^2}{V_\infty^2} + \alpha^2 + 2\alpha \frac{v}{V_\infty} + \frac{v^2}{V_\infty^2} \quad (6-23)$$

Substituting into the C_p relation, Eqn. (6-21), we get:

$$\begin{aligned} C_p &= 1 - \frac{\hat{V}^2}{V_\infty^2} = 1 - \left(1 + 2\frac{u}{V_\infty} + \frac{u^2}{V_\infty^2} + \alpha^2 + 2\alpha \frac{v}{V_\infty} + \frac{v^2}{V_\infty^2} \right) \\ &= 1 - 1 - 2\frac{u}{V_\infty} - \frac{u^2}{V_\infty^2} - \alpha^2 - 2\alpha \frac{v}{V_\infty} - \frac{v^2}{V_\infty^2} \end{aligned} \quad (6-24)$$

and if $\alpha, u/V_\infty$ and v/V_∞ are $\ll 1$, then the last four terms can be neglected in comparison with the third term. The final result is:

$$C_p = -2\frac{u}{V_\infty} \quad (6-25)$$

This is the linearized or “thin airfoil theory pressure formula”. From experience gained comparing various computational results, I’ve found that this formula is a slightly more severe restriction on the accuracy of the solution than the linearized boundary condition. Equation (6-25) shows that under the small disturbance approximation, the pressure is a linear function of u , and we can add the C_p contribution from thickness, camber, and angle of attack by superposition. A similar derivation can be used to show that Eq. (6-25) is also valid for compressible flow up to moderate supersonic speeds.

6.2.5 Delta Cp due to camber/alpha (thickness cancels)

Next, we make use of the result in Eq. (6-25) to obtain a formula for the load distribution on the wing:

$$DC_p = C_{p_{\text{LOWER}}} - C_{p_{\text{UPPER}}} \quad (6-26)$$

Using superposition, the pressures can be obtained as the contributions from wing thickness, camber, and angle of attack effects:

$$\begin{aligned} C_{p_{\text{LOWER}}} &= C_{p_{\text{THICKNESS}}} + C_{p_{\text{CAMBER}}} + C_{p_{\text{ANGLE OF ATTACK}}} \\ C_{p_{\text{UPPER}}} &= C_{p_{\text{THICKNESS}}} - C_{p_{\text{CAMBER}}} - C_{p_{\text{ANGLE OF ATTACK}}} \end{aligned} \quad (6-27)$$

so that:

$$\begin{aligned} DC_p &= \left(C_{p_{\text{THICKNESS}}} + C_{p_{\text{CAMBER}}} + C_{p_{\text{ANGLE OF ATTACK}}} \right) \\ &\quad - \left(C_{p_{\text{THICKNESS}}} - C_{p_{\text{CAMBER}}} - C_{p_{\text{ANGLE OF ATTACK}}} \right) \quad (6-28) \\ &= 2 \left(C_{p_{\text{CAMBER}}} + C_{p_{\text{ANGLE OF ATTACK}}} \right) \end{aligned}$$

Equation (6-28) demonstrates that for cases where the linearized pressure coefficient relation is valid, *thickness does not contribute to lift to 1st order in the velocity disturbance!*

The importance of this analysis is that we have shown:

1. how the lifting effects can be obtained without considering thickness, and
2. that the cambered surface boundary conditions can be applied on a flat coordinate surface, resulting in an easy to apply boundary condition.

The principles demonstrated here for transfer and linearization of boundary conditions can be applied in a variety of situations other than the application in vortex lattice methods. Often this idea can be used to handle complicated geometries that can't easily be treated exactly.

The analysis here produced an entirely consistent problem formulation. This includes the linearization of the boundary condition, the transfer of the boundary condition, and the approximation between velocity and pressure. All approximations are consistent with each other. Improving one of these approximations without improving them all in a consistent manner may actually lead to worse results. Sometimes you can make agreement with data better, sometimes it may get worse. You have to be careful when trying to improve theory on an *ad hoc* basis.

6.3 Vortex Theorems

In using vortex singularities to model lifting surfaces, we need to review some properties of vortices. The key properties are defined by the so-called vortex theorems. These theorems are associated with the names of Kelvin and Helmholtz, and are proven in Karamcheti.¹³ Three important results are:

1. Along a vortex line (tube) the circulation, Γ , is constant.
2. A vortex filament (or line) cannot begin or end abruptly in a fluid. The vortex line must i) be closed, ii) extend to infinity, or iii) end at a solid boundary. Furthermore, the circulation, Γ , about any section is the vortex strength.
3. An initially irrotational, inviscid flow will remain irrotational.

Related to these theorems we state an important result:

- A sheet of vortices can support a jump in tangential velocity [i.e. a force], while the normal velocity is continuous. This means you can use a vortex sheet to represent a lifting surface.

6.4 Biot-Savart Law

We know that a two-dimensional vortex singularity satisfies Laplace's equation (i.e. a point vortex):

$$\mathbf{V} = \frac{\Gamma}{2\pi r} \mathbf{e}_\theta \quad (6-29)$$

where V_θ is the irrotational vortex flow illustrated in Fig. 6-3.

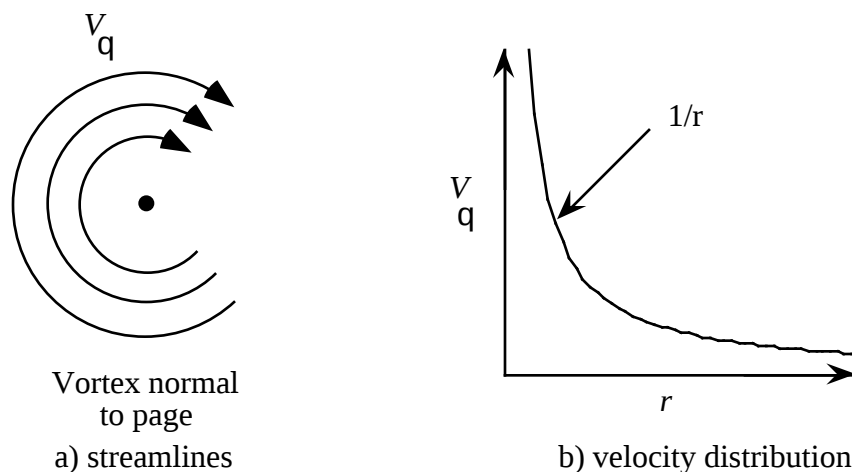


Figure 6-3. The point vortex.

What is the extension of the point vortex idea to the case of a general three-dimensional vortex filament? Consider the flowfield induced by the vortex filament shown in Fig. 6-4, which defines the nomenclature.

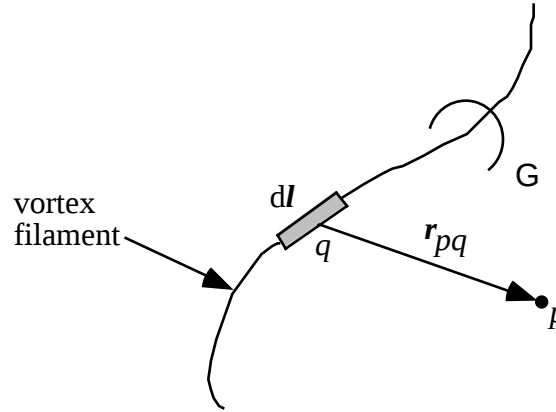


Figure 6-4. General three-dimensional vortex filament.

The mathematical description of the flow induced by this filament is given by the Biot-Savart law (see Karamcheti,¹³ pages 518-534). It states that the increment in velocity $d\mathbf{V}$ at a point p due to a segment of a vortex filament $d\mathbf{l}$ at q is:

$$d\mathbf{V}_p = \frac{G}{4\pi} \frac{d\mathbf{l} \times \mathbf{r}_{pq}}{|\mathbf{r}_{pq}|^3}. \quad (6-30)$$

To obtain the velocity induced by the entire length of the filament, integrate over the length of the vortex filament (or line) recalling that G is constant. We obtain:

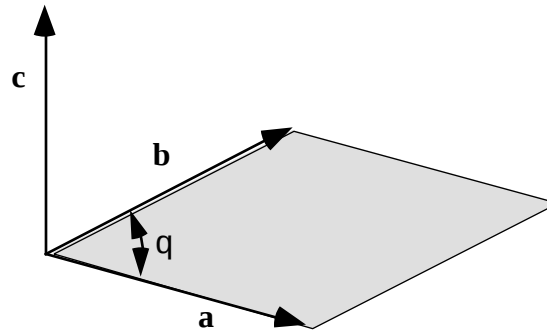
$$\bar{\mathbf{V}}_p = \frac{G}{4\pi} \oint \frac{d\bar{\mathbf{l}} \times \bar{\mathbf{r}}_{pq}}{|\bar{\mathbf{r}}_{pq}|^3}. \quad (6-31)$$

To illustrate the evaluation of this integral we give the details for several important examples. The vector cross product definition is reviewed in the sidebar below. Reviewing the cross product properties we see that the velocity direction $d\mathbf{V}_p$ induced by the segment of the vortex filament $d\mathbf{l}$ is perpendicular to the plane defined by $d\mathbf{l}$ and $\mathbf{r}_{pq}$, and its magnitude is computed from Eq. (6-31).

Case #1: the infinitely long straight vortex.

In the first illustration of the computation of the induced velocity using the Biot-Savart Law, we consider the case of an infinitely long straight vortex filament. The notation is given in Fig. 6-5.

A review: the meaning of the cross product. What does $\mathbf{a} \times \mathbf{b}$ mean? Consider the following sketch:



Here, the vectors $\mathbf{a}$ and $\mathbf{b}$ form a plane, and the result of the cross product operation is a vector $\mathbf{c}$, where $\mathbf{c}$ is perpendicular to the plane defined by $\mathbf{a}$ and $\mathbf{b}$. The value is given by:

$$c = \mathbf{a} \times \mathbf{b} = |\mathbf{a}||\mathbf{b}|\sin q$$

and $\mathbf{e}$ is perpendicular to the plane of $\mathbf{a}$ and $\mathbf{b}$. One consequence of this is that if $\mathbf{a}$ and $\mathbf{b}$ are parallel, then $\mathbf{a} \times \mathbf{b} = 0$.

Also: $|\mathbf{a} \times \mathbf{b}| = \text{area of the parallelogram}$

and

$$\mathbf{a} \times \mathbf{b} = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ A_x & A_y & A_z \\ B_x & B_y & B_z \end{vmatrix} = (A_y B_z - A_z B_y)\mathbf{i} - (A_x B_z - A_z B_x)\mathbf{j} + (A_x B_y - A_y B_x)\mathbf{k}$$

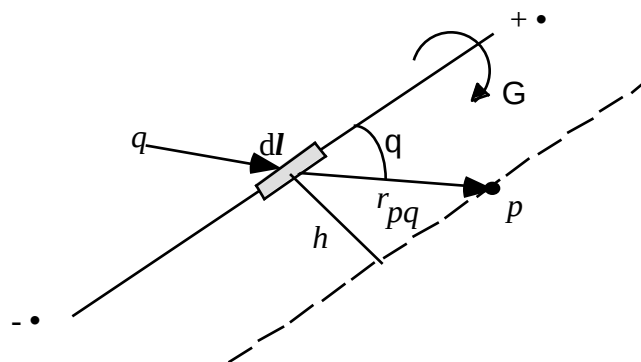


Figure 6-5. The infinitely long straight vortex.

Now consider the numerator in Eq. (6-30) given above using the definition of the cross product:

$$d\mathbf{l} \times \mathbf{r}_{pq} = |d\mathbf{l}| |\mathbf{r}_{pq}| \sin \alpha \frac{\mathbf{V}}{|\mathbf{V}|} \quad (6-32)$$

so that the entire expression becomes

$$\begin{aligned} d\mathbf{V}_p &= \frac{G}{4\pi} \frac{|d\mathbf{l}| |\mathbf{r}_{pq}| \sin \alpha}{|\mathbf{r}_{pq}|^3} \frac{\mathbf{V}}{|\mathbf{V}|} \\ &= \frac{G}{4\pi} \frac{\mathbf{V} \sin \alpha}{|\mathbf{V}|} \frac{dl}{r_{pq}^2} \end{aligned} \quad (6-33)$$

Next, simplify the above expressions so they can be readily evaluated. Use the nomenclature shown in Fig. 6-6.

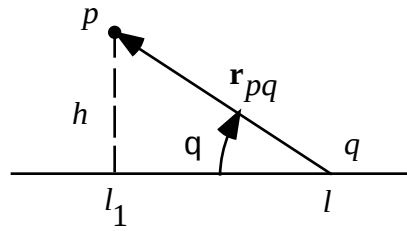


Figure 6-6. Relations for the solution of the Biot-Savart Law.

Using the notation in Fig. 6-6 to find the relations for $\mathbf{r}_{pq}$ and $d\mathbf{l}$. First we see that:

$$h = r_{pq} \sin \alpha \quad (6-34)$$

or

$$r_{pq} = \frac{h}{\sin \alpha} \quad (6-35)$$

and

$$l_1 - l = \frac{h}{\tan \alpha} = h \cot \alpha. \quad (6-36)$$

Next look at changes with α . Start by taking the differential of Eq. (6-36):

$$d(l_1 - l) = d\left(\frac{h}{\tan \alpha}\right) = d(h \cot \alpha)$$

which is

$$-dl = h d(\cot \alpha) \quad (6-37)$$

$$-\frac{dl}{d\alpha} = h \frac{d}{d\alpha}(\cot \alpha) = -h \operatorname{cosec}^2 \alpha$$

and we can now write dl in terms of $d\alpha$:

$$dl = h \operatorname{cosec}^2 \alpha d\alpha \quad (6-38)$$

so that the Biot-Savart Law gives:

$$\begin{aligned}
 d\mathbf{V}_p &= \frac{G}{4\pi} \frac{\mathbf{V}}{|\mathbf{V}|} \frac{\sin q \, dl}{r_{pq}^2} \\
 &= \frac{G}{4\pi} \frac{\mathbf{V}}{|\mathbf{V}|} \frac{\sin q \, h \operatorname{cosec}^2 q \, dq}{\frac{h^2}{\sin^2 q}} \quad (6-39) \\
 &= \frac{G}{4\pi} \frac{\sin q}{h} dq \quad (\text{direction understood})
 \end{aligned}$$

Thus we integrate:

$$\mathbf{V}_p = \frac{G}{4\pi} \int \frac{\sin q}{h} dq = \frac{G}{4\pi h} \int_{q=0}^{q=p} \sin q \, dq \quad (6-40)$$

where here the limits of integration would change if you were to consider a finite straight length.

Carrying out the integration:

$$\begin{aligned}
 \mathbf{V}_p &= \frac{G}{4\pi h} [-\cos q]_{q=0}^{q=p} \\
 &= \frac{G}{4\pi h} [-(-1) - (-1)]
 \end{aligned}$$

and:

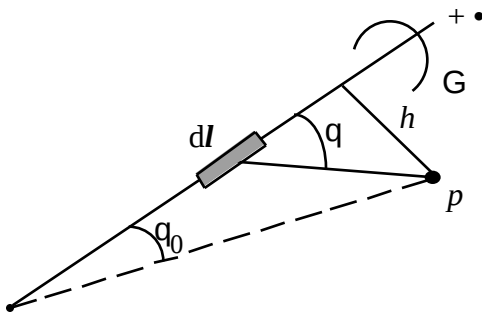
$$\mathbf{V}_p = \frac{G}{2\pi h} \quad (6-41)$$

which agrees with the two dimensional result.

Although a vortex cannot end in a fluid, we can construct expressions for infinitely long vortex lines made up of a series of connected straight line segments by combining expressions developed using the method illustrated here. To do this we simply change the limits of integration. Two cases are extremely useful for construction of vortex systems, and the formulas are given here without derivation.

Case #2: the semi-infinite vortex.

This expression is useful for modeling the vortex extending from the wing to downstream infinity.

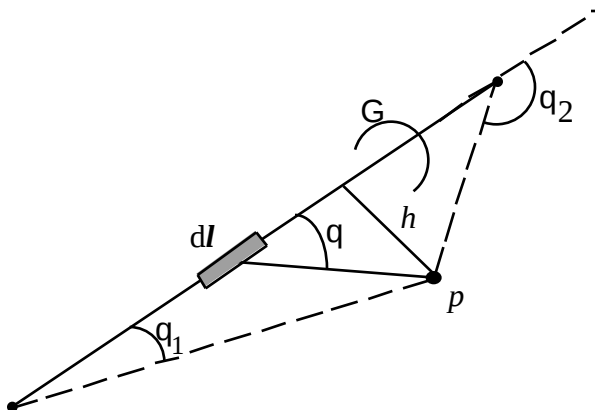


$$\mathbf{V}_p = \frac{G}{4\pi h} (1 + \cos q_0) \mathbf{e} \quad (6-42)$$

Figure 6-7. The semi-infinite vortex.

Case #3: the finite vortex.

This expression can be used to model the vortex on a wing, and can be joined with two semi-infinite vortices to form a vortex of infinite length, satisfying the vortex theorems.



$$\mathbf{V}_p = \frac{G}{4\pi h} (\cos q_1 - \cos q_2) \mathbf{e} \quad (6-43)$$

Figure 6-8. The finite vortex.

Systems of vortices can be built up using Eqns. (6-42) and (6-43) and the vortex theorems. The algebra can become tedious, but there are no conceptual difficulties.

6.5 The Horseshoe Vortex

There is a specific form of vortex used in the traditional vortex lattice method. We now use the expressions developed from the Biot-Savart Law to create a “horseshoe vortex”, which extends from downstream infinity to a point in the field “A”, then from point “A” to point “B,” and another vortex from point “B” downstream to infinity. The velocity induced by this vortex is the sum of the three parts. The basic formulas were presented in the previous section. Here we extend the analysis of the previous section, following the derivation and notation of Bertin and Smith.¹⁴ In particular, the directions of the induced flow are made more precise. Our goal is to obtain the expression for the velocity field at a general point in space (x,y,z) due to the specified horseshoe vortex.

To create a horseshoe vortex we will use three straight line vortices: one finite length vortex and two semi-infinite vortices. This is illustrated in Fig. 6-9. To start our analysis, we rewrite the Biot-Savart Law in the Bertin and Smith notation,¹⁴ which is given in Fig. 6-10.

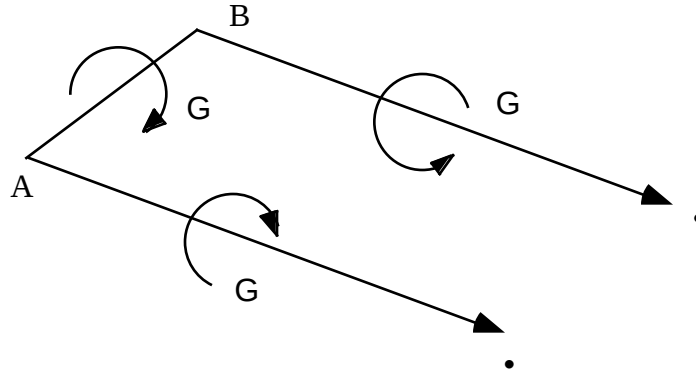


Figure 6-9. The horseshoe vortex.

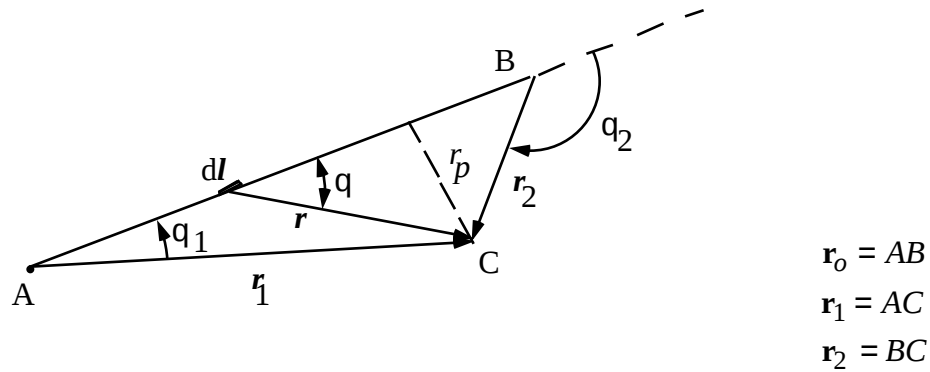


Figure 6-10. Nomenclature for induced velocity calculation.

The next step is to relate the angles to the vector definitions. The definition of the dot product is used:

$$\mathbf{A} \cdot \mathbf{B} = |\mathbf{A}| |\mathbf{B}| \cos q \quad (6-44)$$

so that:

$$\begin{aligned} \cos q_1 &= \frac{\mathbf{r}_0 \cdot \mathbf{r}_1}{|\mathbf{r}_0| |\mathbf{r}_1|} \\ \cos q_2 &= \frac{\mathbf{r}_0 \cdot \mathbf{r}_2}{|\mathbf{r}_0| |\mathbf{r}_2|} \end{aligned} \quad (6-45)$$

Next, we put these relations into the formula for the finite length vortex segment formula for the induced velocity field given above in Eq. (6-43).

$$\mathbf{V}_p = \frac{G}{4\pi |\mathbf{r}_p|} (\cos q_1 - \cos q_2) \mathbf{e} \quad (6-46)$$

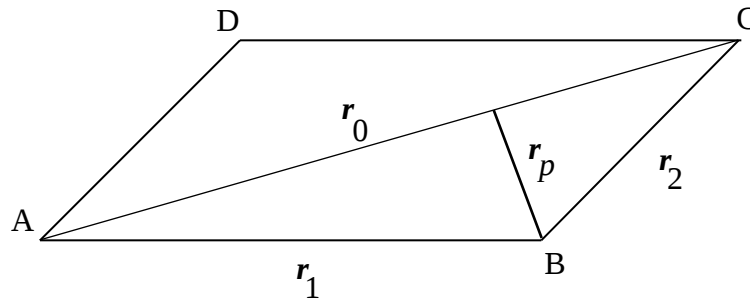
so that, substituting using the definition in Eq. (6-44), we get

$$\mathbf{V}_p = \frac{G}{4\pi} \frac{|\mathbf{r}_0|}{|\mathbf{r}_1| |\mathbf{r}_2|} \frac{\hat{\mathbf{e}}_{\mathbf{r}_0} \cdot \mathbf{r}_1}{|\mathbf{r}_0| |\mathbf{r}_1|} - \frac{\mathbf{r}_0 \cdot \mathbf{r}_2}{|\mathbf{r}_0| |\mathbf{r}_2|} \frac{\hat{\mathbf{e}}_{\mathbf{r}_0} \cdot \mathbf{r}_1}{|\mathbf{r}_1| |\mathbf{r}_2|} \quad (6-47)$$

Bertin and Smith¹⁴ use the relation:

$$|\mathbf{r}_p| = \frac{|\mathbf{r}_1 \times \mathbf{r}_2|}{|\mathbf{r}_0|} |\mathbf{r}_p| = \frac{|\mathbf{r}_1 \times \mathbf{r}_2|}{|\mathbf{r}_0|}$$

and we need to demonstrate that this is true. Consider the parallelogram ABCD shown in the sketch:



by definition: $|\mathbf{r}_1 \times \mathbf{r}_2| = A_p$, is the area of the parallelogram.

Similarly, the area of ABC is:

$$\begin{aligned} A_{ABC} &= \frac{1}{2} b h \\ &= \frac{1}{2} |\mathbf{r}_0| |\mathbf{r}_p| \end{aligned}$$

The total area of the parallelogram can be found from both formulas, and by equating these two areas we obtain the expression we are trying to get:

$$\begin{aligned} 2A_{ABC} &= A_p \\ |\mathbf{r}_0| |\mathbf{r}_p| &= |\mathbf{r}_1 \times \mathbf{r}_2| \end{aligned}$$

which can be rewritten to obtain the expression given by Bertin and Smith:¹⁴

$$|\mathbf{r}_p| = \frac{|\mathbf{r}_1 \times \mathbf{r}_2|}{|\mathbf{r}_0|}$$

Collecting terms and making use of the vector identity provided in the sidebar, we obtain the Bertin and Smith statement of the Biot-Savart Law for a finite length vortex segment:

$$\mathbf{V}_p = \frac{G}{4\pi} \frac{\mathbf{r}_1 \times \mathbf{r}_2}{|\mathbf{r}_1 \times \mathbf{r}_2|^2} \int \hat{\mathbf{r}}_0 \frac{\hat{\mathbf{e}}_1}{|\hat{\mathbf{e}}_1|} - \frac{\mathbf{r}_2}{|\mathbf{r}_2|^3} \cdot \quad (6-48)$$

For a single infinite length horseshoe vortex we will use three segments, each using the formula given above. The nomenclature is given in the sketch below. The primary points are the

connecting points A and B . Between A and B we use a finite length vortex which is considered a “bound” vortex, and from A to infinity and B to infinity we define “trailing” vortices that are parallel to the x -axis.

The general expression for the velocity at a point x, y, z due to a horseshoe vortex at (x_{1n}, y_{1n}, z_{1n}) , (x_{2n}, y_{2n}, z_{2n}) with trailing vortices parallel to the x -axis is (from Bertin and Smith¹⁴):

$$\mathbf{V} = \mathbf{V}_{AB} + \mathbf{V}_{A\infty} + \mathbf{V}_{B\infty} \quad (6-49)$$

where the total velocity is the sum of the contributions from the three separate straight line vortex segments making up the horseshoe vortex, as shown in Fig. 6-11.

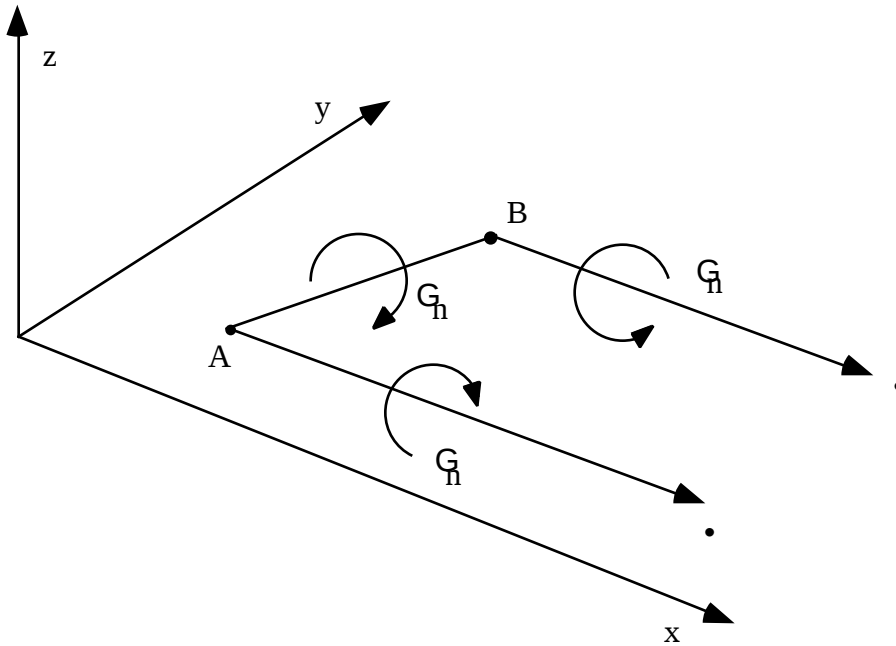


Figure 6-11. Definitions for notation used in induced velocity expressions.

The corner points of the vortex, A and B , are arbitrary, and are given by:

$$\begin{aligned} A &= A(x_{1n}, y_{1n}, z_{1n}) \\ B &= B(x_{2n}, y_{2n}, z_{2n}) \end{aligned} \quad (6-50)$$

We now write the expression for the velocity field at a general point in space (x, y, z) due to the horseshoe vortex system. At $C(x, y, z)$ find the induced velocity due to each vortex segment. Start with AB , and use Fig. 6-12.

Now we define the vectors as:

$$\begin{aligned}
 \mathbf{r}_0 &= (x_{2n} - x_{1n})\mathbf{i} + (y_{2n} - y_{1n})\mathbf{j} + (z_{2n} - z_{1n})\mathbf{k} \\
 \mathbf{r}_1 &= (x - x_{1n})\mathbf{i} + (y - y_{1n})\mathbf{j} + (z - z_{1n})\mathbf{k} \\
 \mathbf{r}_2 &= (x - x_{2n})\mathbf{i} + (y - y_{2n})\mathbf{j} + (z - z_{2n})\mathbf{k}
 \end{aligned}
 \tag{6-51}$$

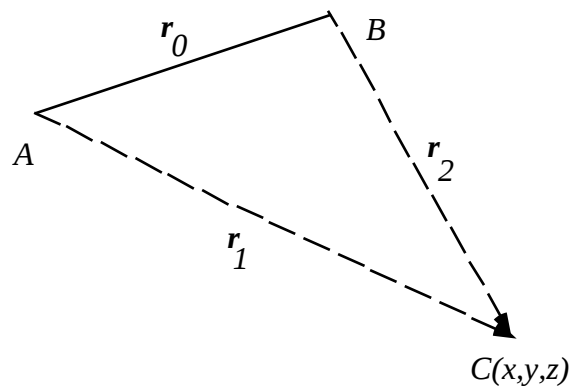


Figure 6-12. Velocity induced at Point C due to the vortex between A and B.

and we simply substitute into:

$$\mathbf{V}_C = \frac{G}{4\pi} \underbrace{\frac{\mathbf{r}_1 \times \mathbf{r}_2}{|\mathbf{r}_1 \times \mathbf{r}_2|^2}}_{\mathbf{Y}} \underbrace{\left[\frac{\hat{\mathbf{e}}_1}{|\mathbf{r}_1|} - \frac{\hat{\mathbf{e}}_2}{|\mathbf{r}_2|} \right]}_{\mathbf{W}}
 \tag{6-52}$$

Considering the bound vortex on AB first we obtain,

$$\mathbf{V}_{AB} = \frac{G}{4\pi} \mathbf{W} \mathbf{Y}
 \tag{6-53}$$

where Y and W are lengthy expressions. By following the vector definitions, Eqn. (6-53) can be written in Cartesian coordinates. The vector Y is:

$$\begin{aligned}
 \mathbf{Y} &= \frac{\mathbf{r}_1 \times \mathbf{r}_2}{|\mathbf{r}_1 \times \mathbf{r}_2|^2} \\
 &= \frac{[(y - y_{1n})(z - z_{2n}) - (y - y_{2n})(z - z_{1n})]\mathbf{i} - [(x - x_{1n})(z - z_{2n}) - (x - x_{2n})(z - z_{1n})]\mathbf{j} + [(x - x_{1n})(y - y_{2n}) - (x - x_{2n})(y - y_{1n})]\mathbf{k}}{[(y - y_{1n})(z - z_{2n}) - (y - y_{2n})(z - z_{1n})]^2 + [(x - x_{1n})(z - z_{2n}) - (x - x_{2n})(z - z_{1n})]^2 + [(x - x_{1n})(y - y_{2n}) - (x - x_{2n})(y - y_{1n})]^2}
 \end{aligned}
 \tag{6-54}$$

The scalar portion of the expression, W , is:

$$W = \frac{\hat{\mathbf{E}}}{\hat{\mathbf{E}}} \cdot \frac{\mathbf{r}_1}{|\mathbf{r}_1|} - \mathbf{r}_o \cdot \frac{\mathbf{r}_2}{|\mathbf{r}_2|} = \frac{[(x_{2n} - x_{1n})(x - x_{1n}) + (y_{2n} - y_{1n})(y - y_{1n}) + (z_{2n} - z_{1n})(z - z_{1n})]}{\sqrt{(x - x_{1n})^2 + (y - y_{1n})^2 + (z - z_{1n})^2}} - \frac{[(x_{2n} - x_{1n})(x - x_{2n}) + (y_{2n} - y_{1n})(y - y_{2n}) + (z_{2n} - z_{1n})(z - z_{2n})]}{\sqrt{(x - x_{2n})^2 + (y - y_{2n})^2 + (z - z_{2n})^2}} \quad (6-55)$$

We find the contributions of the trailing vortex legs using the same formula, but redefining the points 1 and 2. Then, keeping the 1 and 2 notation, define a downstream point, “3” and let x_3 go to infinity. Thus the trailing vortex legs are given by:

$$\mathbf{V}_{A\bullet} = \frac{G_h}{4\pi} \left[\frac{(z - z_{1n})\mathbf{j} + (y_{1n} - y)\mathbf{k}}{[(z - z_{1n})^2 + (y_{1n} - y)^2]^{3/2}} \right]_{x_{1n}}^{x_{2n}} + \frac{x - x_{1n}}{\sqrt{(x - x_{1n})^2 + (y - y_{1n})^2 + (z - z_{1n})^2}} \quad (6-56)$$

and

$$\mathbf{V}_{B\bullet} = -\frac{G_h}{4\pi} \left[\frac{(z - z_{2n})\mathbf{j} + (y_{2n} - y)\mathbf{k}}{[(z - z_{2n})^2 + (y_{2n} - y)^2]^{3/2}} \right]_{x_{2n}}^{x_{3n}} + \frac{x - x_{2n}}{\sqrt{(x - x_{2n})^2 + (y - y_{2n})^2 + (z - z_{2n})^2}} \quad (6-57)$$

Note that G_h is contained linearly in each expression, so that the expression given above can be arranged much more compactly by using (6-49) with (6-52), (6-56), and (6-57) as:

$$\mathbf{V}_m = C_{mn} G_h \quad (6-58)$$

and C_{mn} is an influence coefficient for the n^{th} horseshoe vortex effect at the location m , including all three segments.

Now that we can compute the induced velocity field of a horseshoe vortex, we need to decide where to place the horseshoe vortices to represent a lifting surface.

6.6 Selection of Control Point/Vortex Location

Since we are interested in using the horseshoe vortex defined above to represent a lifting surface, we need to examine exactly how this might be done. In particular: where do you locate the vortex, and where do you locate a control point to satisfy the surface boundary condition? Tradition has been to determine their locations by comparison with known results. In particular, we use two dimensional test cases, and then apply them directly to the three dimensional case.

An alternate distribution based on numerical properties of quadrature formulas has been derived by Lan. Section 6.11 will demonstrate the use of his vortex/control point locations in the inverse case, where the pressure is given, and the shape of the surface is sought.

a) Simplest Approach: A Flat Plate

Consider representing the flow over a flat plate airfoil by a single vortex and control point. Comparing with the known result from thin airfoil theory we determine the spacing between the vortex and control point which produces a lift identical with the thin airfoil theory value.

Consider the flat plate as sketched in Fig. 6-13.

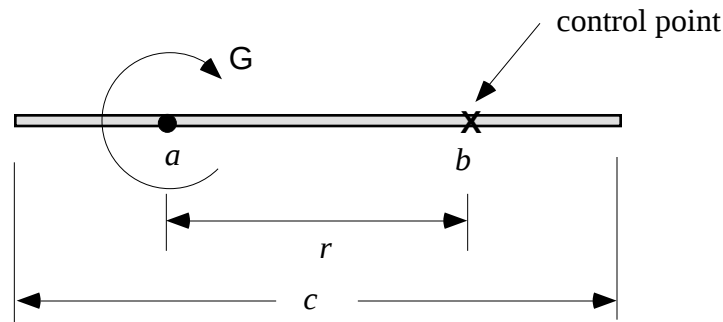


Figure 6-13. The notation for control point and vortex location analysis.

The velocity at the control point, cp , due to the point vortex is:

$$v_{cp} = - \frac{G}{2\pi r} . \quad (6-59)$$

The flow tangency condition was given above as:

$$\frac{v_{BC}}{V_{\infty}} = \frac{df_c}{dx} - a \quad (6-60)$$

and ignoring camber:

$$\frac{v_{BC}}{V_{\infty}} = - a \quad (6-61)$$

or:

$$v_{BC} = - a V_{\infty} . \quad (6-62)$$

Now, we equate v_{BC} and v_{cp} :

$$- \frac{G}{2\pi r} = - a V_{\infty} . \quad (6-63)$$

resulting in the expression for a :

$$a = \frac{G}{2\pi r V_\infty} \quad (6-64)$$

To make use of this relation, recall the Kutta-Joukowski Theorem:

$$L = \Gamma V_\infty \quad (6-65)$$

and the result from thin airfoil theory:

$$L = \underbrace{\frac{1}{2} \rho V_\infty^2}_{q_\infty} \underbrace{c}_{S_{ref}} \underbrace{2\pi a}_{C_L} \quad (6-66)$$

Equate the expressions for lift, Eqns. (6-65) and (6-66) and substitute for a using the expression in Eqn. (6-64) given above:

$$\begin{aligned} \Gamma V_\infty &= \frac{1}{2} \rho V_\infty^2 c 2\pi a \\ &= \frac{1}{2} \rho V_\infty^2 c 2\pi \frac{G}{2\pi r V_\infty} \\ 1 &= \frac{1}{2} \frac{c}{r} \end{aligned} \quad (6-67)$$

and finally:

$$r = \frac{c}{2} \quad (6-68)$$

This defines the relation between the vortex placement and the control point in order for the single vortex model to reproduce the theoretical lift of an airfoil predicted by thin airfoil theory. Since the flat plate has constant (zero) camber everywhere, this case doesn't pin down placement (distance of the vortex from the leading edge) completely. Intuitively, the vortex should be located at the quarter-chord point since that is the location of the aerodynamic center of a thin flat plate airfoil. The next example is used to determine the placement of the vortex.

b) Determine placement of the vortex using parabolic camber model.

Rewrite the velocity at the control point due to the point vortex in a little more detail, where a denotes the location of the vortex, and b the location of the control point:

$$v_{cp} = - \frac{\Gamma}{2\pi(a - b)} \quad (6-69)$$

and the boundary condition remains the same:

$$v_{BC} = V_\infty \frac{d\eta}{dx} - a \frac{d\eta}{dx} \quad (6-70)$$

Equating the above expressions (and dividing by V_∞):

$$-\frac{G}{2\rho(a-b)V_\infty} = \frac{\hat{E}}{E} \frac{df_c}{dx} - a \quad (6-71)$$

For parabolic camber,

$$f_c(x) = 4d \frac{\hat{E}}{E} \frac{x}{c} \left(c - x \right) \quad (6-72)$$

we have the slope,

$$\frac{df_c(x)}{dx} = 4d \frac{\hat{E}}{E} \left(1 - 2 \frac{x}{c} \right) \quad (6-73)$$

so that:

$$-\frac{G}{2\rho(a-b)V_\infty} = 4d \frac{\hat{E}}{E} \left(1 - 2 \frac{x}{c} \right) - a \quad (6-74)$$

Now use the result from thin airfoil theory:

$$L = \frac{1}{2} \rho V_\infty^2 c 2\pi(a + 2d)$$

and substitute for the lift from the Kutta-Joukowski theorem. We thus obtain an expression for the circulation of the vortex in terms of the angle of attack and camber:

$$G = \rho V_\infty c(a + 2d). \quad (6-75)$$

Substitute for G from (6-75) into (6-74), and satisfy the boundary condition at $x = b$:

$$-\frac{\rho V_\infty c(a + 2d)}{2\rho(b-a)V_\infty} = 4d \frac{\hat{E}}{E} \left(1 - 2 \frac{b}{c} \right) - a$$

or:

$$-\frac{1}{2} \frac{\hat{E}}{E} \frac{c}{b-a} (a + 2d) = 4d \frac{\hat{E}}{E} \left(1 - 2 \frac{b}{c} \right) - a \quad (6-76)$$

To be true for arbitrary a, d , the coefficients must be equal:

$$\begin{aligned} -\frac{1}{2} \frac{\hat{E}}{E} \frac{c}{b-a} &= -1 \\ -\frac{\hat{E}}{E} \frac{c}{b-a} &= 4 \frac{\hat{E}}{E} \left(1 - 2 \frac{b}{c} \right) \end{aligned} \quad (6-77)$$

and we solve for a and b . The first relation can be solved for $(b-a)$:

$$(b-a) = \frac{c}{2} \quad (6-78)$$

and we obtain the same results obtained above, validating our previous analysis ($r = c/2$).

Now, rewrite the 2nd equation:

$$-c = 4 \left(1 - \frac{\hat{E}}{2} \right) \frac{\hat{E} b}{E_c} (b - a)$$

or:

$$-c = 4 \left(1 - \frac{\hat{E}}{2} \right) \frac{\hat{E} b}{E_c} \frac{c}{2} \quad (6-79)$$

and solve for b/c :

$$\begin{aligned} -1 &= 2 \left(1 - \frac{\hat{E}}{2} \right) \frac{\hat{E} b}{E_c} \\ -\frac{1}{2} &= 1 - \frac{\hat{E} b}{E_c} \\ 2 \frac{b}{c} &= 1 + \frac{1}{2} = \frac{3}{2} \\ \frac{b}{c} &= \frac{1}{2} \cdot \frac{3}{2} = \frac{3}{4}, \end{aligned} \quad (6-80)$$

and use this to solve for a/c starting with Eqn. (6-78) :

$$(b - a) = \frac{c}{2}$$

or:

$$\frac{b}{c} - \frac{a}{c} = \frac{1}{2} \quad (6-81)$$

and:

$$\frac{a}{c} = \frac{b}{c} - \frac{1}{2} = \frac{3}{4} - \frac{1}{2}.$$

Finally:

$$\frac{a}{c} = \frac{1}{4}. \quad (6-82)$$

Thus the vortex is located at the 1/4 chord point, and the control point is located at the 3/4 chord point. Naturally, this is known as the “1/4 - 3/4 rule.” It’s not a theoretical law, simply a placement that works well and has become a rule of thumb. It was discovered by Italian Pistolesi. Mathematical derivations of more precise vortex/control point locations are available (see Lan⁶), but the 1/4 - 3/4 rule is widely used, and has proven to be sufficiently accurate in practice.

To examine the use of these ideas we present a two-dimensional example. The airfoil is divided into a number of equal size panels. Each panel has a vortex at the quarter chord point and the non-penetration condition is satisfied at the three-quarter chord point. We use the example to illustrate the accuracy of the classical thin airfoil theory formulation. In Fig. 6-14, we compare the results obtained for a 5% circular arc camber.* Three solutions are presented. The linear theory curve uses classical thin airfoil theory with results obtained satisfying the boundary condition on the mean surface. This is compared with numerical results for the case where the boundary condition is applied exactly on the camber line, and the result obtained applying the boundary condition using the approximate method described above. The difference between placing the vortex on the actual camber surface and satisfying the boundary condition on the actual surface, and the more approximate traditional approach of locating the vortex and control point on the mean surface is extremely small.

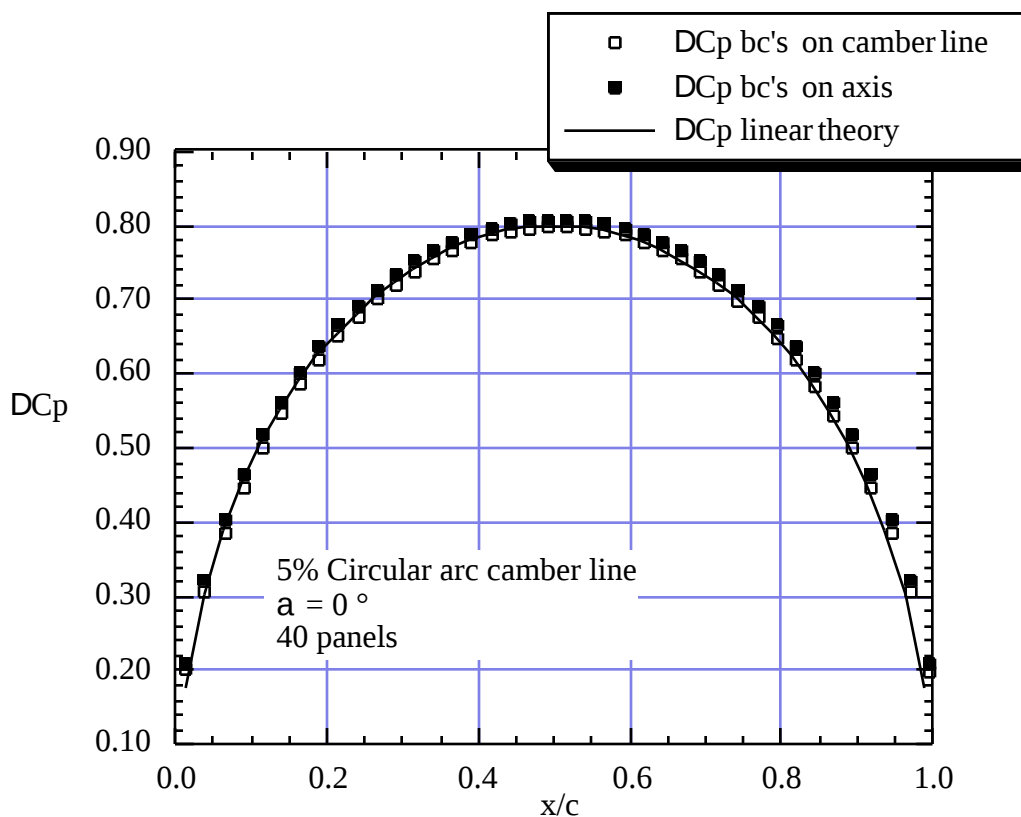


Figure 6-14. Comparison in 2D of the 1/4-3/4 rule for vortex-control point locations with linear theory, and including a comparison between placing the vortex and control point on the camber line or on the axis.

* A relatively large camber for a practical airfoil, the NACA 4412 example we used in Chapter 4 was an extreme case, and it has 4% camber.

6.7 The Classical Vortex Lattice Method

There are many different vortex lattice schemes. In this section we describe the “classical” implementation. Knowing that vortices can represent lift from our airfoil analysis, and that one approach is to place the vortex and then satisfy the boundary condition using the “1/4 - 3/4 rule,” we proceed as follows:

1. Divide the planform up into a lattice of quadrilateral panels, and put a horseshoe vortex on each panel.
2. Place the bound vortex of the horseshoe vortex on the 1/4 chord element line of each panel.
3. Place the control point on the 3/4 chord point of each panel at the midpoint in the spanwise direction (sometimes the lateral panel centroid location is used) .
4. Assume a flat wake in the usual classical method.
5. Determine the strengths of each Γ_n required to satisfy the boundary conditions by solving a system of linear equations. The implementation is shown schematically in Fig. 6-15.

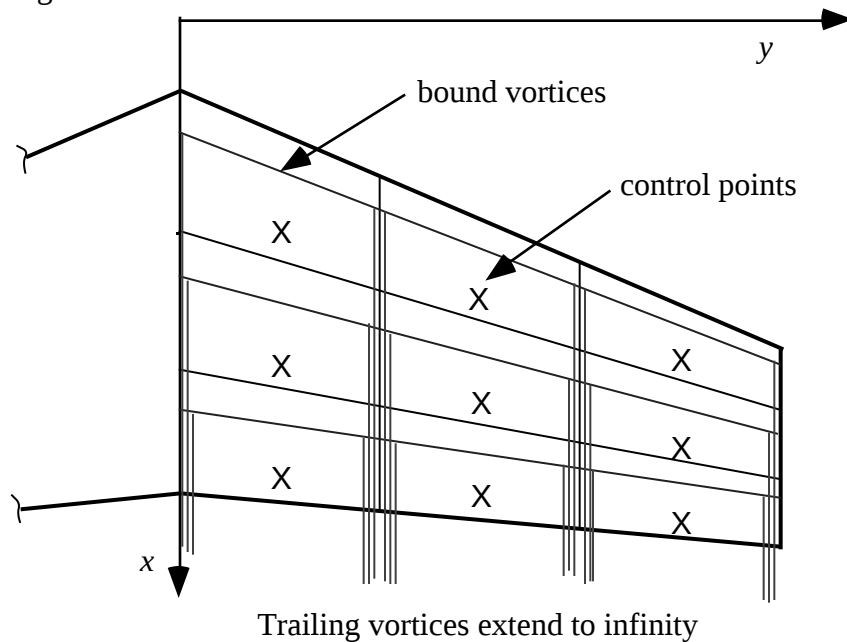


Figure 6-15. The horseshoe vortex layout for the classical vortex lattice method.

Note that the lift is on the bound vortices. To understand why, consider the vector statement of the Kutta-Joukowski Theorem, $\mathbf{F} = \rho \mathbf{V} \times \mathbf{G}$. Assuming the freestream velocity is the primary contributor to the velocity, the trailing vortices are parallel to the velocity vector and hence the force on the trailing vortices are zero. More accurate methods find the wake deformation required to eliminate the force in the presence of the complete induced flowfield.

Next, we derive the mathematical statement of the classical vortex lattice method described above. First, recall that the velocity induced by a single horseshoe vortex is

$$\mathbf{V}_m = \mathbf{C}_{m,n} \mathbf{G}_n. \quad (6-60)$$

This is the velocity induced at the point m due to the n^{th} horseshoe vortex, where $\mathbf{C}_{m,n}$ is a vector, and the components are given by Equations 6-54, 6-58 and 6-59.

The total induced velocity at m due to the $2N$ vortices (N on each side of the planform) is:

$$\mathbf{V}_{m_{ind}} = u_{m_{ind}} \mathbf{i} + v_{m_{ind}} \mathbf{j} + w_{m_{ind}} \mathbf{k} = \sum_{n=1}^{2N} \mathbf{C}_{m,n} \mathbf{G}_n. \quad (6-83)$$

The solution requires the satisfaction of the boundary conditions for the total velocity, which is the sum of the induced and freestream velocity. The freestream velocity is (introducing the possibility of considering vehicles at combined angle of attack and sideslip):

$$\mathbf{V}_\infty = V_\infty \cos \alpha \cos \beta \mathbf{i} - V_\infty \sin \beta \mathbf{j} + V_\infty \sin \alpha \cos \beta \mathbf{k} \quad (6-84)$$

so that the total velocity at point m is:

$$\mathbf{V}_m = (V_\infty \cos \alpha \cos \beta + u_{m_{ind}}) \mathbf{i} + (-V_\infty \sin \beta + v_{m_{ind}}) \mathbf{j} + (V_\infty \sin \alpha \cos \beta + w_{m_{ind}}) \mathbf{k}. \quad (6-85)$$

The values of the unknown circulations, $\mathbf{G}_n$, are found by satisfying the non-penetration boundary condition at all the control points simultaneously. For steady flow this is

$$\mathbf{V} \cdot \mathbf{n} = 0 \quad (6-1)$$

where the surface is described by

$$F(x, y, z) = 0. \quad (6-86)$$

Equation (6-1) can then be written:

$$\mathbf{V} \cdot \frac{\nabla F}{|\nabla F|} = \mathbf{V} \cdot \nabla F = 0. \quad (6-87)$$

This equation provides freedom to express the surface in a number of forms. The most general form is obtained by substituting Eqn. (6-85) into Eqn. (6-87) using Eqn. (6-83). This can be written as:

$$\left[(V_\infty \cos \alpha \cos \beta + u_{m_{ind}}) \mathbf{i} + (-V_\infty \sin \beta + v_{m_{ind}}) \mathbf{j} + (V_\infty \sin \alpha \cos \beta + w_{m_{ind}}) \mathbf{k} \right] \cdot \left(\frac{\partial F}{\partial x} \mathbf{i} + \frac{\partial F}{\partial y} \mathbf{j} + \frac{\partial F}{\partial z} \mathbf{k} \right) = 0 \quad (6-88)$$

or:

$$\hat{\mathbf{E}} \cdot \left(V \cos \alpha \cos \beta + \sum_{n=1}^{2N} \hat{\mathbf{A}}_{m,n_i} C_{m,n_i} \mathbf{G}_h \right) \mathbf{i} + \sum_{n=1}^{2N} \hat{\mathbf{A}}_{m,n_j} C_{m,n_j} \mathbf{G}_h \mathbf{j} + \left(V \sin \alpha \cos \beta + \sum_{n=1}^{2N} \hat{\mathbf{A}}_{m,n_k} C_{m,n_k} \mathbf{G}_h \right) \mathbf{k} \cdot \hat{\mathbf{E}} = 0$$

$$\left(\frac{\partial F}{\partial x} \mathbf{i} + \frac{\partial F}{\partial y} \mathbf{j} + \frac{\partial F}{\partial z} \mathbf{k} \right) \cdot \hat{\mathbf{E}} = 0 \quad (6-89)$$

Carrying out the dot product operation and collecting terms:

$$\frac{\partial F}{\partial x} \left(V \cos \alpha \cos \beta + \sum_{n=1}^{2N} \hat{\mathbf{A}}_{m,n_i} C_{m,n_i} \mathbf{G}_h \right) + \frac{\partial F}{\partial y} \sum_{n=1}^{2N} \hat{\mathbf{A}}_{m,n_j} C_{m,n_j} \mathbf{G}_h \mathbf{j} + \frac{\partial F}{\partial z} \left(V \sin \alpha \cos \beta + \sum_{n=1}^{2N} \hat{\mathbf{A}}_{m,n_k} C_{m,n_k} \mathbf{G}_h \right) = 0 \quad (6-90)$$

Recall that Eqn. (6-90) is applied to the boundary condition at point m . Next, we collect terms to clearly identify the expression for the circulation. The resulting expression defines a system of equations for all the panels, and is the system of linear algebraic equations that is used to solve for the unknown values of the circulation distribution. The result is:

$$\sum_{n=1}^{2N} \hat{\mathbf{A}}_{m,n_i} \frac{\partial F}{\partial x} C_{m,n_i} + \frac{\partial F}{\partial y} C_{m,n_j} + \frac{\partial F}{\partial z} C_{m,n_k} \mathbf{G}_h = -V \left(\cos \alpha \cos \beta \frac{\partial F}{\partial x} - \sin \alpha \frac{\partial F}{\partial y} + \sin \alpha \cos \beta \frac{\partial F}{\partial z} \right) \quad (6-91)$$

$m = 1, \dots, 2N$

This is the general equation used to solve for the values of the circulation. It is arbitrary, containing effects of both angle of attack and sideslip (if the vehicle is at sideslip the trailing vortex system should be yawed to align it with the freestream).

If the surface is primarily in the x - y plane and the sideslip is zero, we can write a simpler form. In this case the natural description of the surface is

$$z = f(x, y) \quad (6-92)$$

and

$$F(x, y, z) = z - f(x, y) = 0. \quad (6-93)$$

The gradient of F becomes

$$\frac{\partial F}{\partial x} = -\frac{\partial f}{\partial x}, \quad \frac{\partial F}{\partial y} = -\frac{\partial f}{\partial y}, \quad \frac{\partial F}{\partial z} = 1. \quad (6-94)$$

Substituting into the statement of the boundary condition, Eqn. (6-91), we obtain:

$$\sum_{n=1}^{2N} \hat{\mathbf{A}}_{m,n_k} C_{m,n_k} - \frac{\partial f}{\partial x} C_{m,n_i} - \frac{\partial f}{\partial y} C_{m,n_j} \mathbf{G}_h = V \left(\cos \alpha \frac{\partial f}{\partial x} - \sin \alpha \frac{\partial f}{\partial y} \right), \quad m = 1, \dots, 2N. \quad (6-95)$$

This equation provides the solution for the vortex lattice problem.

Note that if an essentially vertical surface is of interest, the form of F is more naturally

$$F = y - g(x, z),$$

and this should be used to work out the boundary condition in a similar fashion.

To illustrate the usual method, consider the simple planar surface case, where there is no dihedral. Furthermore, recalling the example in the last section and the analysis at the beginning of the chapter, the thin airfoil theory boundary conditions can be applied on the mean surface, and not the actual camber surface. We also use the small angle approximations. Under these circumstances, Eq. (6-95) becomes:

$$w_m = \sum_{n=1}^{2N} \hat{A}_{C_{m,n_k}} G_n = V \cdot \left[\frac{\hat{E} df_c}{\hat{E} dx} - \hat{a} \right]_m. \quad (6-96)$$

Thus we have the following equation which satisfies the boundary conditions and can be used to relate the circulation distribution and the wing camber and angle of attack:

$$\sum_{n=1}^{2N} \hat{A}_{C_{m,n_k}} \frac{\hat{E} G_n}{\hat{E} V} = \frac{\hat{E} df_c}{\hat{E} dx} - \hat{a}_m \quad m = 1, \dots, 2N. \quad (6-97)$$

Equation (6-97) contains two cases:

1. *The Analysis Problem.* Given camber slopes and $\hat{a}$, solve for the circulation strengths, $(GV \cdot)$ [a system of $2N$ simultaneous linear equations].

or

2. *The Design Problem.* Given $(GV \cdot)$, which corresponds to a specified surface loading, we want to find the camber and $\hat{a}$ required to generate this loading (only requires simple algebra, no system of equations must be solved).

Notice that the way df_c/dx and $\hat{a}$ are combined illustrates that the division between camber, angle of attack and wing twist is arbitrary (twist can be considered a separate part of the camber distribution, and is useful for wing design). However, care must be taken in keeping the bookkeeping straight.

One reduction in the size of the problem is available in many cases. If the geometry is symmetrical and the camber and twist are also symmetrical, then G_n is the same on each side of the planform (but *not* the influence coefficient). Therefore, we only need to solve for N G 's, not $2N$ (this is true also if ground effects are desired, see Katz and Plotkin¹¹). The system of equations for this case becomes:

$$\sum_{n=1}^N \hat{A}_{C_{m,n_k \text{ left}}} + C_{m,n_k \text{ right}} \frac{\hat{E} G_n}{\hat{E} V} = \frac{\hat{E} df_c}{\hat{E} dx} - \hat{a}_m \quad m = 1, \dots, N. \quad (6-98)$$

This is easy. Why not just program it up ourselves? You can, but most of the work is:

- A. Automatic layout of panels for arbitrary geometry. As an example, when considering multiple lifting surfaces, the horseshoe vortices on each surface must “line up”. The downstream leg of a horseshoe vortex cannot pass through the control point of another panel.
- and
- B. Converting G_h to the aerodynamics values of interest, C_L , C_m , etc., and the spanload, is tedious for arbitrary configurations.

Nevertheless, many people (including previous students in the Applied Computational Aerodynamics class) have written **vlm** codes. The method is widely used in industry and government for aerodynamic estimates for conceptual and preliminary design predictions. The method provides good insight into the aerodynamics of wings, including interactions between lifting surfaces.

Typical analysis uses (in a design environment) include

- Predicting the configuration neutral point during initial configuration layout, and studying the effects of wing placement and canard and/or tail size and location.
- Finding the induced drag, C_{Di} , from the spanload in conjunction with farfield methods.
- With care, estimating control and device deflection effectiveness (estimates where viscous effects may be important require calibration. Some examples are shown in the next section. For example, take 60% of the inviscid value to account for viscous losses, and also realize that a deflection of $\delta_f = 20 - 25^\circ$ is about the maximum useful device deflection in practice).
- Investigating the aerodynamics of interacting surfaces.
- Finding the lift curve slope, C_{La} , approach angle of attack, etc.

Typical design applications include:

- Initial estimates of twist to obtain a desired spanload, or root bending moment.
- Starting point for finding a camber distribution in purely subsonic cases.

Before examining how well the method works, two special cases require comments. The first case arises when a control point is in line with the projection of one of the finite length vortex segments. This problem occurs when the projection of a swept bound vortex segment from one side of the wing intersects a control point on the other side. This happens frequently. The velocity induced by this vortex is zero, but the equation as usually written degenerates into a singular form, with the denominator going to zero. Thus a special form should be used. In practice, when this happens the contribution can be set to zero without invoking the special form. Figure 6-16 shows how this happens. Using the Warren 12 planform and 36 vortices on each side of the wing, we see that the projection of the line of bound vortices on the last row of the left hand side of the planform has a projection that intersects one of the control points on the right hand side.

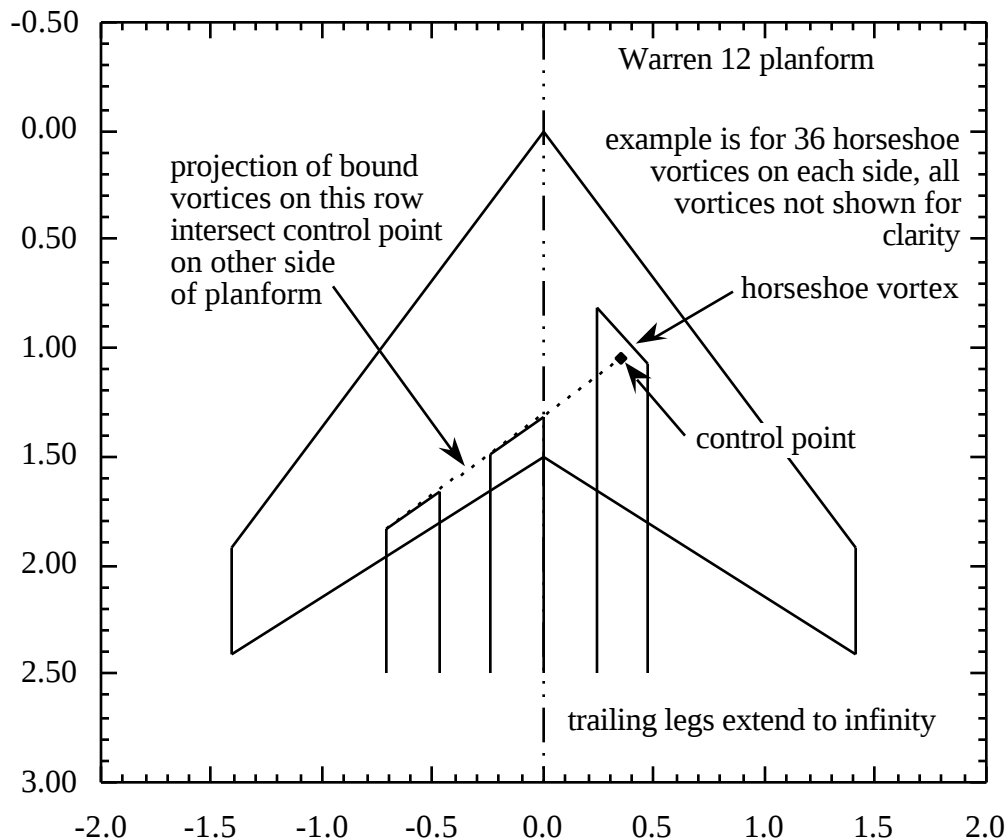


Figure 6-16. Example of case requiring special treatment, the intersection of the projection of a vortex with a control point.

A model problem illustrating this can be constructed for a simple finite length vortex segment. The velocity induced by this vortex is shown in Fig. 6-17. When the vortex is approached directly, $x/l = 0.5$, the velocity is singular for $h = 0$. However, as soon as you approach the axis ($h = 0$) off the end of the segment ($x/l > 1.0$) the induced velocity is zero. This illustrates why you can set the induced velocity to zero when this happens.

This second case that needs to be discussed arises when two or more planforms are used with this method. This is one of the most powerful applications of the vortex lattice method. However, care must be taken to make sure that the trailing vortices from the first surface do not intersect the control points on the second surface. In this case the induced velocity is in fact infinite, and the method breaks down. Usually this problem is solved by using the same spanwise distribution of horseshoe vortices on each surface. This aligns the vortex the legs, and the control points are well removed from the trailing vortices of the forward surfaces.

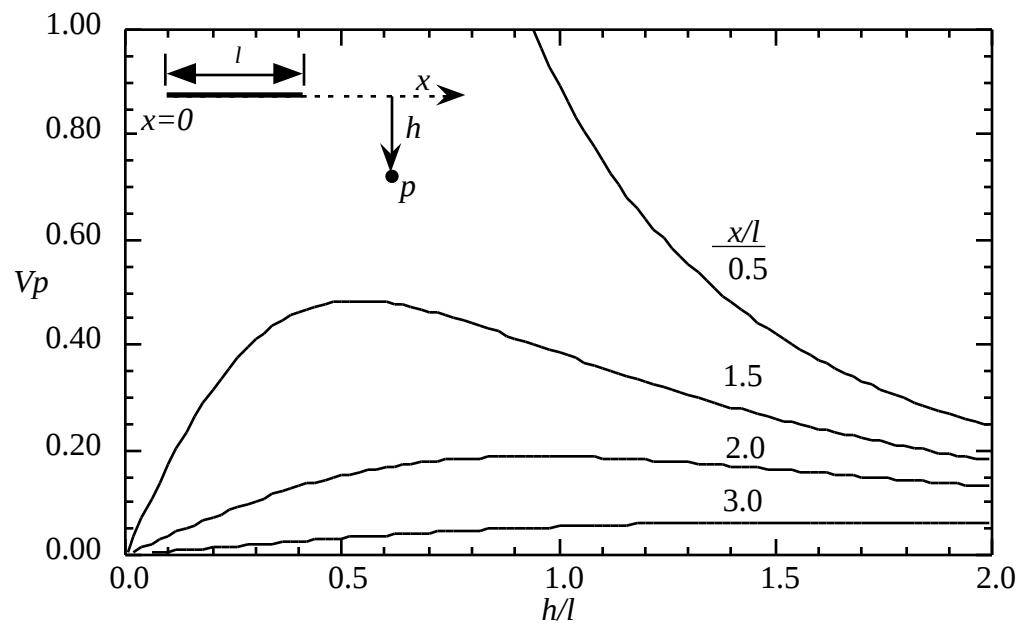


Figure 6-17. Velocity induced by finite straight line section of a vortex.

6.8 Examples of the Use and Accuracy of the vlm Method

How well does the method work? In this section we describe how the method is normally applied, and present some example results obtained using it. More examples and a discussion of the aerodynamics of wings and multiple lifting surfaces are given in Section 6.10.

The vortex lattice layout is clear for most wings and wing-tail or wing-canard configurations. The method can be used for wing-body cases by simply specifying the projected planform of the entire configuration as a flat lifting surface made up of a number of straight line segments. The exact origin of this somewhat surprising approach is unknown. The success of this approach is illustrated in examples given below.

To get good, consistent and reliable results some simple rules for panel layout should be followed. This requires that a few common rules of thumb be used in selecting the planform break points: i) the number of line segments should be minimized, ii) breakpoints should line up streamwise on front and rear portions of each planform, and should line up between planforms, iii) streamwise tips should be used, iv) small spanwise distances should be avoided by making edges streamwise if they are actually very highly swept, and v) trailing vortices from forward surfaces cannot hit the control point of an aft surface. Figure 6-18 illustrates these requirements.

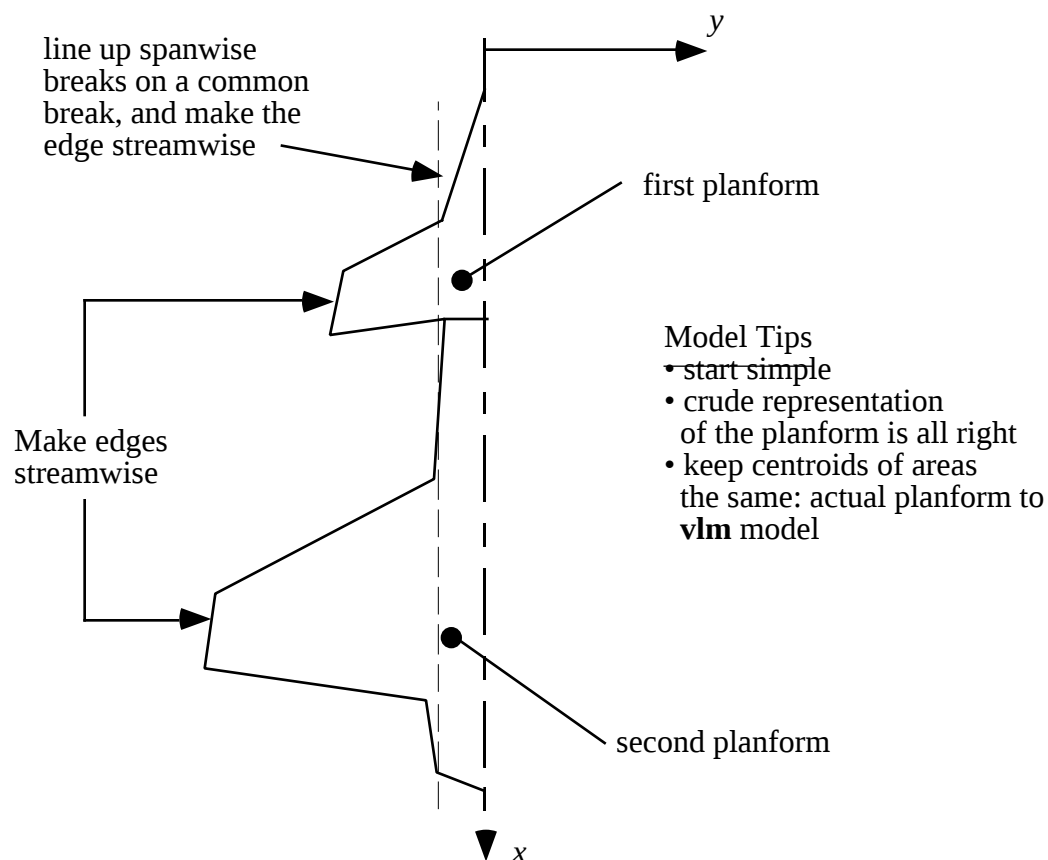


Figure 6-18. Example of a **vlm** model of an aircraft configuration. Note that one side of a symmetrical planform is shown.

Examples from three reports have been selected to illustrate the types of results that can be expected from vortex lattice methods. They illustrate the wide range of uses for the vlm method.

Aircraft configurations examined by John Koegler¹⁵

As part of a study on control system design methods, John Koegler at McDonnell Aircraft Company studied the prediction accuracy of several methods for fighter airplanes. In addition to the vortex lattice method, he also used the PAN AIR and Woodward II panel methods (see Chapter 4 for details of the panel methods). He compared his predictions with the three-surface F-15, which became known as the STOL/Maneuver demonstrator, and the F-18. These configurations are illustrated in Fig. 6-19.

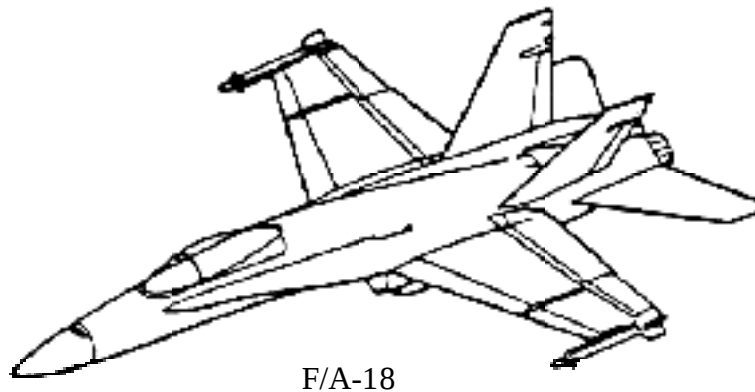
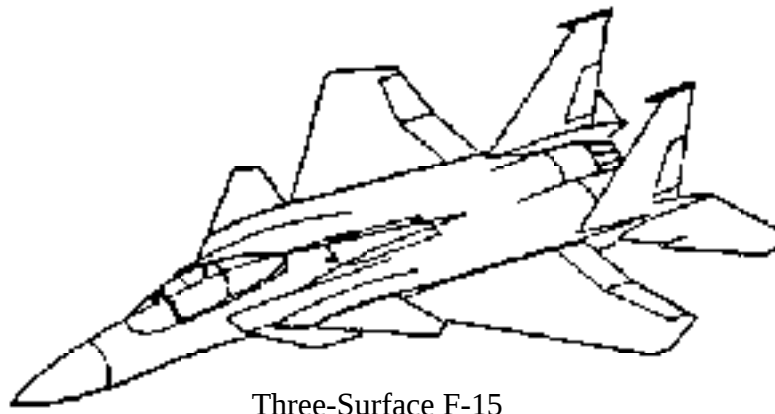


Figure 6-19. Configurations used by McDonnell Aircraft to study **vlm** method accuracy (Reference 15).

Considering the F-15 STOL/Maneuver demonstrator first, the basic panel layout is given in Figure 6-20. This shows how the aircraft was modeled as a flat planform, and the corner points of the projected configuration were used to represent the shape in the vortex lattice method and the panel methods. Note that in this case the rake of the wingtip was included in the computational model. In this study the panel methods were also used in a purely planar surface mode. In the vortex lattice model the configuration was divided into three separate planforms, with divisions at the wing root leading and trailing edges. On this configuration each surface was at a different height and, after some experimentation, the vertical distribution of surfaces shown in Figure 6-21 was found to provide the best agreement with wind tunnel data.

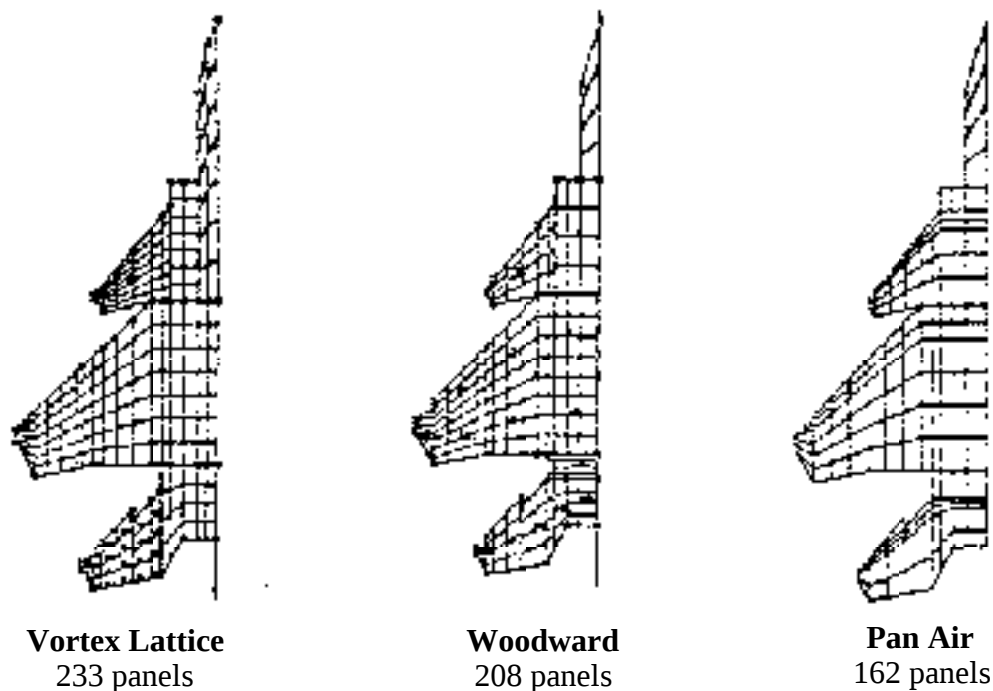


Figure 6-20. Panel models used for the three-surface F-15 (Ref.15).

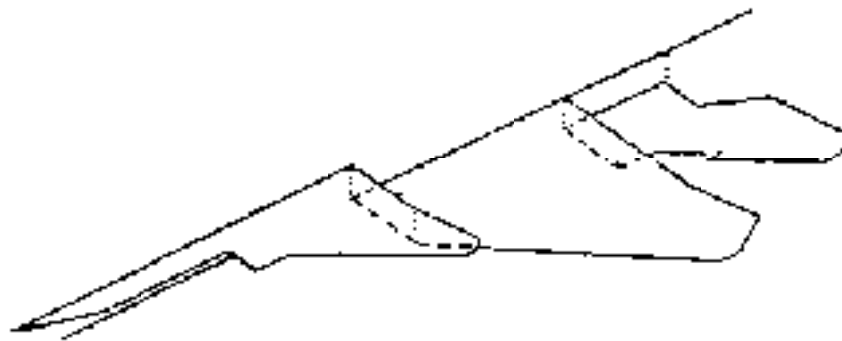


Figure 6-21. Canard and horizontal tail height representation (Ref. 15).

The results from these models are compared with wind tunnel data in Table 6-1. The vortex lattice method is seen to produce excellent agreement with the data for the neutral point location, and lift and moment curve slopes at Mach 0.2.

Subsonic Mach number effects are simulated in **vlm** methods by transforming the Prandtl-Glauert equation which describes the linearized subsonic flow to Laplace's Equation using the Göthert transformation. However, this is only approximately correct and the agreement with wind tunnel data is not as good at the transonic Mach number of 0.8. Nevertheless, the **vlm** method is as good as PAN AIR used in this manner. The **vlm** method is not applicable at supersonic speeds. The wind tunnel data shows the shift in the neutral point between subsonic and supersonic flow. The Woodward method, as applied here, over predicted the shift with Mach number. Note that the three-surface configuration is neutral to slightly unstable subsonically, and becomes stable at supersonic speeds.

Figure 6-22 provides an example of the use of the **vlm** method to study the effects of moving the canard. Here, the wind tunnel test result is used to validate the method and to provide an "anchor" for the numerical study (it would have been useful to have to have an experimental point at -15 inches). This is typical of the use of the **vlm** method in aircraft design. When the canard is above the wing the neutral point is essentially independent of the canard height. However when the canard is below the wing the neutral point varies rapidly with canard height.

Table 6-1 Three-Surface F-15 Longitudinal 1Derivatives				
Data Source		Neutral Point (% mac)	C_{ma} (1/deg)	C_{La} (1/deg)
M = 0.2	Wind Tunnel	15.70	.00623	.0670
	Vortex Lattice	15.42	.00638	.0666
	Woodward	14.18	.00722	.0667
	Pan Air	15.50	.00627	.0660
M = 0.8	Wind Tunnel	17.70	.00584	.0800
	Vortex Lattice	16.76	.00618	.0750
	Pan Air	15.30	.00684	.0705
M = 1.6	Wind Tunnel	40.80	-.01040	.0660
	Woodward	48.39	-.01636	.0700
from reference 15, Appendix by John Koegler				

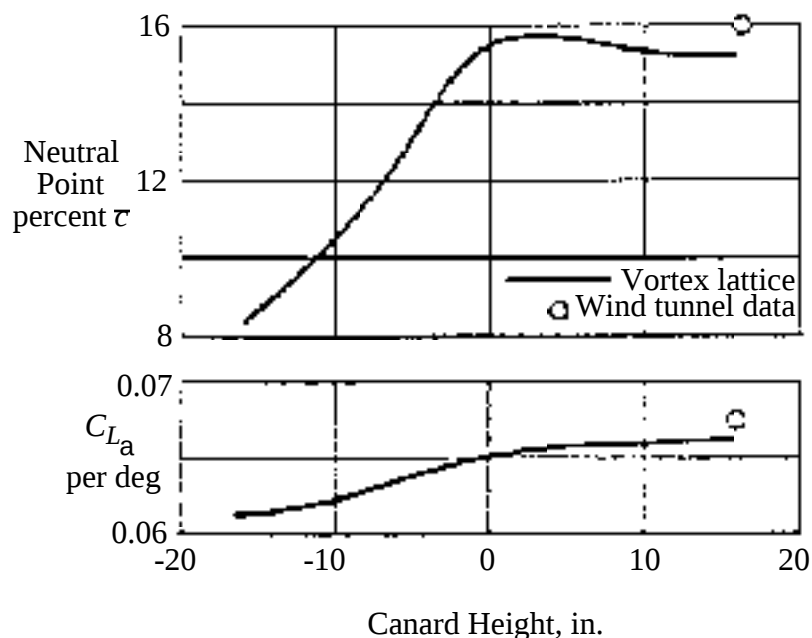


Figure 6-22. Effect of canard height variation on three-surface F-15 characteristics (Ref. 15).

Control effectiveness is also of interest in conceptual and preliminary design, and the **vlm** method can be used to provide estimates. Figure 6-23 provides an apparently accurate example of this capability for F-15 horizontal tail effectiveness. Both C_{Ldh} and C_{mdh} are presented. The **vlm** estimate is within 10% accuracy at both Mach .2 and .8. However, the F-15 has an all moving horizontal tail to provide sufficient control power under both maneuvering and supersonic flight conditions. Thus the tail effectiveness presented here is effectively a measure of the accuracy of the prediction of wing lift and moment change with angle of attack in a non-uniform flowfield, rather than the effectiveness of a flap-type control surface. A flapped device such as a horizontal stabilizer and elevator combination will have significantly larger viscous effects, and the inviscid estimate from a vortex lattice or panel method (or any inviscid method) will overpredict the control effectiveness. This is shown next for an aileron.

The aileron effectiveness for the F-15 presented in Fig. 6-24 is more representative of classical elevator or flap effectiveness correlation between **vlm** estimates and experimental data. This figure presents the roll due to aileron deflection. In this case the device deflection is subject to significant viscous effects, and the figure shows that only a portion of the effectiveness predicted by the **vlm** method is realized in the actual data. The **vlm** method, or *any* method, should always be calibrated with experimental data close to the cases of interest to provide an indication of the agreement between theory and experiment. In this case the actual results are found to be about 60% of the inviscid prediction at low speed.

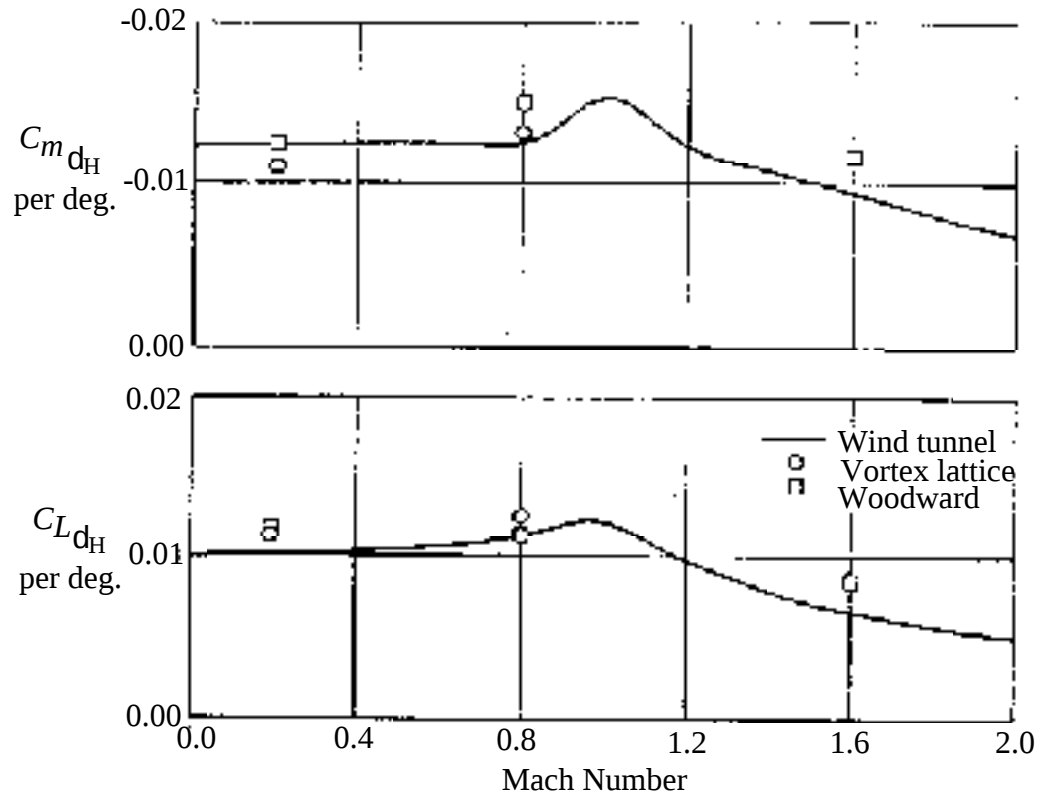


Figure 6-23. F-15 horizontal tail effectiveness (Ref. 15).

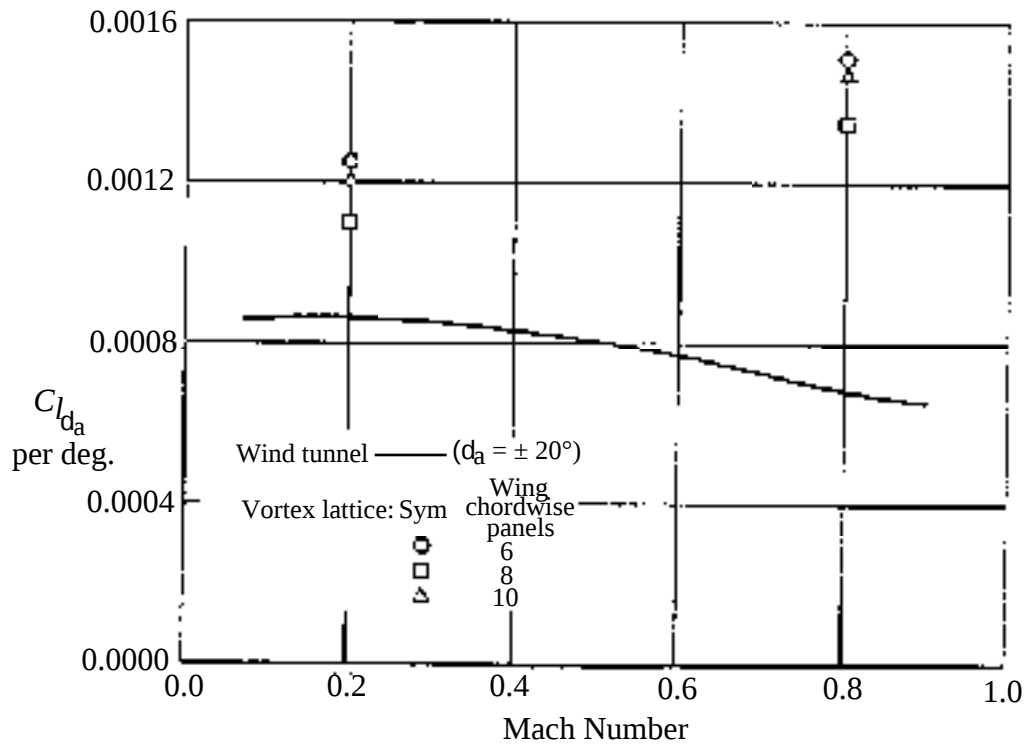


Figure 6-24. F-15 aileron effectiveness (Ref. 15).

The F/A-18 was also considered by Kogler. In this case the contributions to the longitudinal derivatives by the wing-tip missiles and the vertical tail were investigated (the vertical tails are canted outward on the F/A-18). The panel scheme used to estimate the effects of the wing-tip missile and launcher is shown in Figure 6-25. The results are given in Table 6-2. Here the computational increments are compared with the wind tunnel increments. The **vlm** method over predicts the effect of the wing-tip missiles, and under predicts the effects of the contribution of the vertical tail to longitudinal characteristics due to the cant of the tail (recall that on the F/A-18 the rudders are canted inward at takeoff to generate an additional nose up pitching moment) .

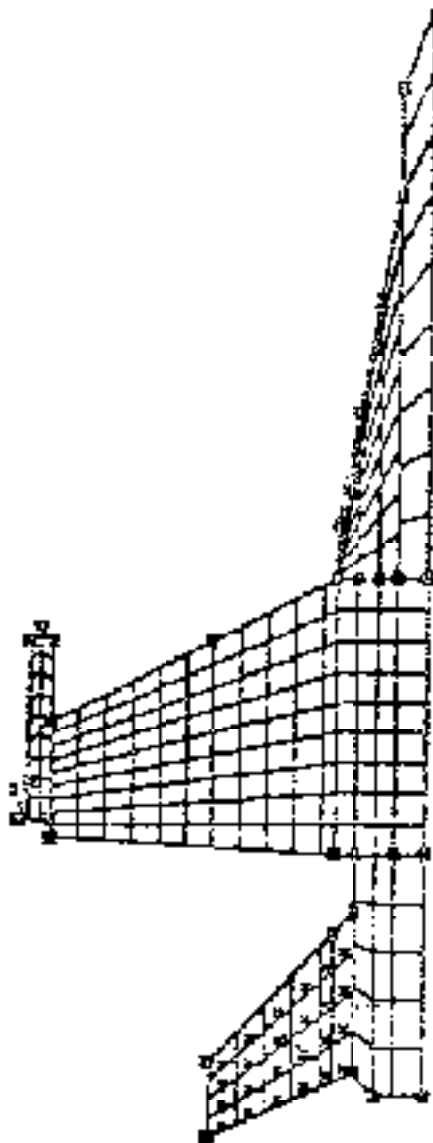
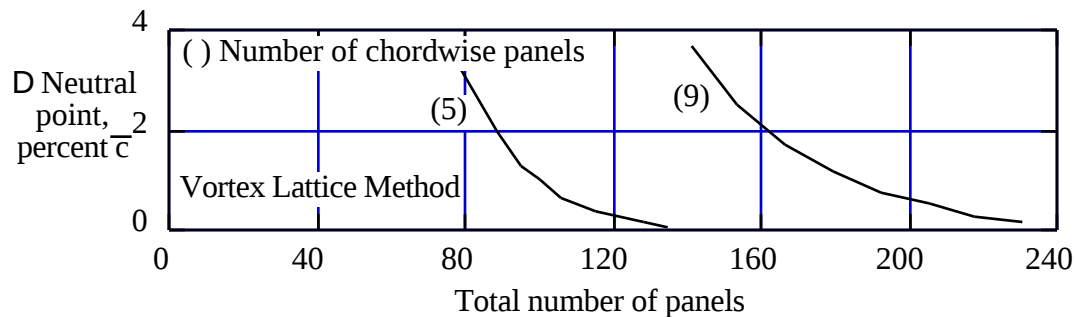


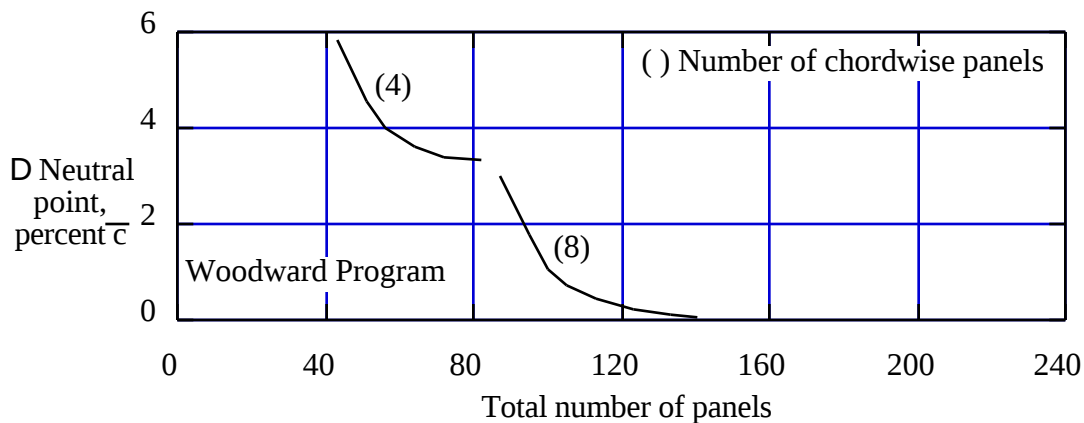
Figure 6-25. F/A-18 panel scheme with wing-tip missile and launcher (Ref. 15).

Table 6-2 F/A-18 Increments Due To Adding Wing-Tip Missiles and Launchers, and Vertical Tails					
		Mach Number	DNeutral Point (% mac)	DC _{Ma} (1/deg)	DC _{La} (1/deg)
Wing Tip Missiles and Launchers	Wind Tunnel	0.2	1.10	-0.00077	0.0020
	"	0.8	1.50	-0.00141	0.0030
	"	1.6	-1.60	0.00148	0.0030
	Vortex Lattice	0.2	1.48	-0.00121	0.0056
	"	0.8	2.11	-0.00198	0.0082
	Woodward	0.2	1.52	-0.00132	0.0053
	"	0.8	1.77	-0.00180	0.0079
	"	1.6	-0.17	0.00074	0.0022
Vertical Tails	Wind Tunnel	0.2	1.50	-0.00110	0.0050
	"	0.8	2.00	-0.00202	0.0080
	Vortex Lattice	0.2	1.11	-0.00080	0.0022
	"	0.8	1.32	-0.00108	0.0026
from reference 15, Appendix by John Koegler					

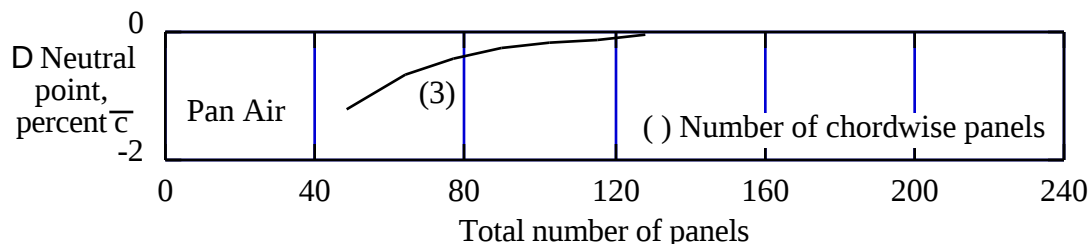
Finally, the effects of the number of panels and the way they are distributed is presented in Figure 6-26. In this case the **vlm** method is seen to take between 130 to 220 panels to produce converged results. For the vortex lattice method it appears important to use a large number of spanwise rows, and a relatively small number of chordwise panels (5 or 6 appear to be enough).



a) vortex lattice sensitivity to number of panels



b) Woodward program sensitivity to number of panels



c) Pan Air program sensitivity to number of panels

Figure 6-26. F/A-18 panel convergence study (Ref. 15).

Although this study has been presented last in this section, a study like this should be conducted before making a large number of configuration parametric studies. Depending on the relative span to length ratio the paneling requirements may vary. The study showed that from about 120 to 240 panels are required to obtain converged results. The vortex lattice methods obtains the best results when many spanwise stations are used, together with a relatively small number of chordwise panels. In that case about 140 panels provided converged results.

Slender lifting body results from Jim Pittman¹⁶

To illustrate the capability of the vortex lattice method for bodies that are more fuselage-like than wing-like, we present the lifting body comparison of the experimental and **vlm** results published by Jim Pittman of NASA Langley. Figure 6-27 shows the configuration used. Figure 6-28 provides the results of the vortex lattice method compared with the experimental data. In this case the camber shape was modeled by specifying camber slopes on the mean surface. The model used 138 vortex panels. The bars at several angles of attack illustrate the range of predictions obtained with different panel arrangements. For highly swept wings, leading edge vortex flow effects are included, as we will describe in Section 6.12. The program **VLMpc** available for this course contains the option of using the leading edge suction analogy to model these effects. Remarkably good agreement with the force and moment data is demonstrated in Fig. 6-28. The nonlinear variation of lift and moment with angle of attack arises due to the inclusion of the vortex lift effects. The agreement between data and computation breaks down at higher angles of attack because the details of the distribution of vortex flow separation are not provided by the leading edge suction analogy. The drag prediction is also very good. The experimental drag is adjusted by removing the zero lift drag, which contains the drag due to friction and separation. The resulting drag due to lift is compared with the **vlm** estimates. The comparisons are good primarily because this planform is achieving, essentially, no leading edge suction, and hence the drag is simply $C_D = C_L \tan \alpha$.

copyrighted figure from the AIAA *Journal of Aircraft*

Figure 6-27. Highly swept lifting body type hypersonic concept (Ref. 16).

copyrighted figure from the *AIAA Journal of Aircraft*

Figure 6-28. Comparison of C_L , C_m , and C_D predictions with data (Ref. 16).

Non-planar results from Kalman, Rodden and Giesing,¹⁷

All of the examples presented above considered essentially planar lifting surface cases. The vortex lattice method can also be used for highly non-planar analysis, and the example cases used at Douglas Aircraft Company in a classic paper¹⁷ have been selected to illustrate the capability. To avoid copyright problems, several of the cases were re-computed using the Virginia Tech code **JKayVLM**, and provide an interesting comparison with the original results from Douglas. Figure 6-29 presents an example of the prediction capability for the pressure loading on a wing. In this case the geometry is complicated by the presence of a wing fence. The pressures are compared with data on the inboard and outboard sides of the fence. The agreement is very good on the inboard side. The comparison is not so good on the outboard side of the fence. This quality of agreement is representative of the agreement that should be expected using vortex lattice methods at low Mach numbers in cases where the flow would be expected to be attached.

Figure 6-30 provides an example of the results obtained for an extreme non-planar case: the ring, or annular, wing. In this case the estimates are compared with other theories, and seen to be very good. The figure also includes the estimate of C_{mq} . Although not included in the present discussion, C_{mq} and C_{lp} can be computed using **vlm** methods, and this capability is included in the vortex lattice method provided here, **VLMpc**.

copyrighted figure from the AIAA *Journal of Aircraft*

Figure 6-29. Comparison of DC_p loading on a wings with a fence (Ref. 17).

copyrighted figure from the AIAA *Journal of Aircraft*

Figure 6-30. Example of aerodynamic characteristics of a ring wing (Ref. 17).

Figure 6-31 provides an example of the effects of the presence of the ground on the aerodynamics of simple unswept rectangular wings. The lift and pitching moment slopes are presented for calculations made using **JKayVLM** and compared with the results published by Kalman, Rodden and Giesing,¹⁷ and experimental data. The agreement between the data and calculations is excellent for the lift curve slope. The $AR = 1$ wing shows the smallest effects of ground proximity because of the three dimensional relief provided around the wing tips. As the aspect ratio increases, the magnitude of the ground effects increases. The lift curve slope starts to increase rapidly as the ground is approached.

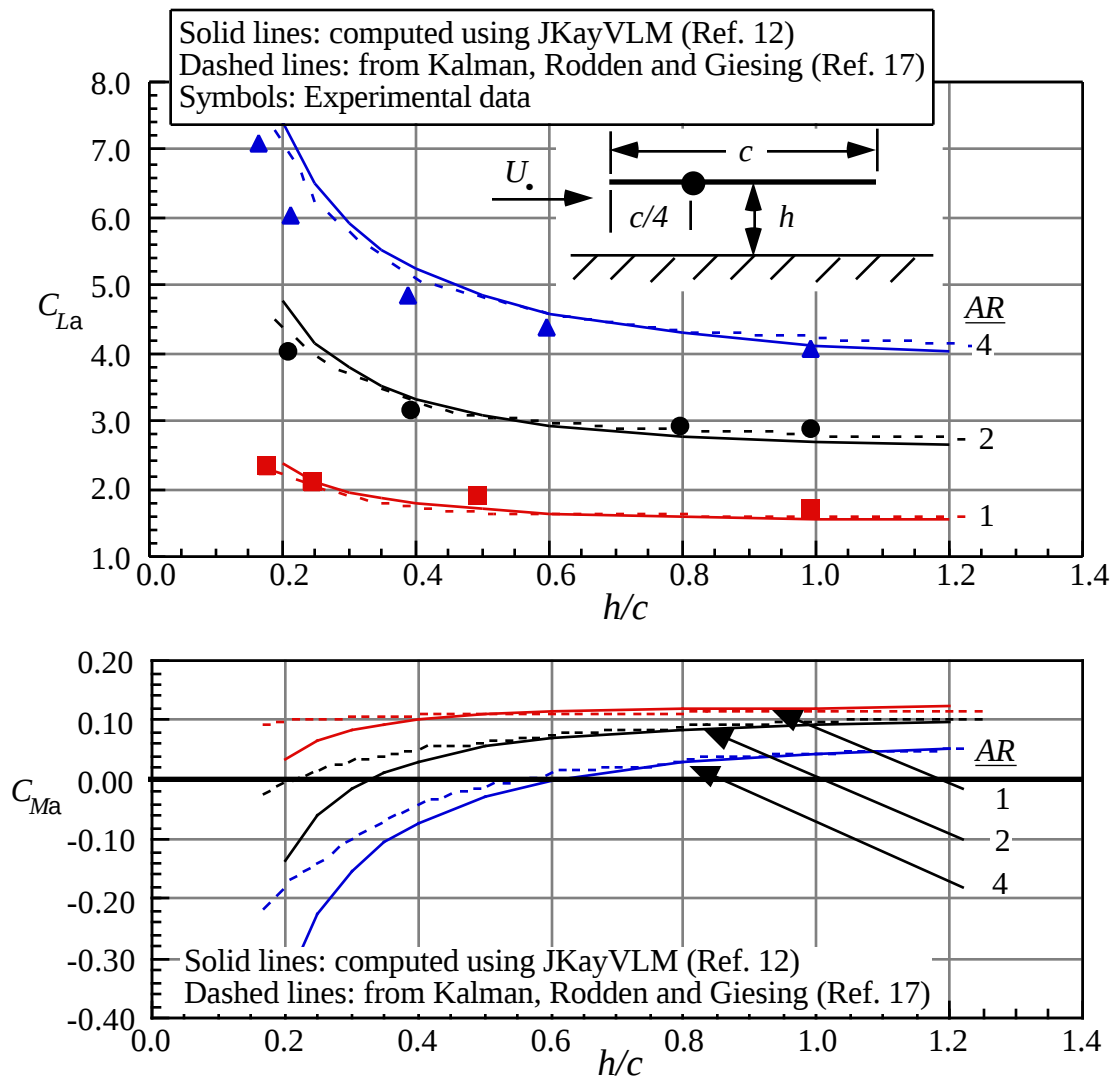


Figure 6-31. Example of ground effects for a simple rectangular wing (a case from Ref. 17).

The wings also experience a significant change in the pitching moment slope (aerodynamic center shift), and this is also shown. Note that the predictions start to differ as the ground is approached. **JKayVLM** actually rotates the entire surface to obtain another solution to use in estimating the lift curve slope. The standard procedure used by most methods is to simply change the slope condition at the mean line, as discussed previously in this chapter. Because of

the proximity to the ground, this might be a case where the transfer of the boundary condition may not be accurate. I have not asked Joe Giesing if he remembers how these calculations were made (nearly 30 years ago!).

Figure 6-32 presents similar information for the effect of dihedral angle on a wing. In this case the effects of anhedral, where the wing tip approach the ground, are extremely large. The results of dihedral changes for a wing out of ground effect are shown for comparison. Both methods agree well with each other, with differences appearing only as the wingtips approach the ground. Here again, **JKayVLM** actually rotates the entire geometry, apparently resulting in an increase in the effects as the tips nearly contact the ground. It also prevents calculations from being obtained as close to the ground as the published results. In making these calculations it was discovered that the wing panel was rotated and not sheared, so that the projected span decreases as the dihedral increases, and this produces much more pronounced changes in the lift curve slope due to the reduction in projected span.

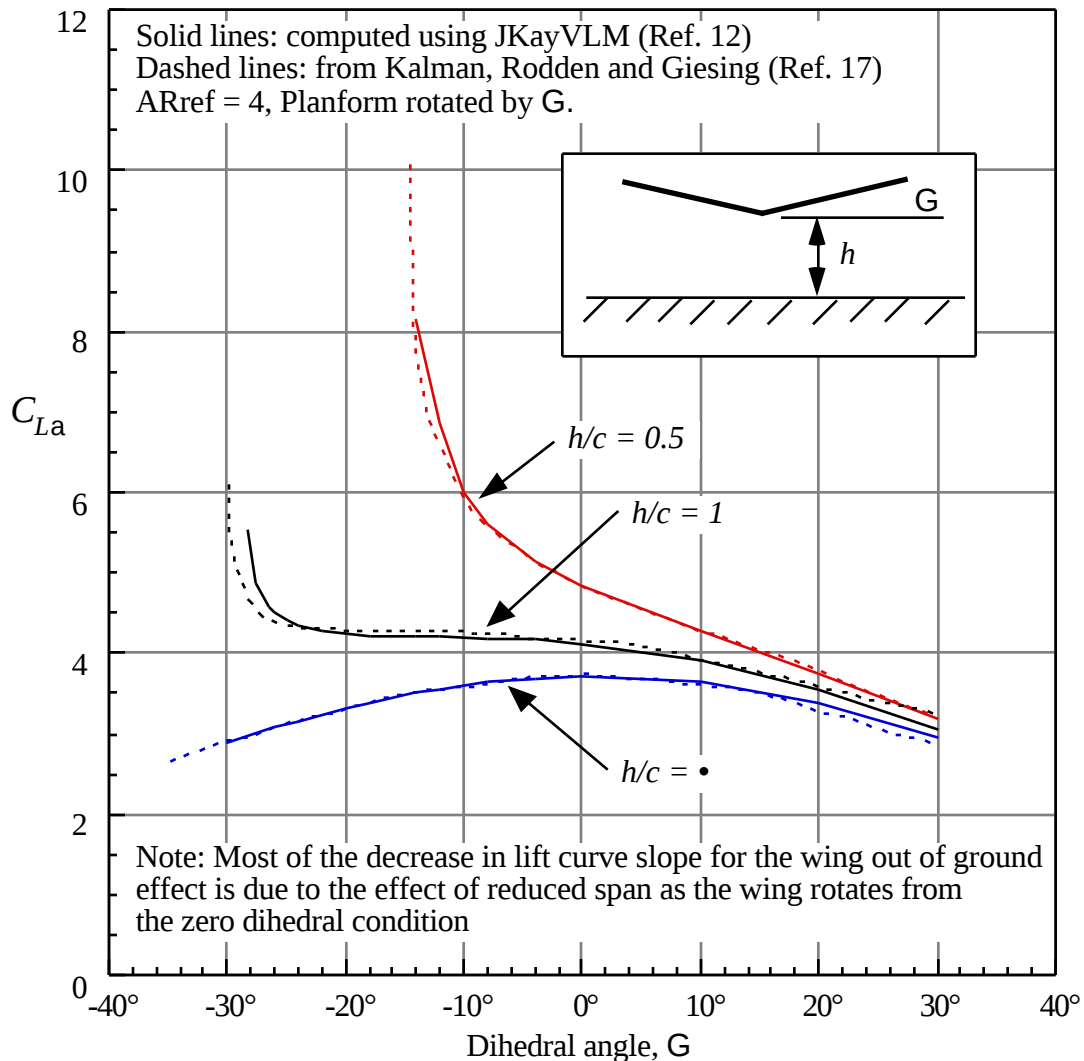


Figure 6-32. Example of ground effects for a wing with dihedral (a case from Ref. 17).

6.9 Program VLMpc and the Warren 12 Test Case

This vortex lattice method can be used on personal computers. The version of the Lamar program described in NASA TN D-7921³ fits easily into personal computers, and is available for student use (students typed this code in from the listing in the TN) as **VLMpc**. The code and input instructions are described in Appendix D.6.

This code is still used in advanced design work, and can be used to investigate many ideas in wing aerodynamics. As shown above, results can be obtained and used before the large time consuming methods of CFD are used to examine a particular idea in detail.

This section defines one reference wing case that is used to check the accuracy of vortex lattice codes. It provides a ready check case for the evaluation of any new or modified code, as well as a check on the panel scheme layout. This wing is known as the Warren 12 planform, and is defined, together with the “official” characteristics from previous calculations, in Fig. 6-33 below.

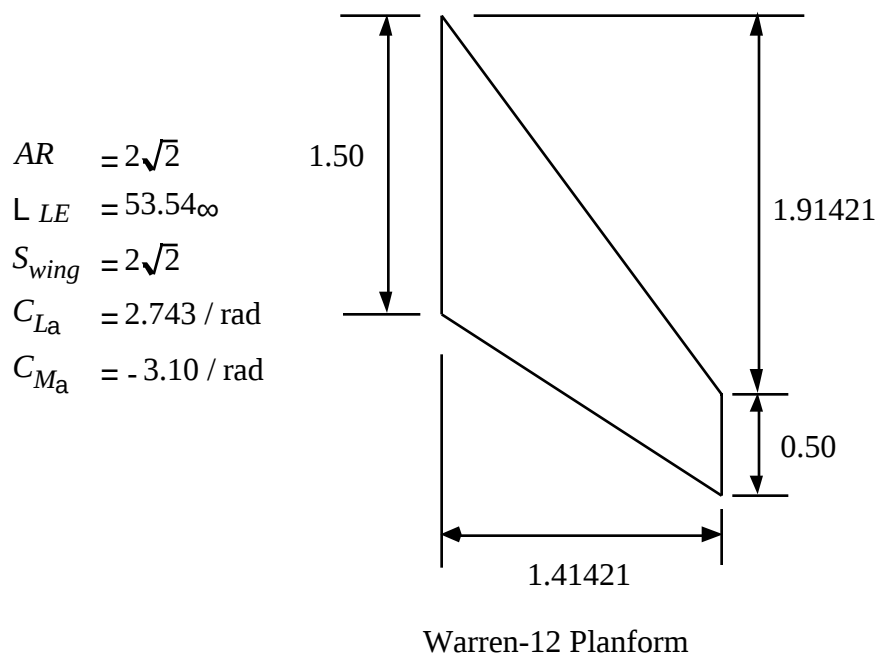


Figure 6-33. Definition and reference results for the Warren-12 wing.

For the results cited above, the reference chord used in the moment calculation is the average chord (slightly nonstandard, normally the reference chord used is the mean aerodynamic chord) and the moment reference point is located at the wing apex (which is also nonstandard).

6.10 Aerodynamics of Wings

With a three-dimensional method available, we can examine the aerodynamics of wings. Most of the results presented in this section were computed using **VLMpc**. One key advantage of the vortex lattice method compared to lifting line theory is the ability to treat swept wings. Classical Prandtl lifting line theory is essentially correct for unswept wings, but is completely erroneous for swept wings. Aerodynamics of unswept wings are closely related to the airfoil characteristics of the airfoil used in the wing. This relationship is less direct for swept wings. Many of the most important wing planform-oriented characteristics of wings arise when the planforms are swept. Even though sweep is used primarily to reduce compressibility effects, the important aerodynamic features of swept wings can be illustrated at subsonic speeds using the **vlm** method.

6.10.1 - Basic Ideas

Wings are designed to satisfy stability and handling characteristics requirements, while achieving low drag at the design conditions (usually cruise and sustained maneuver). They must also attain high maximum lift coefficients to meet field performance and maneuver requirements. Although these requirements might at first appear overwhelming, a small number of key characteristics can provide a basic physical understanding of the aerodynamics of wings.

Aerodynamic Center: The first key characteristic is the aerodynamic center of the wing, defined as location at which $dC_m/dC_L = 0$. The neutral point of the configuration is the aerodynamic center for the entire configuration. The **vlm** method was shown to provide accurate predictions of the neutral point for many configurations in the previous sections. The location of the neutral point is important in initial configuration layout to position the wing and any longitudinal stability and control surfaces at the proper location on the aircraft. Subsequently this information is fundamental in developing the control system. Wing planform shaping, as well as positioning, is used to control the location of the configuration neutral point.

Spanload: The next key consideration is the spanload distribution, cc_l/c_a , where c is the local chord, c_a is the average chord, and c_l is the local section lift coefficient. The spanload controls the location of the maximum section lift coefficient, the induced drag, and the magnitude of the wing root bending moment. The location and value of the maximum section lift coefficient determines where the wing will stall first.* If the wing airfoil stalls in front of a control surface, control will be poor at flight conditions where the control becomes very important. The shape of the spanload, together with the actual value of the wingspan, determines the value of the induced

* For a trapezoidal wing with an elliptic spanload the maximum value of the local lift coefficient occurs at $h = 1 - l$.

drag. For a specified span, the performance of the wing is evaluated by finding the value of the span efficiency factor, e , as described in Chapter 5. Finally, the wing root bending moment provides an indication of the structural loading requirements that the wing structure must be designed to accommodate. When considering the total system, the basic aerodynamic efficiency may be compromised to reduce structural wing weight. The shape of the spanload can be controlled through a combination of planform selection and wing twist. Typical twist distributions required to produce good wing characteristics are presented below.

The simplest example of planform shaping is the selection of wing aspect ratio, AR, wing taper, t , and wing sweep, L . While the aerodynamicist would like to see high values the aspect ratio, several considerations limit aspect ratio. Perhaps the most important limitation is the increase of wing structural weight with increasing aspect ratio. In addition, the lift coefficient required to maximize the benefits of high aspect ratio wings increases with the square root of the aspect ratio. Hence, airfoil performance limits can restrict the usefulness of high aspect ratios, especially for highly swept wings based on airfoil concepts. In recent years advances in both aerodynamics and structures have allowed aircraft to be designed with higher aspect ratios and reduced sweep. Table 6-3 provides some key characteristics of transport wings designed to emphasize efficient cruise while meeting takeoff and landing requirements.

Taper: Several considerations are used in selecting the wing taper. For a straight untwisted, unswept wing, the minimum induced drag corresponds to a taper ratio of about 0.4. However, a tapered wing is more difficult and hence expensive to build than an untapered wing. Many general aviation aircraft wings are built with no taper (all ribs are the same, reducing fabrication cost, and the maximum section C_l occurs at the root, well away from the control surface). To reduce structural weight the wing should be highly tapered, with $t < .4$. However, although highly tapered wings are desirable structurally, the section lift coefficient near the tip may become high. This consideration limits the amount of taper employed (current jet transports use taper ratios in the range of 0.2 to 0.3, as well as progressively increasing twist upward from the tip). As an example, the Aero Commander 500 had an aspect ratio of 9.5 and a taper ratio of 0.5 (it also had -6.5° of twist and the quarter chord of the wing was swept forward 4°).

Sweep: Sweep is used primarily to delay the effects of compressibility and increase the drag divergence Mach number. The Mach number controlling these effects is approximately equal to the Mach number normal to the leading edge of the wing, $M_{\text{eff}} = M_\infty \cos L$. The treatise on swept planforms by Küchemann is very helpful in understanding swept wing aerodynamics.¹⁸ Aerodynamic performance is based on the wingspan, b . For a fixed span, the structural span increases with sweep, $b_s = b/\cos L$, resulting in a higher wing weight. Wing sweep also leads to aeroelastic problems. For aft swept wings flutter becomes an important consideration. If the wing

is swept forward, divergence is a problem. Small changes in sweep can be used to control the aerodynamic center when it is not practical to adjust the wing position on the fuselage (the DC-3 is the most famous example of this approach).

Table 6-3 Typical Planform Characteristics of Major Transport Aircraft					
1st Flight	Aircraft	W/S	AR	$L^\circ(c/4)$	Γ
1957	B707-120	105.6	7.04	35	0.293
1958	DC-8-10	111.9	7.32	30	0.230
1963	B707-320C	110.0	7.06	35	0.250
1970	B747-200B	149.1	6.96	37.5	0.240
1970	L-1011	124.4	8.16	35	0.200
1972	DC-10-30	153.7	7.57	35	0.230
1972	A300 B2	107.9	7.78	28	0.230
1982	A310-100	132.8	8.80	28	0.260
1986	B767-300	115.1	7.99	31.5	0.182
1988	B747-400	149.9	7.61	37.5	0.240
1990	MD-11	166.9	7.57	35	0.230
1992	A330	119.0	9.3	29.74	0.192
data courtesy of Nathan Kirschbaum					

To understand the effects of sweep, the Warren 12 wing is compared with wings of the same span and aspect ratio, but unswept and swept forward. The planforms are shown in Figure 6-34. The wing leading edge sweep of the aft swept wing becomes the trailing edge sweep of the forward swept wing. Figure 6-35 provides the spanload and section lift coefficient distributions from **VLMpc**. The spanload, cc_l / c_a , is given in Fig. 6-35a, where, c is the local chord, c_l is the local lift coefficient, based on the local chord, and c_a is the average chord, S/b . Using this nomenclature, the area under the curve is the total wing lift coefficient. Note that sweeping the wing aft increases the spanload outboard, while sweeping the wing forward reduces the spanload outboard. This follows directly from a consideration of the vortex lattice model of the wing. In both cases, the portion of the wing aft on the planform is operating in the induced upwash flowfield of the wing ahead of it, resulting in an increased spanload. Figure 6-35b shows the corresponding value of the local lift coefficient. Here the effect of sweep is more apparent. The forward swept wing naturally results in a spanload with a nearly constant lift coefficient. This means that a comparatively higher wing lift coefficient can be achieved before the wing stall begins. The program **LIDRAG** can be used to compare the span e 's associated with these spanloads (an exercise for the reader).

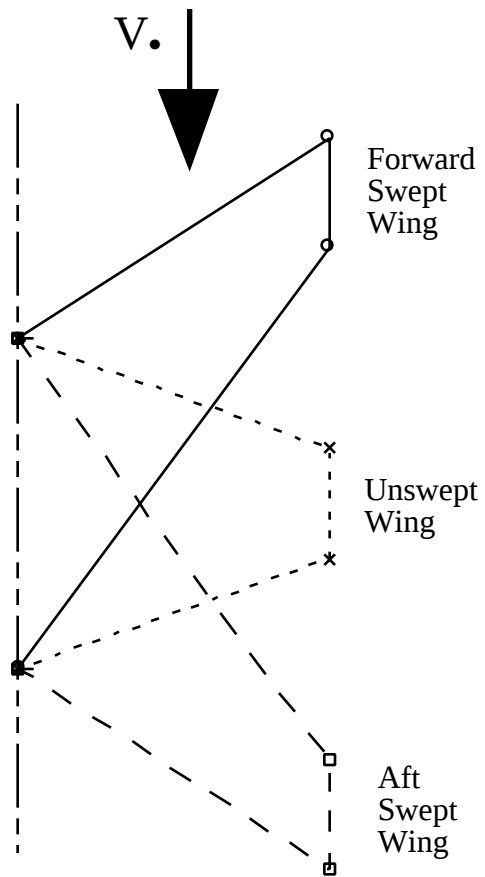
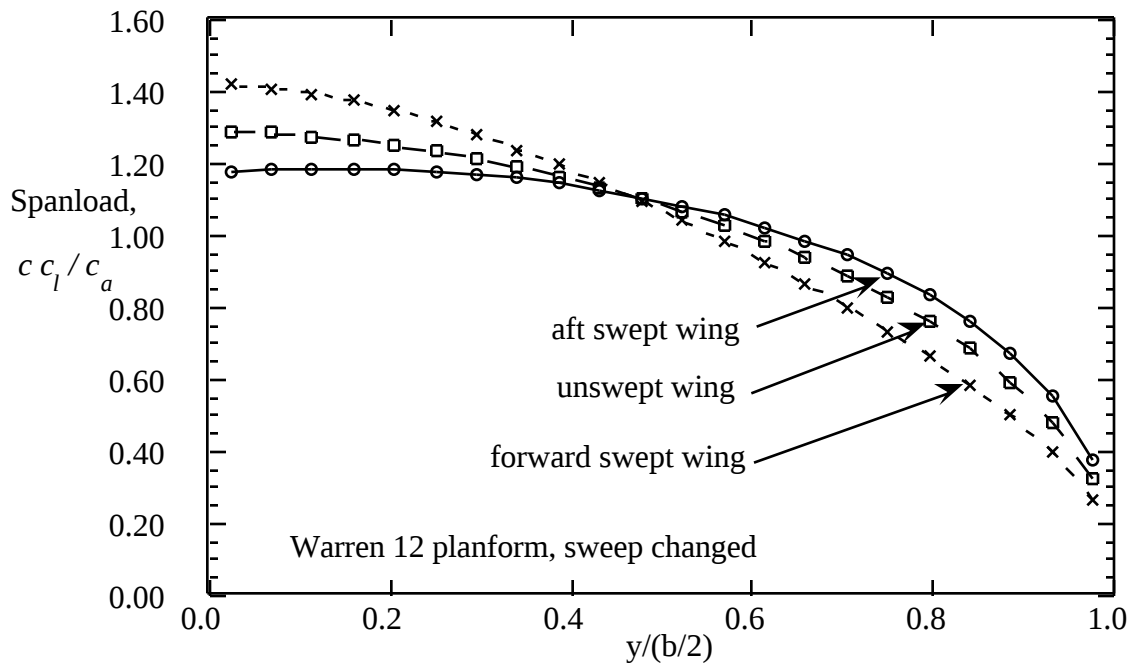
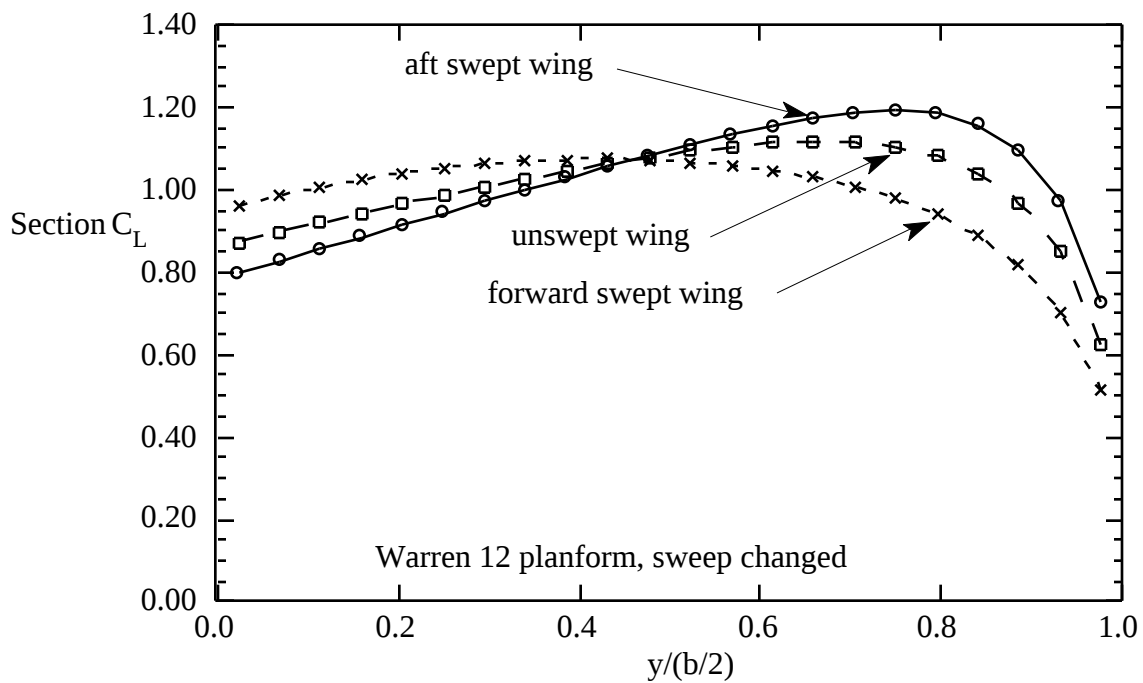


Figure 6-34. Comparison of forward, unswept, and aft swept wing planforms, $AR = 2.8$.



a) comparison of spanloads



b) comparison of section lift coefficients

Figure 6-35. Effects of sweep on planform spanload and lift coefficient distributions, $AR = 2.8$.

Similar results are now presented for a series of wings with larger aspect ratios ($AR = 8$) than the wings used in the study given above. Figure 6-36 shows the planforms used for comparison, and Figure 6-37 presents the results for the spanwise distribution of lift and section lift coefficient. These results are similar to the previous results. However, the trends observed above are in fact exaggerated at the higher aspect ratio.

Aerodynamic problems as well as structural penalties arise when using a swept wing. Because of the high section lift coefficient near the tip, aft swept wings tend to stall near the tip first. Since the lift at the tip is generated well aft, the pitching moment characteristics change when this stall occurs. With the inboard wing continuing to lift, a large positive increase in pitching moment occurs when the wingtip stalls. This is known as *pitchup*, and can be difficult to control, resulting in unsafe flight conditions. Frequently the swept wing pitching moment characteristics are compounded by the effects of flow separation on the outboard control surface. Figure 6-38 provides an example of the pitching moment characteristics of an isolated aspect ratio 10 wing using experimental data.¹⁹ The figure also includes the predictions from **VLMpc**. The agreement is reasonably good at low angle of attack, but deteriorates at high angle of attack as viscous effects become important. This is another reason that sweep is minimized.

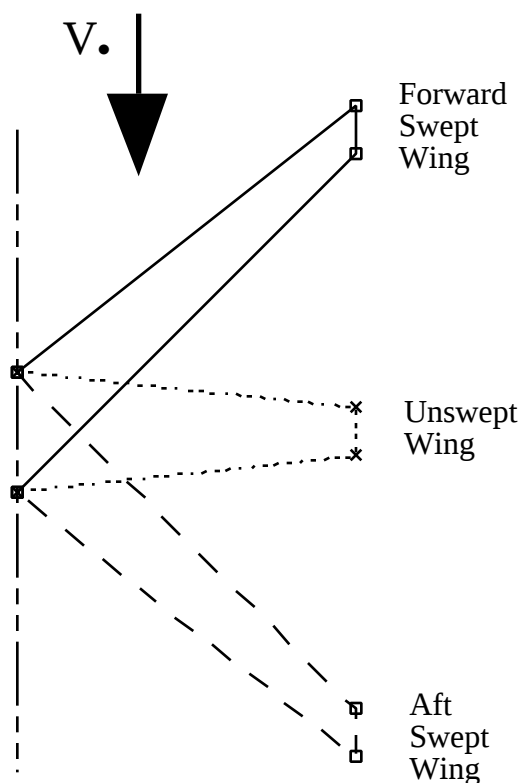
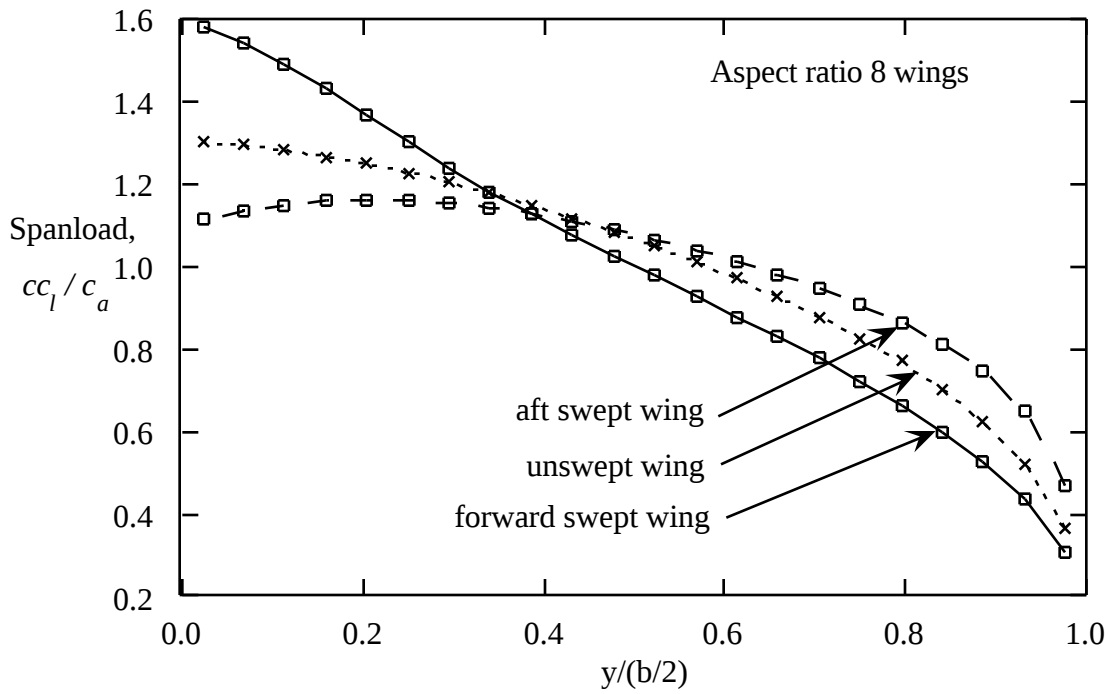
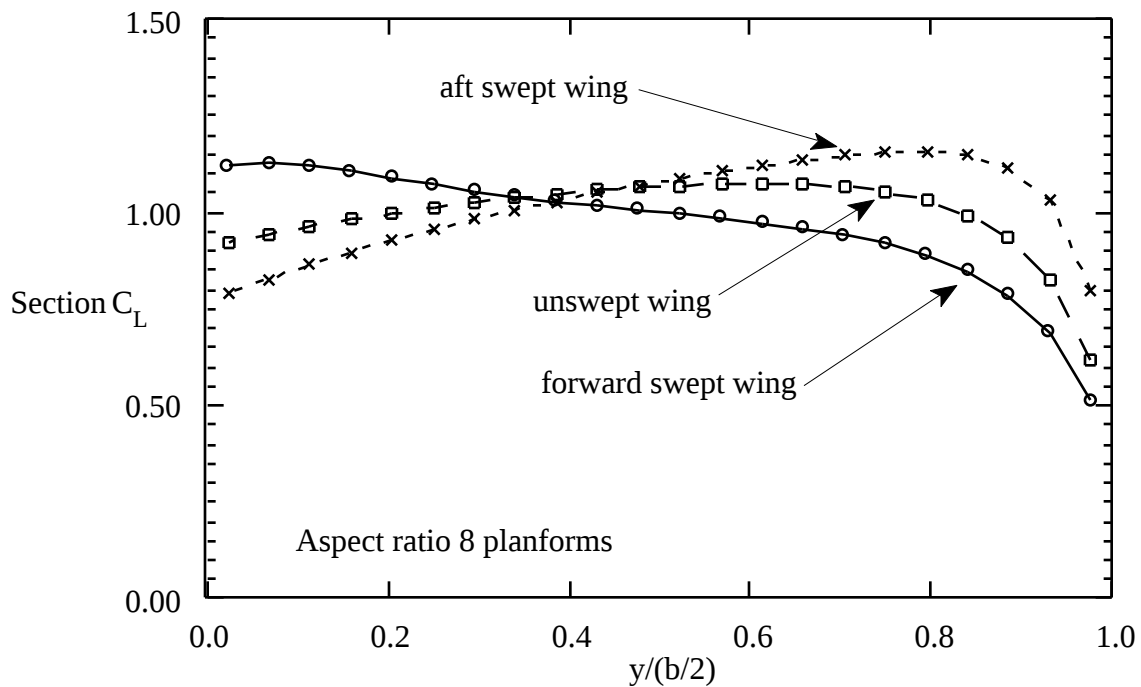


Figure 6-36. Comparison of forward, unswept, and aft swept wing planforms, $AR = 8$.



a) comparison of spanloads



b) comparison of section lift coefficients

Figure 6-37. Effects of sweep on planform spanload and lift coefficient distributions, $AR = 8$.

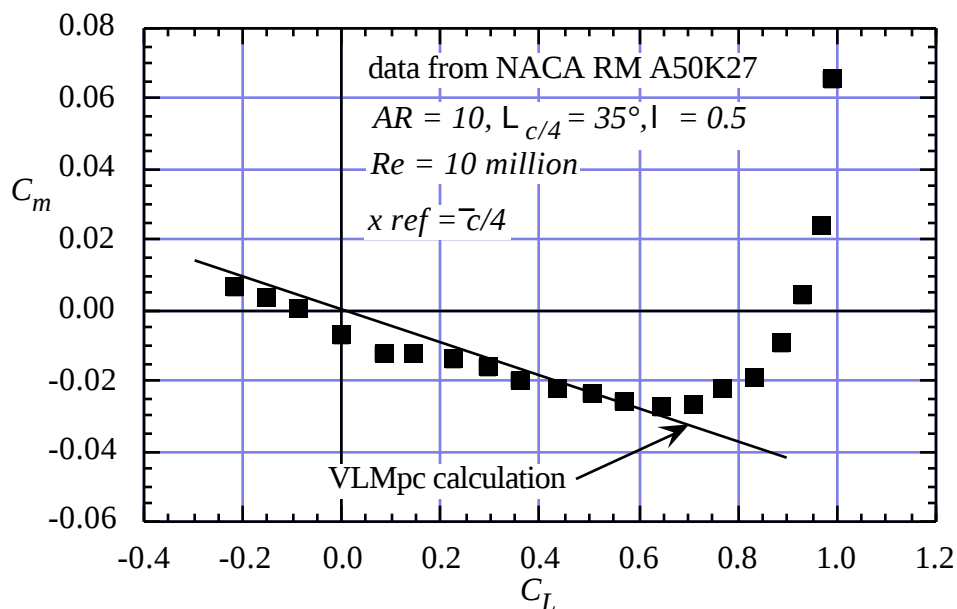


Figure 6-38. Example of isolated wing pitchup: NACA data¹⁹ compared with VLMpc.

To control the spanload, the wing can be twisted. Figure 6-39 shows typical twist distributions for aft and forward swept wings, obtained from John Lamar's program **LamDes**.²⁰ (see Chapter 5 for a description). In each case the twist is used to reduce the highly loaded areas, and increase the loading on the lightly loaded portions of the wing. For an aft swept wing this means the incidence is increased at the wing root, known as washin, and reduced, known as washout, at the wing tip. Just the reverse is true for the forward swept wing. The sudden drop in required twist at the tip for the forward swept wing case is frequently seen in typical design methods and attribute to a weakness in the method and "faired out" when the aerodynamicist gives his design to the lofting group.

Although geometric sweep is used to reduce the effective Mach number of the airfoil, the geometric sweep is not completely effective. The flowfield resists the sweep. In particular, the wing root and tip regions tend to effectively unsweep the wing. Aerodynamicists study lines of constant pressure on the wing planform known as isobars to investigate this phenomenon. Figure 6-40 presents the computed isobars for a typical swept wing,²¹ using a transonic small disturbance method.²² The effect is dramatic. The effective sweep may actually correspond to the isobar line from the wing root trailing edge to the leading edge at the wing tip. To increase the isobar sweep, in addition to geometric sweep and twist, the camber surface and thickness are typically adjusted to move the isobars forward at the wing root and aft at the wing tip. This is a key part of the aerodynamic wing design job, regardless of the computational methodology used to obtain the predicted isobar pattern.

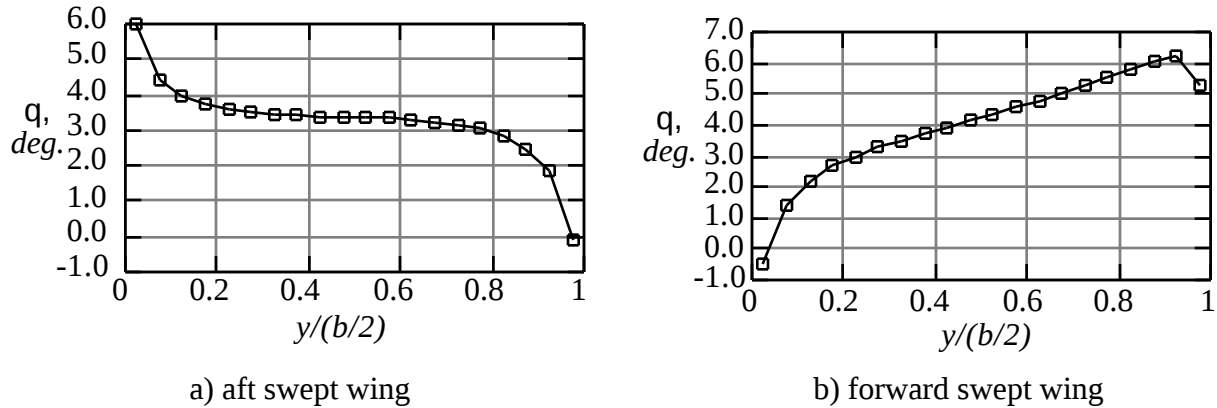


Figure 6-39. Typical twist distribution required to improve spanload on swept wings

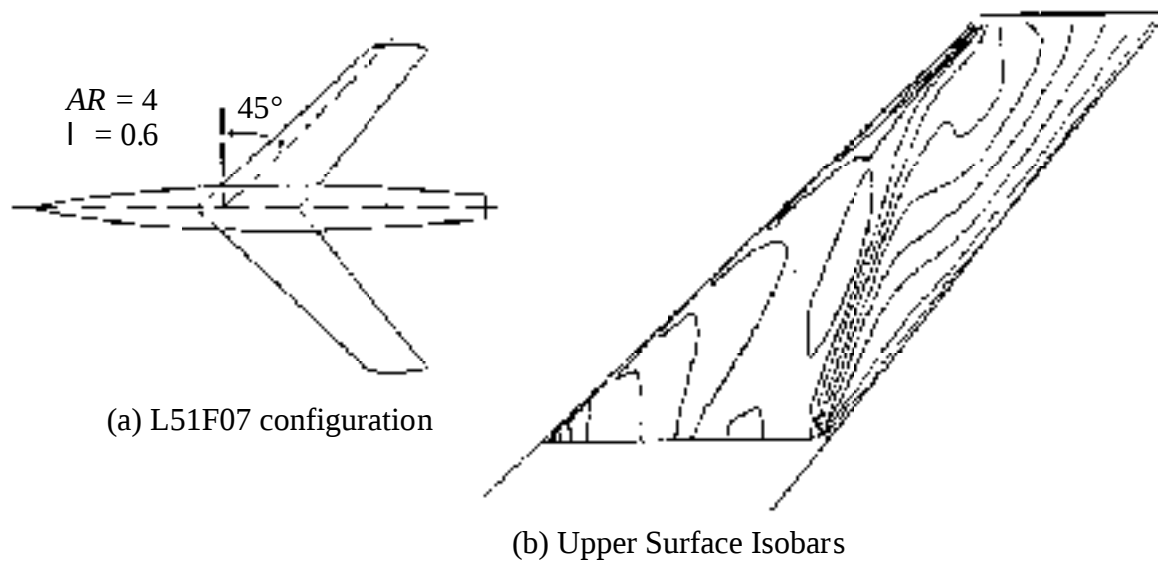


Figure 6-40. Example of the isobar distribution on an untwisted swept wing.²²

Using the wing planform and twist, together with a constant chord loading, Fig. 6-41 provides the camber lines required to support the load near the root, the mid-span and the wing tip. These results were also computed using **LamDes**.²⁰ At each station a similar chord load is specified. Here we clearly see the differences in the camber required. This is an explicit illustration of the modification to an airfoil camberline required to maintain two-dimensional airfoil-type performance when the airfoil is placed in a swept wing. These modifications represent the explicit effects of the three dimensionality of the flowfield.

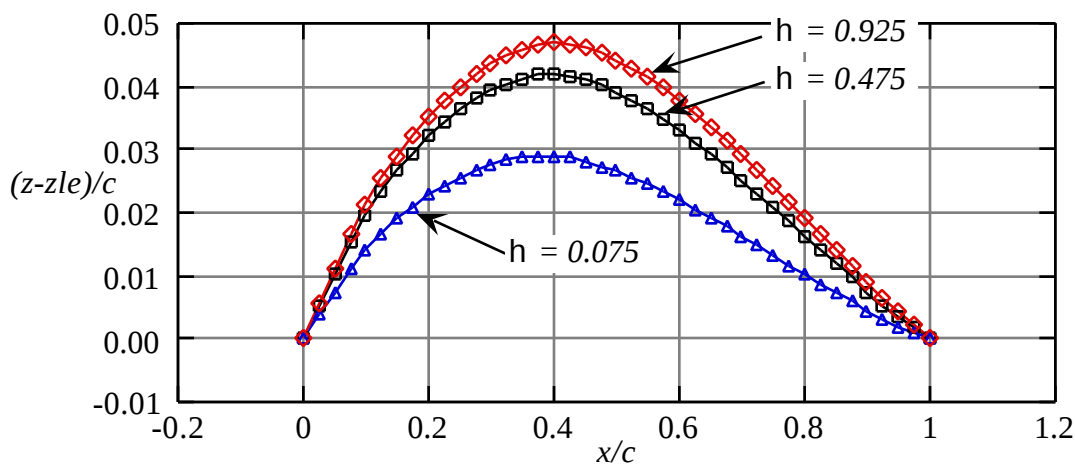


Figure 6-41. Comparison of camber lines required to develop the same chord load shape at the root, mid-span and tip region of an aft swept wing. (from **LamDes**²⁰)

Many other refinements are available to the aerodynamic designer. Insight into both the human and technical aspects of wing design prior to the introduction of computational aerodynamics is available in two recent books describing the evolution of the Boeing series of jet transports.^{23,24} One interesting refinement of swept wings has been the addition of trailing edge area at the wing root. Generally known as a “Yehudi flap”, this additional area arises for at least two reasons. The reason cited most frequently is the need to provide structure to attach the landing gear at the proper location. However, the additional chord lowers the section lift coefficient at the root, where wing-fuselage interference can be a problem, and the lower required section lift makes the design job easier. Douglas introduced this planform modification for swept wings on the DC-8, while Boeing did not incorporate it until the -320 model of the 707. However, the retired Boeing engineer William Cook, in his book,²³ on page 83, says it was first introduced on the B-29 to solve an interference problem between the inboard nacelle and the fuselage. The aerodynamic benefit to the B-29 can be found in the paper by Snyder.²⁵ Cook says, in a letter to me, that the device got its name because each wind tunnel part needed a name and there was a popular radio show at the time that featured the continuing punch line “Who’s Yehudi?” (the Bob Hope Radio show featuring Jerry Colonna, who had the line). Thus, a Boeing engineer decided to call it a Yehudi flap. This slight extra chord is readily apparent when examining the B-29, but is very difficult to photograph.

6.10.2 The Relation Between Airfoils and Swept Wings

Chapter 4 examined the basic aerodynamics of airfoils using panel methods. This chapter has emphasized the planform shape, and its analysis using vortex lattice methods. The connection between the airfoil and planform is important. In most cases the integration of the airfoil concept and the wing planform concept is crucial to the development of a successful configuration. Simple sweep theory can be used to provide, at least approximately, the connection between the airfoil and the planform. The typical aerodynamic design problem for an airfoil in a wing is defined by specifying the streamwise thickness to chord ratio, t/c , the local section lift coefficient, C_{Ldes} , and the Mach number. This three-dimensional problem is then converted to a corresponding two-dimensional problem. The desired two dimensional airfoils are then designed and transformed back to the streamwise section to be used as the wing airfoil section. Examples of the validity of this technique, together with details on other properties, including the “cosine cubed” law for profile drag due to lift are available in the NACA report by Hunton.²⁶ The relations between the streamwise airfoil properties and the chordwise properties (values normal to the leading edge, as shown in Fig. 6-42) are:

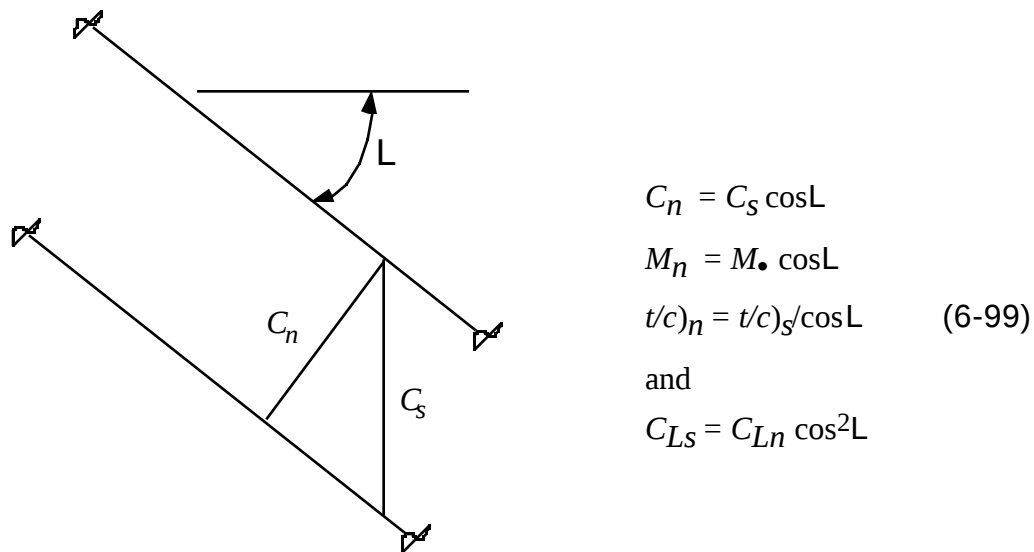


Figure 6-42. Swept wing definitions.

These relations demonstrate that the equivalent two-dimensional airfoil is thicker, operates at a lower Mach number, and at a higher lift coefficient than the three dimensional wing airfoil section. Taper effects on real wings require the selection of an effective sweep angle. Numerous approaches have been used to determine the effective angle, where guidance has been obtained by examining experimental data. The quarter chord sweep or shock sweep are typical choices.

One good example of airfoil/planform matching is the Grumman X-29. In that case wind tunnel testing of advanced transonic maneuver airfoil sections on aft swept wing configurations led the aerodynamicists (Glenn Spacht in particular) to conclude that the proper planform to take advantage of the advanced airfoil section performance should be swept forward.

6.10.3 - Wing/Tail and Canard/Wing Aerodynamics

Additional lifting surfaces are used to provide control over a wide range of conditions. If modern advanced control systems are not used, the extra surface is also designed, together with the rest of the configuration, to produce a stable design. Considering aft tail configurations first, the problem of pitchup described above for isolated wings must be reconsidered for aft tailed configurations. In particular, T-tail aircraft can encounter problems when the horizontal tail interacts with the wake of the wing at stall. Figure 6-43 provides the pitching moment characteristics of the DC-9.²⁷ The initial abrupt nose-down characteristic is the result of careful design, before the large pitchup develops. Note that even though pitch-up is a viscous effect, inviscid calculations clearly show why it happens, and can provide valuable information.

Figure 6-43 shows that a stable trim condition occurs at an angle of attack of 43° . This is an undesirable equilibrium condition, which could result in the vehicle actually trying to “fly” at this angle of attack. If adequate control power is not available, it may even be difficult to dislodge the vehicle from this condition, which is commonly known as a *deep* or *hung stall*. This will result in a rapid loss of altitude due to the very high drag. Although for this configuration full down elevator eliminates the possibility of getting “trapped” in a trimmed flight condition at this angle of attack, the amount of pitching moment available may not be sufficient to affect a rapid recovery from this condition. Examples of pitchup characteristics are not readily available. Aerodynamic designers do not like to admit that their configurations might have this characteristic. This aspect of swept wing and wing-tail aerodynamics is an important part of aerodynamic configuration development.

Even low tail placement cannot guarantee that there will not be a problem. Figure 6-44 shows the pitching moment characteristics for an F-16 type wind tunnel model.²⁸ In this case a deep stall is clearly indicated, and in fact the allowable angle of attack on the F-16 is limited to prevent the airplane from encountering this problem. In this case the pitchup arises because of powerful vortices generated by the strakes, which continue to provide lift as the wing stalls. This type of flowfield is discussed in Section 6.12.

Canard configurations provide another interesting example of multiple lifting surface interaction. The downwash from the canard wake, as it streams over the wing, reduces the effective angle of attack locally, and hence the local lift on the wing behind the canard. Wing

twist is used to counteract this effect. Figure 6-45 illustrates how this interaction occurs. The relative loading of the surfaces is an important consideration in configuration aerodynamics. The induced drag is highly dependent on the relative wing loading, which is determined by the selection of the configuration stability level and the requirement to trim about the center of gravity. in determining the induced drag. Figure 6-46, also computed using **LamDes**,²⁰ shows how the trimmed drag changes with *cg* position. Three different canard heights are shown for a range of *cg* positions, which is equivalent to varying the stability level. Figure 6-47 provides an example of the wing twist required to account for the effect of the canard downwash. Note that the forward swept wing twist increment due to the canard acts to reduce the twist required, which is exactly opposite the effect for the aft swept wing.

a copyrighted figure from the AIAA *Journal of Aircraft*

Figure 6-43. Pitching moment characteristics of the DC-9.²⁷

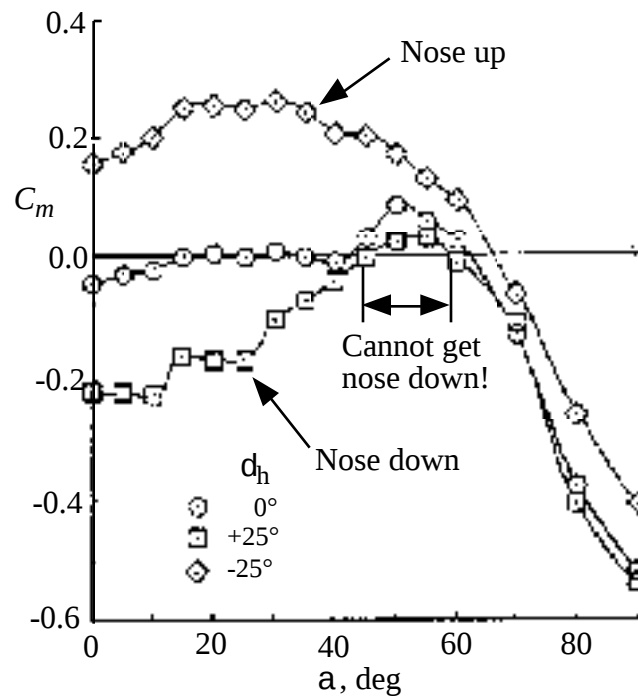


Figure 6-44. Pitching moment characteristics of an F-16 type wind tunnel model.²⁸

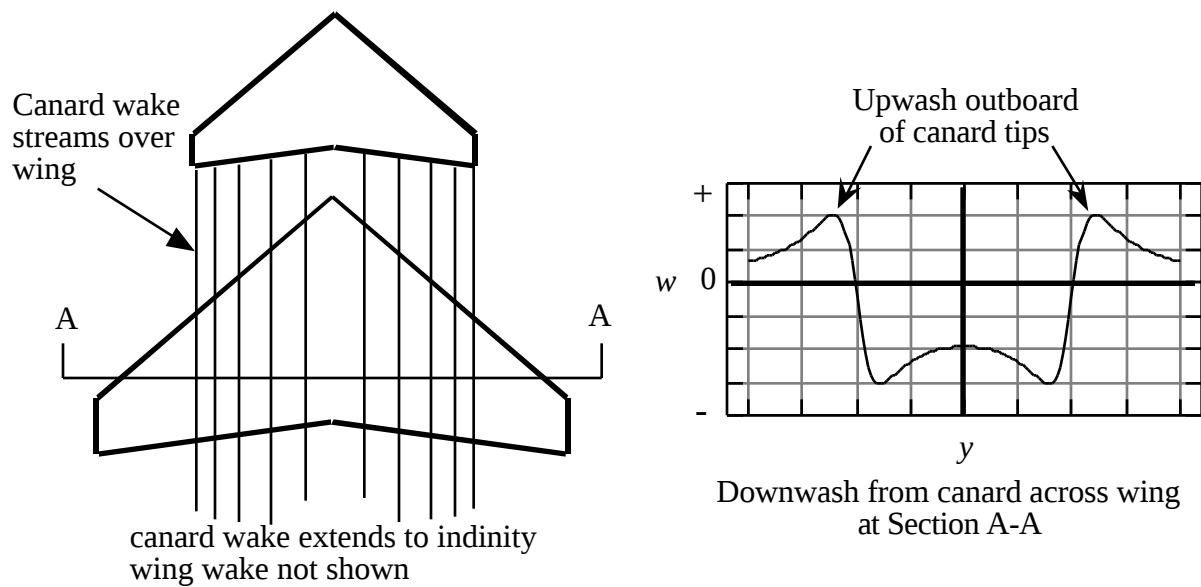
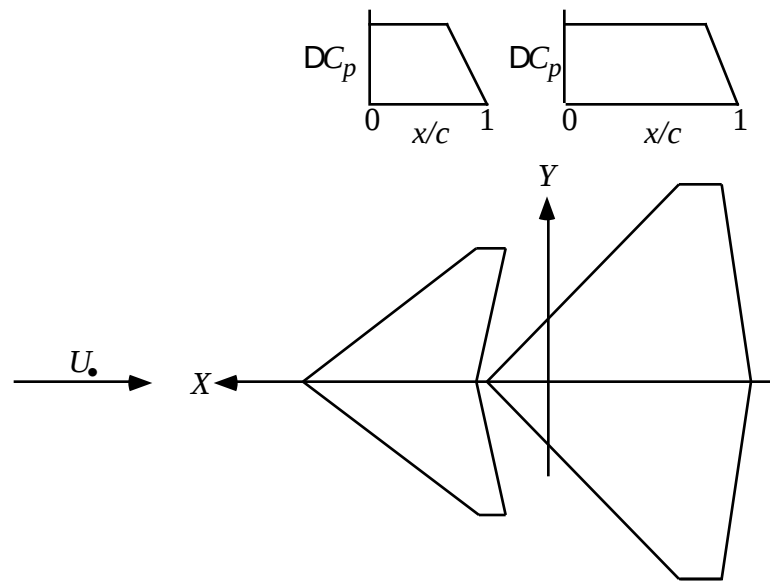
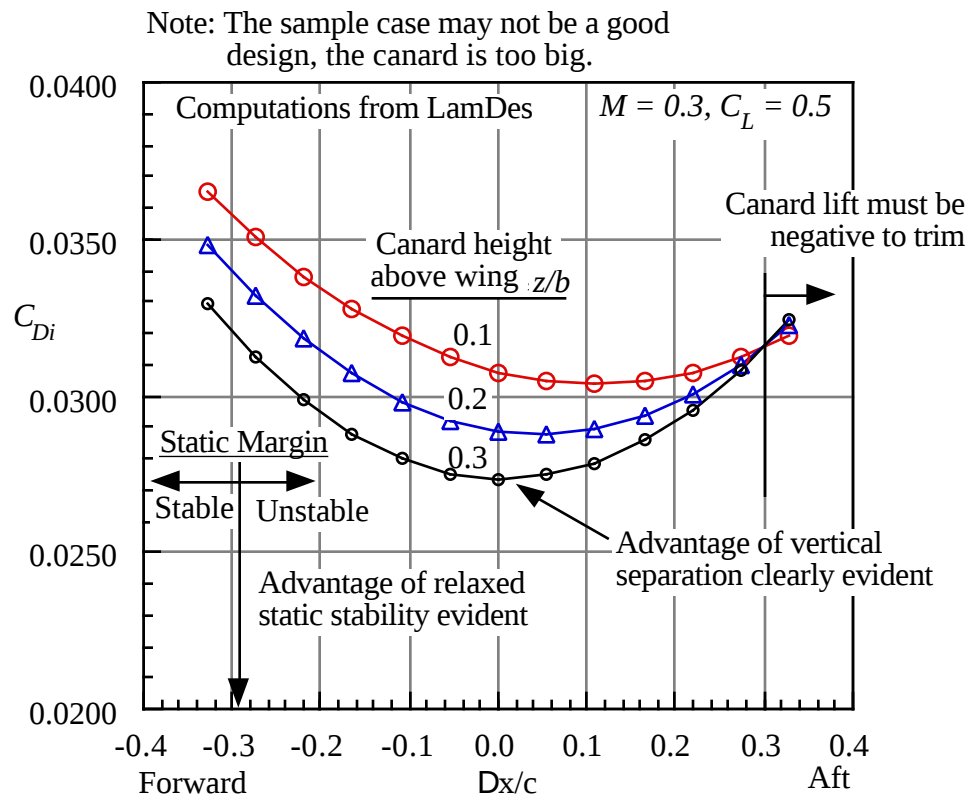


Figure 6-45. Illustration of wing-canard interaction.



a) Canard-wing planform and chord loads used in drag calculation.



b) Minimum trimmed drag variation with trim position and canard-wing separation.

Figure 6-46. Example of relation of minimum trimmed drag to balance (stability).²⁰

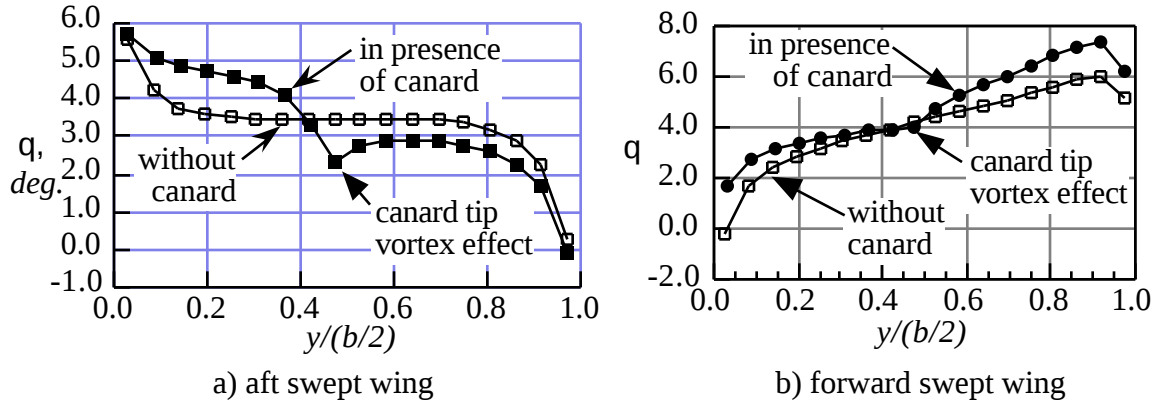


Figure 6-47. Effects of canard on twist requirements. Twist required for minimum drag using LamDes²⁰ (Note: results depend on configuration details, balance).

6.11 Inverse Design Methods and Program DesCam

Although most of the analysis discussed above corresponds to the analysis problem, the design problem can also be treated. In this section we provide one example: the determination of the camber line shape required to obtain a specific chord load in the two dimensional case. We take the opportunity to illustrate a method due to Lan⁶ that uses a mathematically based selection of vortex and control point placements instead of the 1/4 - 3/4 rule used above.

Recall that a line of vortex singularities induces a vertical velocity on the singularity line given by (see chapter 4 and Karamcheti¹³):

$$w(x) = -\frac{1}{2\pi} \int_0^c \frac{g(\xi)}{x - \xi} d\xi \quad (6-100)$$

For thin wing theory the vertical velocity can be related to the slope as shown above in Section 6.2. The vortex strength can be related to the streamwise velocity by $g = u^+ - u^-$. This in turn can be used to relate the vorticity to the change in pressure, DC_p through:

$$DC_p = C_{p_l} - C_{p_u} = -2u^- - 2u^+ = 2(u^+ - u^-) \quad (6-101)$$

which leads to:

$$\frac{DC_p(x)}{2} = g(x) \quad (6-102)$$

resulting in the expression for camber line slope in terms of design chord load:

$$\frac{dz}{dx} = -\frac{1}{4\pi} \int_0^c \frac{DC_p}{x - \xi} d\xi \quad (6-103)$$

Here dz/dx includes the slope due to the angle of attack. Note that the integral contains a singularity, and this singularity introduces the extra complications that require special analysis for numerical integration. The original Lan theory was used to find DC_p (in a slightly different form), but it can also be used to obtain dz/dx from DC_p . To do this, Lan derived a summation formula to obtain the slope. Once the slope is known, it is integrated to obtain the camber line.

Lan showed that the integral in Eq. (6-100) could be very accurately found from the summation:

$$\left. \frac{dz}{dx} \right|_i = -\frac{1}{N} \sum_{k=1}^N \frac{DC_p}{4} \frac{\sqrt{x_k(1-x_k)}}{x_i - x_k} \quad (6-104)$$

where:

$$x_k = \frac{1}{2} \left[1 - \cos \left(\frac{(2k-1)\pi}{2N} \right) \right] \quad k = 1, 2, \dots, N \quad (6-105)$$

and:

$$x_i = \frac{1}{2} \left[1 - \cos \left(\frac{i\pi}{N} \right) \right] \quad i = 0, 1, 2, \dots, N. \quad (6-106)$$

Here $N + 1$ is the number of stations on the camber line at which the slopes are obtained.

Given dz/dx , the camber line is then computed by integration using the trapezoidal rule (marching forward starting at the trailing edge):

$$z_{i+1} = z_i - \frac{x_{i+1} - x_i}{2} \left[\left. \frac{dz}{dx} \right|_i + \left. \frac{dz}{dx} \right|_{i+1} \right] \quad (6-107)$$

The design angle of attack is then:

$$\alpha_{DES} = \tan^{-1} z_0 \quad (6-108)$$

The camber line can then be redefined in standard nomenclature, i.e., $z(x=0) = z(x=1) = 0.0$:

$$\bar{z}_i = z_i - (1 - x_i) \tan \alpha_{DES} \quad (6-109)$$

How well does this work? Program **DesCam** implements the method described here, and the user's manual is provided in App. D.7. Here we compare the results from **DesCam** with the analytic formula given in Appendix A.1 for the NACA 6 Series mean line with $a = .4$. The results are shown in Figure 6-48 below. Notice that the camber scale is greatly enlarged to

demonstrate the excellent comparison. Even though the chord load is constructed by two straight line segments, the resulting required camber line is highly curved over the forward portion of the airfoil. Note also that thin airfoil theory allows only two possible values for the pressure differential at the leading edge, zero or infinity. A close examination of the camber line shape required to produce a finite load reveals a singularity. The slope is infinite. This feature is much easier to study using the analytic solution, as given in Appendix A. This approach can easily be extended to three dimensions. Notice that design problem is direct, in that it does not require the solution of a system of equations.

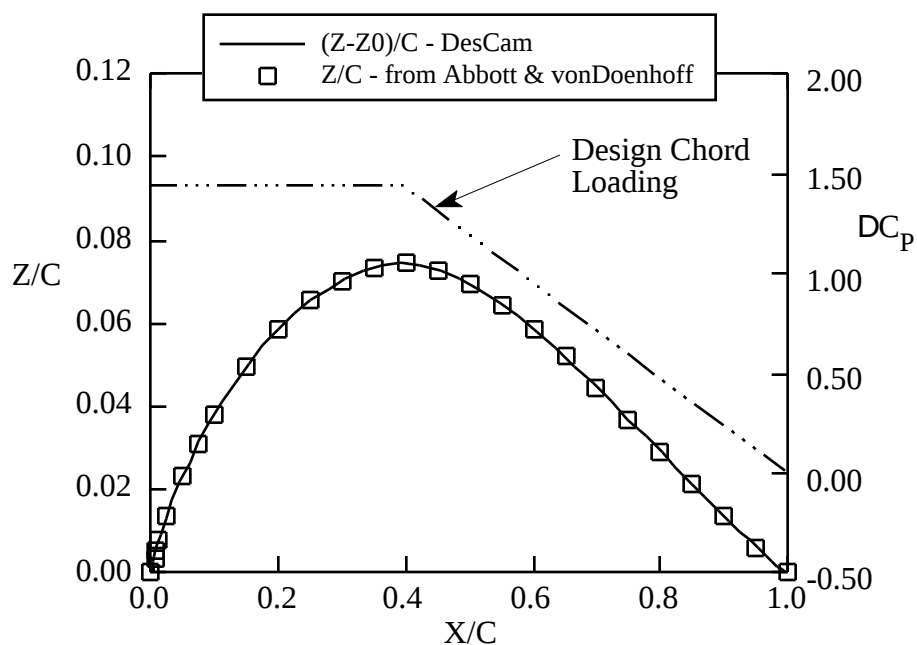
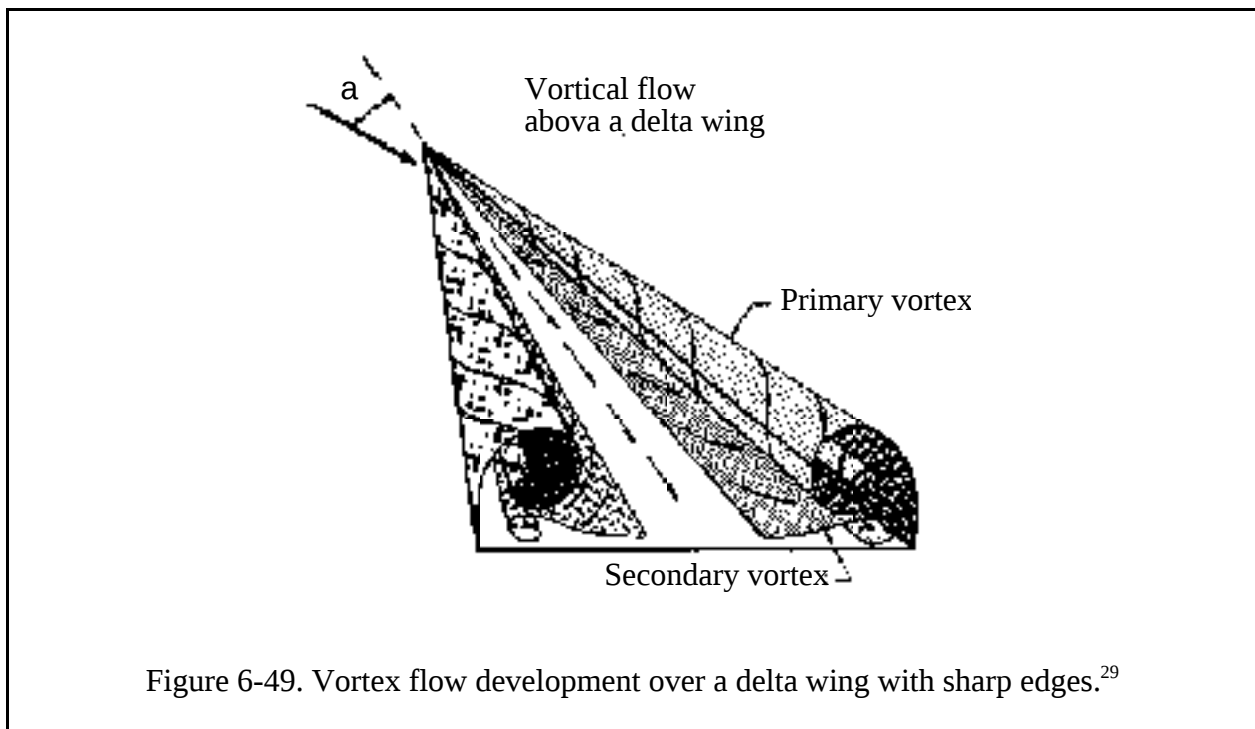


Figure 6-48. Example and verification of camber design using **DesCam**.

6.12 Vortex Flow Effects and the Leading Edge Suction Analogy

For highly swept wings at even moderate angles of attack, the classical attached flow/trailing edge Kutta condition flow model we've adopted is wrong. Instead of the flow remaining attached on the wing and leaving the trailing edge smoothly, the flow separates at the leading edge, forming a well defined vortex. This vortex plays an important role in the design of highly swept, or "slender wing" aircraft. The most notable example of this type of configuration is the Concorde. Sharp leading edges promote this flow phenomena. The basic idea is illustrated in the sketch from Payne and Nelson²⁹ given here in Fig. 6-49.



An important consequence of this phenomena is the change in the characteristics of the lift generation as the wing angle of attack increases. The vortex that forms above the wing provides an additional low pressure force due to the strongly spiraling vortex flow. The low pressure associated with the centrifugal force due to the vortex leads to the lower pressure on the wing surface. As the wing increases its angle of attack the vortex gets stronger, further reducing the pressure on the wing. The resulting increase in lift due to the vortex can be large, as shown in Fig. 6-50, from Polhamus.³⁰

This is an important flow feature. Slender wings have very low attached flow lift curve slopes, and without the additional vortex lift it would be impractical to build configurations with low aspect ratio wings. The low attached flow alone lift curve slope would prevent them from being able to land at acceptable speeds or angle of attack. Vortex lift made the Concorde possible. Another feature of the flow is the high angle of attack at which maximum lift occurs,

and typically a very mild lift loss past maximum lift. These features are a direct result of the leading edge vortex flow structure that occurs on slender wings.

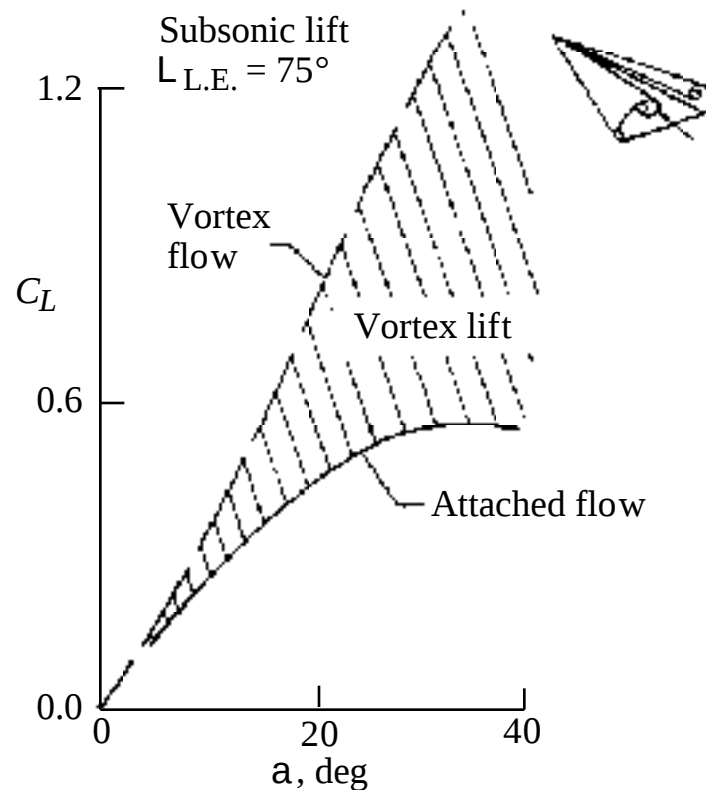


Figure 6-50. Vortex lift changes the characteristic lift development on wings.³⁰

Although the vortex lattice method formulation presented above does not include this effect, vortex lattice methods are often used as the basis for extensions that *do* include the leading edge vortex effects. A remarkable, reasonably accurate, flow model for leading edge vortex flows was introduced by Polhamus^{31,32} at NASA Langley in 1966 after examining lots of data. This flow model is known as the “Leading Edge Suction Analogy.” The concept is quite simple and was invented for sharp edged wings. The leading edge suction that should exist according to attached flow theory (see section 5.8) is assumed to rotate 90° and generate a vortex induced force instead of a suction when leading edge vortex flow exists. Thus the vortex flow force is assumed to be equal to the leading edge suction force. However, the force now acts in the direction normal to the wing surface in the direction of lift rather than in the plane of the wing leading edge. The concept is shown in the Fig. 6-51 from the original Polhamus NASA report.³¹ Further details on the effects of vortex flow effects are also available in the reports by Kulfan.^{33,34}

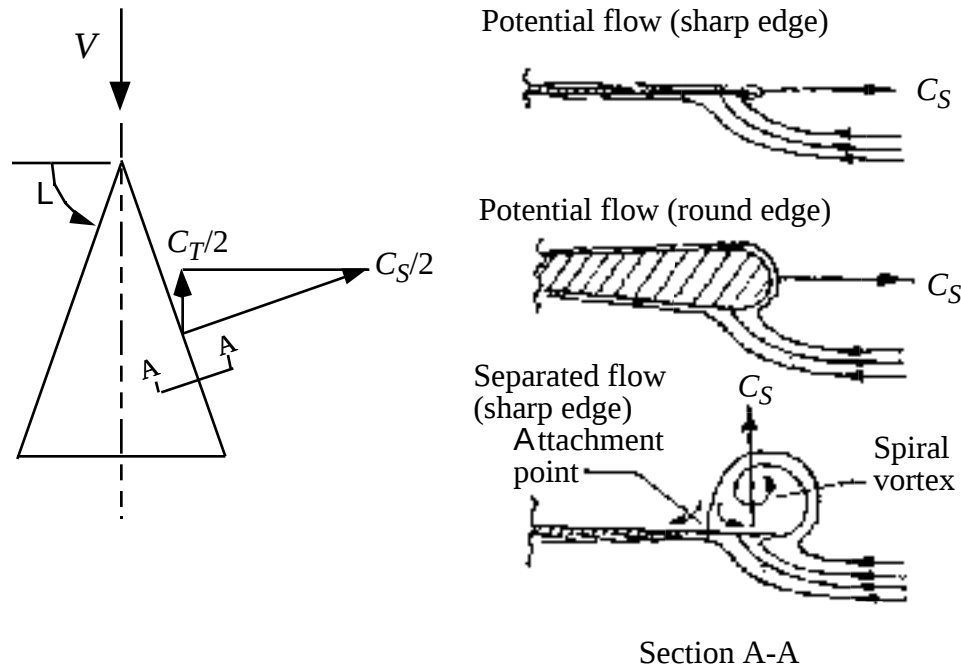


Figure 6-51. The Polhamus leading edge suction analogy.

Polhamus developed charts to compute the suction force for simple wing shapes. For a delta wing with a sharp leading edge, the method is shown compared with the data of Bartlett and Vidal³⁵ in Figure 6-52. The agreement is quite good (my reconstruction doesn't show agreement as good as that presented by Polhamus,³¹ but it is still impressive).

The figure also shows the large size of the vortex lift, and the nonlinear shape of the lift curve when large angles are considered. This characteristic was exploited in the design of the Concorde.

To find the vortex lift using the leading edge suction analogy, an estimate of the leading edge suction distribution is required. However the suction analogy does not result in an actual flowfield analysis including leading edge vortices. The Lamar vortex lattice code (**VLMpc**) optionally includes a fully developed suction analogy based on Polhamus ideas, with extensions to treat side edge suction by John Lamar³ also included.

Other approaches have been developed to compute leading edge vortex flows in more detail. Many of these methods allow vortex filaments, simulating the leading edge vortices, to leave the leading edge. The location of these vortices, and their effect on the wing aerodynamics as they roll up are explicitly computed. Mook¹⁰ and co-workers are leaders in this methodology.

The area of vortex flows in configuration aerodynamics is fascinating, and an entire conference was held at NASA Langley³⁶ devoted to the topic. The references cited above were selected to provide an entry to the literature of these flows. Interest in the area remains strong. The effects of round leading edges have been investigated by Ericsson and Reding³⁷ and Kulfan.³³ The relation between sweep, vortex lift, and vortex strength has been given recently by Hemsch and Luckring.³⁸

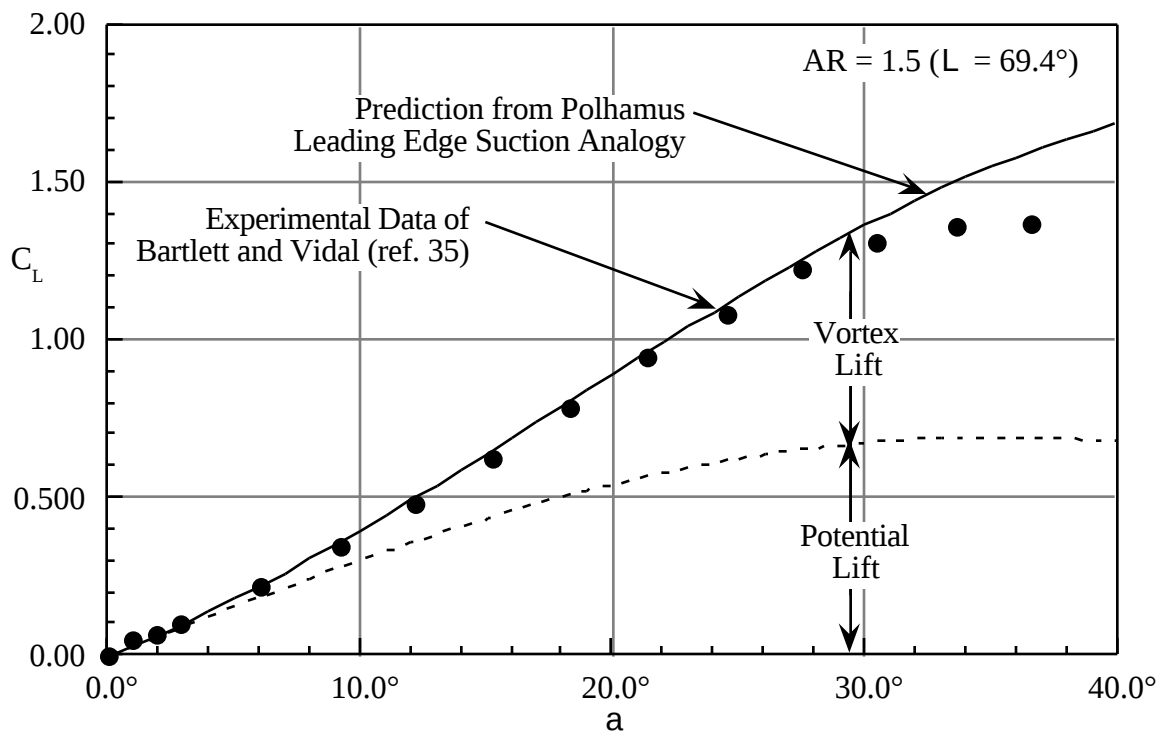


Figure 6-52. Comparison of the leading edge suction analogy with data.

6.13 Alternate and Advanced VLM Methods

Many variations of the vortex lattice method have been proposed. They address both the improvement in accuracy for the traditional case with a planar wake, and extensions to include wake position and rollup as part of the solution. Areas requiring improvement include the ability to predict leading edge suction, the explicit treatment of the Kutta condition, and the improvement in convergence properties with increasing numbers of panels.⁸ The traditional vortex lattice approaches assume that the wing wake remains flat and aligned with the freestream. This assumption is acceptable for most cases. The effect of the wake on the wing that generates it is small unless the wing is highly loaded. However, the interaction between the wake from an upstream surface and a trailing lifting surface can be influenced by the rollup and position.

In the basic case where the wake is assumed to be flat and at a specified location, the primary extensions of the method have been directed toward improving the accuracy using a smaller number of panels. Hough⁷ demonstrated that improvement in accuracy could be achieved by using a lattice that was slightly smaller than the true planform area. Basically, he proposed a 1/4 panel width inset from the tips.

Perhaps the most important revision of the vortex lattice method was proposed by Lan⁶, and called the “quasi vortex lattice method.” In this method Lan used mathematical methods, rather than the more heuristic arguments described above, to find an approximation for the thin airfoil integral in the streamwise direction. The result was, in effect, a method where the vortex and control point locations were established from the theory of Chebychev polynomials to obtain an accurate estimate of the integrals with a small number of panels. The mathematically based approach also led to an ability to compute leading edge suction very accurately.

The wake rollup and position problem has been addressed by Mook¹⁰ among others, and his work should be consulted for details. A method similar to Mook’s has been presented recently in the book by Katz and Plotkin.¹¹ They propose a vortex ring method, which has advantages when vortices are placed on the true surface of a highly cambered shapes.

Unsteady flow extension

Analogous extensions have been made for unsteady flow. For the case of an assumed flat wake the extension to harmonically oscillating surfaces was given by Albano and Rodden.³⁹ When the vortex is augmented with an oscillating doublet, the so-called *doublet-lattice method* is obtained. The doublet-lattice method is widely used for subsonic flutter calculations. Kalman, Giesing and Rodden¹⁷ provide additional details and examples (they also included the steady flow examples given above).

General unsteady flows calculations, including wake location as well as the incorporation of leading edge vortices, have been carried out by Mook among others. The resulting codes have the potential to be used to model time accurate aerodynamics of vehicles in arbitrary maneuvering flight, including the high angle of attack cases of interest in fighter aerodynamics. These codes are currently being used in studies where the aircraft aerodynamics is coupled with advanced control systems. In this case the active control is incorporated and the dynamics of the maneuver change dramatically due to the incorporation of a stability and control augmentation system.

To sum up

The summary provided above illustrates the current state of affairs. Vortex lattice methods, *per se*, are not being developed. However, they are being used in advanced methods where several disciplines are being studied simultaneously and an affordable model of the aerodynamics is required.

6.14 Exercises

1. Get a copy of **VLMpc** from the web site. The detailed instructions for this program are included in Appendix D.6. Install the program on your personal computer and repeat the sample case, checking that your output is the same as the sample output files on the web. Study the output to familiarize yourself with the variety of information generated. Turn in a report describing your efforts (*not* the output), including any mods required to make the code run on your computer.
2. How good is thin airfoil theory? Compare the thin airfoil theory DC_p for a 2D flat plate airfoil with program **VLMpc**.

Flat plate thin airfoil theory:

$$DC_p = 4a \sqrt{\frac{(1 - x/c)}{x/c}}$$

- i. Pick an aspect ratio 10 unswept wing at $\alpha = 3^\circ$ and 12° and run **VLMpc**.
 - ii. Plot $(DC_p)/a$ as a function of x/c at the wing root.
 - iii. How many panels do you need to get a converged solution from VLM?
 - iv. What conclusions do you reach?
3. Compare the validity of an aerodynamic strip theory using **VLMpc**. Consider an uncambered, untwisted wing, $AR = 4$, $l = .4$, $L_{le} = 50^\circ$, at a lift coefficient of 1. Plot the spanload, and the DC_p distribution at approximately the center section, the midspan station, and the 85% semispan station. Compare your results with a spanload constructed assuming that the wing flow is approximated as 2D at the angle of attack required to obtain the specified lift. Also compare the chordloads, DC_p , at the three span stations. How many panels do you need to obtain converged results. Document your results. Do you consider this aerodynamic strip theory valid based on this investigation? Comment.
 4. Compare the wing aerodynamic center location relative to the quarter chord of the *mac* for the wing in exercise 3, as well as similar wings. Consider one wing with zero sweep on the quarter chord, and a forward swept wing with a leading edge sweep of -50° . Compare the spanloads. Document and analyze these results. What did you learn from this comparison?
 5. For the wings in exercise 4, compare the section lift coefficients. Where would each one stall first? Which wing appears to be able to reach the highest lift coefficient before the section stalls.
 6. For the problem in exercise 5, add twist to each wing to obtain near elliptic spanloads. Compare the twist distributions required in each case.
 7. Pick a NASA or NACA report describing wind tunnel results for a simple one or two lifting surface configuration at subsonic speeds. Compare the lift curve slope and stability level predicted by **VLMpc** with wind tunnel data. Submit a report describing your work and assessing the results.
 8. Add a canard to the aft and forward swept wings analyzed in exercise 4. Plot the sum of the spanloads. How does the canard effect the wing spanload.
 9. Consider the wings in exercise 8. How does lift change with canard deflection? Add an equivalent tail. Compare the effect of tail or canard deflection on total lift and moment. Did you learn anything? What?
 10. Construct a design code using the 1/4 - 3/4 rule and compare with **DESCAM**.
 11. Construct a little 2D code to study ground effects.
 12. Compare wing and wing/tail(canard) results for C_{La} with standard analytic formulas.

6.15 References

- ¹ *Vortex Lattice Utilization Workshop*, NASA SP-405, May, 1976
- ² Margason, R.J., and Lamar, J.E., "Vortex-Lattice FORTRAN Program for Estimating Subsonic Aerodynamic Characteristics of Complex Planforms," NASA TN D-6142, 1971
- ³ Lamar, J.E., and Gloss, B.B., "Subsonic Aerodynamic Characteristics of Interacting Lifting Surfaces With Separated Flow Around Sharp Edges Predicted by a Vortex-Lattice Method," NASA TN D-7921, 1975
- ⁴ Lamar, J.E., and Herbert, H.E., "Production Version of the Extended NASA-Langley Vortex Lattice FORTRAN Computer Program," Vol. I, User's Guide, (requires update packet, July, 1984) NASA TM 83303, 1982
- ⁵ Herbert, H.E., and Lamar, J.E., "Production Version of the Extended NASA-Langley Vortex Lattice FORTRAN Computer Program," Vol. II, Source Code, NASA TM 83304, 1982
- ⁶ Lan, C.E., "A Quasi-Vortex-Lattice Method in Thin Wing Theory", *Journal of Aircraft*, Vol. 11, No. 9, September 1974, pp. 518-527.
- ⁷ Hough, Gary R., "Remarks on Vortex-Lattice Methods," *Journal of Aircraft*, Vol. 10, No. 5, May 1973, pp. 314-317.
- ⁸ DeJarnette, F.R., "Arrangement of Vortex Lattices on Subsonic Wings," in *Vortex Lattice Utilization Workshop*, NASA SP-405, May, 1976. pp. 301-319.
- ⁹ Frink, Neal T., "Lifting-Surface Theory for Skewed and Swept Subsonic Wings," *Journal of Aircraft*, Vol. 19, No. 7, July 1982, pp. 519-524.
- ¹⁰ Mook, D.T., and Nayfeh, A.H., "Application of the Vortex-Lattice Method to High-Angle-of-Attack Subsonic Aerodynamics," SAE Paper No. 851817, October, 1985.
- ¹¹ Katz, J., and Plotkin, A., *Low-Speed Aerodynamics: From Wing Theory to Panel Methods*, McGraw-Hill, Inc., New York, 1991.
- ¹² Jacob Kay, W.H. Mason, W. Durham, F. Lutze and A. Benoliel, "Control Power Issues in Conceptual Design: Critical Conditions, Estimation Methodology, Spreadsheet Assessment, Trim and Bibliography," VPI-Aero-200, November 1993.
- ¹³ Karamcheti, K., *Principles of Ideal Fluid Aerodynamics*, John Wiley & Sons, New York, 1966, pp: 518
- ¹⁴ Bertin, J.J., and Smith, M.L., *Aerodynamics for Engineers*, Prentice-Hall, Inc., Englewood Cliffs, 2nd Ed., 1989, pp: 261-282, 3rd Ed. 1998, pp: 291-311.
- ¹⁵ Thomas, R.W., "Analysis of Aircraft Stability and Control Design Methods," AFWAL-TR-84-3038, Vol. II, App. B., "Evaluation of Aerodynamic Panel Methods," by John Koegler, May, 1984.
- ¹⁶ Pittman, J.L., and Dillon, J.L., "Vortex Lattice Prediction of Subsonic Aerodynamics of Hypersonic Vehicle Concepts," *Journal of Aircraft*, Vol. 14, No. 10, pg 1017, October, 1977.
- ¹⁷ Kalman, T.P., Rodden, W.P., and Giesing, J., "Application of the Doublet-Lattice Method to Nonplanar Configurations in Subsonic Flow," *Journal of Aircraft*, Vol. 8, No. 6, June 1971, pp. 406-415.
- ¹⁸ Küchemann, D., *The Aerodynamic Design of Aircraft*, Pergamon Press, Oxford, 1978.
- ¹⁹ Tinling, B.E., and Kolk, W. R., "The Effects of Mach Number and Reynolds Number on the Aerodynamic Characteristics of Several 12-Percent-Thick Wings Having 35° of Sweepback and Various Amounts of Camber," NACA RM A50K27, Feb. 1951.

- ²⁰ Lamar, J.E., "A Vortex Lattice Method for the Mean Camber Shapes of Trimmed Non-Coplanar Planforms with Minimum Vortex Drag," NASA TN D-8090, June 1976.
- ²¹ Loving, D.L., and Estabrooks, B.B., "Transonic Wing Investigation in the Langley Eight Foot High Speed Tunnel at High Subsonic Mach Numbers and at Mach Number of 1.2," NACA RM L51F07, 1951.
- ²² Mason, W.H., MacKenzie, D.A., Stern, M.A., Ballhaus, W.F., Jr., and Frick, J., "Automated Procedure for Computing the Three-Dimensional Transonic Flow Over Wing-Body Combinations, including Viscous Effects," Vol. I, Description of Methods and Applications, AFFDL-TR-77-122, February 1978.
- ²³ Cook, William C., *The Road to the 707*, TYC Publishing, Bellevue, 1991.
- ²⁴ Irving, Clive, *Wide-Body: The Triumph of the 747*, William Morrow, New York, 1993.
- ²⁵ Snyder, George, "Structural Design Problems in the B-29 Airplane," *Aeronautical Engineering Review*, Feb. 1946, pp. 9-12.
- ²⁶ Hunton, Lynn W., "A Study of the Application of Airfoil Section Data to the Estimation of the High-Subsonic-Speed Characteristics of Swept Wings," NACA RM A55C23, June 1955.
- ²⁷ Shevell, R.S., and Schauffele, R.D., "Aerodynamic Design Features of the DC-9," *Journal of Aircraft*, Vol. 3, No. 6, Nov.-Dec. 1966, pp. 515-523.
- ²⁸ Nguyen, L.T., Ogburn, M.E., Gilbert, W.P., Kibler, K.S., Brown, P.W., and Deal, P.L., "Simulator Study of Stall/Post-Stall Characteristics of a Fighter Airplane With Relaxed Longitudinal Static Stability," NASA TP 1538, Dec. 1979.
- ²⁹ Payne, F.W., and Nelson, R.C., "An Experimental Investigation of Vortex Breakdown on a Delta Wing," in *Vortex Flow Aerodynamics*, NASA CP 2416, 1985.
- ³⁰ Polhamus, E.C., "Application of Slender Wing Benefits to Military Aircraft," AIAA Wright Brothers Lecture, AIAA-83-2566, 1983..
- ³¹ Polhamus, E.C., "A Concept of the Vortex Lift on Sharp Edge Delta Wings Based on a Leading-edge Suction Analogy," NASA-TN 3767, 1966.
- ³² Polhamus, E.C., "Prediction of Vortex Lift Characteristics by a Leading-edge Suction Analogy," *Journal of Aircraft*, Vol. 8, No. 4, 1971, pp. 193-199.
- ³³ Kulfan, R.M., "Wing Airfoil Shape Effects on the Development of Leading-Edge Vortices," AIAA 79-1675, 1979.
- ³⁴ Kulfan, R.M., "Wing Geometry Effects on Leading Edge Vortices," AIAA Paper No. 79-1872, 1979.
- ³⁵ Bartlett, G.E., and Vidal, R.J., "Experimental Investigation of Influence of Edge Shape on the Aerodynamic Characteristics of Low Aspect Ratio Wings at Low Speeds," *Journal of the Aeronautical Sciences*, Vol. 22, No. 8, August, 1955, pp.517-533,588.
- ³⁶ *Vortex Flow Aerodynamics*, NASA SP-2416 (volume I), and NASA SP-2417 (volume II), October, 1985.
- ³⁷ Ericsson, L.E., and Reding, J.P., "Nonlinear Slender Wing Aerodynamics," AIAA Paper No. 76-19, January, 1976.
- ³⁸ Hemsch, M.J., and Luckring, J.M., "Connection Between Leading-Edge Sweep, Vortex Lift, and Vortex Strength for Delta Wings," *Journal of Aircraft*, Vol. 27, No. 5, May 1990, pp. 473-475.
- ³⁹ Albano, E., and Rodden, W.P., "A Doublet-Lattice Method for Calculating Lift Distributions on Oscillating Surfaces in Subsonic Flows," *AIAA J.*, Vol. 7, No. 2, February 1969, pp. 279-285; errata *AIAA J.*, Vol. 7, No. 11, November 1969, p. 2192.

8. Introduction to Computational Fluid Dynamics

We have been using the idea of distributions of singularities on surfaces to study the aerodynamics of airfoils and wings. This approach was very powerful, and provided us with methods which could be used easily on PCs to solve real problems. Considerable insight into aerodynamics was obtained using these methods. However, the class of effects that could be examined was somewhat restricted. In particular, practical methods for computing fundamentally nonlinear flow effects were excluded. This includes both inviscid transonic and boundary layer flows.

In this chapter we examine the basic ideas behind the direct numerical solution of differential equations. This approach leads to methods that can handle nonlinear equations. The simplest methods to understand are developed using numerical approximations to the derivative terms in the partial differential equation (PDE) form of the governing equations. Direct numerical solutions of the partial differential equations of fluid mechanics constitute the field of computational fluid dynamics (CFD). Although the field is still developing, a number of books have been written.^{1,2,3,4,5,6} In particular, the book by Tannehill *et al*,¹ which appeared in 1997 as a revision of the original 1984 text, covers most of the aspects of CFD theory used in current codes and reviewed here in Chapter 14. Fundamental concepts for solving partial differential equations in general using numerical methods are presented in a number of basic texts. Smith⁷ and Ames⁸ are good references.

The basic idea is to model the derivatives by finite differences. When this approach is used the entire flowfield must be discretized, with the field around the vehicle defined in terms of a mesh of grid points. We need to find the flowfield values at every mesh (or grid) point by writing down the discretized form of the governing equation at each mesh point. Discretizing the equations leads to a system of simultaneous algebraic equations. A large number of mesh points is usually required to accurately obtain the details of the flowfield, and this leads to a very large system of equations. Especially in three dimensions, this generates demanding requirements for computational resources. To obtain the solution over a complete three dimensional aerodynamic configuration millions of grid points are required!

In contrast to the finite difference idea, approximations to the integral form of the governing equations result in the *finite volume* approach. A book has been written recently devoted solely to this approach,⁹ and we will cover this approach briefly here.

Thus CFD is usually associated with computers with large memories and high processing speeds. In addition, massive data storage systems must be available to store computed results, and ways to transmit and examine the massive amounts of data associated with a computed result must be available. Before the computation of the solution is started, the mesh of grid points must be established. Thus the broad area of CFD leads to many different closely related but nevertheless specialized technology areas. These include:

- grid generation
- flowfield discretization algorithms
- efficient solution of large systems of equations
- massive data storage and transmission technology methods
- computational flow visualization

Originally, CFD was only associated with the 2nd and 3rd items listed above. Then the problem with establishing a suitable mesh for arbitrary geometry became apparent, and the specialization of grid generation emerged. Finally, the availability of large computers and remote processing led to the need for work in the last two items cited. Not generally included in CFD per se, a current limiting factor in the further improvement in CFD capability is development of accurate turbulence models, discussed in Chapter 10.

This chapter provides an introduction to the concepts required for developing discretized forms of the governing equations and a discussion of the solution of the resulting algebraic equations. For the most part, we adopt the viewpoint of solving equilibrium (elliptic) problems. This in contrast to the more frequent emphasis on solving hyperbolic systems. Although the basic idea of CFD appears straightforward, once again we find that a successful numerical method depends on considerable analysis to formulate an accurate, robust, and efficient solution method. We will see that the classification of the mathematical type of the governing equations (Sec. 2.8) plays an important role in the development of the numerical methods. Although we adopt finite difference/finite volume methods to solve nonlinear equations, to establish the basic ideas we consider only linear equations. Application to nonlinear equations is addressed in Chapters 10, 11 and 12, where additional concepts are introduced and applied to the solution of nonlinear equations. Chapter 13 describes the most advanced approaches currently in use.

8.1 Approximations to partial derivatives

There are many ways to obtain finite difference representations of derivatives. Figure 8-1 illustrates the approach intuitively. Suppose that we use the values of f at a point x_0 and a point a

distance Δx away. Then we can approximate the slope at x_0 by taking the slope between these points. The sketch illustrates the difference between this simple slope approximation and the actual slope at the point x_0 . Clearly, accurate slope estimation depends on the method used to estimate the slope and the use of suitably small values of Δx .

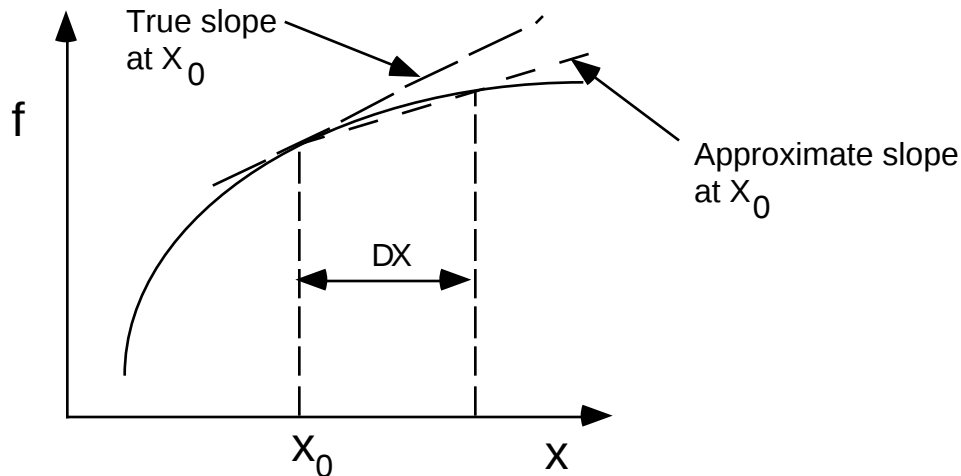


Figure 8-1. Example of slope approximation using two values of the function.

Approximations for derivatives can be derived systematically using Taylor series expansions. The simplest approach is to find an estimate of the derivative from a single series. Consider the following Taylor series:

$$f(x_0 + \Delta x) = f(x_0) + \Delta x \left. \frac{df}{dx} \right|_{x_0} + \frac{(\Delta x)^2}{2} \left. \frac{d^2f}{dx^2} \right|_{x_0} + \frac{(\Delta x)^3}{6} \left. \frac{d^3f}{dx^3} \right|_{x_0} + \dots \quad (8-1)$$

and rewrite it to solve for $\left. \frac{df}{dx} \right|_{x_0}$:

$$\left. \frac{df}{dx} \right|_{x_0} = \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x} - \Delta x \left. \frac{1}{2} \frac{d^2f}{dx^2} \right|_{x_0} - \dots$$

or:

$$\left. \frac{df}{dx} \right|_{x_0} = \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x} + \underbrace{O(\Delta x)}_{\text{Truncation Error}} \quad (8-2)$$

where the last term is neglected and called the truncation error. In this case it is $O(\Delta x)$. The term “truncation error” means that the error of the approximation vanishes as Δx goes to zero.* The

* This assumes that the numerical results are exactly accurate. There is a lower limit to the size of the difference step in Δx due to the use of finite length arithmetic. Below that step size, roundoff error becomes important. In most

form of the truncation error term is frequently important in developing numerical methods. When the order of the truncation error is $O(\Delta x)$, the approximation is described as a “first order accurate” approximation, and the error is directly proportional to Δx . The other characteristic of this representation is that it uses only the information on one side of x_0 , and is thus known as a one-sided difference approximation. Finally, because it uses information ahead of x_0 , it’s known as a forward difference. Thus, Eq.(8-2) is a first order, one sided, forward difference approximation to the derivative.

We could also write the approximation to the derivative using information prior to the point of interest. The corresponding first order accurate one sided backward difference approximation is obtained by expanding the Taylor series to a point prior to the point about which the expansion is carried out. The resulting expansion is:

$$f(x_0 - \Delta x) = f(x_0) - \Delta x \left. \frac{df}{dx} \right|_{x_0} + \frac{(\Delta x)^2}{2} \left. \frac{d^2f}{dx^2} \right|_{x_0} - \frac{(\Delta x)^3}{6} \left. \frac{d^3f}{dx^3} \right|_{x_0} + \dots \quad (8-3)$$

Solving for the first derivative in the same manner we used above, we obtain:

$$\left. \frac{df}{dx} \right|_{x_0} = \frac{f(x_0) - f(x_0 - \Delta x)}{\Delta x} + O(\Delta x), \quad (8-4)$$

the first order accurate, one sided, backward difference approximation.

Note from Fig. 8-1 above that one sided differences can lead to a fairly large truncation error. In many cases a more accurate finite difference representation would be useful. To obtain a specified level of accuracy, the step size Δx must be made small. If a formula with a truncation term of $O(\Delta x)^2$ is used,* the required accuracy can be obtained with significantly fewer grid points. A second order, $O(\Delta x)^2$, approximation can be obtained by subtracting the Taylor series expansions, Eq.(8-3) from Eq.(8-1):

$$f(x_0 + \Delta x) - f(x_0 - \Delta x) = +2\Delta x \left. \frac{df}{dx} \right|_{x_0} + \frac{(\Delta x)^3}{3} \left. \frac{d^3f}{dx^3} \right|_{x_0} + \dots$$

Here the $O(\Delta x)$ terms cancel in the subtraction. When we divide by $2\Delta x$ and solve for the first derivative, we get an expression with a truncation error of $O(\Delta x)^2$. The resulting expression for the derivative is:

cases the stepsize used for practical finite difference calculations is larger than the limit imposed by roundoff errors. We can’t afford to compute using grids so finely spaced that roundoff becomes a problem.

* With Δx small, Δx^2 is much smaller than Δx

$$\left. \frac{df}{dx} \right|_{x_0} = \frac{f(x_0 + Dx) - f(x_0 - Dx)}{2Dx} + O(Dx)^2. \quad (8-5)$$

This is a second order accurate *central* difference formula since information comes from both sides of x_0 . Numerous other approximations can be constructed using this approach. It's also possible to write down second order accurate forward and backward difference approximations.

We also need the finite difference approximation to the second derivative. Adding the Taylor series expressions for the forward and backward expansions, Eq.(8-1) and Eq.(8-3), results in the following expression, where the odd order terms cancel:

$$f(x_0 + Dx) + f(x_0 - Dx) = 2f(x_0) + (Dx)^2 \left. \frac{d^2 f}{dx^2} \right|_{x_0} + O(Dx)^4$$

Solving for the second derivative yields:

$$\left. \frac{d^2 f}{dx^2} \right|_{x_0} = \frac{f(x_0 + Dx) - 2f(x_0) + f(x_0 - Dx)}{(Dx)^2} + O(Dx)^2. \quad (8-6)$$

The formulas given above are the most frequently used approximations to the derivatives using finite difference representations. Other methods can be used to develop finite difference approximations. In most cases we want to use no more than two or three function values to approximate derivatives.

Forward and backward finite difference approximations for the second derivative can also be derived. Note that formally these expressions are only first order accurate. They are:

- a forward difference expression:

$$\left. \frac{d^2 f}{dx^2} \right|_{x_0} = \frac{f(x_0) - 2f(x_0 + Dx) + f(x_0 + 2Dx)}{(Dx)^2} + O(Dx) \quad (8-7)$$

- a backward difference expression:

$$\left. \frac{d^2 f}{dx^2} \right|_{x_0} = \frac{f(x_0) - 2f(x_0 - Dx) + f(x_0 - 2Dx)}{(Dx)^2} + O(Dx). \quad (8-8)$$

In addition, expressions can be derived for cases where the points are not evenly distributed. In general the formal truncation error for unevenly spaced points is not as high as for the evenly spaced point distribution. In practice, for reasonable variations in grid spacing, this may not be a serious problem. We present the derivation of these expressions here. A better way of handling non-uniform grid points is presented in the next chapter. The one sided first derivative expressions Eq.(8-2) and Eq.(8-4) are already suitable for use in unevenly spaced situations. We need to obtain a central difference formula for the first derivative, and an expression for the

second derivative. First consider the Taylor expansion as given in Eqs. (8-1) and (8-3). However, the spacing will be different in the two directions. Use Dx^+ and Dx^- to distinguish between the two directions. Eqs. (8-1) and (8-3) can then be rewritten as:

$$f(x_0 + Dx^+) = f(x_0) + Dx^+ \left. \frac{df}{dx} \right|_{x_0} + \frac{(Dx^+)^2}{2} \left. \frac{d^2f}{dx^2} \right|_{x_0} + \frac{(Dx^+)^3}{6} \left. \frac{d^3f}{dx^3} \right|_{x_0} + \dots \quad (8-9)$$

$$f(x_0 - Dx^-) = f(x_0) - Dx^- \left. \frac{df}{dx} \right|_{x_0} + \frac{(Dx^-)^2}{2} \left. \frac{d^2f}{dx^2} \right|_{x_0} - \frac{(Dx^-)^3}{6} \left. \frac{d^3f}{dx^3} \right|_{x_0} + \dots \quad (8-10)$$

Define $Dx^+ = a Dx^-$. To obtain the forms suitable for derivation of the desired expressions, replace Dx^+ in Eq. (8-9) with $a Dx^-$, and multiply Eq. (8-10) by a . The resulting expressions are:

$$f(x_0 + Dx^+) = f(x_0) + a Dx^- \left. \frac{df}{dx} \right|_{x_0} + \frac{(a Dx^-)^2}{2} \left. \frac{d^2f}{dx^2} \right|_{x_0} + \frac{(a Dx^-)^3}{6} \left. \frac{d^3f}{dx^3} \right|_{x_0} + \dots \quad (8-11)$$

$$a f(x_0 - Dx^-) = a f(x_0) - a Dx^- \left. \frac{df}{dx} \right|_{x_0} + a \frac{(Dx^-)^2}{2} \left. \frac{d^2f}{dx^2} \right|_{x_0} - a \frac{(Dx^-)^3}{6} \left. \frac{d^3f}{dx^3} \right|_{x_0} + \dots \quad (8-12)$$

To obtain the expression for the first derivative, subtract Eq (8-12) from Eq. (8-11).

$$\begin{aligned} f(x_0 + Dx^+) - a f(x_0 - Dx^-) &= f(x_0) - a f(x_0) + 2a Dx^- \left. \frac{df}{dx} \right|_{x_0} \\ &\quad + \frac{(a Dx^-)^2}{2} \left. \frac{d^2f}{dx^2} \right|_{x_0} - a \frac{(Dx^-)^2}{2} \left. \frac{d^2f}{dx^2} \right|_{x_0} + \dots \end{aligned} \quad (8-13)$$

and rearrange to solve for df/dx :

$$\left. \frac{df}{dx} \right|_{x_0} = \frac{f(x_0 + Dx^+) + (a - 1)f(x_0) - a f(x_0 - Dx^-)}{2a Dx^-} + O(Dx^-) \quad (8-14)$$

To obtain the expression for the second derivative, add (8-11) and (8-12):

$$f(x_0 + Dx^+) + a f(x_0 - Dx^-) = f(x_0) + a f(x_0) + \frac{(a Dx^-)^2}{2} \left. \frac{d^2f}{dx^2} \right|_{x_0} + a \frac{(Dx^-)^2}{2} \left. \frac{d^2f}{dx^2} \right|_{x_0} + O(Dx)^3 \dots$$

which is then solved for d^2f/dx^2 :

$$\left. \frac{d^2 f}{dx^2} \right|_{x_0} = \frac{f(x_0 + Dx^+) - (1+a)f(x_0) + af(x_0 - Dx^-)}{\frac{a}{2}(1+a)(Dx^-)^2} + O(Dx^-) \quad (8-15)$$

Note that both Eqs. (8-14) and (8-15) reduce to the forms given in Eq.(8-5) and Eq.(8-6) when the grid spacing is uniform.

Finally, note that a slightly more sophisticated analysis (Tannehill, *et al.*,¹ pages 61-63) will lead to a second order expression for the first derivative on unevenly spaced points:

$$\left. \frac{df}{dx} \right|_{x_0} = \frac{f(x_0 + Dx^+) + (a^2 - 1)f(x_0) - a^2 f(x_0 - Dx^-)}{a(a+1)Dx^-} + O(Dx^-)^2 \quad (8-16)$$

Tannehill, *et al.*,¹ give additional details and a collection of difference approximations using more than three points and difference approximations for mixed partial derivatives (Tables 3-1 and 3-2 on their pages 52 and 53). Numerous other methods of obtaining approximations for the derivatives are possible. The most natural one is the use of a polynomial fit through the points. Polynomials are frequently used to obtain derivative expressions on non-uniformly spaced grid points.

These formulas can also be used to represent partial derivatives. To simplify the notation, we introduce a grid and a notation common in finite difference formulations. Figure 8-2 illustrates this notation using $Dx = Dy = \text{const}$ for these examples.

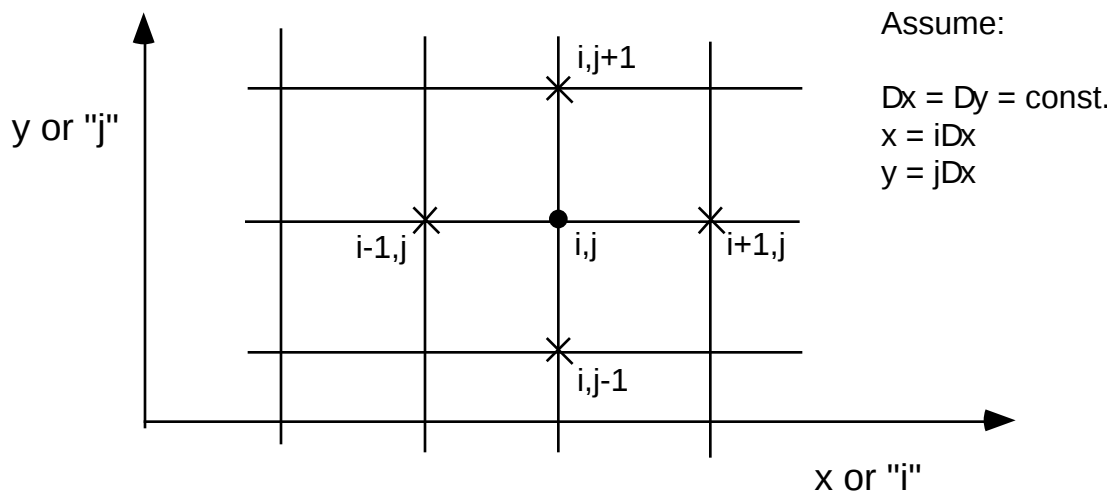


Figure 8-2. Nomenclature for use in partial differential equation difference expressions.

In this notation the following finite difference approximations for the first derivatives are:

$$\frac{\partial f}{\partial x} = \frac{f_{i+1,j} - f_{i,j}}{\Delta x} + O(\Delta x) \quad \text{1st order forward difference} \quad (8-17)$$

$$\frac{\partial f}{\partial x} = \frac{f_{i,j} - f_{i-1,j}}{\Delta x} + O(\Delta x) \quad \text{1st order backward difference} \quad (8-18)$$

$$\frac{\partial f}{\partial x} = \frac{f_{i+1,j} - f_{i-1,j}}{2\Delta x} + O(\Delta x)^2 \quad \text{2nd order central difference} \quad (8-19)$$

and the second derivative is:

$$\frac{\partial^2 f}{\partial x^2} = \frac{f_{i+1,j} - 2f_{i,j} + f_{i-1,j}}{(\Delta x)^2} + O(\Delta x)^2 \quad \text{2nd order central difference} \quad (8-20)$$

Similar expressions can be written for the y derivatives. To shorten the expressions, various researchers have introduced different shorthand notations to replace these expressions. The shorthand notation is then used in further operations on the difference expressions.

8.2 Finite difference representation of Partial Differential Equations (PDE's)

We can use the approximations to the derivatives obtained above to replace the individual terms in partial differential equations. The following figure provides a schematic of the steps required, and some of the key terms used to ensure that the results obtained are in fact the solution of the original partial differential equation. We will define each of these new terms below.

Steps and Requirements To Obtain a Valid Numerical Solution

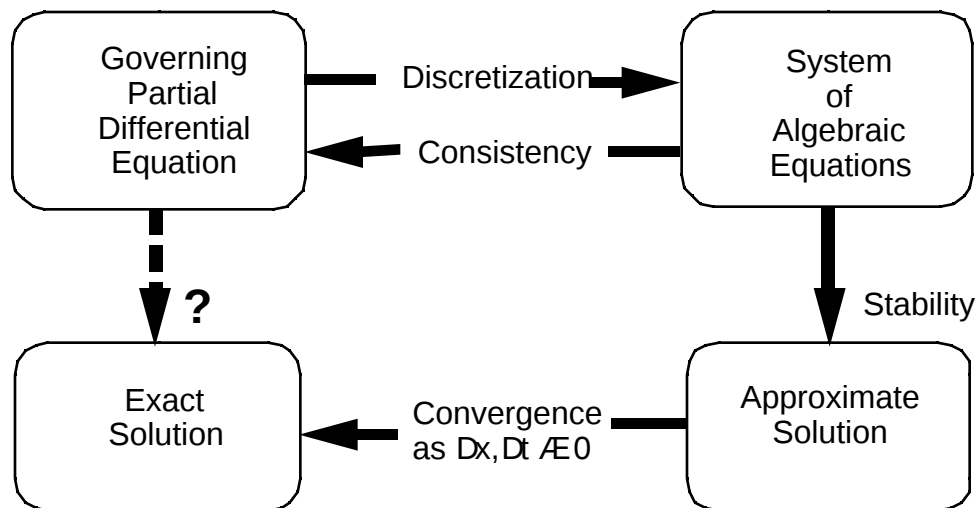


Figure 8-3. Overall procedure used to develop a CFD solution procedure.

Successful numerical methods for partial differential equations demand that the physical features of the PDE be reflected in the numerical approach. The selection of a particular finite difference approximation depends on the physics of the problem being studied. In large part the *type* of the PDE is crucial, and thus a determination of the type, *i.e.* elliptic, hyperbolic, or parabolic is extremely important. The mathematical type of the PDE must be used to construct the numerical scheme for approximating partial derivatives. Some advanced methods obscure the relationship, but it must exist. Consider the example given in Fig. 8-4 illustrating how information in a grid must be used.

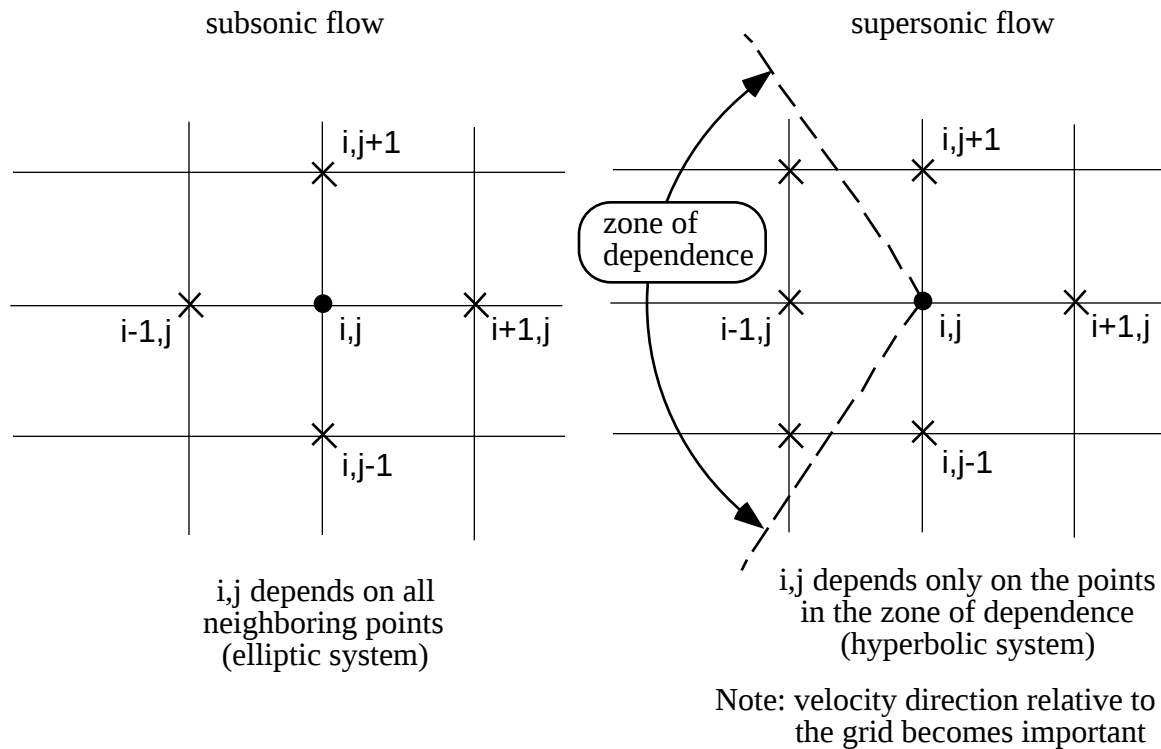


Figure 8-4. Connection between grid points used in numerical method and equation type.

Any scheme that fails to represent the physics correctly will fail when you attempt to obtain a solution. Furthermore, remember, in this case we were looking at a uniformly spaced cartesian grid. In actual “real life” applications we have to consider much more complicated non-uniform grids in non-Cartesian coordinate systems. In this section we use simple uniform Cartesian grid systems to illustrate the ideas. The necessary extensions of the methods illustrated in this chapter are outlined in the next chapter.

In Fig. 8-3 above, we introduced several important terms requiring definition and discussion:

- discretization
- consistency
- stability
- convergence

Before defining the terms, we provide an example using the heat equation:

$$\frac{\partial u}{\partial t} = a \frac{\partial^2 u}{\partial x^2}. \quad (8-21)$$

We discretize the equation using a forward difference in time, and a central difference in space following the notation shown in the following sketch:

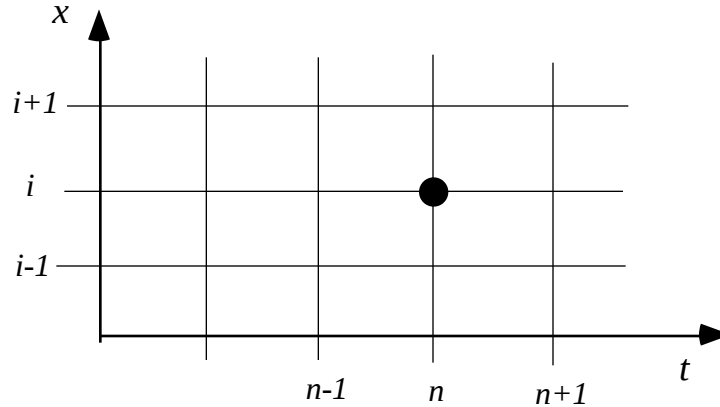


Figure 8-5. Grid nomenclature for discretization of heat equation.

The heat equation can now be written as:

$$\underbrace{\frac{\partial u}{\partial t} - a \frac{\partial^2 u}{\partial x^2}}_{\text{PDE}} = \underbrace{\frac{u_i^{n+1} - u_i^n}{\Delta t} - \frac{a}{(\Delta x)^2} (u_{i+1}^n - 2u_i^n + u_{i-1}^n)}_{\text{FDE}} + \underbrace{\left[\frac{\partial^2 u}{\partial t^2} \right]_i^n \frac{\Delta t}{2} + a \left[\frac{\partial^4 u}{\partial x^4} \right]_i^n \frac{(\Delta x)^2}{12} + \dots}_{\text{Truncation Error}} = 0 \quad (8-22)$$

where we use the superscript to denote time and the subscript to denote spatial location. In Eq. (8-18) the partial differential equation (PDE) is converted to the related finite difference equation (FDE). The truncation error is $O(\Delta t) + O(\Delta x)^2$ or $O[\Delta t(\Delta x)^2]$. An understanding of the truncation error for a particular scheme is important.

Using the model equation give here, we define the terms in the schematic given above:

discretization

This is the process of replacing derivatives by finite difference approximations. Replace continuous derivatives with an approximation at a discrete set of points (the mesh). This introduces an error due to the truncation error arising from the finite difference approximation

and any errors due to treatment of BC's. A reexamination of the Taylor series representation is worthwhile in thinking about the possible error arising from the discretization process:

$$\frac{\partial f}{\partial x} = \frac{f(x_0 + \Delta x) - f(x_0 - \Delta x)}{2\Delta x} + \underbrace{\frac{\Delta x^2}{6} \frac{\partial^3 f}{\partial x^3}}_{\text{formally valid for } \Delta x \rightarrow 0, \text{ but when } \Delta x = \text{finite, } \partial^3 f / \partial x^3 \text{ can be big for rapidly changing solutions (shock wave cases)}}. \quad (8-5a)$$

Thus we see that the size of the truncation error will depend locally on the solution. In most cases we expect the discretization error to be larger than round-off error.

consistency

A finite-difference representation of a PDE is *consistent* if the difference between the PDE and its difference representation vanishes as the mesh is refined, i.e.,

$$\lim_{\text{mesh} \rightarrow 0} (\text{PDE} - \text{FDE}) = \lim_{\text{mesh} \rightarrow 0} (\text{T.E.}) = 0 \quad (8-23)$$

When might this be a problem? Consider a case where the truncation error is $O(\Delta t / \Delta x)$. In this case we must let the mesh go to zero just such that:

$$\lim_{\Delta t, \Delta x \rightarrow 0} \frac{\Delta t}{\Delta x} = 0 \quad (8-24)$$

Some finite difference representations have been tried that weren't consistent. An example cited by Tannehill, *et al*,¹ is the DuFort-Frankel differencing of the wave equation.

stability

A stable numerical scheme is one for which errors from any source (round-off, truncation) are not permitted to grow in the sequence of numerical procedures as the calculation proceeds from one marching step, or iteration, to the next, thus:

errors grow $\Rightarrow$ unstable
errors decay $\Rightarrow$ stable

and

- Stability is normally thought of as being associated with marching problems.
- Stability requirements often dictate allowable step sizes.
- In many cases a stability analysis can be made to define the stability requirements.

convergence

The solution of the FDE's should approach the solution of the PDE as the mesh is refined. In the case of a linear equation there is a theorem which proves that the numerical solution to the FDE is in fact the solution of the original partial differential equation.

Lax Equivalence Theorem¹ (linear, initial value problem): For a properly posed problem, with a consistent finite difference representation, stability is the necessary and sufficient condition for convergence.

In practice, numerical experiments must be conducted to determine if the solution appears to be converged with respect to mesh size. * Machine capability and computing budget (time as well as money) dictate limits to the mesh size. Many, many results presented in the literature are not completely converged with respect to the mesh.

So far we have represented the PDE by an FDE at the point i,n . The PDE is now a set of algebraic equations written at each mesh point. If the grid is (in three dimensions) defined by a grid with $IMAX$, $JMAX$ and $KMAX$ mesh points in each direction, then we have a grid with $IMAX \times JMAX \times KMAX$ grid points. This can be a very large number. A typical recent case computed by one of my students was for the flow over a simple aircraft forebody. The calculation required 198,000 grid points. Thus the ability to carry out aerodynamic analysis using finite difference methods depends on the ability to solve large systems of algebraic equations efficiently.

We need to obtain the solution for the values at each grid point. We now consider how this is actually accomplished. Since the computer requirements and approach are influenced by the mathematical type of the equation being solved, we illustrate the basic types of approaches to the solution with two examples.

1st example - typical parabolic/hyperbolic PDEs

Explicit Scheme: Consider the finite difference representation of the heat equation given above in Eq. (8-18). Using the notation shown in the Fig. 8-6 below, we write the finite difference representation as:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{a}{(\Delta x)^2} (u_{i+1}^n - 2u_i^n + u_{i-1}^n) \quad (8-25)$$

and the solution at time step n is known. At time $n+1$ there is only one unknown.

* This is convergence with respect to grid. Another convergence requirement is associated with the satisfaction of the solution of a system of equations by iterative methods on a fixed grid.

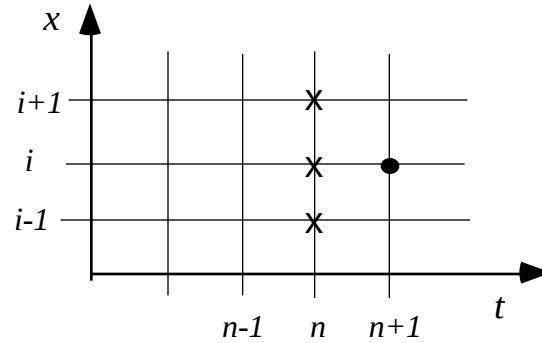


Figure 8-6. Grid points used in typical explicit calculation.

We solve for the value of u at i and the $n+1$ time step:

$$u_i^{n+1} = u_i^n + a \frac{\hat{E}}{\hat{E}(Dx)^2} \frac{Dt}{Dx} \left(u_{i+1}^n - 2u_i^n + u_{i-1}^n \right) \quad (8-26)$$

and thus at each i on $n+1$ we can solve for u_i^{n+1} algebraically, without solving a system of equations. This means that we can solve for each new value explicitly in terms of known values from the previous time step. This type of algorithm is known as an explicit scheme. It is a very straight forward procedure. To summarize:

- The algebra is simple.
- The bad news for non-vector computers: stability requirements require very small steps sizes.
- The good news: this scheme is easily vectorized* and a natural for massively parallel computation.

Implicit Scheme: Now consider an alternate finite difference representation of the heat equation given above in Eq. (8-18). Use the notation shown in the Fig. 8-7 below to define the location of grid points used to define the finite difference representation.

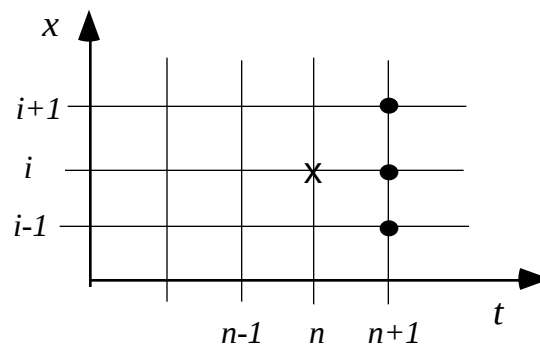


Figure 8-7. Grid points used in typical implicit calculation.

* see Chapter 3 for a brief discussion of vectorization.

Now we write the finite difference representation as:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{a}{(\Delta x)^2} (u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}) \quad (8-27)$$

where we use the spatial derivative at time $n+1$. By doing this we obtain a system where at each i on $n+1$, u_i^{n+1} depends on all the values at $n+1$. Thus we need to find the values along $n+1$ simultaneously. This leads to a system of algebraic equations that must be solved. For our model problem this system is linear. We can see this more clearly by rearranging Eq. (8-27). Defining

$$l = a \frac{\Delta t}{(\Delta x)^2} \quad (8-28)$$

we can re-write Eq.(8-27) after some minor algebra as:

$$-l u_{i-1}^{n+1} + (1 + 2l) u_i^{n+1} - l u_{i+1}^{n+1} = u_i^n \quad \text{for } i = 1, \dots, N. \quad (8-29)$$

This can be put into a matrix form to show that it has a particularly simple form:

$$\begin{pmatrix} 1+2l & -l & 0 & 0 & 0 & 0 & 0 \\ -l & 1+2l & -l & 0 & \ddots & 0 & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ \ddots & \ddots & -l & 1+2l & -l & \ddots & \ddots \\ \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & 0 & \ddots & 0 & -l & 1+2l & -l \\ 0 & 0 & 0 & 0 & 0 & -l & 1+2l \end{pmatrix} \begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ \vdots \\ u_i^{n+1} \\ \vdots \\ u_{N-1}^{n+1} \\ u_N^{n+1} \end{pmatrix} = \begin{pmatrix} u_1^n \\ u_2^n \\ \vdots \\ u_i^n \\ \vdots \\ u_{N-1}^n \\ u_N^n \end{pmatrix} \quad (8-30)$$

Equation (8-30) is a special type of matrix form known as a tridiagonal form. A particularly easy solution technique is available to solve this form. Known as the Thomas algorithm, the details are described in Section 8.5 and a routine called **tridag** is described in Appendix H-1. Many numerical methods are tailored to be able to produce this form.

The approach that leads to the formulation of a problem requiring the simultaneous solution of a system of equations is known as an *implicit* scheme. To summarize:

- The solution of a system of equations is required at each step.
- The good news: stability requirements allow a large step size.
- The not so good news: this scheme is harder to vectorize/parallelize.

A common feature for both explicit and implicit methods for parabolic and hyperbolic equations:

- A large number of mesh points can be treated, you only need the values at a small number of marching stations at any particular stage in the solution. This means you can obtain the solution with a large number of grid points using a relatively small amount of memory. Curiously, some recent codes don't take advantage of this last fact.

2nd Example - elliptic PDE

We use Laplace's equation as the model problem for elliptic PDE's:

$$f_{xx} + f_{yy} = 0 \quad (8-31)$$

and consider the grid shown below in Figure 8-8.

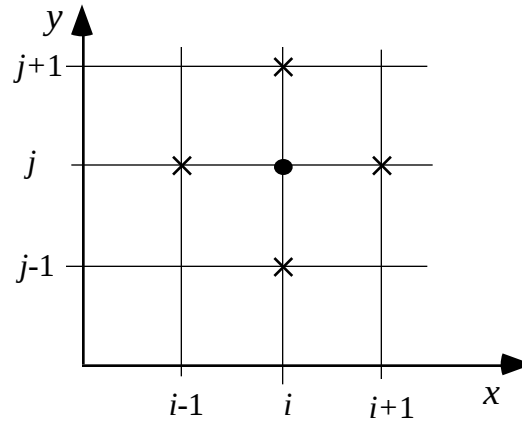


Figure 8-8. Grid points used in a typical representation of an elliptic equation.

Use the second order accurate central difference formulas at i,j :

$$f_{xx} = \frac{f_{i+1,j} - 2f_{i,j} + f_{i-1,j}}{(Dx)^2} + O(Dx)^2 \quad (8-32)$$

and:

$$f_{yy} = \frac{f_{i,j+1} - 2f_{i,j} + f_{i,j-1}}{(Dy)^2} + O(Dy)^2, \quad (8-33)$$

and substitute these expressions into the governing equation:

$$\frac{f_{i+1,j} - 2f_{i,j} + f_{i-1,j}}{(Dx)^2} + \frac{f_{i,j+1} - 2f_{i,j} + f_{i,j-1}}{(Dy)^2} = 0 \quad (8-34)$$

Solve this equation for f_{ij} :

$$f_{i,j} = \frac{(Dy)^2}{2[(Dx)^2 + (Dy)^2]} (f_{i+1,j} + f_{i-1,j}) + \frac{(Dx)^2}{2[(Dx)^2 + (Dy)^2]} (f_{i,j+1} + f_{i,j-1}) \quad (8-35)$$

where if $Dx = Dy$:

$$f_{i,j} = \frac{1}{4} (f_{i+1,j} + f_{i-1,j} + f_{i,j+1} + f_{i,j-1}) \quad (8-36)$$

This expression illustrates the essential physics of flows governed by elliptic PDE's:

- f_{ij} depends on all the values around it
- *all* values of f must be found simultaneously
- computer storage requirements are much greater than those required for parabolic/hyperbolic PDE's

Because of the large number of mesh points required to resolve the flowfield details, it is generally not practical to solve the system of equations arising from applying the above equation at each mesh point directly. Instead, an iterative procedure is usually employed. In this procedure an initial guess for the solution is made and then each mesh point in the flowfield is updated repeatedly until the values satisfy the governing equation. This iterative procedure can be thought of as having a time-like quality, which has been exploited in many solution schemes to find the steady flowfield.

A Note on Conservation Form

Care must be taken if the flowfield has discontinuities (shocks). In that case the correct solution of the partial differential equation will only be obtained if the conservative forms of the governing equations are used.

8.3 Other approaches, including the finite volume technique

Finite difference methods are the most well known methods in CFD. However other methods have also proven successful, and one method in particular, the finite volume technique, actually forms the basis for most current successful codes. The other methods in use are categorized as finite element and spectral. Each method eventually leads to a large set of algebraic equations, just as with the finite difference methods. See References 1 and 3 for more details of the latter two methods. In US aircraft aerodynamics work they don't currently have an impact.

The finite volume method *is* important. Instead of discretizing the PDE, select the integral form of the equations. Recall that each conservation law had both differential and integral statements. The integral form is more fundamental, not depending on continuous partial derivatives.

Example of Finite Volume Approach (Fletcher,³ vol. I, pg.105-116, Tannehill, *et al*,¹ pg 71-76)

Consider the general conservation equation (in two dimensions for our example analysis):

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = 0. \quad (8-37)$$

Pick the particular form to be conservation of mass:

$$\begin{aligned}
 \mathbf{q} &= r \\
 \mathbf{F} &= r u, \\
 \mathbf{G} &= r v
 \end{aligned}
 \tag{8-38}$$

and recall that this conservation law could also come from the integral statement:

$$\frac{\partial}{\partial t} \iiint_V r dV = - \oint_S r \mathbf{V} \cdot \mathbf{n} dS.
 \tag{8-39}$$

Introducing the notation defined above and assuming two dimensional flow, the conservation law can be rewritten as:

$$\frac{\partial}{\partial t} \iiint_V r dV + \oint_S \mathbf{H} \cdot \mathbf{n} dS = 0
 \tag{8-40}$$

where

$$\mathbf{H} = (\mathbf{F}, \mathbf{G}) = r \mathbf{V}
 \tag{8-41}$$

and

$$\begin{aligned}
 H_x &= F = r u \\
 H_y &= G = r v
 \end{aligned}
 \tag{8-42}$$

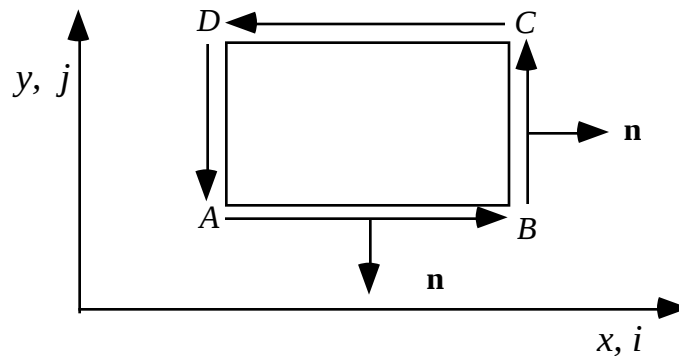


Figure 8-9. Basic nomenclature for finite volume analysis.

Using the definition of $\mathbf{n}$ in Cartesian coordinates, and considering for illustration the Cartesian system given in Fig. 8-9, we can write:

$$\begin{aligned}
 \mathbf{H} \cdot \mathbf{n} dS &= (H_x \mathbf{i} + H_y \mathbf{j}) \cdot \mathbf{n} dS \\
 &= (F \mathbf{i} + G \mathbf{j}) \cdot \mathbf{n} dS
 \end{aligned}
 \tag{8-43}$$

along AB, $\mathbf{n} = -\mathbf{j}$, $dS = dx$, and:

$$\mathbf{H} \cdot \mathbf{n} dS = - G dx
 \tag{8-44}$$

along BC, $\mathbf{n} = \mathbf{i}$, $dS = dy$, and:

$$\mathbf{H} \cdot \mathbf{n} dS = F dy
 \tag{8-45}$$

or in general:

$$\mathbf{H} \cdot \mathbf{n} dS = F dy - G dx. \quad (8-46)$$

Using the general grid shown in the Fig. 8-10, our integral statement, Eq. (8-40) can be written as:

$$\frac{\partial}{\partial t} (A q_{j,k}) + \hat{\mathbf{A}}_{AB}^{DA} (F Dy - G Dx) = 0. \quad (8-47)$$

Here A is the area of the quadrilateral $ABCD$, and $q_{i,j}$ is the average value of q over $ABCD$.

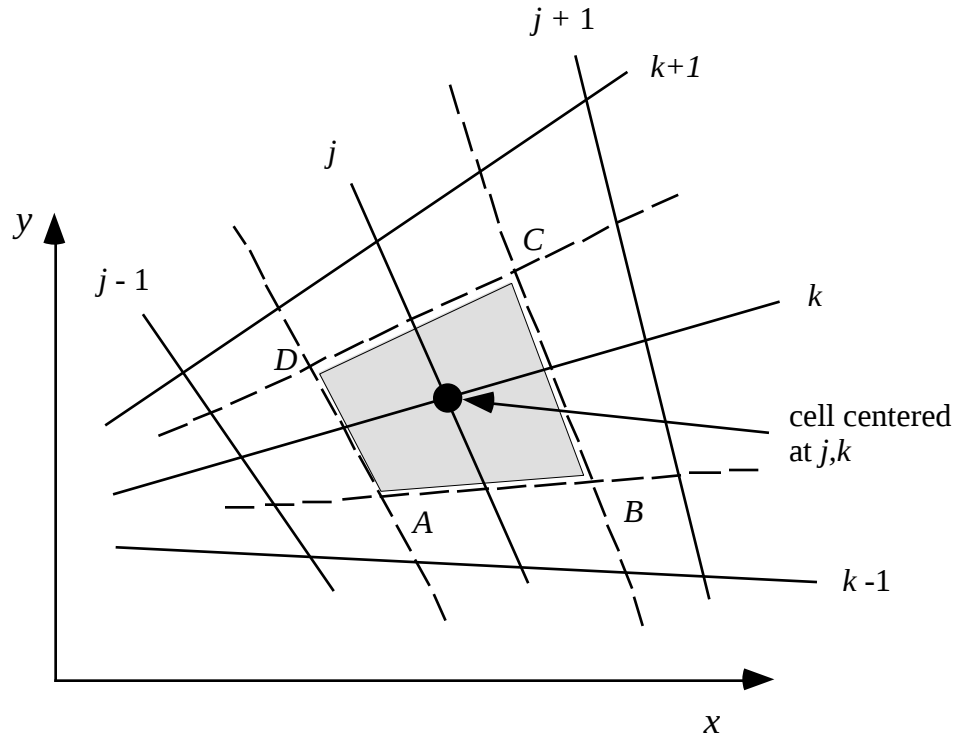


Figure 8-10. Circuit in a general grid system.

Now define the quantities over each face. For illustration consider AB :

$$\begin{aligned} Dy_{AB} &= y_B - y_A \\ Dx_{AB} &= x_B - x_A \\ F_{AB} &= \frac{1}{2} (F_{j,k-1} + F_{j,k}), \\ G_{AB} &= \frac{1}{2} (G_{j,k+1} + G_{j,k}) \end{aligned} \quad (8-48)$$

and so on over the other cell faces.

Assuming A is not a function of time, and combining:

$$\begin{aligned}
 A \frac{\partial q_{j,k}}{\partial t} + \frac{1}{2} (F_{j,k-1} + F_{j,k}) Dy_{AB} - \frac{1}{2} (G_{j,k-1} + G_{j,k}) Dx_{AB} \\
 + \frac{1}{2} (F_{j,k} + F_{j+1,k}) Dy_{BC} - \frac{1}{2} (G_{j,k} + G_{j+1,k}) Dx_{BC} \\
 + \frac{1}{2} (F_{j,k} + F_{j,k+1}) Dy_{CD} - \frac{1}{2} (G_{j,k} + G_{j,k+1}) Dx_{CD} \\
 + \frac{1}{2} (F_{j-1,k} + F_{j,k}) Dy_{DA} - \frac{1}{2} (G_{j-1,k} + G_{j,k}) Dx_{DA} \\
 = 0
 \end{aligned} \quad (8-49)$$

Supposing the grid is regular cartesian as shown in Fig. 8-11. Then $A = Dx Dy$, and along:

$$\begin{aligned}
 AB: \quad Dy = 0, \quad Dx_{AB} = Dx \\
 BC: \quad Dx = 0, \quad Dy_{BC} = Dy \\
 CD: \quad Dy = 0, \quad Dx_{CD} = -Dx \\
 DA: \quad Dx = 0, \quad Dy_{DA} = -Dy
 \end{aligned} \quad (8-50)$$

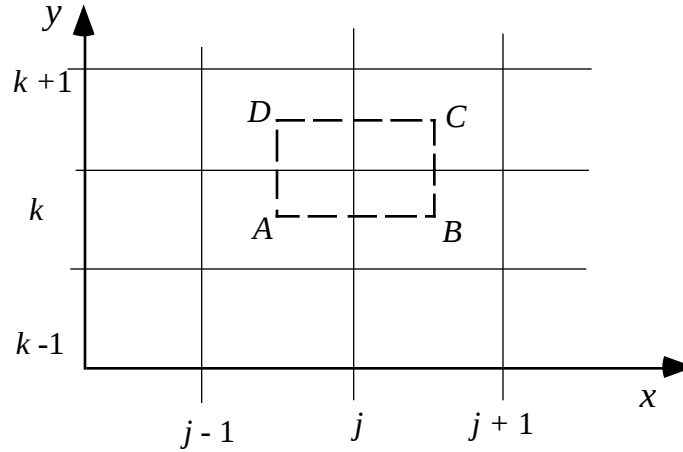


Figure 8-11. General finite volume grid applied in Cartesian coordinates.

Thus, in Eq. (8-49) we are left with:

$$\begin{aligned}
 Dx Dy \frac{\partial q_{j,k}}{\partial t} - \frac{1}{2} (G_{j,k-1} + G_{j,k}) Dx + \frac{1}{2} (F_{j,k} + F_{j+1,k}) Dy \\
 + \frac{1}{2} (G_{j,k} + G_{j,k+1}) Dx - \frac{1}{2} (F_{j-1,k} + F_{j,k}) Dy = 0
 \end{aligned} \quad (8-51)$$

Collecting terms:

$$\frac{\partial q_{j,k}}{\partial t} + \underbrace{\frac{F_{j+1,k} - F_{j-1,k}}{2\Delta x} + \frac{G_{j,k+1} - G_{j,k-1}}{2\Delta y}}_{\substack{\text{for this reversion to Cartesian} \\ \text{coordinates the equation just reduces} \\ \text{to simple central differences of the original} \\ \text{partial differential equation}}} = 0 \quad (8-52)$$

or:

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = 0. \quad (8-53)$$

Thus, and at first glance remarkably, the results of the finite volume approach can lead to the exact same equations to solve as the finite difference method on a simple Cartesian mesh.

However, the interpretation is different:*

- Finite difference: approximates the governing equation at a point
- Finite volume: " " " " over a volume
- Finite volume is the most physical in fluid mechanics codes, and is actually used in most codes.
- Finite difference methods were developed earlier, the analysis of methods is easier and further developed.

Both the finite difference and finite volume methods are very similar. However, there are differences. They are subtle but important. We cite three points in favor of the finite volume method compared to the finite difference method:

- Good conservation of mass, momentum, and energy using integrals when mesh is finite size
- Easier to treat complicated domains (integral discretization [averaging] easier to figure out, implement, and interpret)
- Average integral concept much better approach when the solution has shock waves (i.e. the partial differential equations assume continuous partial derivatives).

Finally, special considerations are needed to implement some of the boundary conditions in this method. The references, in particular Fletcher,³ should be consulted for more details.

8.4 Boundary conditions

So far we have obtained expressions for interior points on the mesh. However, the actual geometry of the flowfield we wish to analyze is introduced through the boundary conditions. We use an elliptic PDE problem to illustrate the options available for handling boundary conditions.

Consider the flow over a symmetric airfoil at zero angle of attack, as shown in Fig. 8-12.

* Summarized from Professor B. Grossman's unpublished CFD notes.

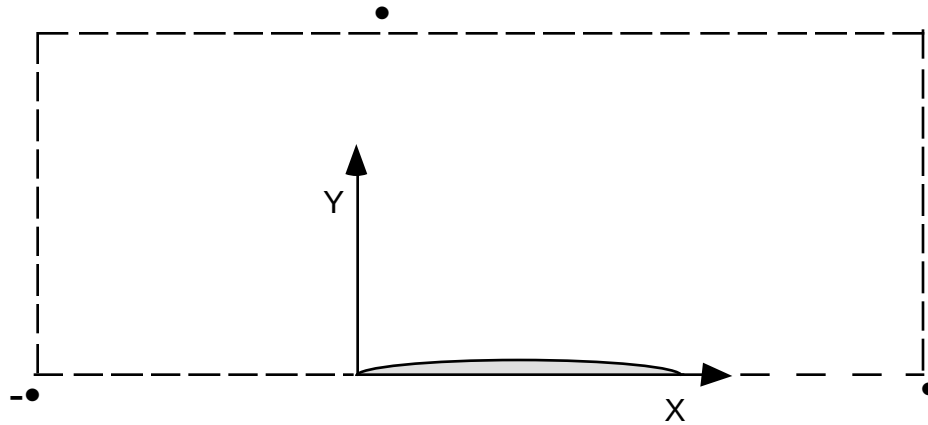


Figure 8-12. Example of boundary condition surface requiring consideration.

Here, because there is no lift, symmetry allows us to solve only the top half of the region. If f is a perturbation potential [see Chap. 2, Eq. (2-123)],

$$\begin{aligned} U &= U_{\infty} + f_x \\ V &= f_y \end{aligned} \quad (8-54)$$

then far away from the surface,

$$u = v = 0 \quad (8 - 55)$$

or

$$f \rightarrow 0 \quad \text{as} \quad x^2 + y^2 \rightarrow \infty. \quad (8 - 56)$$

For a lifting airfoil, the farfield potential must take the form of a potential vortex singularity with a circulation equal to the circulation around the airfoil.

The boundary condition on the surface of primary interest is the flow tangency condition, where the velocity normal to the surface is specified. In most cases the velocity normal to the surface is zero.

Consider ways to handle the farfield BC

There are several possibilities:

- A. “go out” far enough (?) and set $f = 0$ for $f \rightarrow \infty$, as the distance from the body goes to infinity (or $v = 0$, $u = 0$ where these are the perturbation velocities, or $u = U_{\infty}$ if it is the total velocity).

How good is this? This method is frequently used, although clearly it requires numerical experimentation to ensure that the boundary is “far enough” from the body. In lifting cases this can be on the order of 50 chord lengths in two-dimensions. In

addition, this approach leads to excessive use of grid points in regions where we normally aren't interested in the details of the solution.

- B. Transform the equation to another coordinate system, and satisfy the boundary condition explicitly at infinity (details of this approach are given in Chap. 9).

Figure 8-13 demonstrates what we mean. In the x system the physical distance from 0 to infinity is transformed to the range from 0 to 1. Although this approach may lead to efficient use of grid points, the use of the resulting highly stretched grid in the physical plane may result in numerical methods that lose accuracy, and even worse, do not converge during an iterative solution.

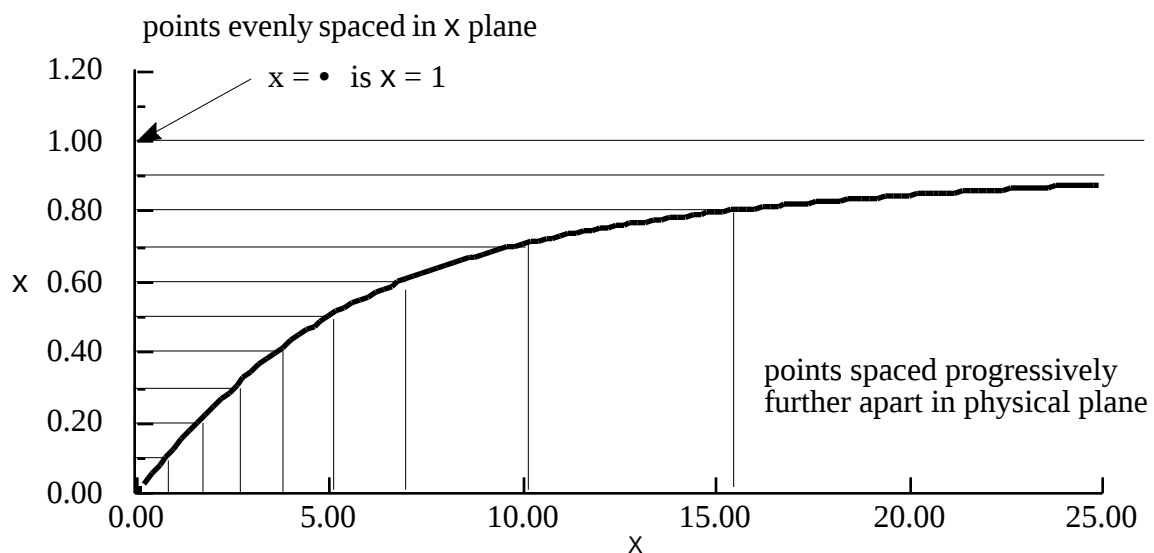


Figure 8-13. One example of a way to handle the farfield boundary condition.

- C. Blocks of Grids are sometimes used, a dense “inner” grid and a “coarse” outer grid. In this approach the grid points are used efficiently in the region of interest. It is a simple version of the adaptive grid concept, where the the grid will adjust automatically to concentrate points in regions of large flow gradients.
- D. Match the numerical solution to an analytic approximation for the farfield boundary condition.

This is emerging as the standard way to handle the farfield boundary conditions. It allows the outer boundary to be placed at a reasonable distance from the body, and properly done, it ensures that the boundary numerical solution reflects the correct physics at the boundary. This has been found to be particularly important in the solution

of the Euler equations. Effort is still underway to determine the best way to implement this approach.

To summarize this discussion on farfield boundary conditions:

- BC's on the FF boundary are important, and can be especially important for Euler codes which march in time to a steady state final solution.
- How to best enforce the FF BC is still under study - research papers are still being written describing new approaches.

Consider ways to handle the nearfield BC

There are also several ways to approach the satisfaction of boundary conditions on the surface. Here we discuss three.

- A. Use a standard grid and allow the surface to intersect grid lines in an irregular manner. Then, solve the equations with BC's enforced between node points. Figure 8-14 illustrates this approach. In the early days of CFD methodology development this approach was not found to work well, and the approach discussed next was developed. However, using the finite volume method, an approach to treat boundary conditions imposed in this manner was successfully developed (primarily by NASA Langley and its contractors and grantees). It has not become a popular approach, and is considered to lead to an inefficient use of grid points. Many grid points end up inside the body.

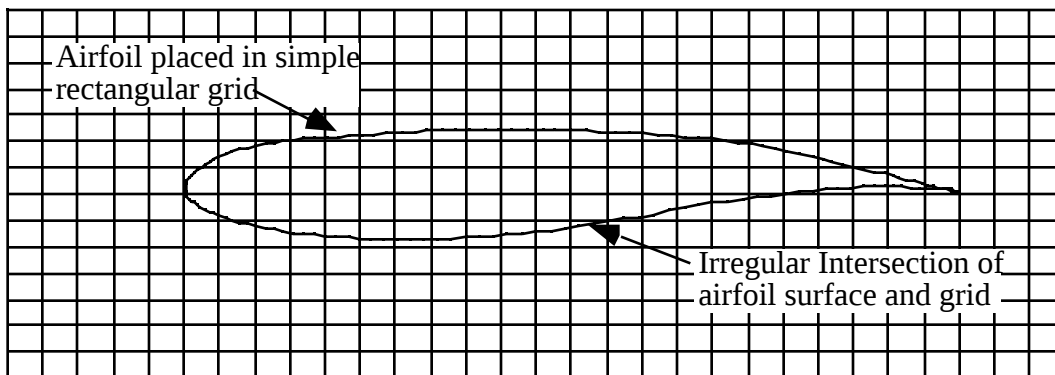


Figure 8-14. Surface passing through a general grid.

- B. The most popular approach to enforcing surface boundary conditions is to use a coordinate system constructed such that the surface of the body is a coordinate surface. An example of this approach is shown in Figure 8-15. This is currently the method of choice and by far the most popular approach employed in CFD. It works well. However, it complicates the problem formulation. To use this approach, grid generation became an area of study by itself. Grid generation is discussed in Chapter 9.

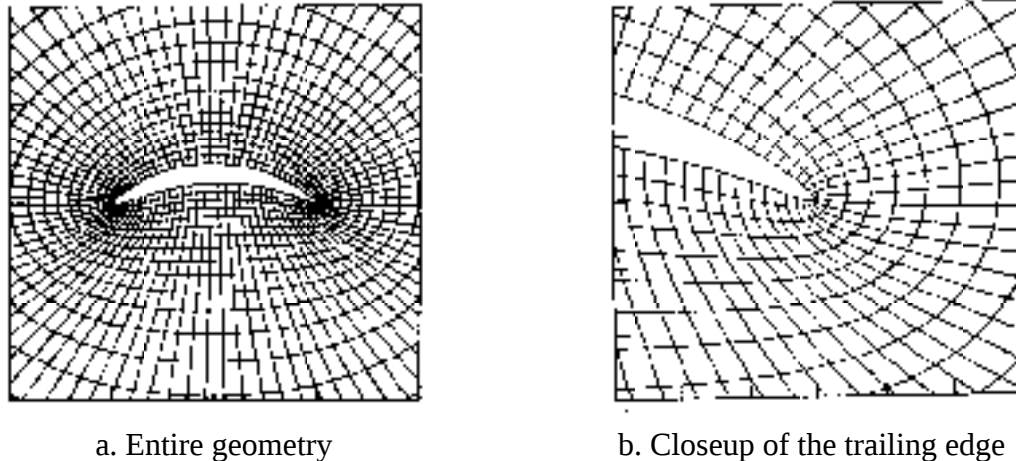


Figure 8-15. Body conforming grid for easy application of BCs on curved surfaces.¹⁰

- C. Another approach is to use thin airfoil theory boundary conditions, as described in detail in Chapter 6. This eliminates many of the problems associated with the first two approaches. It is expedient, but at some loss in accuracy (but very likely not that much, as shown in Chap. 6, Fig. 6-14).

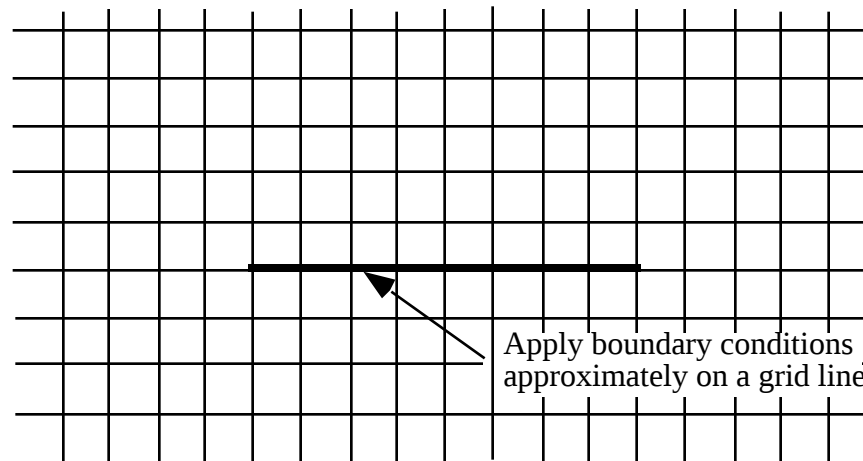


Figure 8-16. Approximate approach to boundary condition specification.

Finite difference representation of the BC's

After defining a coordinate system, the finite difference representation of the boundary condition must be written down. Using Laplace's Equation as an example, consider that there are normally two types of boundary conditions associated with the boundary: 1) the Dirichlet problem, where f is specified on the boundary, and 2) the Neumann problem, where $\partial f / \partial n$ is specified. If the Dirichlet problem is being solved, the value on the boundary is simply specified and no special difference formulas are required.

When the solution requires that the gradient normal to the surface be specified, a so-called “dummy row” is the easiest way to implement the boundary condition. As an example, following Moran,⁴ consider a case where the normal velocity, v , is set to zero at the outer boundary. The boundary is at grid line $j = NY$. Assume that another row is added at $j = NY + 1$, as indicated in Fig. 8-17.

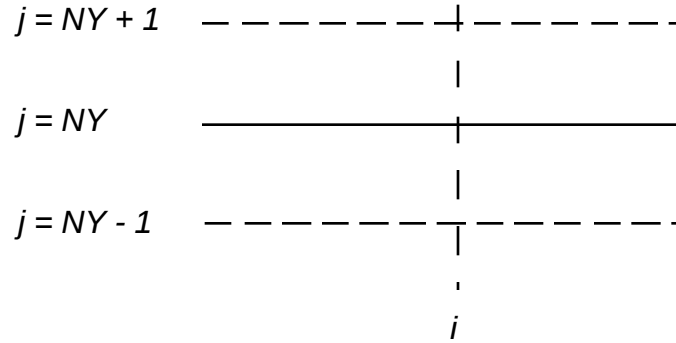


Figure 8-17. Boundary condition at farfield.

The required boundary condition at $j = NY$ is:

$$\frac{\partial f}{\partial n} = 0 = \frac{f_{i,NY+1} - f_{i,NY-1}}{Y_{NY+1} - Y_{NY-1}} + O(DY)^2 \quad (8-57)$$

and to ensure that the boundary condition is satisfied, simply define:

$$f_{i,NY+1} \equiv f_{i,NY-1}. \quad (8-58)$$

The equations are then solved up to Y_{NY} , and whenever you need f at $NY+1$, simply use the value at $NY-1$.

Now, we present an example demonstrating the application of thin airfoil theory boundary conditions at the surface. Recall that the boundary condition is:

$$v = \frac{\partial f}{\partial y} = U_{\infty} \frac{df}{dx} \quad (8-59)$$

where $y_{\text{foil}} = f(x)$. Assuming that in the computer code v has been nondimensionalized by U_{∞} , the boundary condition is:

$$\frac{\partial f}{\partial y} = \frac{df}{dx} \quad (8-60)$$

and the grid near the surface is defined following Fig. 8-18.

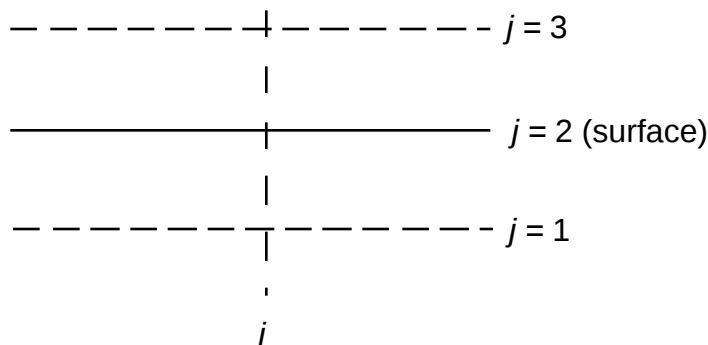


Figure 8-18. Boundary condition at surface.

Writing the derivative in terms of central differences at $j = 2$,

$$\frac{f_{i,3} - f_{i,1}}{Y_3 - Y_1} = \frac{df}{dx} \quad (8-61)$$

we solve for $f_{i,1}$:

$$f_{i,1} = f_{i,3} - (Y_3 - Y_1) \frac{df}{dx}. \quad (8-62)$$

Note that since $j = 1$ is a dummy row, you can select the grid spacing such that the spacing is equal on both sides of $j = 2$, resulting in second order accuracy. Thus, as in the previous example, anytime we need $f_{i,1}$ we use the value given by Eq. (8-62). Using these boundary condition relations, the boundary conditions are identically satisfied. Note also that this approach is the reason that in many codes the body surface corresponds to the second grid line, $j = 2$.

Imposition of boundary conditions is sometimes more difficult than the analysis given here suggests. Specifically, both the surface and farfield boundary conditions for the pressure in the Navier-Stokes and Euler equations can be tricky.

8.5 Solution of Algebraic Equations

We now know how to write down a representation of the PDE at each grid point. The next step is to solve the resulting system of equations. Recall that we have one algebraic equation for each grid point. The system of algebraic equations may, or may not, be linear. If they are nonlinear, the usual approach is to form an approximate linear system, and then solve the system iteratively to obtain the solution of the original nonlinear system. The accuracy requirement dictates the number of the grid points required to obtain the solution. Previously, we assumed that linear equation solution subroutines were available, as discussed in Chapter 3. However, the development of CFD methods requires a knowledge of the types of algebraic systems of equations.

Recall that linear algebraic equations can be written in the standard form:

$$\mathbf{Ax} = \mathbf{b} \quad (8-63)$$

For an inviscid two-dimensional solution, a grid of 100×30 is typical. This is 3000 grid points, and results in a matrix 3000×3000 . In three dimensions, 250,000 ~ 300,000 grid points are common, 500,000 points are not uncommon, and a million or more grid points are often required. Clearly, you can't expect to use classical direct linear equation solvers for systems of this size.

Standard classification of algebraic equations depends on the characteristics of the elements in the matrix $\mathbf{A}$. If $\mathbf{A}$:

1. contains few or no zero coefficients, it is called dense,
2. contains many zero coefficients, it is called sparse,
3. contains many zero coefficients, *and* the non-zero coefficients are close to the main diagonal: the $\mathbf{A}$ matrix is called sparse and banded.

Dense Matrix

For a dense matrix direct methods are appropriate. Gauss elimination is an example of the standard approach to these systems. LU decomposition¹¹ is used in program **PANEL**, and is an example of a standard method for solution of a dense matrix. These methods are not good for large matrices (> 200-400 equations). The run time becomes huge, and the results may be susceptible to round-off error.

Sparse and Banded

Special forms of Gauss elimination are available in many cases. The most famous banded matrix solution applies to so-called tridiagonal systems:

$$\begin{pmatrix} b_1 & c_1 & & & & \\ a_2 & b_2 & c_2 & & & \\ & \ddots & \ddots & \ddots & & \\ & & a_i & b_i & c_i & \\ & & & \ddots & \ddots & \ddots \\ & & & & 0 & a_{N-1} & b_{N-1} & c_{N-1} \\ & & & & & a_N & b_N & c_N \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \\ \vdots \\ x_{N-1} \\ x_N \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_i \\ \vdots \\ d_{N-1} \\ d_N \end{pmatrix} \quad (8-64)$$

The algorithm used to solve Eq. (8-64) is known as the Thomas algorithm. This algorithm is very good and widely used. The Thomas algorithm is given in detail in the sidebar, and a sample subroutine, **tridag**, is described in App H-1.

Solution of tridiagonal systems of equations

The Thomas Algorithm is a special form of Gauss elimination that can be used to solve tridiagonal systems of equations. When the matrix is tridiagonal, the solution can be obtained in $O(n)$ operations, instead of $O(n^3/3)$. The form of the equation is:

$$a_i x_{i-1} + b_i x_i + c_i x_{i+1} = d_i \quad i = 1, \dots, n$$

where a_1 and c_n are zero. The solution algorithm¹² starts with $k = 2, \dots, n$:

$$m = \frac{a_k}{b_{k-1}}$$

$$b_k = b_k - m c_{k-1}.$$

$$d_k = d_k - m d_{k-1}$$

Then:

$$x_n = \frac{d_n}{b_n}$$

and finally, for $k = n - 1, \dots, 1$:

$$x_k = \frac{d_k - c_k x_{k+1}}{b_k}.$$

In CFD methods this algorithm is usually coded directly into the solution procedure, unless machine optimized subroutines are employed on a specific computer.

General Sparse

These matrices are best treated with iterative methods. In this approach an initial estimate of the solution is specified (often simply 0), and the solution is then obtained by repeatedly updating the values of the solution vector until the equations are solved. This is also a natural method for solving nonlinear algebraic equations, where the equations are written in the linear equation form, and the coefficients of the **A** matrix are changed as the solution develops during the iteration. Many methods are available.

There is one basic requirement for iterative solutions to converge. The elements on the diagonal of the matrix should be large relative to the values off the diagonal. The condition can be given mathematically as:

$$|a_{ii}| \geq \sum_{j=1, j \neq i}^n |a_{ij}| \quad (8-65)$$

and for at least one row:

$$|a_{ii}| > \sum_{j=1}^n |a_{ij}| \quad (8-66)$$

A matrix that satisfies this condition is diagonally dominant, and, for an iterative method to converge, the matrix must be diagonally dominant. One example from aerodynamics of a matrix that arises which is not diagonally dominant is the matrix obtained in the monoplane equation formulation for the solution of the lifting line theory problem.

One class of iterative solution methods widely used in CFD is “relaxation.” As an example, consider Laplace’s Equation. Start with the discretized form, Eq.(8-31). The iteration proceeds by solving the equation at each grid point i,j at an iteration $n+1$ using values found at iteration n . Thus the solution at iteration $n+1$ is found from:

$$f_{i,j}^{n+1} = \frac{1}{4} [f_{i+1,j}^n + f_{i-1,j}^n + f_{i,j+1}^n + f_{i,j-1}^n]. \quad (8-67)$$

The values of f are computed repeatedly until they are no longer changing. The “relaxation” of the values of f to final converged values is roughly analogous to determining the solution for an unsteady flow approaching a final steady state value, where the iteration cycle is identified as a time-like step. This is an important analogy. Finally, the idea of “iterating until the values stop changing” as an indication of convergence is not good enough. Instead, we must check to see if the finite difference representation of the partial differential equation using the current values of f actually satisfies the partial differential equation. In this case, the value of the equation should be zero, and the actual value of the finite difference representation is known as the *residual*. When the residual is zero, the solution has converged. This is the value that should be monitored during the iterative process. Generally, as done in **THINFOIL**, the maximum residual and its location in the grid, and the average residual are computed and saved during the iterative process to examine the convergence history.

Note that this method uses all old values of f to get the new value of f . This approach is known as *point Jacobi* iteration. You need to save all the old values of the array as well as the new values. This procedure converges only very slowly to the final converged solution.

A more natural approach to obtaining the solution is to use new estimates of the solution as soon as they are available. Figure 8-19 shows how this is done using a simply programmed systematic sweep of the grid. With a conventional sweep of the grid this becomes:

$$f_{i,j}^{n+1} = \frac{1}{4} [f_{i+1,j}^n + f_{i-1,j}^{n+1} + f_{i,j+1}^n + f_{i,j-1}^{n+1}]. \quad (8-68)$$

This scheme is called the *point Gauss-Seidel* iteration. It also eliminates the need to store all the old iteration values as well as all the new iteration results, which was required with the point Jacoby method.

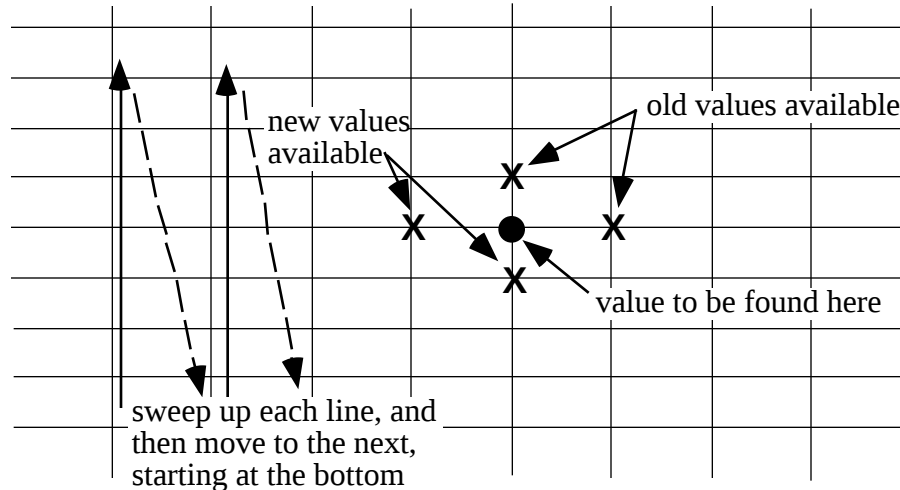


Figure 8-19. Grid sweep approach to implement the Gauss-Seidel solution iteration scheme.

The point Gauss-Seidel iteration procedure also converges slowly. One method of speeding up the convergence is to make the change to the value larger than the change indicated by the normal Gauss-Seidel iteration. Since the methods that have been described are known as relaxation methods, the idea of increasing the change is known as successive over-relaxation, or SOR. This is implemented by defining an intermediate value:

$$\hat{f}_{i,j}^{n+1} = \frac{1}{4} [f_{i+1,j}^n + f_{i-1,j}^{n+1} + f_{i,j+1}^n + f_{i,j-1}^{n+1}] \quad (8-69)$$

and then obtaining the new value as:

$$f_{i,j}^{n+1} = f_{i,j}^n + w(\hat{f}_{i,j}^{n+1} - f_{i,j}^n). \quad (8-70)$$

The parameter w is a relaxation parameter. If it is unity, the basic Gauss-Seidel method is recovered. How large can we make it? For most model problems, a stability analysis (presented in the next section) indicates that $w < 2$ is required to obtain a converging iteration. The best value of w depends on the grid and the actual equation. For most cases of practical interest the best values of w must be determined through numerical experimentation. Figure 8-20 presents an example of the manner in which the solution evolves with iterations. The value of f after 2000 iterations is approached very gradually. The figure also illustrates the time-like nature of the iteration.

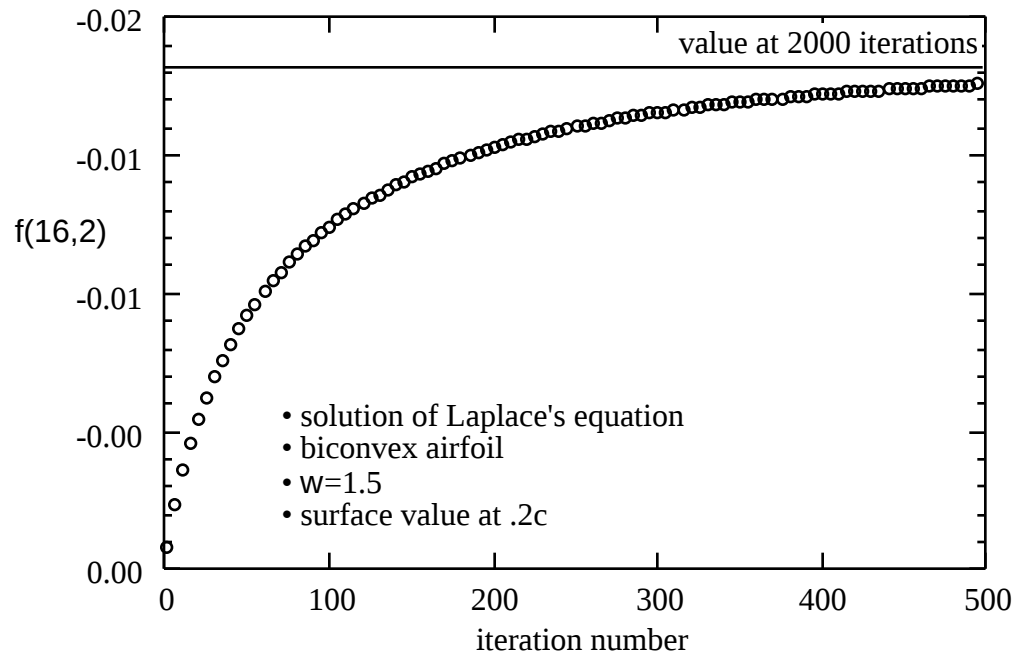


Figure 8-20. Typical variation of f during solution iteration.

Another way to speed up the iteration is to sweep the flowfield a “line” at a time rather than a point at a time. Applying over-relaxation to this process, the so-called successive line over-relaxation, or SLOR, process is obtained. In this method a system of equations must be solved at each line. Figure 8-21 illustrates this approach. The method is formulated so that the system of equations is tridiagonal, and the solution is obtained very efficiently. This approach provides a means of spreading the information from new values more quickly than the point by point sweep of the flowfield. However, all of these approaches result in a very slow approach to the final value during the iterations.

The effect of the value of the over relaxation parameter is shown in Figure 8-22. Here, the convergence level is compared for various values of w . Notice that as convergence requirements are increased, the choice of w becomes much more important. Unfortunately, the choice of w may not only be dependent on the particular numerical method, but also on the particular problem being solved.

Mathematically, the convergence rate of an iterative process depends on the value of the so-called *spectral radius* of the matrix relating the value of the unknowns at one iteration to the values of the unknowns at the previous iteration. The spectral radius is the absolute value of the largest eigenvalue of the matrix. The spectral radius must be less than one for the iterative process to converge. The smaller the value of the spectral radius, the faster the convergence.

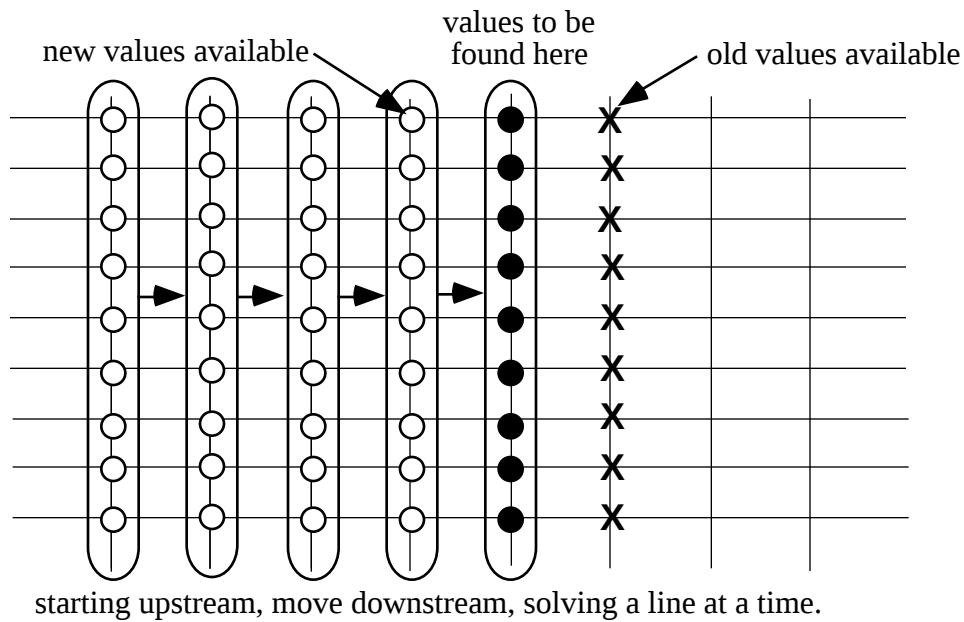
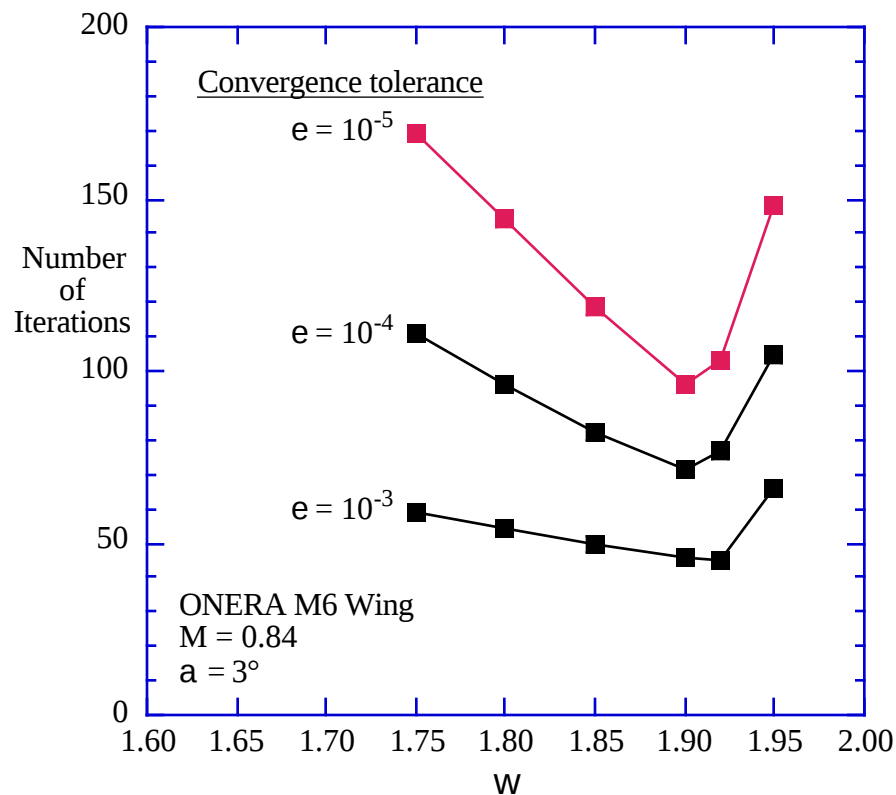


Figure 8-21. Solution approach for SLOR.

Figure 8-22. Effect of the value of w on the number of iterations required to achieve various levels of convergence.¹³

Another way to spread the information rapidly is to alternately sweep in both the x and y direction. This provides a means of obtaining the final answers even more quickly, and is known

as an alternating direction implicit or ADI method. Figure 8-23 illustrates the modification to the SLOR method that is used to implement an ADI scheme. Several different methods of carrying out the details of this iteration are available. The traditional approach for linear equations is known as the Peaceman-Rachford method, and is described in standard textbooks, *e.g.*, Ames⁸ and Isaacson and Keller.¹⁴ This approach is also known as an approximate factorization or “AF” scheme. It is known as AF1 because of the particular approach to the factorization of the operator. A discussion of ADI including a computer program is given in the first edition of the *Numerical Recipes* book.¹⁵

Another approach has been found to be more robust for nonlinear partial differential equations, including the case of mixed sub- and supersonic flow. In this case the time-like nature of the approach to a final value is used explicitly to develop a robust and rapidly converging iteration scheme. This scheme is known as AF2. This method was first proposed for steady flows by Ballhaus, et al.,¹⁶ and Catherall¹⁷ provided a theoretical foundation and results from numerical experiments. A key aspect of ADI or any AF scheme is the use of a sequence of relaxation parameters rather than a single value, as employed in the SOR and SLOR methods. Typically, the sequence repeats each eight to eleven iterations.

Holst¹⁰ has given an excellent review and comparison of these methods. Figure 8-24, from Holst,¹⁰ shows how the different methods use progressively “better” information at a point to find the solution with the fewest possible iterations. The advantage is shown graphically in Figure 8-25, and is tabulated in Table 8-1 (also from Holst¹⁰). Program **THINFOIL**, described in Section 8.7, uses these methods and App. G-1 contains a description of the theoretical implementation of these methods. Further details are given in Chapter 11, Transonic Aerodynamics.

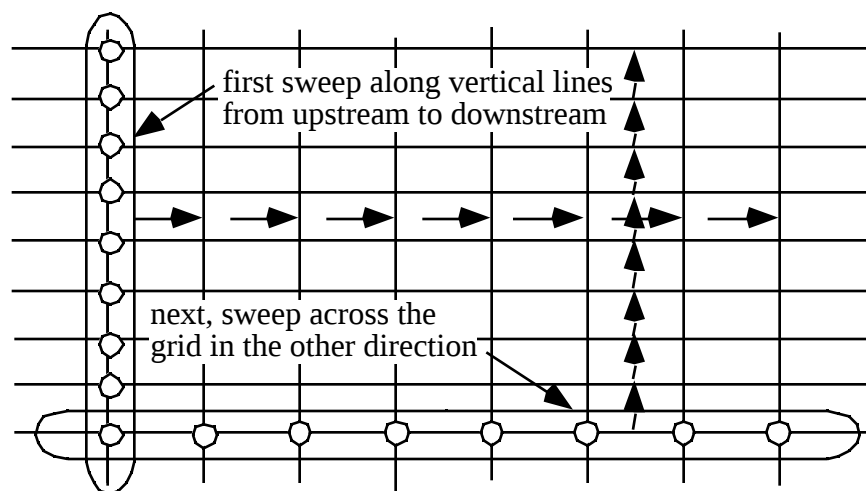
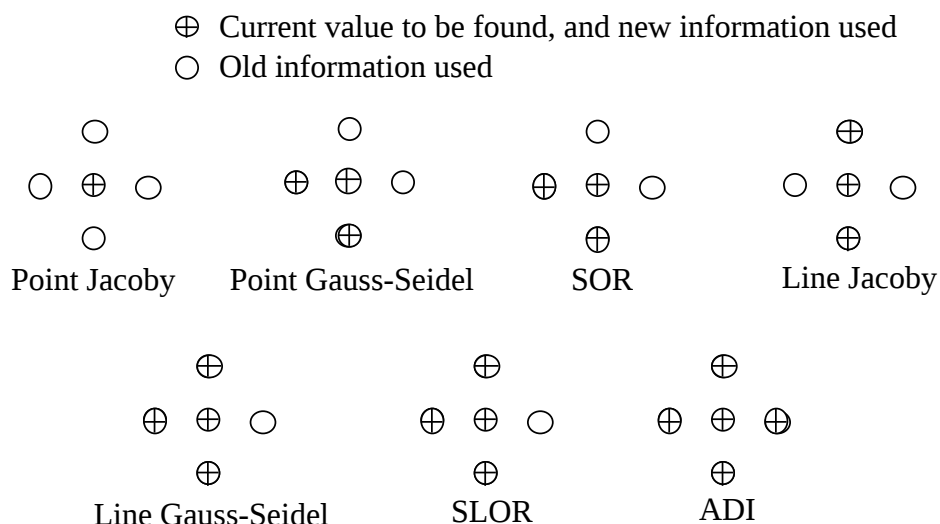


Figure 8-23. ADI Scheme solution approach.

Figure 8-24. Stencil of information (Holst¹⁰)

In addition to these methods, solutions can be obtained more rapidly by using so-called *multigrid* methods. These methods accelerate the convergence iterative procedures by using a sequence of grids of different densities and have become one of the most important techniques used to solve field problems of all types. The overall levels of the solution are established by the solution on a crude grid, while the details of the solution are established on a series of finer grids. Typically, one iteration is made on each successively finer grid, until the finest grid is reached. Then, one iteration is made on each successively coarser grid. This process is repeated until the solution converges. This procedure can reduce the number of fine grid iterations from possibly thousands, as shown above, to from 10 to 30 iterations.

This approach to the solution of partial differential equations was highly developed by Jameson¹⁸ for the solution of computational aerodynamics problems. He used the multigrid approach together with an alternating direction method in an extremely efficient algorithm for the two-dimensional transonic flow over an airfoil.

The details of the multigrid method are, as they say, beyond the scope of this chapter, and the reader should consult the standard literature for more details. This includes the original treatise on the subject by Brandt¹⁹ (which includes an example FORTRAN program), another tutorial which includes a FORTRAN code,²⁰ and more recent presentations by Briggs²¹ and Wesseling.²² The most recent *Numerical Recipes*¹¹ book also includes a brief description and sample program.

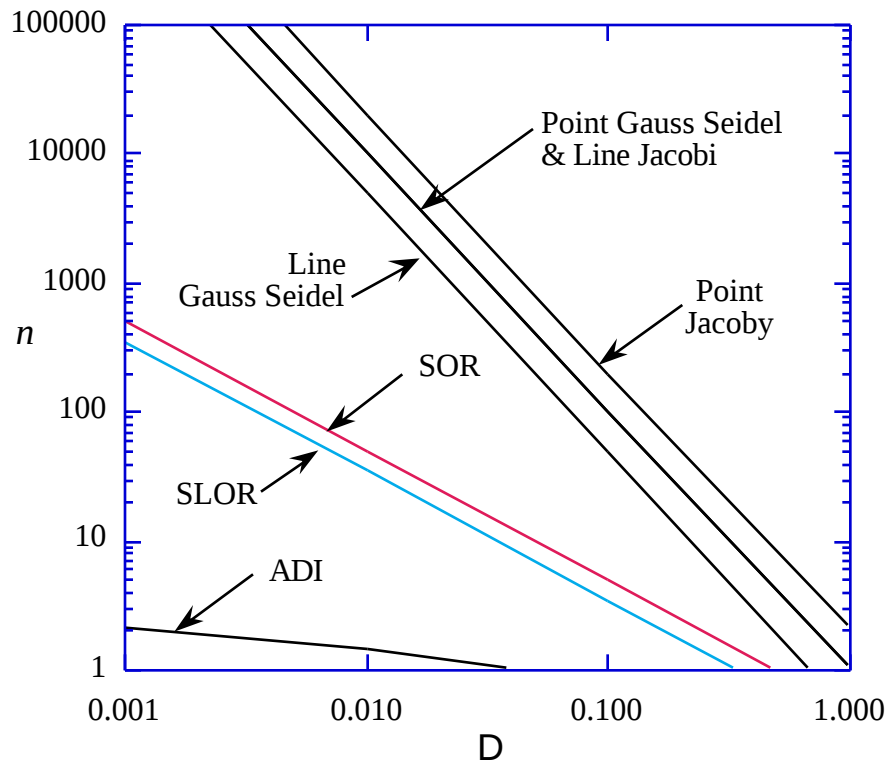


Figure 8-25. Comparison of convergence rates of various relaxation schemes (Holst¹⁰). This is the number of iterations estimated to be required to reduce the residual by one order of magnitude

Table 8-1
Convergence rate estimates for various relaxation schemes (Holst¹⁰)

Algorithm	Number of iterations required for a one order-of-magnitude reduction in error
Point - Jacobi	$2 / D^2$
Point - Gauss - Seidel	$1 / D^2$
SOR	$1 / (2D)$
Line - Jacobi	$1 / D^2$
Line - Gauss - Seidel	$1 / (2D^2)$
SLOR	$1 / (2\sqrt{2}D)$
ADI	$-\log(D/2)/1.55$

To carry out the solution to large systems of equations, the standard numerical procedures require that the approach be generalized slightly from the one given above. Specifically, we define an operator, such that the partial differential equation is written (continuing to use Laplace's equation as an example):

$$Lf = 0 \quad (8-71)$$

where

$$L = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}. \quad (8-72)$$

To solve this equation, we re-write the iteration scheme expressions given above in Equation. (8-70) as:

$$N \underbrace{C_{i,j}^n}_{\substack{n^{th} \text{ iteration} \\ \text{correction}}} + w \underbrace{Lf_{i,j}^n}_{\substack{n^{th} \text{ iteration} \\ \text{residual,} = 0 \\ \text{when converged} \\ \text{solution is achieved}}} = 0. \quad (8-63)$$

This form is known as the standard or delta form. C is given by

$$C_{i,j}^n = f_{1,j}^{n+1} - f_{1,j}^n. \quad (8-64)$$

The actual form of the N operator depends on the specific scheme chosen to solve the problem.

8.6 Stability Analysis

The analysis presented above makes this approach to solving the governing equations for flowfields appear deceptively simple. In many cases it proved impossible to obtain solutions. Frequently the reason was the choice of an inherently *unstable* numerical algorithm. In this section we present one of the classical approaches to the determination of stability criteria for use in CFD. These types of analysis provide insight into grid and stepsize requirements (the term stepsize tends to denote time steps, whereas a grid size is thought of as a spatial size). In addition, this analysis is directly applicable to a linear equation. Applications in nonlinear problems are not as fully developed.

Fourier or Von Neumann Stability Analysis

Consider the heat equation used previously,

$$\frac{\partial u}{\partial t} = a \frac{\partial^2 u}{\partial x^2} \quad (8-17)$$

and examine the stability of the explicit representation of this equation given by Eq. (8-21). Assume at $t = 0$, that an error, possibly due to finite length arithmetic, is introduced in the form:

$$\underbrace{u(x,t)}_{\substack{\text{"error" is} \\ \text{introduced}}} = y(t) \underbrace{e^{jbx}}_{\substack{\text{actually could be a series,} \\ \text{take one term here}}} \quad (8-75)$$

where:

b - a real constant

$$j = \sqrt{-1}$$

Here we restate the explicit finite difference representation,

$$\frac{u(x,t+Dt) - u(x,t)}{Dt} = a \frac{u(x+Dx,t) - 2u(x,t) + u(x-Dx,t)}{(Dx)^2} \quad (8-21)$$

Substitute Eq. (8-75) into this equation, and solve for $y(t+Dt)$. Start with

$$\frac{y(t+Dt)e^{jbx} - y(t)e^{jbx}}{Dt} = a \frac{y(t)}{(Dx)^2} \left\{ e^{jb(x+Dx)} - 2e^{jbx} + e^{jb(x-Dx)} \right\} \quad (8-76)$$

and collecting terms:

$$y(t+Dt)e^{jbx} = y(t)e^{jbx} + a \frac{Dt}{(Dx)^2} y(t)e^{jbx} \underbrace{\left\{ e^{jbDx} - 2 + e^{-jbDx} \right\}}_{\substack{-2 + e^{jbDx} + e^{-jbDx} \\ 2\cos bDx}} \quad (8-77)$$

Note that the e^{jbx} term cancels, and Eq. (8-77) can be rewritten:

$$\begin{aligned} y(t+Dt) &= y(t) \left[1 + a \frac{Dt}{(Dx)^2} (-2 + 2\cos bDx) \right] \\ &= y(t) \left[1 - 2a \frac{Dt}{(Dx)^2} \underbrace{\left(1 - \cos bDx \right)}_{\substack{\text{double angle formula} \\ = 2\sin^2 b \frac{Dx}{2}}} \right] \end{aligned} \quad (8-78)$$

which reduces to:

$$\begin{aligned} y(t+Dt) &= y(t) \left[1 + 2a \frac{Dt}{(Dx)^2} \left(1 - \cos bDx \right) \right] \\ &= y(t) \left[1 - 4a \frac{Dt}{(Dx)^2} \sin^2 b \frac{Dx}{2} \right] \end{aligned} \quad (8-79)$$

Now look at the ratio of $y(t+Dt)$ to $y(t)$, which is defined as an amplification factor G ,

$$G = \frac{y(t+Dt)}{y(t)} = \left[1 - 4a \frac{Dt}{(Dx)^2} \sin^2 b \frac{Dx}{2} \right] \quad (8-80)$$

For stability the requirement is clearly:

$$|G| < 1, \quad (8-81)$$

which means that the error introduced decays. For arbitrary b , what does this condition mean? Observe that the maximum value of the sine term is one. Thus, the condition for stability will be:

$$\left| 1 - 4a \frac{Dt}{(Dx)^2} \right| < 1 \quad (8-82)$$

and the limit will be:

$$|1 - 4a| = 1. \quad (8-83)$$

The largest $|1 - 4a|$ that can satisfy this requirement is:

$$1 - 4a = -1$$

or

$$-4a = -2. \quad (8-84)$$

and

$$a = \frac{1}{2}$$

Thus, the largest $|1 - 4a|$ for $|G| < 1$ means

$$a = \frac{Dt}{(Dx)^2} < \frac{1}{2} \quad (8-85)$$

or:

$$a \frac{Dt}{(Dx)^2} < \frac{1}{2}. \quad (8-86)$$

This sets the condition on Dt and Dx for stability of the model equation. This is a real restriction. It can be applied locally for nonlinear equations by assuming constant coefficients. An analysis of the implicit formulation, Eq. (8-23), demonstrates that the implicit formulation is unconditionally stable.

Is this restriction on Dt and Dx real? Rightmyer and Morton²³ provided a dramatic example demonstrating this criteria. Numerical experiments can quickly demonstrate how important this condition is. Figure 8-26 repeats the analysis of Rightmyer and Morton,²³ demonstrating the validity of the analysis. The initial and boundary conditions used are:

$$u(x,0) = \phi(x) \text{ (given) for } 0 \leq x \leq p$$

$$u(0,t) = 0, \quad u(p,t) = 0 \quad \text{for } t > 0$$

Figure 8-26a presents the development of the solution and shows the particular choice of initial value shape, ϕ , using a value of $\lambda < 1/2$: 5/11. Figure 8-26b-d provide the results for a value of $\lambda > 1/2$: 5/9. Theoretically, this stepsize will lead to an unstable numerical method, and the figure demonstrates that this is, in fact, the case.

Our model problem was parabolic. Another famous example considers a hyperbolic equation. This is the wave equation, where c is the wave speed:

$$\frac{\partial^2 u}{\partial t^2} - c^2 \frac{\partial^2 u}{\partial x^2} = 0. \quad (8-87)$$

This equation represents one-dimensional acoustic disturbances. The two-dimensional small disturbance equation for the potential flow can also be written in this form for supersonic flow. Recall,

$$(1 - M_\infty^2) \phi_{xx} + \phi_{yy} = 0 \quad (8-88)$$

or when the flow is supersonic:

$$\phi_{xx} - \frac{1}{(M_\infty^2 - 1)} \phi_{yy} = 0 \quad (8-89)$$

and we see here that x is the timelike variable for supersonic flow.

Performing an analysis similar to the one above, the stability requirement for Eq. (8-87) is found to result in a specific parameter for stability:

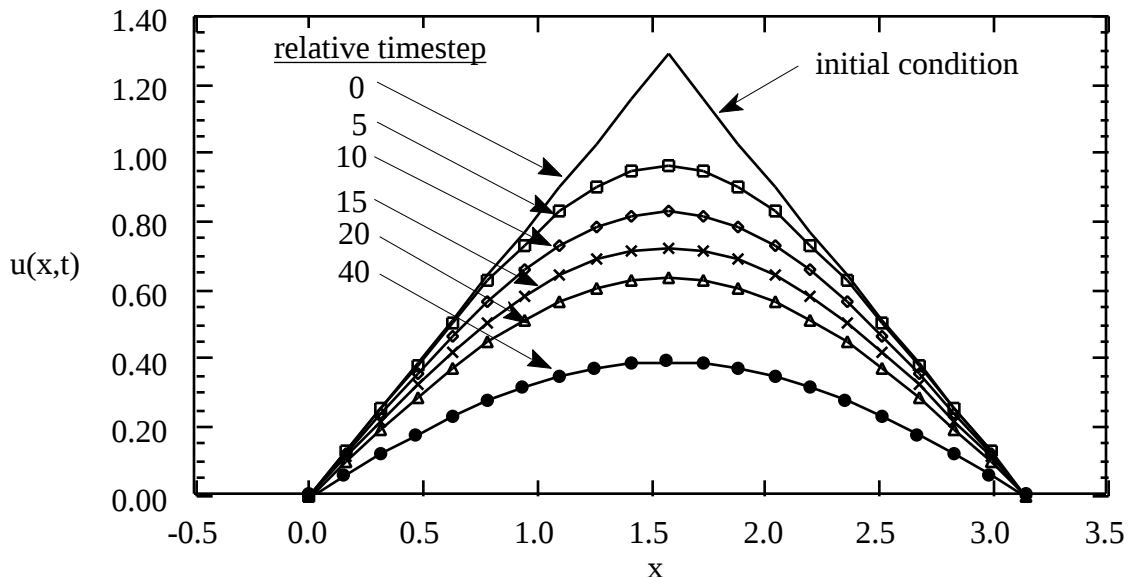
$$n = c \frac{\Delta t}{\Delta x} \quad (8-90)$$

which is known as the Courant number. For many explicit schemes for hyperbolic equations, the stability requirement is found to be

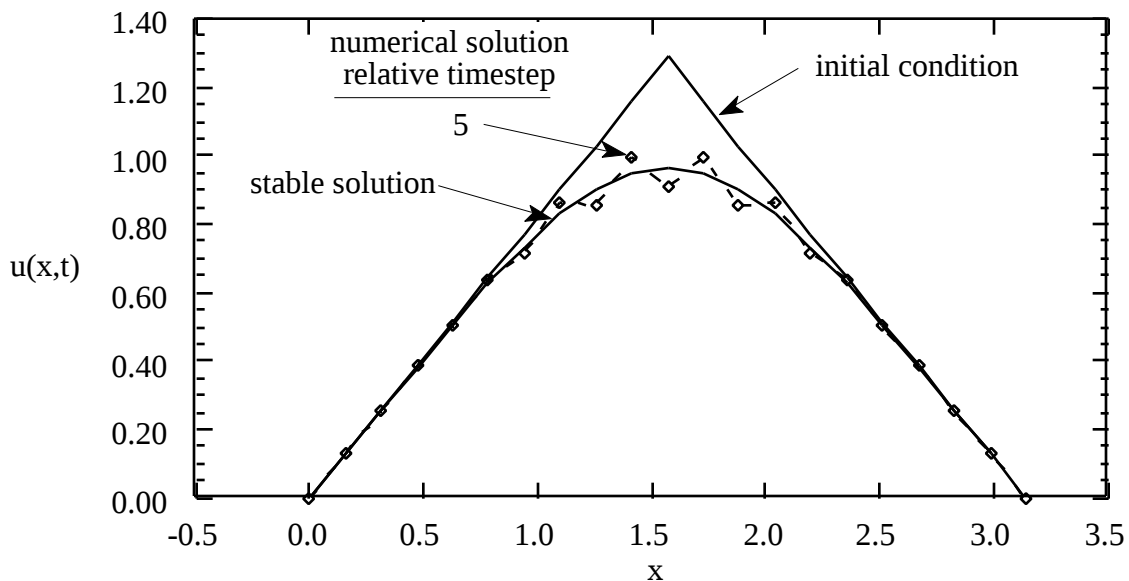
$$|n| \leq 1. \quad (8-91)$$

This requirement is known as the CFL condition, after its discoverers: Courant, Friedrichs, and Levy. It has a physical interpretation. The analytic domain of influence must lie within the numerical domain of influence.

Recalling that the evolution of the solution for an elliptic system had a definite time-like quality, a stability analysis for elliptic problems can also be carried out. For the SOR method, that analysis leads to the requirement that the over-relaxation factor, w , be less than two.

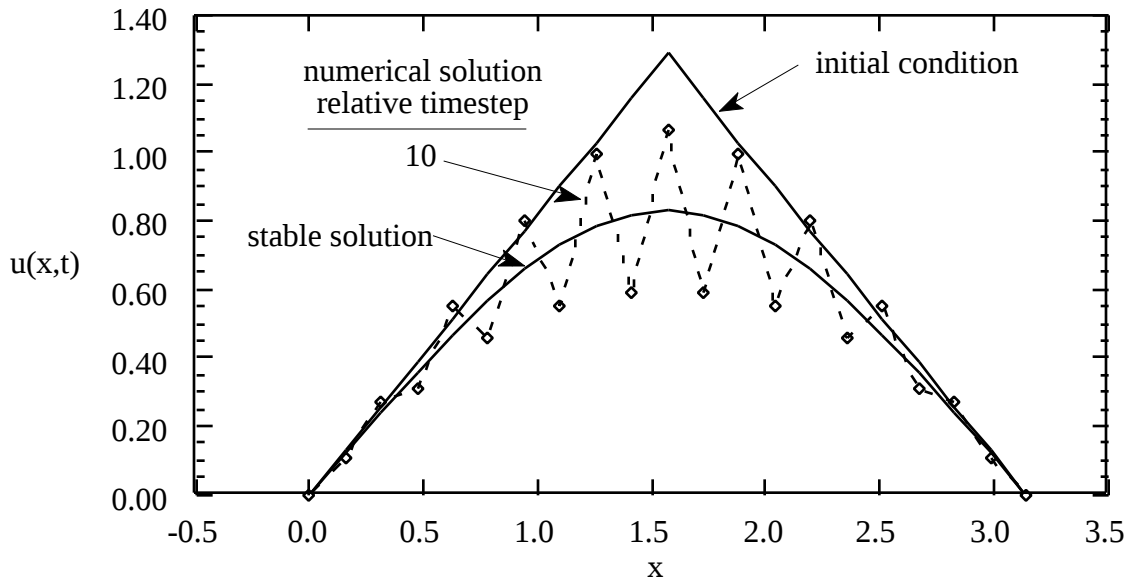


a) numerical solution using a theoretically stable stepsize.

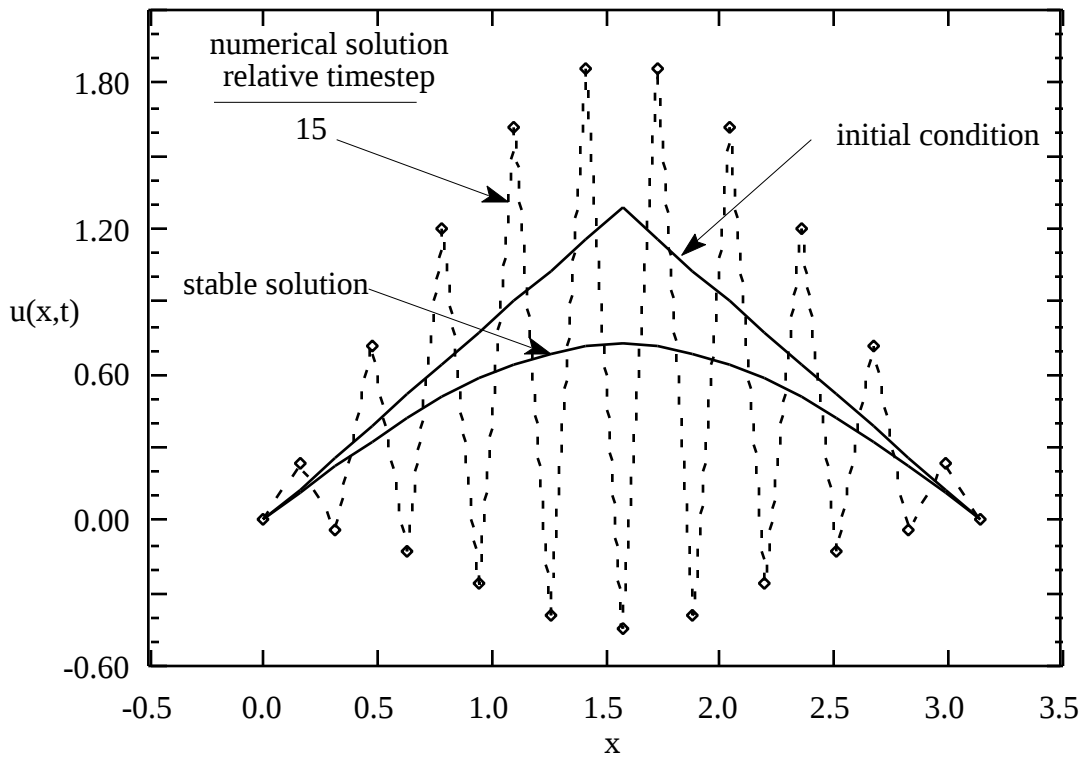


b) numerical solution using a theoretically unstable stepsize, $\frac{a t}{(Dx)^2} = 5$.

Figure 8-26. Demonstration of the step size stability criteria on numerical solutions.



c) numerical solution using a theoretically unstable stepsize, $\frac{a t}{(Dx)^2} = 10$.



d) numerical solution using a theoretically unstable stepsize, $\frac{a t}{(Dx)^2} = 15$.

Figure 8-26. Demonstration of the step size stability criteria on numerical solutions (concluded).

8.7 Program THINFOIL

An example of the solution of Laplace's Equation by finite differences is demonstrated in the program **THINFOIL**. This program offers the users options of SOR, SLOR, AF1 and AF2 to solve the system of algebraic equations for the flow over a biconvex airfoil at zero angle of attack. An unevenly spaced grid is used to concentrate grid points near the airfoil. The program and the theory are described in Appendix G-1. It can be used to study the effects of grid boundary location, number of grid points, and relaxation factor, w .

Figure 8-27 provides the convergence history for the case for which the comparison with the exact solution is given below. Using SOR, this shows that hundreds of iterations are required to reduce the maximum change between iteration approximately three orders of magnitude. This is about the minimum level of convergence required for useful results. A check against results converged further should be made. The reader should compare this with the other iteration options.

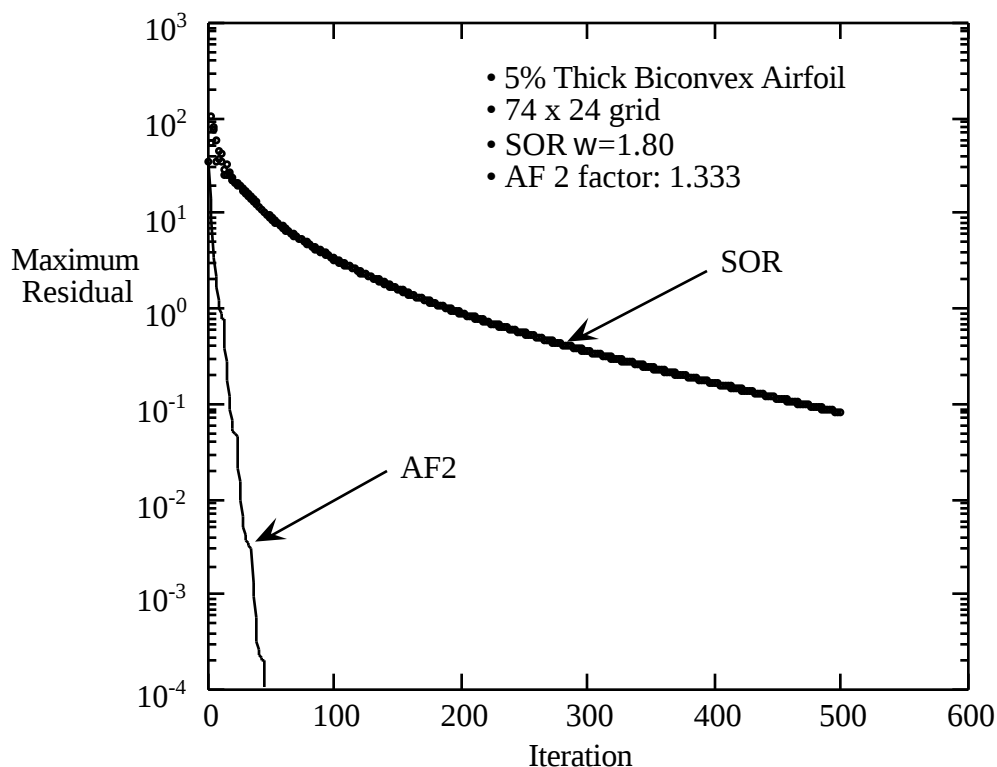


Figure 8-27. Convergence history during relaxation solution.

The convergence history presented above is actually the maximum residual of f at each iteration. The solution is assumed to have converged when the residual goes to zero. Typical

engineering practice is to consider the solution converged when the residual is reduced by 3 or 4 orders of magnitude. However, a check of the solution obtained at a conventional convergence level with a solution obtained at much smaller residual (and higher cost) level should be made before conducting an extensive analysis for a particular study.

The solution for a 5% thick biconvex airfoil obtained with **THINFOIL** is presented in Figure 8-28, together with the exact solution. For this case the agreement with the exact solution is excellent. The exact solution for a biconvex airfoil is given by Milton Van Dyke,²⁴ who cites Milne-Thompson²⁵ for the derivation.

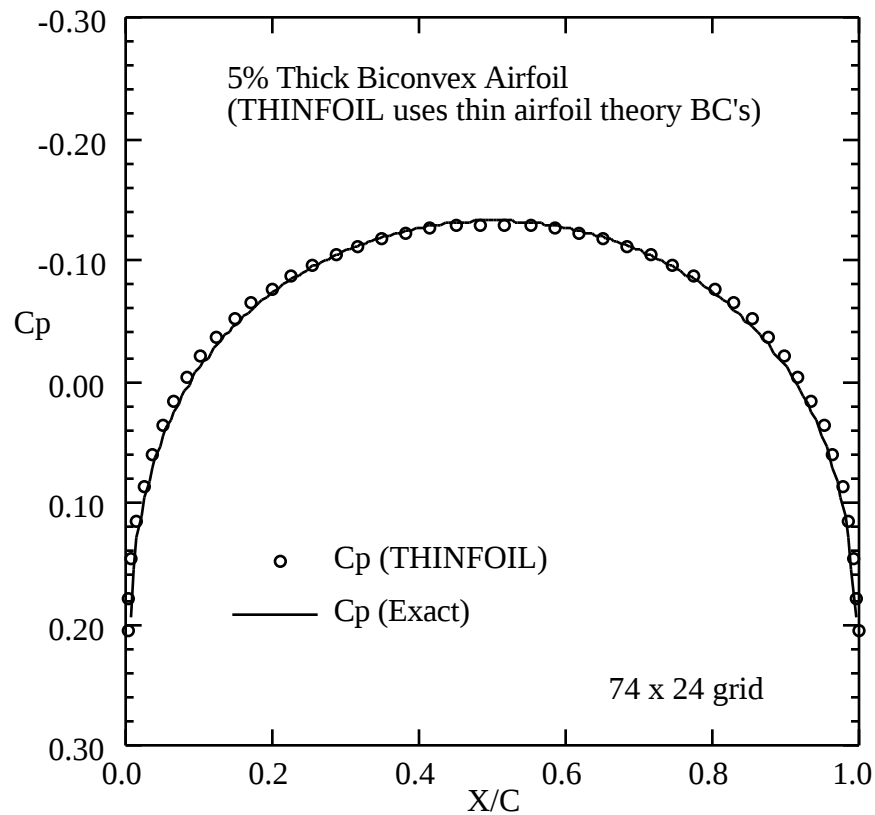


Figure 8-28. Comparison of numerical solution with analytic solution for a biconvex airfoil.

The material covered in this chapter provides a very brief introduction to an area which has been the subject of an incredible amount of research in the last thirty years. Extensions to include ways to treat flows governed by nonlinear partial differential equations are described after some basic problems in establishing geometry and grids are covered in the next chapter.

8.8 Exercises

1. How accurate are finite difference approximations? Over one cycle of a sine wave, compare first and second order accurate finite difference approximations of the 1st derivative and the second order accurate 2nd derivative of the shape with the exact values. How small does the step size have to be for the numerical results to accurate to 2 significant figure? 4 figures? 6? What conclusions about step size can you make?
2. Get some experience with the solution of Laplace's Equation using finite differences.
 - i) Download a copy of **THINFOIL** from the web page
 - ii) Make it run on your PC.
 - iii) Study the program to understand the procedure.Pick as a baseline case: Xmin=-2.2, Xmax=3.2, Ymax=2.4, and NUP=14, NDOWN=14, NON=30, NABOVE=18
 - iv) Run SOR with $w = 1.6$ and see how many iterations to "convergence"
 - v) Run with $w = 1.0, 1.50, 1.75, 1.90, 1.99$ (400 iterations max)
 - vi) Plot the convergence history as a function of iteration for each w . Note that it is standard procedure to plot the log of the residual. See examples in the text.
 - vii) For one w , increase the number of grid points and compare (watch dimensions)
 - convergence rate with the same w case above
 - the surface pressure distribution results for the two grids
 - viii) Draw conclusions about SOR as a numerical method for solving PDE's.
 - ix) Repeat the studies using SLOR, AF1 and AF2. What do you conclude about the relative convergence times and solution accuracy?
3. Examine the effect of the number of grid points on the solution obtained using program **THINFOIL**. How many grid points are required for a grid converged solution?
4. Examine the effect of the location of the farfield boundary condition on the solution obtained using program **THINFOIL**. What do you conclude?
5. Change the farfield boundary condition to set $f = 0$, instead of $\partial f / \partial n = 0$. How does this affect the solution? the convergence rate?
6. Modify program **THINFOIL** to obtain the solution to the flow over an NACA 4-Digit airfoil thickness shape. Address the following issues:
 - i) store the boundary condition values before the calculation begins instead of recomputing each time the BC needs the value
 - ii) recognizing that the slope at the leading edge is infinite, assess two methods of avoiding numerical problems
 - place the leading edge between grid points
 - use Riegels' factor to modify the slope boundary condition, replacing df/dx by

$$\frac{df/dx}{\sqrt{1 + (df/dx)^2}}$$

8.9 References

- ¹ Tannehill, J.C., Anderson, D.A., and Pletcher, R.H., *Computational Fluid Mechanics and Heat Transfer*, 2nd Ed., Taylor & Francis, New York, 1997. (on the first edition the order of the authors was Anderson, D.A., Tannehill, J.C., and Pletcher, R.H)
- ² Hoffman, K.A., and Chiang, S.T., *Computational Fluid Dynamics for Engineers*, in two volumes, Engineering Education System, PO Box 20078, Wichita, KS, 67208-1078. 1993.
- ³ Fletcher, C.A.J., *Computational Techniques for Fluid Dynamics*, Vol. 1: "Fundamental and General Techniques," Vol. II, "Specific Techniques for Different Flow Categories," Springer-Verlag, Berlin, 1988.
- ⁴ Moran, J., *An Introduction to Theoretical and Computational Aerodynamics*, John Wiley & Sons, New York, 1984.
- ⁵ Hirsch, C., *Numerical Computation of Internal and External Flows*, Vol. 1, "Fundamentals of Numerical Discretization," 1988, and Vol. 2, "Computational Methods for Inviscid and Viscous Flows," 1990, John Wiley & Sons, New York.
- ⁶ Anderson, J.D., Jr., *Computational Fluid Dynamics: The Basics with Applications*, McGraw-Hill, Inc., New York, 1995.
- ⁷ Smith, G.D., *Numerical Solution of Partial Differential Equations: Finite Difference Methods*, 3rd Ed., Clarendon Press, Oxford, 1985.
- ⁸ Ames, W.F., *Numerical Methods for Partial Differential Equations*, 2nd Ed., Academic Press, New York, 1977.
- ⁹ Versteeg, H.K., and Malalasekera, W., *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, Addison Wesley Longman, Ltd., Harlow, England, 1995.
- ¹⁰ Holst, T.L., "Numerical Computation of Transonic Flow Governed by the Full-Potential Equation," VKI Lecture Series on Computational Fluid Dynamics, Rhode-St.-Genese, March, 1983.
- ¹¹ Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, 2nd ed., Cambridge University Press, Cambridge, 1992.
- ¹² Conte, S.D., and deBoor, C., *Elementary Numerical Analysis*, McGraw-Hill, New York, 1972.
- ¹³ Mason, W.H., Mackenzie, D., Stern, M., Ballhaus, W.F., and Frick, J., "An Automated Procedure for Computing the Three-Dimensional Transonic Flow Over Wing-Body Combinations, Including Viscous Effects," Vol. II Program User's Manual and Code Description, AFFDL-TR-77-122, Feb. 1978.
- ¹⁴ Isaacson, E., and Keller, H.B., *Analysis of Numerical Methods*, John Wiley, New York, 1966.
- ¹⁵ Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., *Numerical Recipes: The Art of Scientific Computing (FORTRAN Version)*, Cambridge University Press, Cambridge, 1989.
- ¹⁶ Ballhaus, W.F., Jameson, A., and Albert, J., "Implicit Approximate Factorisation Schemes for the Efficient Solution of Steady Transonic Flow Problems," AIAA J., Vol. 16, No. 6, June 1978, pp. 573-579. (also AIAA Paper 77-634, 1977).
- ¹⁷ Catherall, D., "Optimum Approximate-Factorisation Schemes for 2D Steady Potential Flows," AIAA Paper 81-1018, 1981.
- ¹⁸ Jameson, A., "Acceleration of Transonic Potential Flow Calculations on Arbitrary Meshes by the Multiple Grid Method," AIAA Paper 79-1458, *Proceedings of the AIAA 4th Computational*

Fluid Dynamics Conference, AIAA, New York, 1979, pp. 122-146.

¹⁹ Brandt, A., "Multi-Level Adaptive Solutions to Boundary-Value Problems," *Mathematics of Computation*, Vol. 31, No. 138, April 1977, pp. 333-390.

²⁰ Stuben, K., and Trottenberg, U., "Multigrid Methods: Fundamental Algorithms, Model Problem Analysis and Applications," in *Multigrid Methods*, ed. by W. Hackbusch and U. Trottenberg, Lecture Notes in Mathematics Vol. 960, Springer-Verlag, Berlin, 1982. pp. 1-176.

²¹ Briggs, W.L., *A Multigrid Tutorial*, SIAM, Philadelphia, 1987

²² Wesseling, P., *An Introduction to Multigrid Methods*, John Wiley, Chichester, 1992.

²³ Richtmyer, R.D., and Morton, K.W., *Difference Methods for Initial-Value Problems*, 2nd Ed., Interscience, New York, 1967 pp. 4-9.

²⁴ Van Dyke, M., *Perturbation Methods in Fluid Mechanics, Annotated Edition*, The Parabolic Press, Stanford, 1975, problem 4.5, page 74.

²⁵ Milne-Thompson, *Theoretical Hydrodynamics*, 5th ed., (1968), reprinted by Dover Publications, Inc., New York, 1996, Section 6.51, pp. 177-179.

Appendix A Geometry for Aerodynamicists

Aerodynamicists control the flowfield through geometry definition, and are always interested in possible geometric shapes that would be useful in design. This appendix provides the detailed definition of many of the classic shapes frequently specified in aerodynamics. It is not encyclopedic. Section A.1.1 gives some other sources for airfoils.

A.1 Airfoil Geometry

The NACA Airfoils

The NACA airfoils were designed during the period from 1929 through 1947 under the direction of Eastman Jacobs at the NACA's Langley Field Laboratory. Most of the airfoils were based on simple geometrical descriptions of the section shape, although the 6 and 6A series were developed using theoretical analysis and don't have simple shape definitions. Although a new generation of airfoils has emerged as a result of improved understanding of airfoil performance and the ability to design new airfoils using computer methods, the NACA airfoils are still useful in many aerodynamic design applications. A number of references have been included to allow the reader to study both the older NACA literature and the new airfoil design ideas. Taken together, this literature provides a means of obtaining a rather complete understanding of the ways in which airfoils can be shaped to obtain desired performance characteristics.

The NACA airfoils are constructed by combining a thickness envelope with a camber or mean line. The equations which describe this procedure are:

$$\begin{aligned}x_u &= x - y_t(x)\sin\theta \\ y_u &= y_c(x) + y_t(x)\cos\theta\end{aligned}\tag{A-1}$$

and

$$\begin{aligned}x_l &= x + y_t(x)\sin\theta \\ y_l &= y_c(x) - y_t(x)\cos\theta\end{aligned}\tag{A-2}$$

where $y_t(x)$ is the thickness function, $y_c(x)$ is the camber line function, and

$$\theta = \tan^{-1}\left(\frac{dy_c}{dx}\right)\tag{A-3}$$

is the camber line slope. It is not unusual to neglect the camber line slope, which simplifies the equations and makes the reverse problem of extracting the thickness envelope and mean line for a given airfoil straightforward.

The primary reference volume for all the NACA subsonic airfoil studies remains:

Abbott, I.H., and von Doenhoff, A.E., *Theory of Wing Sections*, Dover, 1959.

The following paragraphs provide a brief history of the development of the NACA Airfoils. Appendix B provides references to the development of the NASA advanced airfoils, which were developed from 1966- approx. 1977.

<u>Evolution of the NACA airfoils</u>	<u>Primary NACA Report</u>	<u>Authors</u>	<u>Date</u>
1. <i>The 4-digit foils</i> : According to Abbott, Pinkerton found that the thickness distribution of the Clark Y and Gottingen 398 airfoils were similar, and Jacobs selected a function to describe this thickness distribution. The mean lines were selected to be described by two parabolic arcs which were tangent at the position of maximum camber.	R-460	Jacobs, Ward and Pinkerton	1933
2. <i>The 4-digit modified foils</i> : The camber lines were identical to the 4-digit series, and a more general thickness distribution was defined, which allowed variations in the leading edge radius and position of maximum thickness to be investigated.	R-492	Stack and von Doenhoff	1934
3. <i>The 5-digit foils</i> : The thickness distribution was kept identical to the 4-digit series, and a new camber line was defined which allowed for camber to be concentrated near the leading edge. A reflexed camber line was designed to produce zero pitching moment, but has generally not been used. These foils were derived to get good high lift with minimum C_{m0} .	R-537 R-610	Jacobs, Pinkerton and Greenberg	1935 1937
4. <i>The 6-series foils</i> : The foils were designed to maintain laminar flow over a large portion of the chord by delaying the adverse pressure gradient. The thickness envelope was obtained using exact airfoil theory, and no simple formulas are available to describe the shapes. The camber lines were designed using thin airfoil theory and simple formulas are available which describe their shape.	R-824*	Abbott, von Doenhoff and Stivers	1945
5. <i>The 6A-series foils</i> : To improve the trailing edge structurally, the foils were designed to provide sections with simple (nearly straight) surface geometry near the trailing edge, while maintaining the same general properties as the 6-series foils. The camber line can be described by a simple alteration of the standard 6-series mean line.	R-903*	Loftin	1948

Historical accounts of the NACA airfoil program are contained in:

Abbott, I.H., "Airfoils," *Evolution of Aircraft Wing Design*, AIAA Dayton Section Symposium, March 1980, AIAA Paper 80-3033.

* Additional section data is contained in NASA R-84, 1958, by Patterson and Braslow.

and

Jones, R.T., "Recollections From an Earlier Period in American Aeronautics," *Annual Review of Fluid Mechanics*, Vol. 9, pp. 1-11, 1977.

NASA has published two reports describing computer programs that produce the NACA airfoil ordinates:

Ladson, C.L., and Brooks, C.W., Jr., "Development of a Computer Program to Obtain Ordinates for the NACA 4-Digit, 4-Digit Modified, 5-Digit, and 16-Series Airfoils," NASA TM X-3284, November 1975.

Ladson, C.L., and Brooks, C.W., Jr., "Development of a Computer Program to Obtain Ordinates for the NACA 6- and 6A-Series Airfoils," NASA TM X-3069, September 1974. This program is included in the utility programs described in App. E, as LADSON. It is not extremely accurate for sections less than 6% thick or greater than 15% thick.

An extensive and excellent survey of the older airfoils is contained in the German book (available in English translation):

Riegels, *Airfoil Sections*, Butterworths, London, 1961. (English language version)

NASA supercritical airfoil development is described in the following references:

Whitcomb, "Review of NASA Supercritical Airfoils," ICAS Paper 74-10, August 1974 (ICAS stands for International Council of the Aeronautical Sciences)

Harris, C.D., "NASA Supercritical Airfoils," NASA TP 2969, March 1990.

Becker, J.V., "The High-Speed Airfoil Program," in *The High Speed Frontier*, NASA SP-445, 1980.

The NACA 4-Digit Airfoil

The numbering system for these airfoils is defined by:

NACA MPXX

where XX is the maximum thickness, t/c , in percent chord.

M is the maximum value of the mean line in hundredths of chord,

P is the chordwise position of the maximum camber in tenths of the chord.

Note that although the numbering system implies integer values, the equations can provide 4 digit foils for arbitrary values of M, P, and XX.

An example: NACA 2412

- a 12% thick airfoil,
- a max value of the camber line of 0.02, at $x/c = 0.4$.

The NACA 4-digit thickness distribution is given by:

$$\frac{y_t}{c} = \left(\frac{t}{c} \right) \left[a_0 \sqrt{x/c} - a_1 (x/c) - a_2 (x/c)^2 + a_3 (x/c)^3 - a_4 (x/c)^4 \right] \quad (\text{A-4})$$

where:

$$\begin{array}{lll} a_0 = 1.4845 & a_2 = 1.7580 & a_4 = 0.5075 \\ a_1 = 0.6300 & a_3 = 1.4215 & \end{array}$$

The maximum thickness occurs at $x/c = 0.30$, and the leading edge radius is

$$\left(\frac{r_{LE}}{c}\right) = 1.1019\left(\frac{t}{c}\right)^2 \quad (A-5)$$

The included angle of the trailing edge is:

$$\delta_{TE} = 2 \tan^{-1} \left\{ 1.16925 \left(\frac{t}{c} \right) \right\} \quad (A-6)$$

It is important to note that the airfoil has a finite thickness at the trailing edge.

The camber line is given by:

$$\left. \begin{aligned} \frac{y_c}{c} &= \frac{M}{P^2} \left[2P(x/c) - (x/c)^2 \right] \\ \frac{dy_c}{dx} &= \frac{2M}{P^2} (P - (x/c)) \end{aligned} \right\} \left(\frac{x}{c} \right) < P \quad (A-7)$$

and

$$\left. \begin{aligned} \frac{y_c}{c} &= \frac{M}{(1-P)^2} \left[1 - 2P + 2P(x/c) - (x/c)^2 \right] \\ \frac{dy_c}{dx} &= \frac{2M}{(1-P)^2} (P - (x/c)) \end{aligned} \right\} \left(\frac{x}{c} \right) \geq P \quad (A-8)$$

The camber line slope is found from (A-3) using (A-7) and (A-8), and the upper and lower surface ordinates resulting from the combination of thickness and camber are then computed using equations (A-1) and (A-2).

The NACA 5-Digit Airfoil

This airfoil is an extension of the 4 digit series which provides additional camber lines. The numbering system for these airfoils is defined by:

NACA LPQXX

where XX is the maximum thickness, t/c , in percent chord.

- L is the amount of camber; the design lift coefficient is $3/2 L$, in tenths
- P is the designator for the position of maximum camber, x_f , where $x_f = P/2$, and P is given in tenths of the chord
- Q = 0; standard 5 digit foil camber
 = 1; “reflexed” camber

An example: NACA 23012, is a 12% thick airfoil, the design lift coefficient is 0.3, the position of max camber is located at $x/c = 0.15$, and the “standard” 5 digit foil camber line is used.

The thickness distribution is the same as the NACA 4 digit airfoil thickness distribution described above in equation (A-4).

The standard five-digit series camber line is given by:

$$\left. \begin{aligned} \frac{y_c}{c} &= \frac{K_1}{6} \left[(x/c)^3 - 3m(x/c)^2 + m^2(3-m)(x/c) \right] \\ \frac{dy_c}{dx} &= \frac{K_1}{6} \left[3(x/c)^2 - 6m(x/c) + m^2(3-m) \right] \end{aligned} \right\} 0 \leq (x/c) \leq m \quad (\text{A-9})$$

and

$$\left. \begin{aligned} \frac{y_c}{c} &= \frac{K_1}{6} m^3 [1 - (x/c)] \\ \frac{dy_c}{dx} &= -\frac{K_1}{6} m^3 \end{aligned} \right\} m < (x/c) \leq 1 \quad (\text{A-10})$$

where m is not the position of maximum camber, but is related to the maximum camber position by:

$$x_f = m \left(1 - \sqrt{\frac{m}{3}} \right) \quad (\text{A-11})$$

and m is found from a simple fixed point iteration for a given x_f . K_1 is defined to avoid the leading edge singularity for a prescribed C_{li} and m :

$$K_1 = \frac{6C_{li}}{Q} \quad (\text{A-12})$$

where:

$$Q = \frac{3m - 7m^2 + 8m^3 - 4m^4}{\sqrt{m(1-m)}} - \frac{3}{2}(1-2m) \left[\frac{\pi}{2} - \sin^{-1}(1-2m) \right] \quad (\text{A-13})$$

Note that K_1 is a linear function of C_{li} and the K_1 's were originally tabulated for $C_{li} = .3$. The tabulated K_1 's are multiplied by $(C_{li}/.3)$ to get values at other C_{li} . To compute the camber line, the values of Q and K_1 must be determined. In some cases the computed values of K_1 and Q differ slightly from the official tabulated values (remember these were computed in the 1930s). The tabulated values should be used to reproduce the official ordinates. The following table illustrates the differences.

Mean Line	x_f	$\frac{m}{K_1}$		tabulated	K_1	
		tabulated	computed		using tabulated m	using computed m
210	0.05	0.0580	0.0581	361.4	351.56	350.332
220	0.10	0.1260	0.1257	51.65	51.318	51.578
230	0.15	0.2025	0.2027	15.65	15.955	15.920
240	0.20	0.2900	0.2903	6.643	6.641	6.624
250	0.25	0.3910	0.3913	3.230	3.230	3.223

Once the camberline parameters are chosen, the airfoil is constructed using the equations given above.

Camber lines designed to produce zero pitching moment.

The reflexed mean line equations were derived to produce zero pitching moment about the quarter chord.

$$\frac{y_c}{c} = \frac{K_1}{6} \left[\{(x/c) - m\}^3 - \frac{K_2}{K_1} (1-m)^3 (x/c) - m^3 (x/c) + m^3 \right] \quad 0 \leq (x/c) \leq m \quad (\text{A-14})$$

$$= \frac{K_1}{6} \left[\frac{K_2}{K_1} \{(x/c) - m\}^3 - \frac{K_2}{K_1} (1-m)^3 (x/c) - m^3 (x/c) + m^3 \right] \quad m < (x/c) \leq 1 \quad (\text{A-15})$$

where

$$\frac{K_2}{K_1} = \frac{3(m - x_f)^2 - m^3}{(1-m)^3} \quad (\text{A-16})$$

The parameters are defined as follows: i) given x_f , find m to give $C_{mc/4} = 0$ from thin airfoil theory; ii) given x_f and m , calculate K_1 to give $C_{li} = 0.3$.

The tabulated values for these camber lines are:

Mean Line	(P/2) x_f	m	K_1	K_1/K_2
211	.05	-	-	-
221	.10	0.1300	51.99	0.000764
231	.15	0.2170	15.793	0.006770
241	.20	0.3180	6.520	0.030300
251	.25	0.4410	3.191	0.135500

The NACA Modified 4-Digit Airfoil

This airfoil is an extension of the 4-digit series to allow for a variation of leading edge radius and location of maximum thickness. The numbering system is defined by:

NACA MPXX-IT

where MPXX is the standard 4-digit designation and the IT appended at the end describes the modification to the thickness distribution. They are defined as:

I - designation of the leading edge radius

T - chordwise position of maximum thickness in tenths of chord

$$\frac{\eta_e}{c} = 1.1019 \left(\frac{I}{6} \cdot \frac{t}{c} \right)^2 \quad \text{for } I \leq 8 \quad (\text{A-17})$$

and

$$\frac{\eta_e}{c} = 3 \times 1.1019 \left(\frac{t}{c} \right)^2 \quad \text{for } I = 9 \quad (\text{A-18})$$

I = 6 produces the leading edge radius of the standard 4-digit airfoils.

An example: NACA 0012-74 denotes an uncambered 12% thick airfoil, with a maximum thickness at $x/c = 0.40$ and a leading edge radius of 0.0216, which is 36% larger than the standard 4-digit value.

The NACA 16 series is a special case of the modified 4-digit airfoil with a leading edge radius index of I = 4 and the maximum thickness located at $x/c = 0.5$ (T = 5). As an example, the NACA 16-012 is equivalent to an NACA 0012-45.

The thickness distribution is given by:

$$\frac{y_t}{c} = 5 \left(\frac{t}{c} \right) \left[a_0 \sqrt{\frac{x}{c}} + a_1 \left(\frac{x}{c} \right) + a_2 \left(\frac{x}{c} \right)^2 + a_3 \left(\frac{x}{c} \right)^3 \right] \quad 0 < \frac{x}{c} < T \quad (\text{A-19})$$

and

$$\frac{y_t}{c} = 5 \left(\frac{t}{c} \right) \left[.002 + d_1 \left(1 - \frac{x}{c} \right) + d_2 \left(1 - \frac{x}{c} \right)^2 + d_3 \left(1 - \frac{x}{c} \right)^3 \right] \quad T < \frac{x}{c} \leq 1 \quad (\text{A-20})$$

The coefficients are determined by solving for the d 's first, based on the trailing edge slope and the condition of maximum thickness at $x/c = T$. Once these coefficients are found, the a 's are found by relating a_0 to the specified leading edge radius, the maximum thickness at $x/c = T$, and the condition of continuity of curvature at $x/c = T$. These constants are all determined for $t/c = 0.2$, and then scaled to other t/c values by multiplying by $5(t/c)$. The value of d_1 controls the trailing edge slope and was originally selected to avoid reversals of curvature. In addition to the tabulated values, Riegels has provided an interpolation formula.

The official (tabulated) and Riegels approximate values of d_1 are given in the following table.

$\underline{T}$	<u>Tabulated d_1</u>	<u>Approximate d_1</u>
0.2	0.200	0.200
0.3	0.234	0.234
0.4	0.315	0.314
0.5	0.465	0.464
0.6	0.700	0.722

where the Riegels approximation is given by:

$$d_1 \cong \frac{(2.24 - 5.42T + 12.3T^2)}{10(1 - 0.878T)} \quad (\text{A-21})$$

Once the value of d_1 is known, d_2 and d_3 are found from the relations given by Riegels:

$$d_2 = \frac{0.294 - 2(1 - T)d_1}{(1 - T)^2} \quad (\text{A-22})$$

and

$$d_3 = \frac{-0.196 + (1 - T)d_1}{(1 - T)^3} \quad (\text{A-23})$$

With the d 's determined, the a 's can be found. a_0 is based on the leading edge radius:

$$a_0 = 0.296904 \cdot \chi_{LE} \quad (\text{A-24})$$

where

$$\begin{aligned} \chi_{LE} &= \frac{I}{6} && \text{for } I \leq 8 \\ &= 10.3933 && \text{for } I = 9 \end{aligned} \quad (\text{A-25})$$

Defining:

$$\rho_1 = \left(\frac{1}{5}\right) \frac{(1 - T)^2}{[0.588 - 2d_1(1 - T)]} \quad (\text{A-26})$$

the rest of the a 's can be found from:

$$a_1 = \frac{0.3}{T} - \frac{15}{8} \cdot \frac{a_0}{\sqrt{T}} - \frac{T}{10\rho_1} \quad (\text{A-27})$$

$$a_2 = -\frac{0.3}{T^2} + \frac{5}{4} \cdot \frac{a_0}{T^{3/2}} + \frac{1}{5\rho_1} \quad (\text{A-28})$$

$$a_3 = \frac{0.1}{T^3} - \frac{0.375a_0}{T^{5/2}} - \frac{1}{10\rho_1 T} \quad (\text{A-29})$$

The camber lines are identical to the standard 4-digit airfoils described previously. The upper and lower ordinates are then computed using the standard equations.

The NACA 6 and 6A-Series Mean Lines*

The 6-series mean lines were designed using thin airfoil theory to produce a constant loading from the leading edge back to $x/c = a$, after which the loading decreases linearly to zero at the trailing edge. Theoretically, the loading at the leading edge must be either zero or infinite within the context of thin airfoil theory analysis. The violation of the theory by the assumed finite leading edge loading is reflected by the presence of a weak singularity in the mean line at the leading edge, where the camber line has an infinite slope. Therefore, according to Abbott and von Doenhoff, the 6-series airfoils were constructed by holding the slope of the mean line constant in front of $x/c = 0.005$, with the value at that point. For round leading edges the camberline values are essentially not used at points ahead of the origin of the leading edge radius. The theory is discussed by Abbott and von Doenhoff on pages 73-75, 113, and 120. Tabulated values are contained on pages 394-405. The derivation of this mean line is a good exercise in thin airfoil theory.

By simply adding various mean lines together, other load distributions can be constructed.

From Abbott and von Doenhoff: “The NACA 6-series wing sections are usually designated by a six-digit number together with a statement showing the type of mean line used. For example, in the designation NACA 65,3-218, $a = 0.5$, the 6 is the series designation. The 5 denotes the chordwise position of minimum pressure in tenths of the chord behind the leading edge for the basic symmetrical section at zero lift. The 3 following the comma (sometimes this is a subscript or in parenthesis) gives the range of lift coefficient in tenths above and below the design lift coefficient in which favorable pressure gradients exist on both surfaces. The 2 following the dash gives the design lift coefficient in tenths. The last two digits indicate the thickness of the wing section in percent chord. The designation $a = 0.5$ shows the type of mean line used. When the mean-line is not given, it is understood that the uniform-load mean line ($a = 1.0$) has been used.”

The 6A series airfoils employed an empirical modification of the $a = 0.8$ camberline to allow the airfoil to be constructed of nearly straight line segments near the trailing edge. This camberline is described by Loftin in NACA R-903.

Basic Camberline Equations

When $a = 1$ (uniform loading along the entire chord):

$$\frac{y}{c} = -\frac{C_{li}}{4\pi} \left[\left(1 - \frac{x}{c}\right) \ln \left(1 - \frac{x}{c}\right) + \frac{x}{c} \ln \left(\frac{x}{c}\right) \right] \quad (\text{A-30})$$

and

$$\frac{dy}{dx} = \frac{C_{li}}{4\pi} \left[\ln \left(1 - \frac{x}{c}\right) - \ln \left(\frac{x}{c}\right) \right] \quad (\text{A-31})$$

* Only the mean lines have analytical definitions. The thickness distributions are the result of numerical methods which produced tabulated coordinates. In addition to the values tabulated in the NACA reports, the closest approximation for the thickness distributions is available in program LADSON, see App. E.

where C_{li} is the “ideal” or design lift coefficient, which occurs at zero angle-of-attack. For $a < 1$,

$$\frac{y}{c} = \frac{C_{li}}{2\pi(1+a)} \left\{ \frac{1}{1-a} \left[\frac{1}{2} \left(a - \frac{x}{c} \right)^2 \ln \left| a - \frac{x}{c} \right| - \frac{1}{2} \left(1 - \frac{x}{c} \right)^2 \ln \left(1 - \frac{x}{c} \right) \right] + \frac{1}{4} \left(1 - \frac{x}{c} \right)^2 - \frac{1}{4} \left(a - \frac{x}{c} \right)^2 - \frac{x}{c} \ln \left(\frac{x}{c} \right) + g - h \frac{x}{c} \right\} \quad (A-32)$$

with

$$g = \frac{-1}{(1-a)} \left[a^2 \left(\frac{1}{2} \ln a - \frac{1}{4} \right) + \frac{1}{4} \right] \quad (A-33)$$

$$h = (1-a) \left[\frac{1}{2} \ln(1-a) - \frac{1}{4} \right] + g \quad (A-34)$$

and

$$\frac{dy}{dx} = \frac{C_{li}}{2\pi(1+a)} \left\{ \frac{1}{1-a} \left[\left(1 - \frac{x}{c} \right) \ln \left(1 - \frac{x}{c} \right) - \left(a - \frac{x}{c} \right) \ln \left(a - \frac{x}{c} \right) \right] - \ln \left(\frac{x}{c} \right) - 1 - h \right\} \quad (A-35)$$

The associated angle-of-attack is:

$$\alpha_i = \frac{C_{li} h}{2\pi(1+a)} \quad (A-36)$$

$a = .8$ (modified), the 6A-series mean line

For $0 < x/c < .87437$, use the basic $a = .8$ camberline, but with a modified value of the ideal lift coefficient, $C_{limod} = C_{li}/1.0209$. For $.87437 < x/c < 1$, use the linear equation:

$$\frac{y_{c/c}}{C_{li}} = 0.0302164 - 0.245209 \left(\frac{x}{c} - 0.87437 \right) \quad (A-37)$$

and

$$\frac{dy}{dx} = -0.245209 C_{li} \quad (A-38)$$

Note that at $x/c = 1$, the foregoing approximate relation gives $y/c = -0.000589$, indicating an α shift of $.034^\circ$ for $C_{li} = 1$.

Other airfoil definition procedures

Interest in defining airfoils by a small number of parameters for use in numerical optimization has led to several recent proposed parametric representations that might be useful. In particular, the work by August and co-workers at McDonnell Douglas in St. Louis, MO, uses Chebyshev functions to obtain functions which can represent very general airfoil shapes with from 5 to 20 coefficients required. This work is described in AIAA Papers 93-0099 and 93-0100, "An Efficient Approach to Optimal Aerodynamic Design," Parts 1 and 2.

Another approach using Bezier methods frequently used in CAD surface representation software has been used by Ventkataraman. This approach uses 14 design variables to represent the airfoil, and is described in AIAA Paper 95-1875, "A New Procedure for Airfoil Definition," and AIAA Paper 95-1876, "Optimum Airfoil Design in Viscous Flows." Smith and co-workers at NASA Langley have used a similar approach based on non-uniform rational B-splines (NURBS). A description of their approach appears in AIAA Paper 93-0195, "Grid and Design Variables Sensitivity Analysis for NACA Four-Digit Wing-Sections."

A.1.1. Tabulated Airfoil Definition and the Airfoil Library

Most modern airfoils are not described by equations, but are defined by a table of coordinates. Frequently, these coordinates are the results of a computational aerodynamic design program, and simple algebraic formulas can not be used to define the shape (this was the case with the NACA 6-series airfoils described above). The following table provides a list of the tabulated airfoils currently available on the class disk.. The subsequent tables provide a guide to these airfoils. A standard 2F10 format (the Jameson input format) is used with each set of coordinates, in the form used as input in PANELv2. See App. D.2 for an exact description.

Airfoil Library Disk Files:

	<u>file name</u>	<u>comments</u>
<i>NACA 4 digit airfoils</i>		
NACA 0010	N0010.DAT	
NACA 0010-35	N001035.DAT	(Abbott & VonDoenhoff)
NACA 0012	N0012.DAT	
NACA 4412	N44122.DAT	
<i>NACA 6 & 6A airfoils</i>		
NACA 63(2)-215	N632215.DAT	NASA TM 78503
NACA 63(2)-215 mod B	N632215m.DAT	
NACA 64A010	N64A010.DAT	
NACA 64A410	N64A410.DAT	
NACA 64(3)-418	N643418.DAT	
NACA 65(1)-012	N651012.DAT	
NACA 65(1)-213	N651213.DAT	
NACA 65(1)A012	N65A012.DAT	
	N658299M.DAT	
	N658299R.DAT	
NACA 65(2)-215	N652215.DAT	
NACA 66(3)-018	N663018.DAT	
<i>NASA General Aviation Series</i>		
LS(1)-0417	GAW1.DAT	originally known as: GA(W)-1
LS(1)-0417 mod	LS10417M.DAT	
LS(1)-0413	GAW2.DAT	originally known as: GA(W)-2
LS(1)-0013	LS10013.DAT	
<i>NASA Medium Speed Series</i>		
MS(1)-0313	MS10313.DAT	
MS(1)-0317	MS10317.DAT	
<i>NASA Laminar Flow Series</i>		
NLF(1)-1215F	NL11215F.DAT	
NLF(1)-0414F	NL10414F.DAT	
NLF(1)-0416	NL10416.DAT	
NLF(1)-0414Fmod	NL0414FD.DAT	drooped le
NLF(2)-0415	NL20415.DAT	
HSNLF(1)-0213	HSN0213.DAT	
HSNLF(1)-0213mod	HSN0213D.DAT	drooped le

NASA Supercritical Airfoils

SC(2)-0402	SC20402.DAT	
SC(2)-0403	SC20403.DAT	
SC(2)-0503	SC20503.DAT	
SC(2)-0404	SC20404.DAT	
SC(2)-0406	SC20406.DAT	
SC(2)-0606	SC20606.DAT	
SC(2)-0706	SC20706.DAT	
SC(2)-1006	SC21006.DAT	
SC(2)-0010	SC20010.DAT	
SC(2)-0410	SC20410.DAT	
SC(2)-0610	SC20610.DAT	
SC(2)-0710	SC20710.DAT	also known as Foil 33
SC(2)-1010	SC21010.DAT	
SC(2)-0012	SC20012.DAT	
SC(2)-0412	SC20412.DAT	
SC(2)-0612	SC20612.DAT	
SC(2)-0712	SC20712.DAT	
SC(3)-0712(B)	SC20712B.DAT	
SC(2)-0414	SC20414.DAT	
SC(2)-0614	SC20614.DAT	
SC(2)-0714	SC20714.DAT	Raymer, Ref. NASA TP 2890
SC(2)-0518	SC20518.DAT	
FOIL31	FOIL31.DAT	
SUPER11	SUPER11.dat	11% thick, from ICAS paper
SUPER14	SUPER14.dat	14% thick, NASA TM X-72712

NYU Airfoils

82-06-09	K820609.DAT	
79-03-12	K790312.DAT	
72-06-16	K720616.DAT	
71-08-14	K710814.DAT	
70-10-13	K701013.DAT	
65-14-08	K651408.DAT	
65-15-10	K651510.DAT	
75-06-12	KORN.DAT	the "Korn" Airfoil
75-07-15	K750715.DAT	

Miscellaneous Transonic Airfoils

CAST 7	CAST7.DAT	
DSMA 523	DSMA523.DAT	from AIAA Papre 75-880
NLR HT 731081	NLRHT73.DAT	from AGARD AR-138
ONERA M6	ONERAM6.DAT	
RAE 2822	RAE2822.DAT	
WILBY A	WILBYA.DAT	
WILBY B	WILBYB.DAT	
WILBY C	WILBYC.DAT	
WILBY R	WILBYR.DAT	
SUPER10	NASA10SC.DAT	AGARD AR-138
	MBB-A3.DAT	AGARD AR-138

Eppler Airfoils

EPPLER 662	EPP662.DAT	Raymer's book, ref NASA CP 2085
EPPLER 746	EPP746.DAT	Raymer's book, ref NASA CP 2085

Wortman Airfoils

FX-63-137-ESM	FX63137.DAT
FX-72-MS-150A	FX72M15A.DAT
FX-72-MS-150B	FX72M15B.DAT

Miscellaneous Foils

ClarkY	CLARKY.DAT	
Early Liebeck High Lift	RHLHILFT.DAT	
NLR-1	NLR1.DAT	Rotorcraft airfoil (NASA CP 2046, Vol. II)
RAE 100	RAE100.DAT	
RAE 101	RAE101.DAT	
RAE 102	RAE102.DAT	
RAE 103	RAE103.DAT	
RAE 104	RAE104.DAT	

VariEze Airfoils

VariEze wing bl23	VEZBL32.DAT
VariEze winglet root	VEZWLTR.DAT
VariEze winglet tip	VEZWLT.T.DAT
VariEze canard	VEZCAN.DAT

Human powered aircraft airfoils

DAE 11	DAE11.DAT	Daedalus airfoils (Mark Drela)
DAE 21	DAE21.DAT	
DAE 31	DAE31.DAT	
DAE 51	DAE51.DAT	(propeller foil?)
Lissaman 7769	LISS769.DAT	Gossamer Condor airfoil

Other airfoils are available on the world wide web, check App. F for sources. In particular, the Applied Aerodynamics group at the University of Illinois, under the direction of Prof. Michael Selig has established a massive online source for airfoil definitions and includes data from wind tunnel tests on the airfoils. Their focus is directed toward airfoils designed for low speeds and low Reynolds numbers. Finally, Richard Eppler has published an entire book of his airfoils, *Airfoil design and data*, Springer-Verlag, 1990.

The NASA low and medium speed airfoil program:

NASA Low Speed, Medium Speed, and Natural Laminar Flow Airfoil Chart							
Airfoil Designation	Design Lift	Design Thickness	Design Mach	Test?	Ordinates in Airfoil Library?	Ref.	Comment
GA(W)-1	.4/1.0	.17		✓	✓	TN D-7428	Low Speed
LS(1)-0417mod		.17			✓		
GA(W)-2		.13		✓	✓	TM X-72697	"
mod		.13		✓		TM X-74018	"
?		.21		✓		TM 78650	"
LS(1)-0013		.13		✓	✓	TM-4003	"
MS(1)-0313		.13		✓	✓	TP-1498	Medium Speed
MS(1)-0317	.30	.17	.68	✓	✓	TP-1786	"
mod		.17		✓		TP-1919	"
NLF(1)-0215F	.20 ?	.15	?		✓	Raymer's Book	Natural Laminar Flow
NLF(1)-0414F					✓		
NLF(1)-0416					✓		
NLF(1)-0414F drooped L.E.					✓		
NLF(2)-0415	.40 ?	.15 ?	?	✓	✓	Raymer's Book	"
HSNLF(1)-0213	.20 ?	.13 ?	?		✓	TM-87602	"
HSNLF(1)-0213 mod					✓		

The NASA Phase 2 supercritical airfoils are listed in the following chart.

NASA Supercritical Airfoils - Phase 2 ¹							
Airfoil Designation	Design Lift	Design Thickness	Design Mach	Test ?	Ordinates in Airfoil Library?	Ref.	Comment
SC(2)-0402	0.40	.02			✓		
SC(2)-0403	0.40	.03			✓		
SC(2)-0503	0.50	.03			✓		
SC(2)-0404	0.40	.04			✓		
SC(2)-0406	0.40	.06		✓	✓	unpubl.	
SC(2)-0606	0.60	.06			✓		
SC(2)-0706	0.70	.06	.795	✓	✓	unpubl.	
SC(2)-1006	1.00	.06		✓	✓	unpubl.	
SC(2)-0010	0.00	.10			✓		
SC(2)-0410	0.40	.10	.785		✓		
SC(2)-0610	0.60	.10	.765		✓		
SC(2)-0710	0.70	.10	.755	✓	✓	TM X-72711	Airfoil 33
SC(2)-1010	1.00	.10	.700		✓		
SC(2)-0012	0.00	.12		?	✓	TM-89102	
SC(2)-0412	0.40	.12			✓		
SC(2)-0612	0.60	.12			✓		
SC(2)-0712	0.70	.12	.735	?	✓	TM-86370	TM-86371
SC(2)-0414	0.40	.14			✓		
SC(2)-0614	0.60	.14			✓		
SC(2)-0714	0.70	.14	.715	✓	✓ (Raymers)	TM X-72712	Low Speed TM-81912
SC(2)-0518	1.00	.18			✓		
¹ Tabulated in NASA TP 2969, March 1990, by Charles D. Harris							

Several transonic airfoils were developed at New York University by a group led by Paul Garabedian. The following table provides a list of the airfoils they published.

Garabedian and Korn Airfoil Chart							
Airfoil Designation	Design Lift	Design Thickness	Design Mach	Test?	Ordinates in Airfoil Library?	Pages in Ref. Korn II Book	Comment
79-03-12	.293	.123	.790		✓	37,41-43	
72-06-16	.609	.160	.720		✓	48,52-54	
71-08-14	.799	.144	.710		✓	55,59-61	
70-10-13	.998	.127	.700		✓	62,66-68	
65-14-08	1.409	.083	.650		✓	73,77-79	
65-15-10	1.472	.104	.650		✓	80,84-86	
82-06-09	0.590	.092	.820		✓	91,95	
75-06-12	0.629	.117	.750	✓	✓	96,99-101	"The Korn"
75-07-15	0.668	.151	.750		✓	102,106	

Their airfoils are included in:

Bauer, F., Garabedian, P., and Korn, D., *A Theory of Supercritical Wing Sections with Computer Programs and Examples*, Lecture Notes in Economics and Mathematical Systems, Vol. 66, Springer-Verlag, 1972.

Bauer, F., Garabedian, P., Jameson, A. and Korn, D., *Supercritical Wing Sections II, A Handbook*, Lecture Notes in Economics and Mathematical Systems, Vol. 108, Springer-Verlag, 1975.

Bauer, F., Garabedian, P., and Korn, D., *Supercritical Wing Sections III*, Lecture Notes in Economics and Mathematical Systems, Vol. 150, Springer-Verlag, 1977.

A.2 Classic Bodies of Revolution

Bodies of revolution form the basis for a number of shapes used in aerodynamic design and are also often used in comparing computational methods. The bodies defined in this section are generally associated with supersonic aerodynamics.

a. Summary of Relations

The body radius r is given as a function of x , $r/l = f(x/l)$. Once r is known, a number of other values characterizing the shape can be determined.

The cross-sectional area and derivatives are:

$$S(x) = \pi r^2 \quad (\text{A-39})$$

$$\frac{dS}{dx} = 2\pi r \frac{dr}{dx} \quad (\text{A-40})$$

$$\frac{d^2S}{dx^2} = 2\pi \left[\left(\frac{dr}{dx} \right)^2 + r \frac{d^2r}{dx^2} \right] \quad (\text{A-41})$$

Basic integrals are:

Volume,

$$V = \int_0^l S(x) dx \quad (\text{A-42})$$

Surface area,

$$S_{wet} = 2\pi \int_0^l r(x) dx \quad (\text{A-43})$$

Length along the contour,

$$p(\bar{x}) = \int_0^l \sqrt{1 + \left(\frac{dr}{dx} \right)^2} dx \quad (\text{A-44})$$

Note that the incremental values can be found by changing the lower limit of the integrals. The local longitudinal radius of curvature is given by:

$$R(x) = \frac{\left[1 + \left(\frac{dr}{dx} \right)^2 \right]^{3/2}}{\left| \frac{d^2r}{dx^2} \right|} \quad (\text{A-45})$$

Several simple shapes are also of interest in addition to those presented in detail. They are:

Parabolic Spindle:

$$\frac{r}{l} = 4 \frac{r_{mid}}{l} \frac{x}{l} \left(1 - \frac{x}{l} \right) \quad (\text{A-46})$$

Ellipsoid of revolution:

$$\frac{r}{l} = 2 \frac{r_{mid}}{l} \sqrt{\frac{x}{l} \left(1 - \frac{x}{l}\right)} \quad (\text{A-47})$$

and the power law body:

$$\frac{r}{l} = \frac{r_0}{l} \left(\frac{x}{x_N} \right)^n \quad (\text{A-48})$$

where x_N is the nose length, and r_0 is the radius at $x = x_N$. The nose is blunt for $0 < n < 1$.

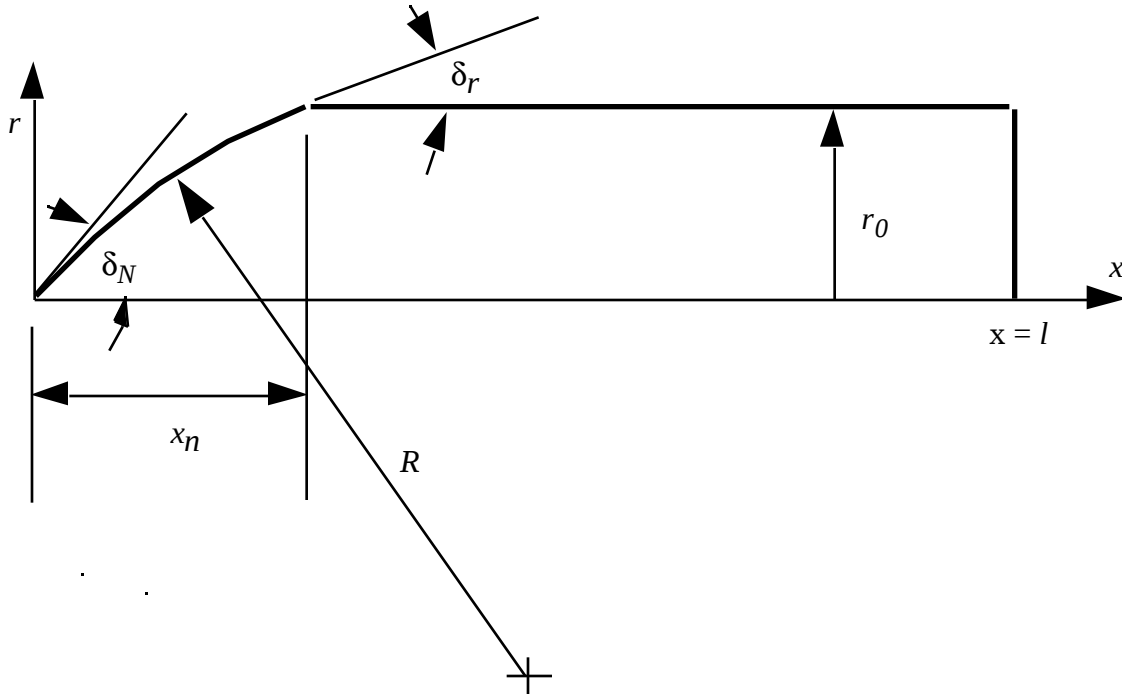
Another common shape is the spherical nose cap, and is discussed in detail in the reference by Krasnov. References that discuss geometry of bodies of revolution are:

Krasnov, N.F., *Aerodynamics of Bodies of Revolution*, edited and annotated by D.N. Morris, American Elsevier, New York, 1970.

Handbook of Supersonic Aerodynamics, Volume 3, Section 8, "Bodies of Revolution, NAVWEPS Report 1488, October 1961.

b). Tangent/Secant Ogives

The tangent or secant ogives are frequently used shapes in supersonic aerodynamics. The nomenclature is illustrated in the following sketch.



Note that the ogive is actually the arc of a circle and when $\delta_r = 0$ the ogive ends tangent to the body, so that $\delta_r = 0$ represents the tangent ogive body. If $\delta_r = \delta_N$, the cone-cylinder is recovered. If $\delta_r = 0$ and $\delta_N = 90^\circ$, the spherical cap case is obtained.

The expression for the radius r is determined using three basic constants for a particular case:

$$A = \frac{r_0}{l} \left(\frac{\cos \delta_N}{\cos \delta_r - \cos \delta_N} \right) \quad (\text{A-49})$$

$$B = 2 \frac{r_0}{l} \left(\frac{\sin \delta_N}{\cos \delta_r - \cos \delta_N} \right) \quad (\text{A-50})$$

and

$$C = \frac{r_0}{l} \quad (\text{A-51})$$

The radius is then given by:

$$\begin{aligned} \frac{r}{l} &= \sqrt{A^2 + B \left(\frac{x}{l} \right) - \left(\frac{x}{l} \right)^2} - A & 0 < \frac{x}{l} < \frac{x_N}{l} \\ &= C & \frac{x_N}{l} < \frac{x}{l} < 1 \end{aligned} \quad (\text{A-52})$$

where x_N is found as follows.

For a tangent ogive ($\delta_r = 0$), the ogive can be defined by specifying either x_N/r_0 or δ_N . The other value can then be found using:

Given δ_N ,

$$\frac{x_N}{r_0} = \frac{\sin \delta_N}{1 - \cos \delta_N} \quad (\text{A-53})$$

Or given x_N/r_0 ,

$$\delta_N = \cos^{-1} \left[\frac{\left(\frac{x_N}{r_0} \right)^2 - 1}{\left(\frac{x_N}{r_0} \right)^2 + 1} \right] \quad (\text{A-54})$$

For the secant ogive, the simplest analytical procedure is to define the ogive in terms of δ_N and δ_r , and then find x_N/l from:

$$\frac{x_N}{l} = \frac{r_0}{l} \left(\frac{\sin \delta_N - \sin \delta_r}{\cos \delta_r - \cos \delta_N} \right) \quad (\text{A-55})$$

If x_N/l is not satisfactory, δ_N and δ_r can be adjusted by trial and error to obtain the desired nose length. A program can be set up to handle this process quite simply.

The first and second derivatives are then given by:

$$\frac{d(r/l)}{d(x/l)} = \frac{B - 2(x/l)}{2[(r/l) + A]} \quad (\text{A-56})$$

and

$$\frac{d^2(r/l)}{d(x/l)^2} = -\frac{[B - 2(x/l)]^2}{4[(r/l) + A]^3} - \frac{1}{[(r/l) + A]} \quad (\text{A-57})$$

The relationships between radius and area derivatives given in section *a*) are then used to complete the calculation.

c) The von Kármán Ogive

The shape that produces minimum wave drag for a specified base area and length, according to slender body theory. This ogive has a very slightly blunted nose, and is described by Ashley and Landahl, *Aerodynamics of Wings and Bodies*, Addison-Wesley, 1965, pp. 178-181.

In this case it is convenient to work with the cross-sectional area and a new independent variable:

$$\theta = \cos^{-1} \left[2 \left(\frac{x}{x_N} \right) - 1 \right] \quad (\text{A-58})$$

or

$$\frac{x}{x_N} = \frac{1}{2} (1 + \cos \theta) \quad (\text{A-59})$$

where the nose is at $\theta = \pi$, and the base is located at $\theta = 0$.

Here we use x_N to denote the “nose length” or length of the ogive, and allow this shape to be part of an ogive-cylinder geometry.

The shape is then given as:

$$\frac{S(x)}{l^2} = \frac{S_B}{l^2} \left[1 - \frac{\theta}{\pi} + \frac{\sin 2\theta}{2\pi} \right] \quad (\text{A-60})$$

and

$$\frac{r}{l} = \sqrt{\frac{S/l^2}{\pi}} \quad (\text{A-61})$$

where S_B is the prescribed base area and l is the total length.

Defining

$$\bar{S} = \frac{S}{l^2}, \quad \bar{x} = \frac{x}{l}, \quad (\text{A-62})$$

We have

$$\frac{d\bar{S}}{d\theta} = -\frac{\bar{S}_B}{\pi} [1 - \cos 2\theta] \quad (\text{A-63})$$

$$\frac{d^2\bar{S}}{d\theta^2} = -\frac{2}{\pi} \bar{S}_B \sin 2\theta \quad (\text{A-64})$$

and

$$\begin{aligned}\frac{d\bar{S}}{d\bar{x}} &= \bar{S}' = \frac{4}{\pi} \left(\frac{l}{x_N} \right) \bar{S}_B \sin \theta \\ \frac{d^2\bar{S}}{d\bar{x}^2} &= \bar{S}'' = -\frac{8}{\pi} \left(\frac{l}{x_N} \right)^2 \frac{\bar{S}_B}{\tan \theta}\end{aligned}\tag{A-65}$$

The radius derivatives are then computed by:

$$\frac{d\bar{r}}{d\bar{x}} = \frac{\bar{S}'}{2\pi\bar{r}}, \quad \frac{d^2\bar{r}}{d\bar{x}^2} = \frac{\bar{S}''}{2\pi\bar{r}} - \frac{\bar{r}'^2}{\bar{r}}\tag{A-66}$$

d) *The Sears-Haack Body*

This is the minimum wave drag shape for a given length and volume according to slender body theory. The body is closed at both ends and has a very slightly blunted nose, and is symmetric about the mid-point. It is described by Ashley and Landahl, *Aerodynamics of Wings and Bodies*, Addison-Wesley, 1965, pp. 178-181.

Although the notation used in section c) for the von Kármán Ogive section could be used, it is more common to describe the Sears-Haack body in the manner presented below. This form uses the fineness ratio, $f = l/d_{max}$ to scale the shape. However, it is important to realize that the Sears-Haack shape is the minimum drag body for a specified volume and length, not for a specified fineness ratio. The minimum drag body for a specified fineness ratio is described below in section e) below.

Defining

$$\varsigma = 1 - 2\left(\frac{x}{l}\right),\tag{A-67}$$

the Sears-Haack body is defined as

$$\frac{r}{l} = \frac{1}{2f} \left(1 - \varsigma^2\right)^{3/4}.\tag{A-68}$$

The derivatives are given by:

$$\frac{d(r/l)}{d(x/l)} = \frac{3\varsigma}{1 - \varsigma^2} \left(\frac{r}{l}\right)\tag{A-69}$$

and

$$\frac{d^2(r/l)}{d(x/l)^2} = -\left(\frac{1}{1 - \varsigma^2}\right) \left[\varsigma \frac{d(r/l)}{d(x/l)} + 6\left(\frac{r}{l}\right) \right].\tag{A-70}$$

The fineness ratio is related to the length and volume by:

$$f = \sqrt{\frac{3\pi^2}{64} \frac{l^3}{V}}.\tag{A-71}$$

In terms of f and either V or l , the other value can be found from the following:

Given f and l :

$$V = \frac{3\pi^2}{64} \frac{l^3}{f^2}. \quad (\text{A-72})$$

Given f and V :

$$l = \left[V \frac{64}{3\pi^2} f^2 \right]^{1/3}. \quad (\text{A-73})$$

The relationships between radius and area derivatives given in Section a) are then used to complete the calculation.

e) The Haack-Adams Bodies

The Haack-Adams bodies define a number of minimum drag shapes, as described by M.C. Adams in "Determination of Shapes of Boattail Bodies of Revolution for Minimum Wave Drag," NACA TN 2550, November 1951. These bodies correspond to the following cases:

- I. Given length, base area, and contour passing through a specifically located radius.
- II. Given length, base area, and maximum area.
- III. Given length, base area, and volume.

In case I, the specified radius will not necessarily be the maximum radius.

The notation used in TN 2550 is employed in the equations, leading to the following definitions:

$$S = 4 \frac{\bar{S}(x)}{l^2}, \quad B = 4 \frac{S_{BASE}}{l^2}, \quad A = 4 \frac{S_A}{l^2}, \quad V = 8 \left(\frac{\bar{V}}{l^3} \right) \quad (\text{A-74})$$

where $S(x)$ is the area, S_A corresponds to either the specified area at a given location, or the maximum area, and V is the volume. The independent variable is defined with its origin at the body mid-point:

$$\zeta = 2 \left(\frac{x}{l} \right) - 1 \quad (\text{A-75})$$

and the location of the specified radius (Case I) and maximum radius (Case II) is designated C and given in ζ coordinates. When referred to the x coordinate, this value is designated C_x .

The equation for each case can be written in a standard form:

Case I — Given S_{BASE} , S_A , C_x :

$$\begin{aligned} \frac{\pi S}{B} = & \left[\frac{\pi A}{B} - \cos^{-1}(-c) \right] \frac{\sqrt{1-\zeta^2}(1-c\zeta)}{(1-c^2)^{3/2}} + \frac{\sqrt{1-\zeta^2}(\zeta-c)}{(1-c^2)} \\ & + \left[\frac{\pi A}{B} - \cos^{-1}(c) - c\sqrt{1-c^2} \right] \frac{(\zeta-c)^2}{(1-c)^2} \ln N + \cos^{-1}(-\zeta) \end{aligned} \quad (\text{A-76})$$

where

$$N = \frac{1 - c\zeta - \sqrt{1 - c^2} \sqrt{1 - \zeta^2}}{|\zeta - c|}. \quad (\text{A-77})$$

Case II — Given S_{BASE} , S_{MAX} :

First find the location of the maximum thickness from the implicit relation

$$f(c) = 0 = \frac{\pi A}{B} c - \sqrt{1 - c^2} - c \cos^{-1}(-c). \quad (\text{A-78})$$

Use Newton's iteration

$$c^{i+1} = c^i - \frac{f(c^i)}{f'(c^i)} \quad (\text{A-79})$$

where

$$f'(c) = \frac{\pi A}{B} - \cos^{-1}(-c). \quad (\text{A-80})$$

An initial guess of $c = 0$ is sufficient to start the iteration. Given c , the relation for the area is:

$$\frac{\pi S}{B} = \frac{\sqrt{1 - \zeta^2}}{c} + \frac{(\zeta - c)^2}{c \sqrt{1 - c^2}} \ln N + \cos^{-1}(-\zeta) \quad (\text{A-81})$$

where N is the same function as given in Case I.

Case III — Given S_{BASE} and V :

$$\frac{\pi S}{B} = \frac{8}{3} \left[\frac{V}{B} - 1 \right] \left(1 - \zeta^2 \right)^{3/2} + \zeta \sqrt{1 - \zeta^2} + \cos^{-1}(-\zeta) \quad (\text{A-82})$$

The maximum thickness for this case is located at:

$$e = \frac{1}{4(V/B - 1)} \quad (\text{A-83})$$

and in x coordinates

$$e_x = \frac{1}{2}(1 + e) \quad (\text{A-84})$$

Note that if $S_{BASE} = 0$, the Sears-Haack body is recovered.

A.3 Cross-Section Geometries for Bodies

The axisymmetric bodies described above can be used to define longitudinal lines for aerodynamic bodies. However, many aerodynamic bodies are not axisymmetric (the fuselage cross section is not round). In this section we define a class of cross section shapes that can be used to develop more realistic aerodynamic models. In particular, they have been used to study geometric shaping effects on forebody aerodynamic characteristics using an analytical forebody model with the ability to produce a wide variation of shapes. This generic model makes use of the equation of a super-ellipse to define cross sectional geometry. The super-ellipse, used previously to control flow expansion around wing leading edges, can recover a circular cross section, produce elliptical cross sections and can also produce chine-shaped cross sections. Thus it can be used to define a variety of different cross sectional shapes.

The super-ellipse equation for a cross section is:

$$\left(\frac{z}{b}\right)^{2+n} + \left(\frac{y}{a}\right)^{2+m} = 1 \quad (\text{A-85})$$

where n and m are adjustable coefficients that control the surface slopes at the top and bottom plane of symmetry and chine leading edge. The constants a and b correspond to the maximum half-breadth (the maximum width of the body) and the upper or lower centerlines respectively. Depending on the value of n and m , the equation can be made to produce all the shapes described above. The case $n = m = 0$ corresponds to the standard ellipse. The body is circular when $a = b$.

When $n = -1$ the sidewall is linear at the maximum half breadth line, forming a distinct crease line. When $n < -1$ the body cross section takes on a cusped or chine-like shape. As n increases, the cross-section starts to become rectangular.

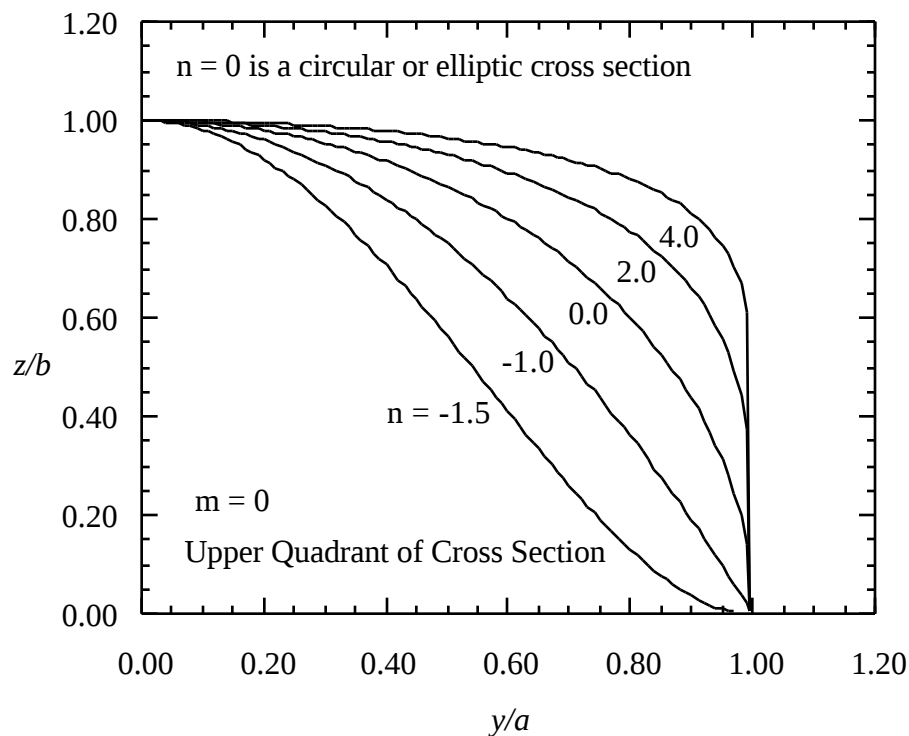
The derivative of z/b with respect to y/a is:

$$\frac{d\bar{z}}{d\bar{y}} = - \frac{\left(\frac{2+m}{2+n}\right)}{\left[1 - \bar{y}^{(2+m)}\right]^{\left(\frac{1+n}{2+n}\right)}} \quad (\text{A-86})$$

where $\bar{z} = z/b$ and $\bar{y} = y/a$. As $\bar{y} \rightarrow 1$, the slope becomes :

$$\frac{d\bar{z}}{d\bar{y}} = \begin{cases} \infty & n > -1 \\ 0 & n < -1 \\ -(2+m)\bar{y}^{1+m} & n = -1 \end{cases} \quad (\text{A-87})$$

The following sketch shows a quadrant of the cross section for various values of n ranging from a chine to a rectangle.



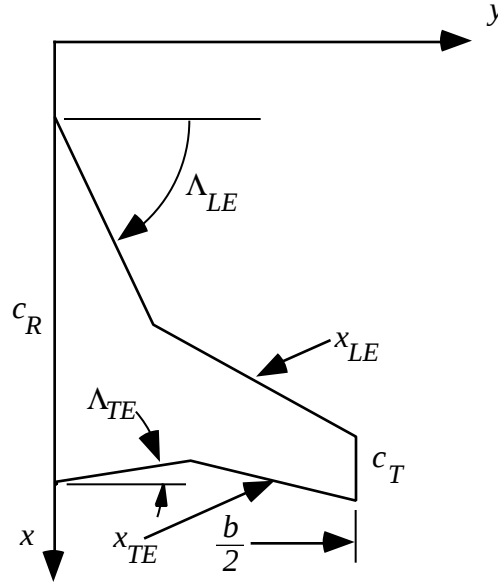
Different cross sections can be used above and below the maximum half-breadth line. Even more generality can be provided by allowing n and m to be functions of the axial distance x . The parameters a and b can also be functions of the planform shape and varied to study planform effects. Notice that when $n = -1$ the value of m can be used to control the slope of the sidewall at the crease line. Also, observe that large positive values of n drives the cross section shape to approach a rectangular or square shape.

Connecting various cross section shapes is part of the subject of lofting, described here in Chapter 9. One of the few other textbook discussions is contained in Raymer, *Aircraft Design: A Conceptual Approach*, published by the AIAA, in Chapter 7. Dan Raymer worked at North American Aviation, where Liming literally wrote the book on the analytic definition of aircraft lines.

A.4 Planform Analysis

Several local and integral planform properties are of interest in aerodynamic analysis. They are summarized in this section. (Note: Biplanes use the total area of both wings as the reference area). For a more complete presentation see DATCOM.

The local values are the leading and trailing edge locations, $x_{LE}(y)$ and $x_{TE}(y)$, the local chord, $c(y)$, and the leading and trailing edge sweep angles: $\Lambda_{LE}(y)$ and $\Lambda_{TE}(y)$. The following sketch illustrates the standard nomenclature.



The integral properties are (assuming the planform is symmetric):

1. Planform Area, S

$$S = 2 \int_0^{b/2} c(y) dy \quad (\text{A-88})$$

2. Mean aerodynamic chord, $\bar{c}$

$$\bar{c} = \frac{2}{S} \int_0^{b/2} c^2(y) dy \quad (\text{A-89})$$

3. X position of centroid of area, x_{cen}

$$x_{cen} = \frac{2}{S} \int_0^{b/2} c(y) \left\{ x_{LE}(y) + \frac{c(y)}{2} \right\} dy \quad (\text{A-90})$$

4. Spanwise position of mac

$$y_{mac} = \frac{2}{S} \int_0^{b/2} y c(y) dy \quad (A-91)$$

5. Leading edge location of mac .

$$x_{LEmac} = \frac{2}{S} \int_0^{b/2} x_{LE}(y) c(y) dy \quad (A-92)$$

In addition, the following derived quantities are often of interest:

Aspect Ratio:

$$AR = \frac{b^2}{S_{ref}} \quad (A-93)$$

Average Chord:

$$c_A = \frac{S_{ref}}{b} \quad (A-94)$$

Taper Ratio:

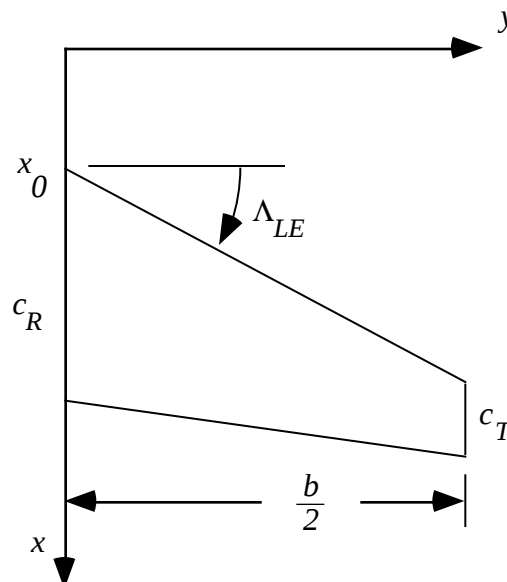
$$\lambda = \frac{c_T}{c_R} \quad (A-95)$$

S_{ref} is usually chosen to be equal to the area of a basic reference trapezoidal planform, and thus the actual planform area, S , may not equal S_{ref} .

When considering two areas, recall that the centroid of the combined surfaces is:

$$\begin{aligned} S\bar{x} &= S_1\bar{x}_1 + S_2\bar{x}_2 \\ S\bar{y} &= S_1\bar{y}_1 + S_2\bar{y}_2 \end{aligned} \quad (A-96)$$

For a standard trapezoidal wing it is convenient to collect the following formulas, where the sketch shows the nomenclature:



$$\begin{aligned} x_{LE}(y) &= x_{LE_0} + y \tan \Lambda_{LE}(y) \\ x_{TE}(y) &= x_{TE_0} + y \tan \Lambda_{TE}(y) \end{aligned} \quad (\text{A-97})$$

and the local chord is:

$$\frac{c(y)}{c_R} = 1 - (1 - \lambda)\eta \quad (\text{A-98})$$

where:

$$y = \frac{b}{2} \eta \quad \text{or} \quad \eta = \frac{y}{b/2} \quad \text{and} \quad \lambda = \frac{c_T}{c_R}. \quad (\text{A-99})$$

The sweep at any element line can be found in terms of the sweep at any other by:

$$\tan \Lambda_n = \tan \Lambda_m - \frac{4}{AR} \left[(n - m) \left(\frac{1 - \lambda}{1 + \lambda} \right) \right] \quad (\text{A-100})$$

where n, m are fractions of the local chord. An alternate formula is available using the trailing edge sweep angle:

$$\tan \Lambda_n = (1 - n) \tan \Lambda_{LE} + n \tan \Lambda_{TE} \quad (\text{A-101})$$

The integral and other relations are given by:

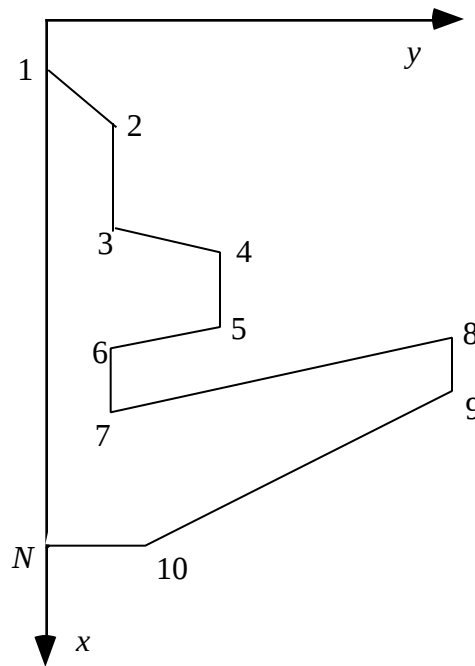
$$\begin{aligned} S &= \frac{b}{2} c_R (1 + \lambda) \\ c_{ave} &= \frac{S}{b} \\ \frac{\bar{c}}{c_R} &= \frac{2}{3} \left(\frac{1 + \lambda + \lambda^2}{1 + \lambda} \right) \\ AR &= \frac{b^2}{S} = \frac{b/2}{c_R} \left(\frac{4}{1 + \lambda} \right) \\ y_{mac} &= \frac{b}{6} \left(\frac{1 + 2\lambda}{1 + \lambda} \right) \\ \frac{x_{LE_{mac}}}{c_R} &= \frac{x_{LE_0}}{c_R} + \left(\frac{1 + 2\lambda}{12} \right) AR \tan \Lambda_{LE} \\ x_{cen} &= x_{LE_{mac}} + \frac{\bar{c}}{2} \\ AR &= \frac{b^2}{S} = \frac{b/2}{c_R} \left(\frac{4}{1 + \lambda} \right) \end{aligned} \quad (\text{A-102})$$

When computing the projected planform area of an entire configuration, the following formula is useful:

$$S = \sum_{k=1}^{k=N} (y_{k+1} + y_k)(x_{k+1} - x_k) \quad (\text{A-103})$$

where the sketch below defines the nomenclature.

At $k = N$, y_{k+1}, x_{k+1} refer to the initial points y_1, x_1 . For normal planforms, $y_{N+1} = y_1 = 0$, so that the summation can be terminated at $N-1$. This formula assumes planform symmetry and provides the total planform area with only one side of the planform used in the computation.



A.5 Conical Camber

An important class of camber distributions is associated with the planform, and not the airfoil. Conical camber has been widely used. Many forms have been used, however the NACA defined a specific type of conical camber that is known as NACA conical camber. The most recent example of NACA conical camber is the F-15 wing. It improves the drag characteristics of wings in the subsonic and transonic flow region even though it was developed to reduce the drag at supersonic speeds!

The key references are:

Hall, C.F., "Lift, Drag, and Pitching Moment of Low Aspect Ratio Wings at Subsonic and Supersonic Speeds," NACA RM A53A30, 1953.

This report provided the original mathematical definition of NACA conical camber. It also provided a large range of test conditions for which the camber was effective.

Boyd, J.W., Migotsky, E., and Wetzel, B.E., "A Study of Conical Camber for Triangular and Swept Back Wings," NASA RM A55G19, Nov. 1955.

This report provided more details of the derivation of the formulas for NACA conical camber, and corrected errors in the equations presented in the first report. Additional experimental results were also presented.

A.6 Three-Dimensional Wing Geometry

Wing geometry is often defined by interpolating between airfoil section-specified at particular spanwise stations. Some care must be taken to interpolate properly. See Chapter 9, Geometry and Grids for a discussion of wing lofting. Program WNGLFT is described in App. E, Utility Codes. This program provides an example of a lofting scheme to provide wing ordinates at any desired location. It can be used to provide wing ordinates for a wide class of wings. It in fact will produce a very good approximation to the wing design employed by a successful Navy airplane.

Appendix C Preparation of Written Material

Effective engineering requires good communication skills. Documentation and presentation of results are two important aspects of computational aerodynamics. This requires good use of both text and graphics. This appendix provides guidelines for student aerodynamicists. The first impression you make on the job is extremely important. Learn and practice good written communication. That is the way bosses “up-the-line” will see your work. You cannot do good written work without practice. This is especially true in aerodynamics, where good plots are crucial. You can’t play in the band or on the basketball team without developing skills through practice. It is even more important to a career to develop good graphics skills while you are in school.

Text: Analysis and calculations must be documented with enough detail to settle any question that arises long after the calculations are made. This includes defining the precise version of the code used, the configuration geometric description, grid details, program input and output. Often questions arise (sometimes years later) where the documentation is insufficient to figure out with certainty exactly what happened. Few of us can remember specific details even after a few months, and particularly when being grilled because something doesn’t “look right” (this is the situation when the flight test data arrives). Two personal examples from wind tunnel testing include inadequate documentation of the exact details of transition fixing and the sign convention for deflection of surfaces (at high angle-of-attack it may not be at all obvious what effect a “plus” or “minus” deflection would produce on the aerodynamic results). Good documentation is also crucial since a typical set of computations might cost many hundreds of thousands of dollars, and the results might be examined for effects that weren’t of specific interest when the initial calculations were made. An unfortunate, but frequent, occurrence in practice is that the time and budget expire before the reporting is completed. Since the report is done last, budget overruns frequently result in poor final documentation. It is best if the documentation can be put together while the computations are being conducted. Computational aerodynamics work should copy wind tunnel test procedures and maintain a test notebook. This approach can minimize the problem.

When writing a memo describing the results be accurate, neat and precise. In a page or two, outline the problem, what you did to resolve it, and your conclusion. What do the results mean? What are the implications for your organization? Provide key figures together with the description of how you arrived at your conclusion. Additional details should be included in an appendix, possibly with limited distribution. When writing your memo or report provide specifics, not generalities, *i.e.*, rather than “greater than,” say “12% greater than.” What do the results mean? When writing the analysis, **do not** simply provide tables of numbers and demand

that the reader do the interpretation. You must tell the reader exactly what you think the results mean. The conclusion to be drawn from the each figure must be precisely stated. Providing computer program output and expecting someone else (your boss or your teacher) to examine and interpret the results **is totally unacceptable**. This is the difference between an engineer and an engineering aide.

Plots and Graphs: To make good plots using the computer, you must understand how a plot is supposed to be made. Hand plotting defines the standards. When plotting by hand use real graph paper. For A size (8 1/2 x 11) plots this means K&E* Cat. No. 46 1327 for 10x10 to the half inch, and an equivalent type for 10x10 to the centimeter. There is an equivalent catalog number for B size paper.** This is Albanene tracing paper. It is the paper that was actually used in engineering work, and it's expensive. The University Bookstore will stock this graph paper until it's no longer available (I think they keep it separated, stored under my name). You should use it carefully, and not waste it. With high quality tracing paper, where the grid is readily visible on the back side, you plot on the back. This allows you to make erasures and produces a better looking plot. Orange graph paper is standard, and generally works better with copy machines, especially when you plot on the back. Before computer data bases were used, tracing paper allowed you to keep reference data on a set of plots and easily overlay other results for comparison (remembering to allow for overlay comparisons by using the same scale for your graphs).

Always draw the axis well inside the border, leaving room for labels inside the border of the paper. Labels should be well inside the page margins. In reports, figure titles go on the bottom. For overhead presentations, the figure titles go on the top. Data plots should contain at least:

- Reference area, reference chord and span as appropriate (include units).
- Moment reference center location.
- Reynolds number, Mach number, and transition information.
- Configuration identification.

If the plots are not portrait style, and must be turned to use landscape style, make sure that they are attached properly. This means placing the bottom of the figure on the right hand side of the paper. This is exactly opposite the way output for landscape plots is output from printers. However, this is the way it must be done.

Use proper scales: Use of "Bastard Scales" is grounds for bad grades in class and much, much worse on the job. This means using the "1,2, or 5 rule". It simply says that the smallest division on the axis of the plot must be easily read. Major ticks should be separated by an

* This paper is very high quality paper. With computers replacing hand plotting, this paper is being discontinued by K&E. Most art supplies stores (sometimes erroneously also claiming to be engineering supply stores) don't stock good graph paper. Cheap paper will not be transparent, preventing easy tracing from one plot to another.

** Wind tunnel data, especially drag polars, are often plotted on B size paper (11 x 17).

increment that is an even multiple of 1, 2 or 5. For example, 10, 0.2, 50 and 0.001 are all good increments between major ticks because it makes interpolation between ticks easy. Increments of 40, 25, 0.125 and 60 are poor choices of increments, and don't obey the 1,2, or 5 rule. The Boeing Scale Selection Rules chart illustrates the rule, and our version of it* is included as Fig. C-1. Label plots neatly and fully. Use good line work. In putting lines on the page, use straight edges and ship's curves to connect points, *no freehand lines*. Ship's curves and not French curves are used by aeronautical engineers when working with force and moment data. Some engineering supply catalogs call them aeronautical engineering curves. Today, pressure distributions are usually plotted directly by computer software because of the density of data. The University Bookstore stocks at least the most common size ship's curve. As a young engineer, I was told that if the wind tunnel data didn't fit the ship's curve, the data were wrong. More often than not this has indeed turned out to be the case!

Drag polars are traditionally plotted with C_D on the abscissa or X -axis, and C_L on the ordinate or Y -axis. Moment curves are frequently included with the C_L -a curve. Figure C-2 provides an example of typical force and moment data plots. The moment axis is plotted from positive to negative, also shown in the figure. This allows the engineer to rotate the graph and examine C_m - C_l in a "normal" way to see the slope. Study the scales on the plot. Also, the drag, moment, and lift results typically require the use of different scales.

The traditional way to plot data and results of calculations was to use symbols for data, and a solid line for calculated results. Recently, and very unfortunately, this style has been reversed when comparing force and moment data. Experimental data may be much more detailed than the computations, which may have been computed at only one or two angles of attack. Nevertheless, I object to using lines for data, and believe that the actual data points should be shown. When comparing pressure distributions, calculations should always be represented by lines, and the experimental data shown as symbols. Also, recall that in aeronautics C_p is plotted with the negative scale upward. Figure C-3 provides a typical example of a C_p plot. When connecting data with curves, they **must** pass through the data points. Connect complicated data with straight lines, as shown in Fig. C-4. If the data points are dense, or a theory is used to compare with data, you don't need to draw lines between points (curves that don't go through data points are assumed to be theoretical results).

More comments on proper plots and graphs are contained in the engineering graphics text, by Giesecke, *et al.* (Ref. C-1). The engineer traditionally puts his initials and date in the lower right hand corner of the plot. One problem frequently arises with plot labeling. In reports, the figure titles go on the bottom. On view graphs and slides the figure titles go on the top. Many graphics

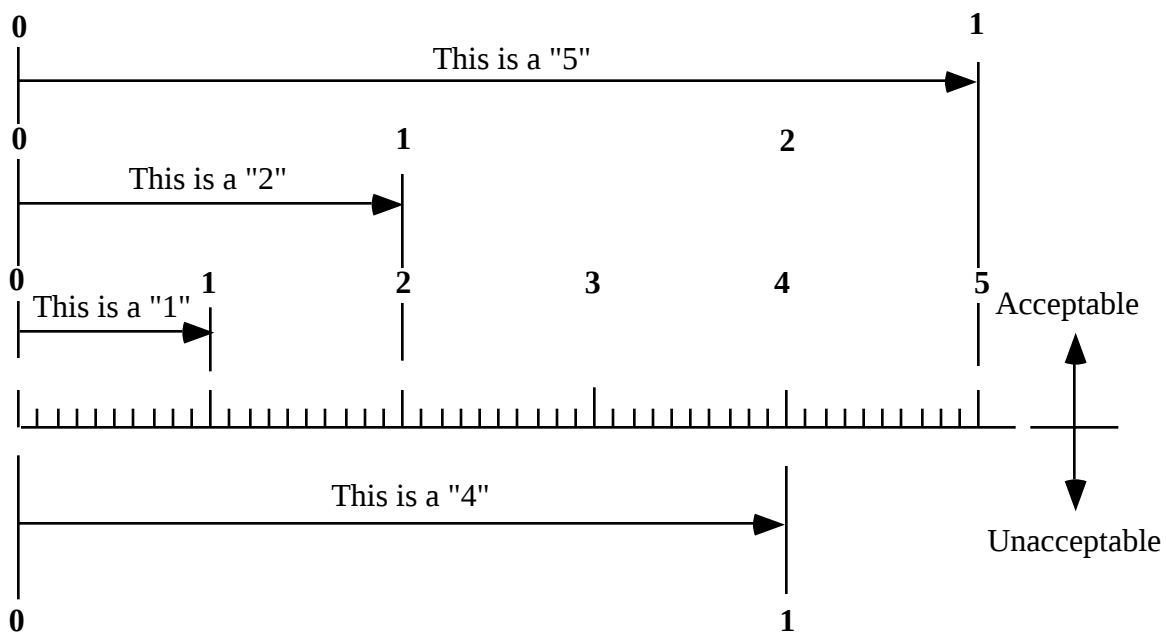
* This "improvement" was conceived by Joel Grassmeyer. It still requires some study.

packages are oriented toward placing the titles on the top. This is unacceptable in engineering reports. Finally, *tables* are labeled on the top for both reports and presentations.

Engineering plots made using your computer must be of engineering quality. To do this you have to understand the requirements given above for hand plots, and should have made enough graphs by hand to be able to identify problems in the computer generated graphs. For force and moment data it is often easier to make plots by hand than to figure out how to get your plotting package to do a good job. Typical problems include poor scale selection, poor quality printout, not being to invert the axis direction, and inability to print the experimental data as symbols and the theory as lines. Another problem that arises is the use of color. While color is important, it presents a major problem if the report is going to be copied for distribution. Most engineering reports don't make routine use of color— yet (electronic reports will make color much easier to distribute).

Reference

C-1 Giesecke, F.E., Mitchell, A., Spencer, H.C., Hill, I.L., Loving, R.O., and Dygdon, J.T., *Principles of Engineering Graphics*, Macmillan Publishing Cop., 1990, pap. 591-613.



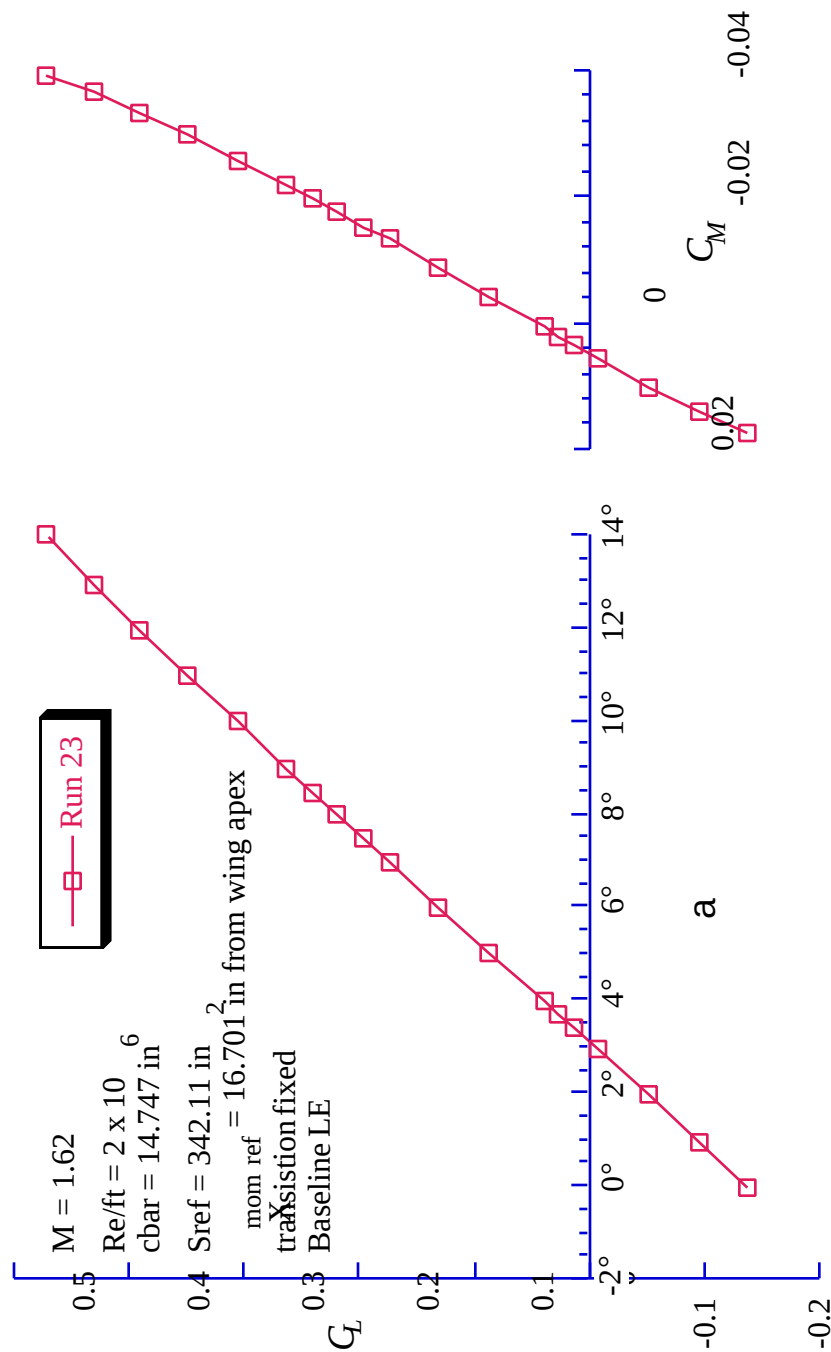
Minor subdivisions of 1, 2, or 5 allow easy interpolation, and are the only acceptable values. A minor division of 4, for example, is very difficult to use.

Scale selection rules for engineering graphs

Originally devised by H.C. Higgins, The Boeing Company, re-interpreted for these notes.

Figure C-1. Boeing scale selection chart
(based on a figure in the *AIAA Student Journal*, April, 1971)

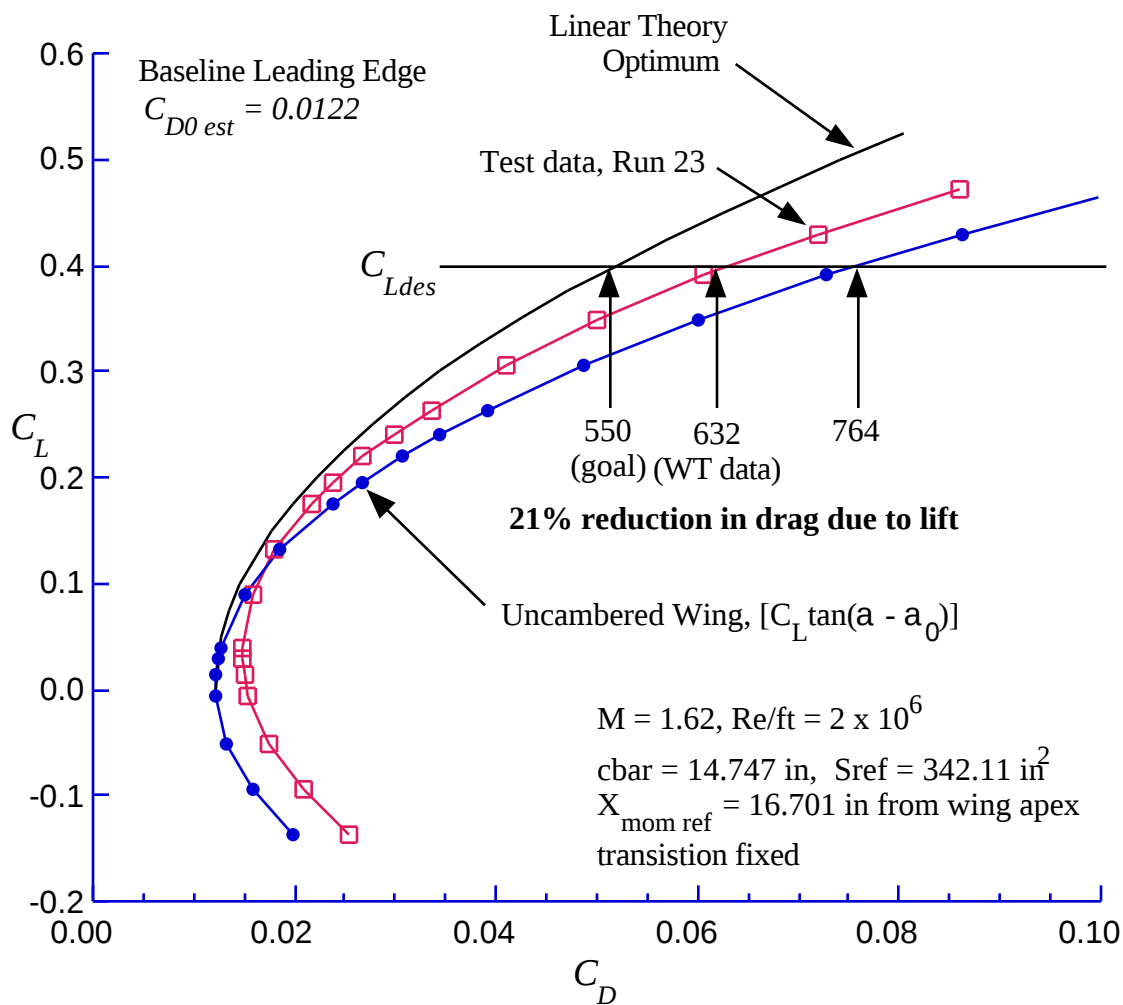
from Grumman Aero Report No. 393-82-02, April, 1982, "Experimental Pressure Distributions and Aerodynamic Characteristics of a Demonstration Wing for a Wing Concept for Supersonic Maneuvering," by W.H. Mason



a) lift and moment

Figure C-2. Examples of wind tunnel data plots.

from Grumman Aero Report No. 393-82-02, April, 1982, "Experimental Pressure Distributions and Aerodynamic Characteristics of a Demonstration Wing for a Wing Concept for Supersonic Maneuvering," by W.H. Mason



Drag Performance of a Demonstration Wing for Supersonic Maneuvering

b) drag polar

Figure C-2. Concluded.

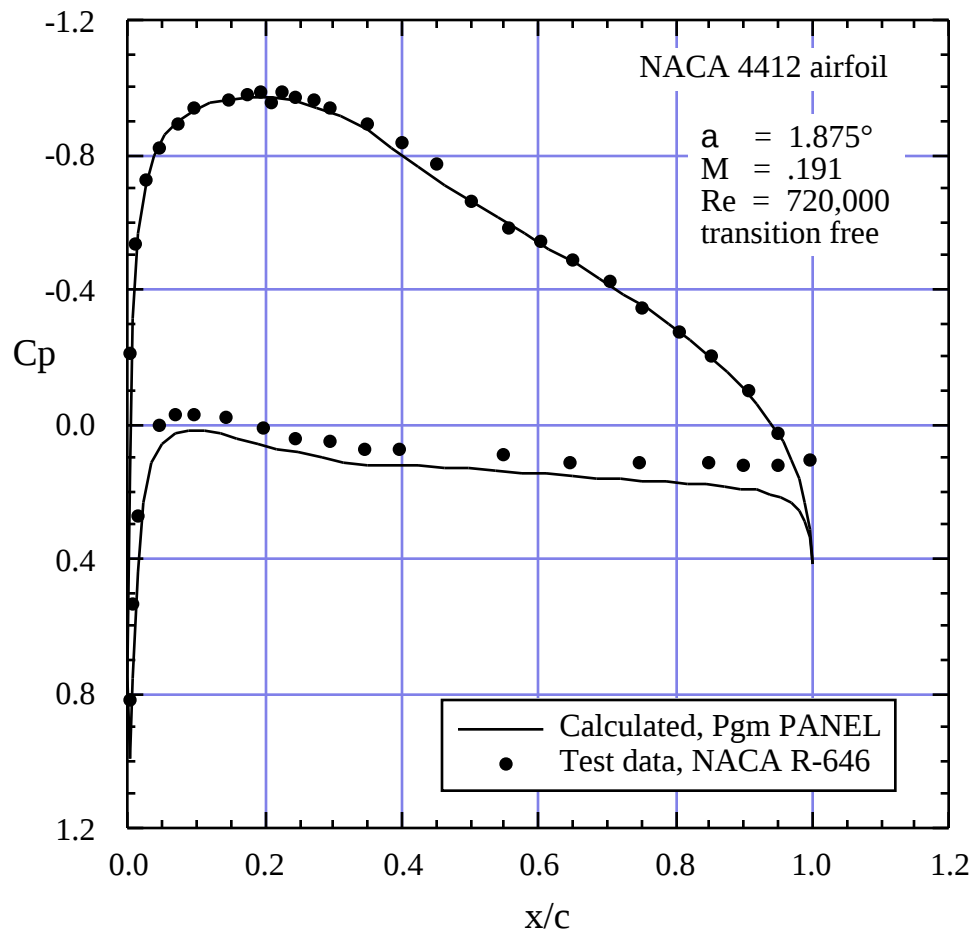


Figure C-3. Example of pressure distribution plot.

from Grumman Memo EG-ARDYN-86-051, 1986.

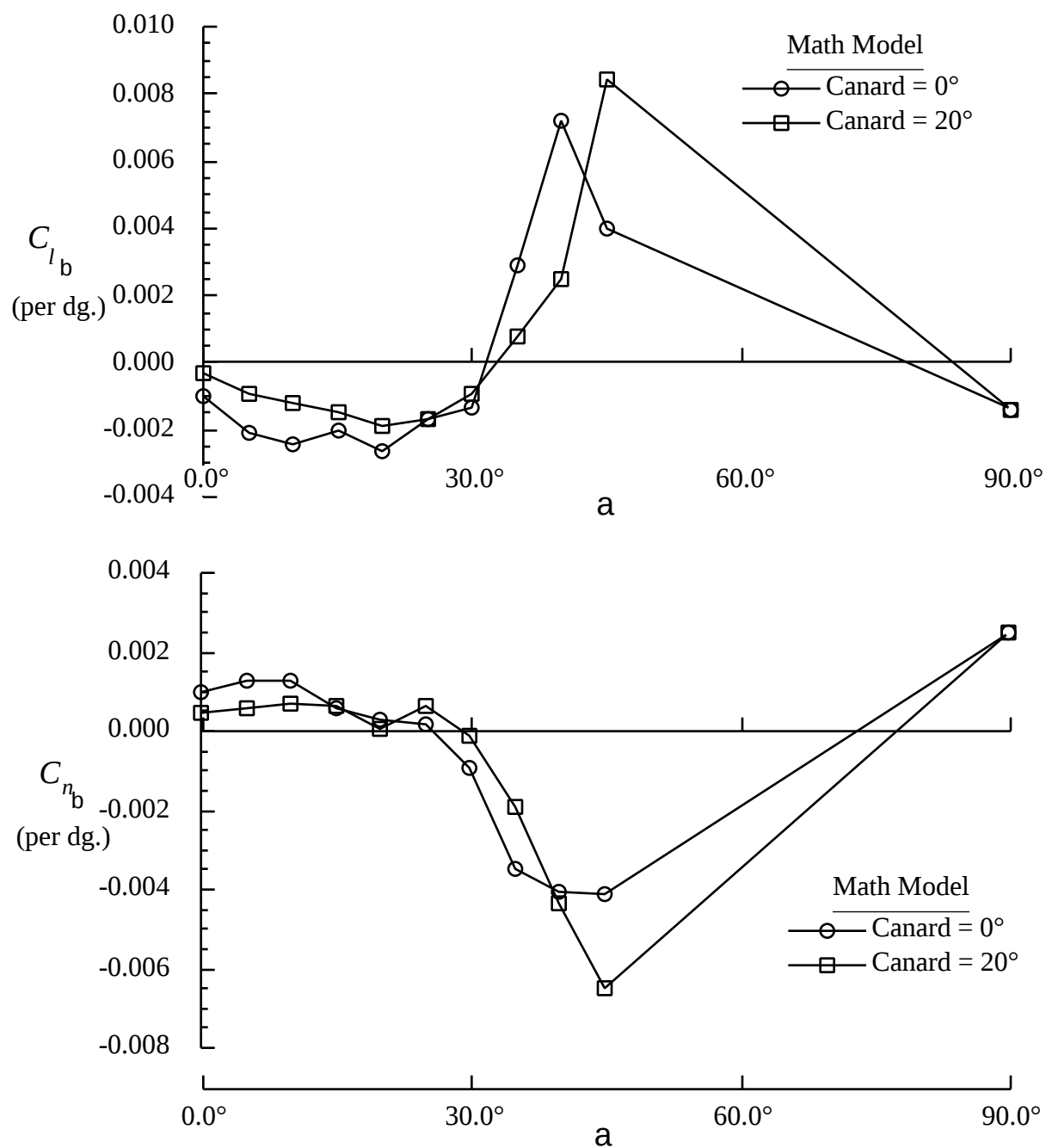


Figure C-4. Example of plotting complicated experimental data.

D.1 PANEL

This is an interactive program directly from Moran, with modifications to improve computational speed for multiple angle of attack cases. A sample output that can be used to verify that the program is working properly is given below.

```

MORAN:   PROGRAM PANEL

INPUT NLOWER, NUPPER

30, 30

INPUT NACA NUMBER

4412

      BODY SHAPE

      I          X          Y
1      1.00000    0.00000
2      0.99721   -0.00002
3      0.98887   -0.00009
4      0.97509   -0.00022
5      0.95603   -0.00041
6      0.93193   -0.00069
7      0.90307   -0.00108
8      0.86980   -0.00162
9      0.83250   -0.00233
10     0.79162   -0.00325
11     0.74760   -0.00441
12     0.70097   -0.00583
13     0.65223   -0.00751
14     0.60193   -0.00942
15     0.55061   -0.01152
16     0.49883   -0.01372
17     0.44715   -0.01592
18     0.39616   -0.01798
19     0.34711   -0.02015
20     0.29972   -0.02250
21     0.25444   -0.02479
22     0.21167   -0.02679
23     0.17183   -0.02825
24     0.13529   -0.02895
25     0.10242   -0.02869
26     0.07358   -0.02732
27     0.04909   -0.02469
28     0.02925   -0.02070
29     0.01432   -0.01529
30     0.00451   -0.00839
31     0.00000    0.00000
32     0.00096    0.00949
33     0.00753    0.01960
34     0.01969    0.03019
35     0.03736    0.04105

```

D-4 Applied Computational Aerodynamics

36	0.06039	0.05187
37	0.08856	0.06233
38	0.12157	0.07207
39	0.15904	0.08074
40	0.20054	0.08799
41	0.24556	0.09354
42	0.29354	0.09715
43	0.34387	0.09866
44	0.39593	0.09797
45	0.44833	0.09541
46	0.50117	0.09150
47	0.55392	0.08637
48	0.60599	0.08018
49	0.65679	0.07312
50	0.70577	0.06538
51	0.75240	0.05719
52	0.79617	0.04877
53	0.83663	0.04036
54	0.87334	0.03220
55	0.90594	0.02452
56	0.93409	0.01756
57	0.95751	0.01152
58	0.97597	0.00661
59	0.98928	0.00298
60	0.99731	0.00075

INPUT ALPHA IN DEGREES:

2.

PRESSURE DISTRIBUTION

I	X	Y	CP
1	0.9986	0.0000	0.38467
2	0.9930	-0.0001	0.30343
3	0.9820	-0.0002	0.25675
4	0.9656	-0.0003	0.22763
5	0.9440	-0.0006	0.20840
6	0.9175	-0.0009	0.19523
7	0.8864	-0.0014	0.18587
8	0.8512	-0.0020	0.17886
9	0.8121	-0.0028	0.17317
10	0.7696	-0.0038	0.16801
11	0.7243	-0.0051	0.16280
12	0.6766	-0.0067	0.15713
13	0.6271	-0.0085	0.15077
14	0.5763	-0.0105	0.14373
15	0.5247	-0.0126	0.13638
16	0.4730	-0.0148	0.12985
17	0.4217	-0.0170	0.12807
18	0.3716	-0.0191	0.12602
19	0.3234	-0.0213	0.11687
20	0.2771	-0.0236	0.10199
21	0.2331	-0.0258	0.08422
22	0.1917	-0.0275	0.06568
23	0.1536	-0.0286	0.04878
24	0.1189	-0.0288	0.03693
25	0.0880	-0.0280	0.03573

26	0.0613	-0.0260	0.05561
27	0.0392	-0.0227	0.11840
28	0.0218	-0.0180	0.27208
29	0.0094	-0.0118	0.60051
30	0.0023	-0.0042	0.98725
31	0.0005	0.0047	0.62389
32	0.0042	0.0145	-0.17221
33	0.0136	0.0249	-0.59380
34	0.0285	0.0356	-0.77475
35	0.0489	0.0465	-0.86631
36	0.0745	0.0571	-0.92155
37	0.1051	0.0672	-0.95698
38	0.1403	0.0764	-0.97726
39	0.1798	0.0844	-0.98317
40	0.2231	0.0908	-0.97429
41	0.2696	0.0953	-0.94986
42	0.3187	0.0979	-0.90860
43	0.3699	0.0983	-0.84568
44	0.4221	0.0967	-0.76754
45	0.4747	0.0935	-0.69391
46	0.5275	0.0889	-0.62775
47	0.5800	0.0833	-0.56419
48	0.6314	0.0766	-0.50177
49	0.6813	0.0692	-0.43979
50	0.7291	0.0613	-0.37777
51	0.7743	0.0530	-0.31529
52	0.8164	0.0446	-0.25196
53	0.8550	0.0363	-0.18742
54	0.8896	0.0284	-0.12130
55	0.9200	0.0210	-0.05316
56	0.9458	0.0145	0.01760
57	0.9667	0.0091	0.09212
58	0.9826	0.0048	0.17280
59	0.9933	0.0019	0.26569
60	0.9987	0.0004	0.38467

AT ALPHA = 2.000

CD = -0.00078 CL = 0.73347 CM = -0.28985

Another angle of attack? (Y/N):

n

STOP

D.2 PANELV2 User's Manual

This manual describes the input for program PANELV2, an extended version of program PANEL from Moran.

This program allows input of arbitrary airfoils for analysis, modification of airfoil shapes using “bumps,” and output of a file for plotting or other analysis. The program runs interactively. The input file for arbitrary airfoils is given below. (the disk with the program includes sample files, identified by ending in “.pan”)

INPUT DESCRIPTION (all numeric input is in 2F10.5 format)

<u>Card</u>	<u>Field</u>	<u>Variable</u>	<u>Description</u>
1	1	Title	Up to 80 characters describing the data set/case (A79)
2	1	FNUP	number of X,Y pairs describing upper surface
	2	FNLOW	" " " lower "
3	dummy card (used for descriptor in input data)		
4	1	X	the upper surface airfoil x/c input station
	2	Y	the y/c value of the upper surface at this x/c
***** CARD 4 is repeated FNUP times *****			
5	dummy card (used for descriptor in input data)		
6	1	X	the lower surface airfoil x/c input station
	2	Y	the y/c value of the lower surface at this x/c
***** CARD 6 is repeated FNLOW times *****			

Notes:

1. Airfoils are input from leading edge to trailing edge.
2. The leading edge point must be input twice: once for the upper surface and once for the lower surface descriptions.

OUTPUT FILE FORMAT

Card	1.	TITLE
	2.	Heading for output
	3.	4 fields: 4F10.4, this card contains
		i) angle of attack, in degrees
		ii) lift coefficient
		iii) moment coefficient (about the quarter chord)
		iv) drag coefficient from surface pressure integration (should be zero)
	4.	Number of points in
	5.	Heading for output
	6.	4 fields: 4F20.7 Note: this card is repeated for each control point
		i) x/c, airfoil ordinate
		ii) y/c, airfoil ordinate
		iii) Cp, pressure coefficient
		iii) Ue/Uinf, the surface velocity at x/c, y/c

A sample input file illustrating the format:

GAW1 - THEORETICAL ORDINATES

38. X	38. Y	(UPPER SURFACE)
0.0	0.0	
0.00200	0.01300	
0.00500	0.02035	
0.01250	0.03069	
0.02500	0.04165	
0.03750	0.04974	
0.05000	0.05600	
0.07500	0.06561	
0.10000	0.07309	
0.12500	0.07909	
0.15000	0.08413	
0.17500	0.08848	
0.20000	0.09209	
0.25000	0.09778	
0.30000	0.10169	
0.35000	0.10409	
0.40000	0.10500	
0.45000	0.10456	
0.50000	0.10269	
0.55000	0.09917	
0.57500	0.09674	
0.60000	0.09374	
0.62500	0.09013	
0.65000	0.08604	
0.67500	0.08144	
0.70000	0.07639	
0.72500	0.07096	
0.75000	0.06517	
0.77500	0.05913	
0.80000	0.05291	
0.82500	0.04644	
0.85000	0.03983	
0.87500	0.03313	
0.90000	0.02639	
0.92500	0.01965	
0.95000	0.01287	
0.97500	0.00604	
1.00000	-0.00074	
LOWER SURFACE		
0.0	0.0	
0.00200	-0.00974	
0.00500	-0.01444	
0.01250	-0.02052	
0.02500	-0.02691	
0.03750	-0.03191	
0.05000	-0.03569	
0.07500	-0.04209	
0.10000	-0.04700	
0.12500	-0.05087	
0.15000	-0.05426	
0.17500	-0.05700	
0.20000	-0.05926	
0.25000	-0.06265	
0.30000	-0.06448	
0.35000	-0.06517	
0.40000	-0.06483	
0.45000	-0.06344	
0.50000	-0.06091	
0.55000	-0.05683	

D-8 Applied Computational Aerodynamics

0.57500	-0.05396
0.60000	-0.05061
0.62500	-0.04678
0.65000	-0.04265
0.67500	-0.03830
0.70000	-0.03383
0.72500	-0.02930
0.75000	-0.02461
0.77500	-0.02030
0.80000	-0.01587
0.82500	-0.01191
0.85000	-0.00852
0.87500	-0.00565
0.90000	-0.00352
0.92500	-0.00248
0.95000	-0.00257
0.97500	-0.00396
1.00000	-0.00783

A sample output from PANELv2:

```

PROGRAM PANELv2
Revised version of Moran code
modifications by W.H. Mason

INPUT NLOWER,NUPPER
(nupper and nlower MUST be equal, and nupper + nlower MUST be less than 100)
40,40

for internally generated ordinates, enter 0
to read an external file of ordinates, enter 1
1

Enter name of file to be read: gaw1.pan

Input file name:gaw1.pan
File title:: GAW1 - THEORETICAL ORDINATES

NU = 38      NL = 38

Upper surface ordinates

index      X/C      Y/C
38      0.000000      0.000000
39      0.002000      0.013000
40      0.005000      0.020350
41      0.012500      0.030690
42      0.025000      0.041650
43      0.037500      0.049740
44      0.050000      0.056000
45      0.075000      0.065610
46      0.100000      0.073090
47      0.125000      0.079090
48      0.150000      0.084130
49      0.175000      0.088480
50      0.200000      0.092090
51      0.250000      0.097780
52      0.300000      0.101690
53      0.350000      0.104090
54      0.400000      0.105000
55      0.450000      0.104560
56      0.500000      0.102690
57      0.550000      0.099170
58      0.575000      0.096740
59      0.600000      0.093740
60      0.625000      0.090130
61      0.650000      0.086040
62      0.675000      0.081440
63      0.700000      0.076390
64      0.725000      0.070960
65      0.750000      0.065170
66      0.775000      0.059130
67      0.800000      0.052910
68      0.825000      0.046440
69      0.850000      0.039830
70      0.875000      0.033130
71      0.900000      0.026390
72      0.925000      0.019650
73      0.950000      0.012870
74      0.975000      0.006040
75      1.000000      -0.000740

```

D-10 Applied Computational Aerodynamics

Lower surface ordinates

index	X/C	Y/C
38	0.000000	0.000000
37	0.002000	-0.009740
36	0.005000	-0.014440
35	0.012500	-0.020520
34	0.025000	-0.026910
33	0.037500	-0.031910
32	0.050000	-0.035690
31	0.075000	-0.042090
30	0.100000	-0.047000
29	0.125000	-0.050870
28	0.150000	-0.054260
27	0.175000	-0.057000
26	0.200000	-0.059260
25	0.250000	-0.062650
24	0.300000	-0.064480
23	0.350000	-0.065170
22	0.400000	-0.064830
21	0.450000	-0.063440
20	0.500000	-0.060910
19	0.550000	-0.056830
18	0.575000	-0.053960
17	0.600000	-0.050610
16	0.625000	-0.046780
15	0.650000	-0.042650
14	0.675000	-0.038300
13	0.700000	-0.033830
12	0.725000	-0.029300
11	0.750000	-0.024610
10	0.775000	-0.020300
9	0.800000	-0.015870
8	0.825000	-0.011910
7	0.850000	-0.008520
6	0.875000	-0.005650
5	0.900000	-0.003520
4	0.925000	-0.002480
3	0.950000	-0.002570
2	0.975000	-0.003960
1	1.000000	-0.007830

internally generated estimate of leading edge point

X(IN)= 0.00200 Y(IN)= -0.00974 IN= 37
XC= 0.02136 YC= -0.00069
leading edge radius, RN = 0.02137

Airfoil shape after interpolation in slopy2

I	X	Y	dY/dX
1	1.00000	-0.00783	-0.20440
2	0.99846	-0.00752	-0.19879
3	0.99384	-0.00664	-0.18137
4	0.98618	-0.00537	-0.15044
5	0.97553	-0.00401	-0.10315
6	0.96194	-0.00300	-0.05058
7	0.94550	-0.00248	-0.01697
8	0.92632	-0.00246	0.01577
9	0.90451	-0.00325	0.05653
10	0.88020	-0.00513	0.09610
11	0.85355	-0.00808	0.12293
12	0.82472	-0.01195	0.14645

13	0.79389	-0.01694	0.17692
14	0.76125	-0.02265	0.17042
15	0.72700	-0.02893	0.18827
16	0.69134	-0.03538	0.17927
17	0.65451	-0.04188	0.17160
18	0.61672	-0.04809	0.15625
19	0.57822	-0.05356	0.12693
20	0.53923	-0.05789	0.09288
21	0.50000	-0.06091	0.06366
22	0.46077	-0.06300	0.04346
23	0.42178	-0.06436	0.02638
24	0.38328	-0.06506	0.01022
25	0.34549	-0.06515	-0.00540
26	0.30866	-0.06467	-0.02049
27	0.27300	-0.06366	-0.03711
28	0.23875	-0.06203	-0.05922
29	0.20611	-0.05976	-0.07942
30	0.17528	-0.05703	-0.09830
31	0.14645	-0.05381	-0.12751
32	0.11980	-0.05011	-0.14721
33	0.09549	-0.04621	-0.17860
34	0.07368	-0.04179	-0.23093
35	0.05450	-0.03693	-0.27172
36	0.03806	-0.03210	-0.34047
37	0.02447	-0.02667	-0.44665
38	0.01382	-0.02131	-0.58795
39	0.00616	-0.01569	-0.99951
40	0.00154	-0.00864	-2.62162
41	0.00000	0.00000	-51.15408
42	0.00154	0.01144	51.67010
43	0.00616	0.02240	3.61813
44	0.01382	0.03207	1.65629
45	0.02447	0.04126	1.02313
46	0.03806	0.05005	0.74407
47	0.05450	0.05796	0.55767
48	0.07368	0.06517	0.42184
49	0.09549	0.07186	0.33808
50	0.11980	0.07794	0.27773
51	0.14645	0.08346	0.22582
52	0.17528	0.08852	0.19111
53	0.20611	0.09289	0.15839
54	0.23875	0.09666	0.12792
55	0.27300	0.09978	0.10354
56	0.30866	0.10221	0.07915
57	0.34549	0.10394	0.05769
58	0.38328	0.10485	0.03569
59	0.42178	0.10497	0.01334
60	0.46077	0.10429	-0.00698
61	0.50000	0.10269	-0.02859
62	0.53923	0.10007	-0.05335
63	0.57822	0.09639	-0.08013
64	0.61672	0.09138	-0.11084
65	0.65451	0.08525	-0.14833
66	0.69134	0.07818	-0.17759
67	0.72700	0.07051	-0.20459
68	0.76125	0.06248	-0.22600
69	0.79389	0.05445	-0.24141
70	0.82472	0.04651	-0.25101
71	0.85355	0.03888	-0.26228
72	0.88020	0.03173	-0.26695
73	0.90451	0.02517	-0.26948
74	0.92632	0.01929	-0.26948
75	0.94550	0.01409	-0.27014
76	0.96194	0.00961	-0.27205

D-12 Applied Computational Aerodynamics

```
77      0.97553   0.00590  -0.27349
78      0.98618   0.00300  -0.27251
79      0.99384   0.00092  -0.27123
80      0.99846  -0.00032  -0.27057
81      1.00000  -0.00074  -0.27028

do you want to modify this airfoil? (Y/N):
y

do you want to add a bump to this airfoil? (Y/N):
y

upper (1) or lower(0) surface?
1

input begining, middle and end of bump
.05,.5,.9

input size of bump:

+ adds to thickness
- subtracts from thickness
.03

Airfoil modification

I          X/C          Y/C baseline   delta Y/C          Y/C
41          0.00000        0.00000        0.00000        0.00000
42          0.00154        0.01144        0.00000        0.01144
43          0.00616        0.02240        0.00000        0.02240
44          0.01382        0.03207        0.00000        0.03207
45          0.02447        0.04126        0.00000        0.04126
46          0.03806        0.05005        0.00000        0.05005
47          0.05450        0.05796        0.00000        0.05796
48          0.07368        0.06517        0.00003        0.06520
49          0.09549        0.07186        0.00021        0.07208
50          0.11980        0.07794        0.00070        0.07864
51          0.14645        0.08346        0.00168        0.08514
52          0.17528        0.08852        0.00330        0.09183
53          0.20611        0.09289        0.00566        0.09854
54          0.23875        0.09666        0.00874        0.10540
55          0.27300        0.09978        0.01243        0.11221
56          0.30866        0.10221        0.01649        0.11870
57          0.34549        0.10394        0.02059        0.12453
58          0.38328        0.10485        0.02434        0.12920
59          0.42178        0.10497        0.02736        0.13234
60          0.46077        0.10429        0.02932        0.13361
61          0.50000        0.10269        0.03000        0.13269
62          0.53923        0.10007        0.02914        0.12922
63          0.57822        0.09639        0.02669        0.12308
64          0.61672        0.09138        0.02297        0.11435
65          0.65451        0.08525        0.01848        0.10372
66          0.69134        0.07818        0.01376        0.09194
67          0.72700        0.07051        0.00935        0.07986
68          0.76125        0.06248        0.00566        0.06813
69          0.79389        0.05445        0.00292        0.05737
70          0.82472        0.04651        0.00119        0.04770
71          0.85355        0.03888        0.00031        0.03920
72          0.88020        0.03173        0.00003        0.03176
73          0.90451        0.02517        0.00000        0.02517
74          0.92632        0.01929        0.00000        0.01929
75          0.94550        0.01409        0.00000        0.01409
76          0.96194        0.00961        0.00000        0.00961
77          0.97553        0.00590        0.00000        0.00590
```

78	0.98618	0.00300	0.00000	0.00300
79	0.99384	0.00092	0.00000	0.00092
80	0.99846	-0.00032	0.00000	-0.00032

do you want to deflect the trailing edge? (Y/N):
y

What is the x/c of the start of the deflection?
.8

what is the deflection, in degrees?
15.

Lower Surface deflected

i	x(i)	y-old	delta y	y-new
1	1.00000	-0.00783	0.05359	-0.06142
2	0.99846	-0.00752	0.05318	-0.06070
3	0.99384	-0.00664	0.05194	-0.05858
4	0.98618	-0.00537	0.04989	-0.05526
5	0.97553	-0.00401	0.04703	-0.05105
6	0.96194	-0.00300	0.04339	-0.04639
7	0.94550	-0.00248	0.03899	-0.04147
8	0.92632	-0.00246	0.03385	-0.03630
9	0.90451	-0.00325	0.02800	-0.03125
10	0.88020	-0.00513	0.02149	-0.02662
11	0.85355	-0.00808	0.01435	-0.02243
12	0.82472	-0.01195	0.00662	-0.01858

Upper Surface deflected

i	x(i)	y-old	delta y	y-new
70	0.82472	0.04770	0.00662	0.04108
71	0.85355	0.03920	0.01435	0.02485
72	0.88020	0.03176	0.02149	0.01027
73	0.90451	0.02517	0.02800	-0.00283
74	0.92632	0.01929	0.03385	-0.01455
75	0.94550	0.01409	0.03899	-0.02489
76	0.96194	0.00961	0.04339	-0.03378
77	0.97553	0.00590	0.04703	-0.04114
78	0.98618	0.00300	0.04989	-0.04689
79	0.99384	0.00092	0.05194	-0.05102
80	0.99846	-0.00032	0.05318	-0.05350
81	1.00000	-0.00074	0.05359	-0.05433

setting up coefficient matrix - takes some time

Computing LU decomposition - may take awhile

input alpha in degrees
2.

Pressure and Velocity distributions

I	X	Y	CP	U/Ue
1	0.9992	-0.0611	0.41670	-0.7637
2	0.9962	-0.0596	0.51302	-0.6978
3	0.9900	-0.0569	0.60839	-0.6258
4	0.9809	-0.0532	0.63092	-0.6075
5	0.9687	-0.0487	0.60386	-0.6294
6	0.9537	-0.0439	0.57453	-0.6523
7	0.9359	-0.0389	0.57245	-0.6539
8	0.9154	-0.0338	0.57498	-0.6519

D-14 Applied Computational Aerodynamics

9	0.8924	-0.0289	0.57777	-0.6498
10	0.8669	-0.0245	0.59221	-0.6386
11	0.8391	-0.0205	0.63641	-0.6030
12	0.8093	-0.0178	0.67164	-0.5730
13	0.7776	-0.0198	0.58930	-0.6409
14	0.7441	-0.0258	0.53214	-0.6840
15	0.7092	-0.0322	0.47252	-0.7263
16	0.6729	-0.0386	0.41531	-0.7646
17	0.6356	-0.0450	0.35948	-0.8003
18	0.5975	-0.0508	0.30927	-0.8311
19	0.5587	-0.0557	0.27221	-0.8531
20	0.5196	-0.0594	0.25583	-0.8627
21	0.4804	-0.0620	0.25355	-0.8640
22	0.4413	-0.0637	0.25645	-0.8623
23	0.4025	-0.0647	0.26273	-0.8586
24	0.3644	-0.0651	0.27258	-0.8529
25	0.3271	-0.0649	0.28574	-0.8451
26	0.2908	-0.0642	0.30098	-0.8361
27	0.2559	-0.0628	0.32177	-0.8235
28	0.2224	-0.0609	0.35146	-0.8053
29	0.1907	-0.0584	0.38532	-0.7840
30	0.1609	-0.0554	0.42540	-0.7580
31	0.1331	-0.0520	0.47686	-0.7233
32	0.1076	-0.0482	0.52766	-0.6873
33	0.0846	-0.0440	0.59616	-0.6355
34	0.0641	-0.0394	0.68317	-0.5629
35	0.0463	-0.0345	0.76489	-0.4849
36	0.0313	-0.0294	0.87733	-0.3502
37	0.0191	-0.0240	0.96586	-0.1848
38	0.0100	-0.0185	0.99394	0.0779
39	0.0038	-0.0122	0.71924	0.5299
40	0.0008	-0.0043	0.00089	0.9996
41	0.0008	0.0057	-0.84325	1.3577
42	0.0038	0.0169	-1.79413	1.6716
43	0.0100	0.0272	-2.45163	1.8579
44	0.0191	0.0367	-2.56550	1.8883
45	0.0313	0.0457	-2.52528	1.8776
46	0.0463	0.0540	-2.36679	1.8349
47	0.0641	0.0616	-2.11734	1.7656
48	0.0846	0.0686	-1.89645	1.7019
49	0.1076	0.0754	-1.71369	1.6473
50	0.1331	0.0819	-1.57093	1.6034
51	0.1609	0.0885	-1.49116	1.5783
52	0.1907	0.0952	-1.45244	1.5660
53	0.2224	0.1020	-1.45256	1.5661
54	0.2559	0.1088	-1.49447	1.5794
55	0.2908	0.1155	-1.56755	1.6024
56	0.3271	0.1216	-1.66552	1.6326
57	0.3644	0.1269	-1.77142	1.6648
58	0.4025	0.1308	-1.86810	1.6935
59	0.4413	0.1330	-1.94601	1.7164
60	0.4804	0.1331	-1.99347	1.7302
61	0.5196	0.1310	-1.98282	1.7271
62	0.5587	0.1261	-1.89361	1.7011
63	0.5975	0.1187	-1.71802	1.6486
64	0.6356	0.1090	-1.49638	1.5800
65	0.6729	0.0978	-1.27652	1.5088
66	0.7092	0.0859	-1.09776	1.4484
67	0.7441	0.0740	-0.99374	1.4120
68	0.7776	0.0628	-1.12848	1.4589
69	0.8093	0.0492	-0.82110	1.3495
70	0.8391	0.0330	-0.48770	1.2197
71	0.8669	0.0176	-0.31486	1.1467
72	0.8924	0.0037	-0.20068	1.0958

73	0.9154	-0.0087	-0.11007	1.0536
74	0.9359	-0.0197	-0.04075	1.0202
75	0.9537	-0.0293	0.01110	0.9944
76	0.9687	-0.0375	0.03867	0.9805
77	0.9809	-0.0440	0.04913	0.9751
78	0.9900	-0.0490	0.09990	0.9487
79	0.9962	-0.0523	0.23071	0.8771
80	0.9992	-0.0539	0.41671	0.7637
I	X	Y	CP	U/Ue

AT ALPHA = 2.000

CL = 1.82147 CM(l.e.) = -0.76764 Cm(c/4) = -0.31257
 CD = -0.00344 (theoretically zero)

send output to a file? (Y/N):

y

enter file name: gaw1.out

enter file title: GAW 1 airfoil with upper surface mod and trailing edge deflected

Another angle of attack? (Y/N):

n

STOP

The output disk file generated from the above is given here (for a 44,44 panel case):

GAW 1 airfoil with upper surface mod and trailing edge deflected

Alpha	CL	cmc4	CD		
2.0000	1.8253	-0.3139	-0.0034		
90.0000000					
	X/C	Y/C	Cp	U/UE	
	1.0000000	-0.0614198	0.4218349	-0.7603717	
	0.9987261	-0.0608210	0.4894220	-0.7145474	
	0.9949107	-0.0590636	0.5828183	-0.6458961	
	0.9885734	-0.0562688	0.6198552	-0.6165588	
	0.9797465	-0.0526529	0.6121694	-0.6227604	
	0.9684749	-0.0485395	0.5824285	-0.6461977	
	0.9548160	-0.0441873	0.5709316	-0.6550332	
	0.9388395	-0.0396023	0.5742147	-0.6525223	
	0.9206268	-0.0348995	0.5779232	-0.6496744	
	0.9002706	-0.0303699	0.5814999	-0.6469159	
	0.8778748	-0.0262270	0.5949170	-0.6364613	
	0.8535534	-0.0224278	0.6267794	-0.6109178	
	0.8274304	-0.0189073	0.7059953	-0.5422220	
	0.7996389	-0.0159320	0.6026480	-0.6303586	
	0.7703204	-0.0211078	0.5561817	-0.6661969	
	0.7396245	-0.0265297	0.4989320	-0.7078615	
	0.7077075	-0.0324513	0.4452444	-0.7448192	
	0.6747321	-0.0383475	0.3940417	-0.7784333	
	0.6408663	-0.0441870	0.3435679	-0.8102050	
	0.6062826	-0.0496877	0.2998845	-0.8367290	
	0.5711573	-0.0544341	0.2689947	-0.8549885	
	0.5356696	-0.0582109	0.2556977	-0.8627295	
	0.5000000	-0.0609100	0.2539230	-0.8637575	
	0.4643304	-0.0628407	0.2562625	-0.8624022	
	0.4288425	-0.0641589	0.2613289	-0.8594598	
	0.3937173	-0.0649298	0.2692938	-0.8548135	
	0.3591337	-0.0651852	0.2802141	-0.8484020	
	0.3252679	-0.0649567	0.2932411	-0.8406895	
	0.2922925	-0.0642823	0.3083775	-0.8316385	
	0.2603754	-0.0631488	0.3301547	-0.8184408	

D-16 Applied Computational Aerodynamics

0.2296796	-0.0614624	0.3589480	-0.8006572
0.2003612	-0.0592899	0.3902262	-0.7808802
0.1725696	-0.0567585	0.4273256	-0.7567526
0.1464466	-0.0538145	0.4750733	-0.7245182
0.1221252	-0.0504530	0.5206525	-0.6923493
0.0997294	-0.0469538	0.5783747	-0.6493268
0.0793732	-0.0430568	0.6562179	-0.5863293
0.0611605	-0.0387003	0.7307052	-0.5189362
0.0451840	-0.0343281	0.8209081	-0.4231924
0.0315251	-0.0296737	0.9181719	-0.2860562
0.0202535	-0.0247229	0.9870794	-0.1136685
0.0114266	-0.0198401	0.9721012	0.1670296
0.0050893	-0.0145438	0.6358421	0.6034549
0.0012739	-0.0078949	-0.0395444	1.0195805
0.0000000	0.0000000	-0.8027386	1.3426610
0.0012739	0.0104299	-1.6840084	1.6382943
0.0050893	0.0205191	-2.3751559	1.8371598
0.0114266	0.0295006	-2.5661790	1.8884330
0.0202535	0.0379599	-2.5584900	1.8863961
0.0315251	0.0461391	-2.4712462	1.8631281
0.0451840	0.0537536	-2.2740183	1.8094249
0.0611605	0.0606522	-2.0435944	1.7445900
0.0793732	0.0671033	-1.8532349	1.6891521
0.0997294	0.0732909	-1.6915706	1.6406007
0.1221252	0.0792271	-1.5635967	1.6011236
0.1464466	0.0851389	-1.4915837	1.5784751
0.1725696	0.0912161	-1.4534186	1.5663393
0.2003612	0.0973116	-1.4457374	1.5638853
0.2296796	0.1035252	-1.4751878	1.5732729
0.2603755	0.1097611	-1.5325123	1.5913869
0.2922925	0.1158108	-1.6132200	1.6165457
0.3252679	0.1214624	-1.7096776	1.6461098
0.3591337	0.1263813	-1.8051412	1.6748556
0.3937174	0.1302163	-1.8901043	1.7000307
0.4288426	0.1327159	-1.9578166	1.7198304
0.4643304	0.1336181	-1.9988899	1.7317303
0.5000000	0.1326900	-1.9909948	1.7294493
0.5356696	0.1296440	-1.9168615	1.7078822
0.5711575	0.1243886	-1.7714504	1.6647674
0.6062827	0.1169697	-1.5728503	1.6040107
0.6408663	0.1077561	-1.3658797	1.5381416
0.6747321	0.0973867	-1.1803569	1.4766032
0.7077075	0.0864435	-1.0399497	1.4282681
0.7396246	0.0755168	-0.9781721	1.4064751
0.7703204	0.0650848	-1.1870403	1.4788646
0.7996388	0.0555373	-0.7750797	1.3323212
0.8274304	0.0395294	-0.4784321	1.2159079
0.8535534	0.0248468	-0.3243269	1.1507940
0.8778748	0.0115267	-0.2185851	1.1038954
0.9002706	-0.0005504	-0.1328125	1.0643367
0.9206268	-0.0114915	-0.0640643	1.0315349
0.9388395	-0.0212971	-0.0094062	1.0046921
0.9548160	-0.0299273	0.0304447	0.9846600
0.9684749	-0.0373215	0.0494097	0.9749822
0.9797465	-0.0434152	0.0697593	0.9644899
0.9885734	-0.0481765	0.1362851	0.9293627
0.9949107	-0.0515903	0.2603474	0.8600306
0.9987261	-0.0536442	0.4218340	0.7603723

D.3 Program LIDRAG

This program computes the span e for a single planar lifting surface given the spanload. It uses the spanload to determine the “ e ” using a Fast Fourier Transform. Numerous other methods could be used. For reference, note that the “ e ” for an elliptic spanload is 1.0, and the “ e ” for a triangular spanload is .72. The code is in the file LIDRAG.F. The sample input is also on the disk and is called B2LDG.INP. The program prompts the user for the name of the input file.

The program was written by Dave Ives, and entered the public domain through the code contained in AFFDL-TR-77-122, “An Automated Procedure for Computing the Three Dimensional Transonic Flow over Wing-Body Combinations, Including Viscous Effects,” Feb. 1978.

The input is the spanload obtained from any method. The output is the Trefftz plane induced drag e and the integral of the spanload, which produces the C_L . This is the “span” e . You should include a point at $\eta = 0$ and at $\eta = 1$ you should include a point with zero spanload. See the sample input for an example.

The input instruction:

<u>Card</u>	<u>Field</u>	<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1	1	1-10	FSPN	Number of spanwise stations of input
2	1	1-10	ETA	The spanwise location of input, $y/(b/2)$.
2	1	1-20	CCLCA	The spanload, ccl/ca (the local chord times the local lift coefficient divided by the average chord)

Note: card 2 is repeated FSPN times

Sample input: (from the output of the **VLMpc** sample case for the B-2, and in the file B2LDG.INP on the disk))

```

20.
0.0      0.58435
0.01805  0.58435
0.06388  0.57919
0.11943  0.56800
0.17664  0.55739
0.23385  0.54709
0.30271  0.52459
0.37158  0.48623
0.42713  0.44590
0.48269  0.40097
0.53925  0.36490
0.59581  0.34718
0.65137  0.33280
0.70693  0.31865
0.76248  0.30225
0.81804  0.27971
0.86735  0.24229
0.91667  0.18494
0.97222  0.09480
1.000    0.000

```

D.4 LAMDES User's Manual

This is the Lamar design program, LamDes2.f. It can be used as a non-planar LIDRAG to get span e for multiple lifting surface cases when user supplies spanload. It has also been called the Lamar/Mason optimization code. It finds the spanload to minimize the sum of the induced and pressure drag, including canards or winglets. It also provides the associated camber distribution for subsonic flow. Since two surfaces are included, it can find the minimum trimmed drag while satisfying a pitching moment constraint.

The program will prompt you for the input file name. A sample input file called lamdes.inp is on the disk, and the output obtained from this case is included here.

References:

J.E. Lamar, "A Vortex Lattice Method for the Mean Camber Shapes of Trimmed Non-Coplanar Planforms with Minimum Vortex Drag," NASA TN D-8090, June, 1976.

W.H. Mason, "Wing-Canard Aerodynamics at Transonic Speeds - Fundamental Considerations on Minimum Drag Spanloads," AIAA Paper No. 82-0097, January 1982.

Input Instructions:

The program assumes the load distribution is constant chordwise until a designated chordwise location (XCFW on the first surface and XCFT on the second surface). The loading then decreases linearly to the trailing edge. This corresponds to a 6 & 6A series camber distribution (the value for the 6A series is usually 0.8). If airfoil polars are used to model the effects of viscosity, the polars are input in a streamwise coordinate system. The user is responsible for adjusting them from 2D to 3D.

This program uses an input file that is very similar to, *but not the same as*, the VLMpcv2 code. It is based on the same geometry and coordinate system ideas. Section D.6 should be consulted for a discussion of the geometry system.

<u>Card #</u>	<u>Format</u>	<u>Field</u>	<u>Name</u>	<u>Remarks</u>
1	Literal		DATA	Title card for the data set
2	8F10.6	1	PLAN	Number of lifting surfaces for the configuration; use 1 or 2.
		2	XMREF	c.g. shift from origin of input planform coordinate system (the program originally trimmed the configuration about the input planform origin). + is a c.g. shift forward - is a c.g. shift aft
		3	CREF	reference chord of the configuration, used only to nondimensionalize the pitching moment coefficients.
		4	SREF	reference area of the configuration

5	TDKLUE	minimization clue = 0 - minimize induced drag only = 1 - minimize induced plus pressure drag
6	CASE	options for the drag polar = 0, model polar, same a , CLmin, CD0 for each surface(see note 3 below). = 1, model polar, each surface has its own a , CLmin, CD0 = 2, one general polar for entire config. = 3, one general polar for each surface
7	SPNKLU	spanload clue = 0 spanload is internally computed using the minimization = 1, no minimization is done, spanload is read in, and e and pressure drag are computed.

Geometric/Planform Data - see the VLMpc section (D.6) for more details

<u>Card #</u>	<u>Format</u>	<u>Field</u>	<u>Name</u>	<u>Remarks</u>
1-P	8F10.6	1	AAN(IT)	# of straight lines defining this surface
		2	XS(IT)	= 0. (not used in this code)
		3	YS(IT)	= 0. (not used in this code)
		4	RTCDHT(IT)	root chord height (- is "higher")
		5	PDRG1(IT)	CLmin
		6	PDRG2(IT)	" a "
		7	PDRG3(IT)	CD0
2-P	8F10.6	1	XREG	X point of line segment (positive is forward)
		2	YREG	Y point of line segment (positive is forward)
		3	DIH	dihedral angle of line
		4	AMCD	sweep wing move code, set = 1 for this program

- Note:
- Card 2-P is read in AAN + 1 times. Surface description starts at forward centerline and works outboard and around, returning to the aft centerline of the surface.
 - Cards 1-P and 2-P are read in as a set for each lifting surface (see VLM4997 for clarification)
 - The model polar is given by: $C_d = a (C_l - C_{lmin})^2 + C_{D0}$

Control Data (corresponding to “Group Two” data in Lamar’s nomenclature)

<u>Card #</u>	<u>Format</u>	<u>Field</u>	<u>Name</u>	<u>Remarks</u>
1-C	6F5.3,2F10.6	1	CONFIG	arbitrary configuration number or ID (may include up to four digits)
		2	SCW	Number of chordwise horseshoe vortices to be used to represent the wing; a maximum of 20 may be used, do not set to zero.
		3	VIC	nominal number of spanwise rows at which chordwise horseshoe may be located; a maximum of 50 may be used. The product of SCW and SSW cannot exceed 400 (see VLM4997 chapter for details of vortex layout).
		4	XMCH	Mach number, used to apply Prandtl- Glauert compressibility correction factor.
		5	CLDES	design lift coefficient for lifting system
		6	XITMAX	Maximum number of iterations allowed in finding the solution for minimum + pressure drag with arbitrary polars input. Must be less than 50. 20 is sufficient for most cases.
		7	EPSMAX	The convergence criteria for the general polar case. A value of .0005 appears to be reasonable.
2-C	6F10.4	1	XCFW	The chord fraction “ <i>a</i> ” at which the chord load shape changes from rooftop to a linear decrease to zero at the trailing edge on the first planform. See the introduction to this section for more discussion.
		2	XCFT	Same as XCFW, except applies to the second planform.
		3	FKON	Clue for constraints = 0 body moment constraint = 1 no constraints = 2 root bending moment constraint = 3 both moment and root bending moment constraints.
		4	CMB	The design wing CM when FKON = 0
		5	FICAM	Camber computation clue. = 0, no cambers computed = 1, wing cambers computed

3-C	8F10.6	6	PUNCH	clue to punch cambers out = 0 - no punch file created = 1 - cards output (unit 7)
		7	CRBMT	Design root bending moment for FKON = 2.
		1	RELAX	The under-relaxation factor for the general polar solution. RELAX = .03 to .3 is satisfactory for most applications.
		2	FIOUTW	Output clue. = 0 - full iteration history is output = 1 - only final results are output
		3	CD0	Basic drag coefficient that will be added to the drag computed by summing the induced drag and the profile drag contained in the input polars.

Arbitrary Polar Input (the following cards are read only if CASE $\geq$ 2.)

<u>Card #</u>	<u>Format</u>	<u>Field</u>	<u>Name</u>	<u>Remarks</u>
1-D	Literal		TITLE	The identifying title for the input drag polar for this surface.
2-D	8F10.5	1	FNCLCD	The number of CL,CD pairs used to define the input polar.
3-D	8F10.5	1	FQCL	The value of streamwise lift coefficient for this point on the drag polar.
		2	FQCD	The value of streamwise drag coefficient for the given lift coefficient.

- Note: 1. Card 3-D is read FNCLCD times
2. Cards 1-D, 2-D and 3-D are read for each planform if CASE = 3.

Spanload Input (the following cards are read only if SPNKLU = 1)

<u>Card #</u>	<u>Format</u>	<u>Field</u>	<u>Name</u>	<u>Remarks</u>
1-S	Literal		TITLE	This is the title card for the input spanloads.
2-S	7F10.5	1	FSPNPT	Number of points on the spanload to be read in for this planform.
3-S	7F10.5	1	YSPNPT	Span location in physical coordinates at which ccl/ca is input (y is positive here!)
		2	CLSPNP	The spanload at YSPNPT

- Note: 1. Card 3-S is read FSPNPT times
2. Cards 2-S and 3-S are read for each planform as a set.

Sample Input: (note: it is important to put data in proper columns!)

```

Lamar program sample input - revised forward swept wing
2.000 -8.000 89.50 26640. 1.0 3.0 0.0
5.000 0.0 0.0 -8.8 0.0 0.0
68.95 0.0 0.0 1.0
68.95 -34.0
49.61 -65.30 0.0 1.0
25.64 -65.30 0.0 1.0
22.25 -34.00
22.25 0.00
5.0 0.0 0.0 0.0 0.0 0.0
-25.90 0.0 0.0 1.0
-25.90 -34.0
38.10 -164.0 0.0 1.0
-2.40 -164.0 0.0 1.0
-147.90 -20.0
-147.90 0.0
1.0 10.0 20. 0.9 0.90 40.0 0.0006
0.0 0.65 0.0 -0.10 1.0
0.030 1.0 0.0 0.0 0.0 0.0
drag polar on canard (conv. sec)
18.0
0.00 0.0000
0.10 0.0000
0.25 0.0002
0.30 0.00078
0.40 0.00175
0.50 0.00315
0.55 0.0040
0.60 0.00535
0.65 0.00685
0.70 0.00880
0.75 0.01125
0.80 0.01485
0.85 0.01975
0.88 0.02400
0.915 0.03600
1.00 0.0880
1.20 0.2680
1.80 0.9880
drag polar
22.0
0.000 0.0003
0.200 0.0003
0.300 0.0005
0.400 0.0008
0.500 0.00125
0.600 0.00178
0.700 0.00244
0.800 0.00324
0.900 0.00442
0.950 0.00528
0.970 0.00570
0.990 0.00621
1.000 0.00650
1.020 0.00730
1.040 0.00820
1.060 0.00930
1.080 0.01090
1.100 0.01280
1.125 0.02400
1.130 0.03600
1.200 0.20400
2.000 2.12400

```

D-24 Applied Computational Aerodynamics

Sample Output:

enter name of input file: lamdes.inp

Lamar Design Code mods by W.H. Mason

Lamar program sample input - revised forward swept wing

plan = 2.0 xmref = -8.0000 cref = 89.5000
tdklue = 1.0 case = 3.0 spnklu = 0.0
sref = 26640.0000

1st REFERENCE PLANFORM HAS 5 CURVES
ROOT CHORD HEIGHT = -8.8000

POINT	X REF	Y REF	SWEEP ANGLE	DIHEDRAL ANGLE
1	76.9500	0.0000	0.00000	0.00000
2	76.9500	-34.0000	31.71155	0.00000
3	57.6100	-65.3000	90.00000	0.00000
4	33.6400	-65.3000	-6.18142	0.00000
5	30.2500	-34.0000	0.00000	0.00000
6	30.2500	0.0000		

2nd REFERENCE PLANFORM HAS 5 CURVES
ROOT CHORD HEIGHT = 0.0000

POINT	X REF	Y REF	SWEEP ANGLE	DIHEDRAL ANGLE
1	-17.9000	0.0000	0.00000	0.00000
2	-17.9000	-34.0000	-26.21138	0.00000
3	46.1000	-164.0000	90.00000	0.00000
4	5.6000	-164.0000	-45.29687	0.00000
5	-139.9000	-20.0000	0.00000	0.00000
6	-139.9000	0.0000		

scw = 10.0 vic = 20.0
xitmax = 40.0 epsmax = 0.00060

CONFIGURATION NO. 1.
delta ord shift for moment = -8.0000

CURVE 1 IS SWEEP 0.0000 DEGREES ON PLANFORM 1
CURVE 1 IS SWEEP 0.0000 DEGREES ON PLANFORM 2

BREAK POINTS FOR THIS CONFIGURATION

POINT	X	Y	Z	SWEEP ANGLE	DIHEDRAL ANGLE
1	76.9500	0.0000	-8.8000	0.0000	0.0000
2	76.9500	-20.0000	-8.8000	0.0000	0.0000
3	76.9500	-34.0000	-8.8000	31.7116	0.0000
4	57.6100	-65.3000	-8.8000	90.0000	0.0000
5	33.6400	-65.3000	-8.8000	-6.1814	0.0000
6	30.2500	-34.0000	-8.8000	0.0000	0.0000
7	30.2500	0.0000	-8.8000		
SECOND PLANFORM BREAK POINTS					
1	-17.9000	0.0000	0.0000	0.0000	0.0000
2	-17.9000	-34.0000	0.0000	-26.2114	0.0000
3	-2.4908	-65.3000	0.0000	-26.2114	0.0000
4	46.1000	-164.0000	0.0000	90.0000	0.0000
5	5.6000	-164.0000	0.0000	-45.2969	0.0000
6	-139.9000	-20.0000	0.0000	0.0000	0.0000
7	-139.9000	0.0000	0.0000		

PLANFORM	TOTAL	SPANWISE
1	80	8
2	200	20

10. HORSESHOE VORTICES IN EACH CHORDWISE ROW

xcfw = 0.00	xcft = 0.65	fkcn = 0.00
ficam = 1.00	punch = 0.00	crbmnt = 0.0000
cmb = -.10	iflag = 1	
relax = 0.03	fioutw = 1.00	cd0 = 0.0000
firbm = 0.00	yrbm = 0.0000	zrbm = 0.0000

drag polar on canard (conv. sec)

there are 1.0 polars on this surface

18.0 points this polar planform 1

qcl	qcd
0.0000	0.0000
0.1000	0.0000
0.2500	0.0002
0.3000	0.0008
0.4000	0.0018
0.5000	0.0032
0.5500	0.0040
0.6000	0.0054
0.6500	0.0069
0.7000	0.0088
0.7500	0.0113
0.8000	0.0148
0.8500	0.0198
0.8800	0.0240
0.9150	0.0360
1.0000	0.0880
1.2000	0.2680
1.8000	0.9880

drag polar

there are 1.0 polars on this surface

22.0 points this polar planform 2

qcl	qcd
0.0000	0.0003
0.2000	0.0003
0.3000	0.0005
0.4000	0.0008
0.5000	0.0012
0.6000	0.0018
0.7000	0.0024
0.8000	0.0032
0.9000	0.0044
0.9500	0.0053
0.9700	0.0057
0.9900	0.0062
1.0000	0.0065
1.0200	0.0073
1.0400	0.0082

D-26 Applied Computational Aerodynamics

1.0600	0.0093
1.0800	0.0109
1.1000	0.0128
1.1250	0.0240
1.1300	0.0360
1.2000	0.2040
2.0000	2.1240

LM = 70	IL = 71	JM = 72	IM = 73	TSPAN = -164.000	TSPANA = -65.300
BOTL = 164.000	BOL = 65.300	SNN = 1.6400	DELTYB = 3.2800		
NMA(KBOT) = 50	KBOT = 2	NMA(KBIT) = 20	KBIT = 1		

induced drag cd = 0.06815	pressure drag cdpt = 0.01665
induced drag cd = 0.06818	pressure drag cdpt = 0.01441
induced drag cd = 0.06827	pressure drag cdpt = 0.01255
induced drag cd = 0.06839	pressure drag cdpt = 0.01139
induced drag cd = 0.06850	pressure drag cdpt = 0.01053
induced drag cd = 0.06863	pressure drag cdpt = 0.00976
induced drag cd = 0.06876	pressure drag cdpt = 0.00915
induced drag cd = 0.06885	pressure drag cdpt = 0.00886
induced drag cd = 0.06893	pressure drag cdpt = 0.00868
induced drag cd = 0.06898	pressure drag cdpt = 0.00856
induced drag cd = 0.06902	pressure drag cdpt = 0.00847
induced drag cd = 0.06905	pressure drag cdpt = 0.00841
induced drag cd = 0.06907	pressure drag cdpt = 0.00836
induced drag cd = 0.06909	pressure drag cdpt = 0.00832
induced drag cd = 0.06911	pressure drag cdpt = 0.00829
induced drag cd = 0.06913	pressure drag cdpt = 0.00826
induced drag cd = 0.06915	pressure drag cdpt = 0.00823
induced drag cd = 0.06916	pressure drag cdpt = 0.00821
induced drag cd = 0.06917	pressure drag cdpt = 0.00819
induced drag cd = 0.06918	pressure drag cdpt = 0.00817
induced drag cd = 0.06919	pressure drag cdpt = 0.00816
induced drag cd = 0.06920	pressure drag cdpt = 0.00815
induced drag cd = 0.06921	pressure drag cdpt = 0.00814
induced drag cd = 0.06921	pressure drag cdpt = 0.00813
induced drag cd = 0.06922	pressure drag cdpt = 0.00812
induced drag cd = 0.06923	pressure drag cdpt = 0.00811

induced drag cd = 0.06923 pressure drag cdpt = 0.00810
 induced drag cd = 0.06924 pressure drag cdpt = 0.00810
 induced drag cd = 0.06924 pressure drag cdpt = 0.00809
 induced drag cd = 0.06924 pressure drag cdpt = 0.00809
 induced drag cd = 0.06925 pressure drag cdpt = 0.00808

pressure drag iteration has converged

k	eps	cl	cdi	cdp	cdi+cdp
1	28.66362	0.90000	0.06815	0.01665	0.08480
2	0.05789	0.90000	0.06818	0.01441	0.08260
3	0.05278	0.90000	0.06827	0.01255	0.08082
4	0.04274	0.90000	0.06839	0.01139	0.07978
5	0.03408	0.90000	0.06850	0.01053	0.07903
6	0.03155	0.90000	0.06863	0.00976	0.07839
7	0.02773	0.90000	0.06876	0.00915	0.07791
8	0.02043	0.90000	0.06885	0.00886	0.07772
9	0.01549	0.90000	0.06893	0.00868	0.07761
10	0.01218	0.90000	0.06898	0.00856	0.07754
11	0.00994	0.90000	0.06902	0.00847	0.07749
12	0.00847	0.90000	0.06905	0.00841	0.07746
13	0.00724	0.90000	0.06907	0.00836	0.07743
14	0.00616	0.90000	0.06909	0.00832	0.07741
15	0.00519	0.90000	0.06911	0.00829	0.07740
16	0.00442	0.90000	0.06913	0.00826	0.07739
17	0.00371	0.90000	0.06915	0.00823	0.07738
18	0.00310	0.90000	0.06916	0.00821	0.07737
19	0.00263	0.90000	0.06917	0.00819	0.07736
20	0.00221	0.90000	0.06918	0.00817	0.07736
21	0.00183	0.90000	0.06919	0.00816	0.07735
22	0.00154	0.90000	0.06920	0.00815	0.07735
23	0.00131	0.90000	0.06921	0.00814	0.07734
24	0.00112	0.90000	0.06921	0.00813	0.07734
25	0.00095	0.90000	0.06922	0.00812	0.07734
26	0.00084	0.90000	0.06923	0.00811	0.07734
27	0.00076	0.90000	0.06923	0.00810	0.07733
28	0.00069	0.90000	0.06924	0.00810	0.07733
29	0.00064	0.90000	0.06924	0.00809	0.07733
30	0.00061	0.90000	0.06924	0.00809	0.07733
31	0.00057	0.90000	0.06925	0.00808	0.07733

induced + pressure drag was minimized on this run

ref. chord = 89.500 c average = 81.2195 true area = 32771.566
 ref. area = 26640.000 b/2 = 164.0000 ref ar = 4.0384
 true ar = 3.2828 Mach number = 0.9000

first planform cl = 0.17126 cm = 0.11493 cb = -0.01502
 second planform cl = 0.72874 cm = -0.21493 cb = -0.18341

1st planform CL = 0.1713 CDP = 0.0042 CM = 0.1150 CB = -0.0151
 2nd planform CL = 0.7292 CDP = 0.0038 CM = -0.2149 CB = 0.0000

no root bending moment constraint

CL DES = 0.90000 CL COMPUTED = 0.9005 CM = -0.0999
 CD I = 0.06925 E = 0.9230
 CDPRESS = 0.00804 CDTOTAL = 0.07729

D-28 Applied Computational Aerodynamics

first planform

Y	CL*C/CAVE	C/CAVE	CL	CD
-61.2000	0.21189	0.33178	0.63862	0.00651
-53.0000	0.33566	0.40510	0.82857	0.01765
-44.8000	0.41311	0.47842	0.86348	0.02166
-37.3500	0.46740	0.54503	0.85757	0.02082
-29.9000	0.49499	0.57498	0.86088	0.02129
-22.9000	0.50260	0.57498	0.87411	0.02317
-15.9000	0.50504	0.57498	0.87835	0.02377
-5.9000	0.50631	0.57498	0.88056	0.02419

second planform

-159.9000	0.33879	0.52480	0.64556	0.00208
-151.7000	0.53136	0.57711	0.92072	0.00478
-143.5000	0.64513	0.62942	1.02495	0.00752
-135.3000	0.72403	0.68173	1.06206	0.00946
-127.1000	0.78509	0.73404	1.06954	0.01006
-118.9000	0.83563	0.78635	1.06267	0.00951
-110.7000	0.87760	0.83866	1.04644	0.00855
-102.5000	0.91055	0.89096	1.02198	0.00739
-94.3000	0.93428	0.94327	0.99047	0.00622
-86.1000	0.94681	0.99558	0.95101	0.00530
-77.9000	0.94347	1.04789	0.90036	0.00443
-69.5500	0.90911	1.10116	0.82559	0.00354
-61.2000	0.82859	1.15442	0.71775	0.00258
-53.0000	0.74419	1.20673	0.61670	0.00189
-44.8000	0.67721	1.25904	0.53788	0.00145
-37.3500	0.63142	1.30656	0.48327	0.00117
-29.9000	0.60043	1.37894	0.43543	0.00096
-22.9000	0.58289	1.46602	0.39760	0.00079
-15.9000	0.57323	1.50210	0.38162	0.00074
-5.9000	0.56730	1.50210	0.37767	0.00073

mean camber lines to obtain the spanload

(subsonic linear theory)

y= -61.2000 y/(b/2) = -0.3732 chord= 26.9474

slopes, dz/dx, at control points, from front to rear

x/c	dz/dx
0.0750	0.1295
0.1750	0.0672
0.2750	0.0194
0.3750	-0.0200
0.4750	-0.0522
0.5750	-0.0775
0.6750	-0.0960
0.7750	-0.1077
0.8750	-0.1122
0.9750	-0.1081

mean camber shape (interpolated to 41 points)

x/c	z/c	delta x	delta z	(z-zle)/c
0.0000	-0.0299	0.0000	-0.8067	0.0000
0.0250	-0.0332	0.6737	-0.8944	-0.0040
0.0500	-0.0365	1.3474	-0.9831	-0.0080
0.0750	-0.0398	2.0211	-1.0717	-0.0121
0.1000	-0.0429	2.6947	-1.1558	-0.0159
0.1250	-0.0457	3.3684	-1.2310	-0.0195
0.1500	-0.0480	4.0421	-1.2945	-0.0226

0.1750	-0.0499	4.7158	-1.3456	-0.0252
0.2000	-0.0514	5.3895	-1.3857	-0.0275
0.2250	-0.0526	6.0632	-1.4166	-0.0294
0.2500	-0.0534	6.7368	-1.4399	-0.0310
0.2750	-0.0540	7.4105	-1.4563	-0.0323
0.3000	-0.0544	8.0842	-1.4660	-0.0334
0.3250	-0.0545	8.7579	-1.4689	-0.0343
0.3500	-0.0544	9.4316	-1.4651	-0.0349
0.3750	-0.0540	10.1053	-1.4548	-0.0353
0.4000	-0.0534	10.7790	-1.4383	-0.0354
0.4250	-0.0525	11.4526	-1.4160	-0.0353
0.4500	-0.0515	12.1263	-1.3884	-0.0351
0.4750	-0.0503	12.8000	-1.3556	-0.0346
0.5000	-0.0489	13.4737	-1.3181	-0.0339
0.5250	-0.0474	14.1474	-1.2760	-0.0331
0.5500	-0.0456	14.8211	-1.2297	-0.0322
0.5750	-0.0438	15.4948	-1.1794	-0.0310
0.6000	-0.0418	16.1684	-1.1254	-0.0298
0.6250	-0.0396	16.8421	-1.0679	-0.0284
0.6500	-0.0374	17.5158	-1.0074	-0.0269
0.6750	-0.0350	18.1895	-0.9440	-0.0253
0.7000	-0.0326	18.8632	-0.8781	-0.0236
0.7250	-0.0301	19.5369	-0.8100	-0.0218
0.7500	-0.0275	20.2105	-0.7400	-0.0200
0.7750	-0.0248	20.8842	-0.6682	-0.0181
0.8000	-0.0221	21.5579	-0.5950	-0.0161
0.8250	-0.0193	22.2316	-0.5205	-0.0141
0.8500	-0.0165	22.9053	-0.4452	-0.0120
0.8750	-0.0137	23.5790	-0.3696	-0.0100
0.9000	-0.0109	24.2527	-0.2942	-0.0079
0.9250	-0.0081	24.9263	-0.2196	-0.0059
0.9500	-0.0054	25.6000	-0.1458	-0.0039
0.9750	-0.0027	26.2737	-0.0728	-0.0020
1.0000	0.0000	26.9474	0.0000	0.0000

y= -53.0000 y/(b/2) = -0.3232 chord= 32.9022

slopes, dz/dx, at control points, from front to rear

x/c	dz/dx
0.0750	0.0783
0.1750	-0.0034
0.2750	-0.0572
0.3750	-0.0982
0.4750	-0.1306
0.5750	-0.1557
0.6750	-0.1740
0.7750	-0.1854
0.8750	-0.1898
0.9750	-0.1845

mean camber shape (interpolated to 41 points)

x/c	z/c	delta x	delta z	(z-zle)/c
0.0000	-0.1036	0.0000	-3.4093	0.0000
0.0250	-0.1056	0.8226	-3.4745	-0.0046
0.0500	-0.1076	1.6451	-3.5414	-0.0092
0.0750	-0.1097	2.4677	-3.6080	-0.0138
0.1000	-0.1115	3.2902	-3.6674	-0.0182
0.1250	-0.1128	4.1128	-3.7122	-0.0222
0.1500	-0.1136	4.9353	-3.7381	-0.0255
0.1750	-0.1138	5.7579	-3.7444	-0.0283
0.2000	-0.1135	6.5804	-3.7339	-0.0306
0.2250	-0.1128	7.4030	-3.7102	-0.0325
0.2500	-0.1117	8.2256	-3.6761	-0.0340

D-30 Applied Computational Aerodynamics

0.2750	-0.1104	9.0481	-3.6333	-0.0353
0.3000	-0.1089	9.8707	-3.5819	-0.0363
0.3250	-0.1070	10.6932	-3.5220	-0.0371
0.3500	-0.1050	11.5158	-3.4534	-0.0376
0.3750	-0.1026	12.3383	-3.3766	-0.0379
0.4000	-0.1001	13.1609	-3.2920	-0.0379
0.4250	-0.0973	13.9834	-3.2003	-0.0377
0.4500	-0.0943	14.8060	-3.1020	-0.0373
0.4750	-0.0911	15.6285	-2.9975	-0.0367
0.5000	-0.0878	16.4511	-2.8872	-0.0359
0.5250	-0.0842	17.2737	-2.7715	-0.0350
0.5500	-0.0806	18.0962	-2.6505	-0.0339
0.5750	-0.0767	18.9188	-2.5247	-0.0327
0.6000	-0.0728	19.7413	-2.3945	-0.0313
0.6250	-0.0687	20.5639	-2.2601	-0.0298
0.6500	-0.0645	21.3864	-2.1219	-0.0282
0.6750	-0.0602	22.2090	-1.9804	-0.0265
0.7000	-0.0558	23.0315	-1.8358	-0.0247
0.7250	-0.0513	23.8541	-1.6886	-0.0228
0.7500	-0.0468	24.6766	-1.5391	-0.0209
0.7750	-0.0422	25.4992	-1.3875	-0.0189
0.8000	-0.0375	26.3218	-1.2341	-0.0168
0.8250	-0.0328	27.1443	-1.0792	-0.0147
0.8500	-0.0281	27.9669	-0.9233	-0.0125
0.8750	-0.0233	28.7894	-0.7671	-0.0104
0.9000	-0.0186	29.6120	-0.6114	-0.0082
0.9250	-0.0139	30.4345	-0.4569	-0.0061
0.9500	-0.0092	31.2571	-0.3038	-0.0041
0.9750	-0.0046	32.0796	-0.1517	-0.0020
1.0000	0.0000	32.9022	0.0000	0.0000

Note this output is repeated for each span station. Most other stations are omitted

y= -5.9000 y/(b/2) = -0.0360 chord= 122.0000

slopes, dz/dx, at control points, from front to rear

x/c	dz/dx
0.0750	-0.0501
0.1750	-0.0505
0.2750	-0.0495
0.3750	-0.0500
0.4750	-0.0537
0.5750	-0.0623
0.6750	-0.0814
0.7750	-0.0975
0.8750	-0.1077
0.9750	-0.1097

mean camber shape (interpolated to 41 points)

x/c	z/c	delta x	delta z	(z-zle)/c
0.0000	-0.0697	0.0000	-8.5090	0.0000
0.0250	-0.0685	3.0500	-8.3562	-0.0005
0.0500	-0.0672	6.1000	-8.2034	-0.0010
0.0750	-0.0660	9.1500	-8.0506	-0.0015
0.1000	-0.0647	12.2000	-7.8975	-0.0020
0.1250	-0.0635	15.2500	-7.7440	-0.0024
0.1500	-0.0622	18.3000	-7.5900	-0.0029
0.1750	-0.0609	21.3500	-7.4358	-0.0034
0.2000	-0.0597	24.4000	-7.2818	-0.0039

0.2250	-0.0584	27.4500	-7.1286	-0.0044
0.2500	-0.0572	30.5000	-6.9763	-0.0049
0.2750	-0.0559	33.5500	-6.8249	-0.0054
0.3000	-0.0547	36.6000	-6.6742	-0.0059
0.3250	-0.0535	39.6500	-6.5237	-0.0064
0.3500	-0.0522	42.7000	-6.3728	-0.0069
0.3750	-0.0510	45.7500	-6.2210	-0.0074
0.4000	-0.0497	48.8000	-6.0676	-0.0079
0.4250	-0.0485	51.8500	-5.9121	-0.0084
0.4500	-0.0472	54.9000	-5.7537	-0.0088
0.4750	-0.0458	57.9500	-5.5919	-0.0092
0.5000	-0.0445	61.0000	-5.4262	-0.0096
0.5250	-0.0431	64.0500	-5.2558	-0.0100
0.5500	-0.0416	67.1000	-5.0791	-0.0102
0.5750	-0.0401	70.1500	-4.8940	-0.0105
0.6000	-0.0385	73.2000	-4.6978	-0.0106
0.6250	-0.0368	76.2500	-4.4878	-0.0106
0.6500	-0.0349	79.3000	-4.2627	-0.0105
0.6750	-0.0330	82.3500	-4.0221	-0.0103
0.7000	-0.0309	85.4000	-3.7669	-0.0100
0.7250	-0.0287	88.4500	-3.4982	-0.0095
0.7500	-0.0264	91.5000	-3.2174	-0.0089
0.7750	-0.0240	94.5500	-2.9253	-0.0083
0.8000	-0.0215	97.6000	-2.6231	-0.0076
0.8250	-0.0189	100.6500	-2.3115	-0.0067
0.8500	-0.0163	103.7000	-1.9920	-0.0059
0.8750	-0.0137	106.7500	-1.6662	-0.0049
0.9000	-0.0110	109.8000	-1.3359	-0.0040
0.9250	-0.0082	112.8500	-1.0031	-0.0030
0.9500	-0.0055	115.9000	-0.6690	-0.0020
0.9750	-0.0027	118.9500	-0.3345	-0.0010
1.0000	0.0000	122.0000	0.0000	0.0000

twist table

i	y	y/(b/2)	twist
1	-61.20000	-0.37317	1.71469
2	-53.00000	-0.32317	5.91587
3	-44.80000	-0.27317	7.36720
4	-37.35000	-0.22774	10.25835
5	-29.90000	-0.18232	9.47910
6	-22.90000	-0.13963	7.60813
7	-15.90000	-0.09695	6.49868
8	-5.90000	-0.03598	5.91663
9	-159.89999	-0.97500	14.45816
10	-151.70001	-0.92500	16.44655
11	-143.50000	-0.87500	14.38027
12	-135.30002	-0.82500	12.36750
13	-127.10001	-0.77500	10.75520
14	-118.90002	-0.72500	9.51973
15	-110.70002	-0.67500	8.46040
16	-102.50002	-0.62500	7.34168
17	-94.30003	-0.57500	6.13154
18	-86.10003	-0.52500	4.67249
19	-77.90003	-0.47500	2.88238
20	-69.55002	-0.42409	1.36595
21	-61.20000	-0.37317	3.52797
22	-53.00000	-0.32317	4.51491
23	-44.80000	-0.27317	4.49845
24	-37.35000	-0.22774	3.79378
25	-29.90000	-0.18232	3.77474
26	-22.90000	-0.13963	3.11226
27	-15.90000	-0.09695	3.52109
28	-5.90000	-0.03598	3.98970

STOP

Sample output:

Program LIDRAG

enter name of input data file
b2ldg.inp

LIDRAG - LIFT INDUCED DRAG ANALYSIS

INPUT SPANLOAD

N	Y/(B/2)	CCLCA
1	0.00000	0.58435
2	0.01805	0.58435
3	0.06388	0.57919
4	0.11943	0.56800
5	0.17664	0.55739
6	0.23385	0.54709
7	0.30271	0.52459
8	0.37158	0.48623
9	0.42713	0.44590
10	0.48269	0.40097
11	0.53925	0.36490
12	0.59581	0.34718
13	0.65137	0.33280
14	0.70693	0.31865
15	0.76248	0.30225
16	0.81804	0.27971
17	0.86735	0.24229
18	0.91667	0.18494
19	0.97222	0.09480
20	1.00000	0.00000

Span e = 0.94708 CL = 0.399

STOP

D.5 Program FRICTION

FRICTION provides an estimate of laminar and turbulent skin friction suitable for use in aircraft preliminary design. It is an entirely new program, but has its roots in a program by Ron Hendrickson at Grumman. It runs on any computer. The input requires geometric information and either the Mach and altitude combination, or the Mach and Reynolds number at which the results are desired. The skin friction is found using the Eckert Reference Temperature method for laminar flow and the van Driest II formula for turbulent flow. The basic formulas are valid from subsonic to hypersonic speeds, but the implementation makes assumptions that limit the validity to moderate supersonic speeds (about Mach 3). The key assumption is that the vehicle surface is at the adiabatic wall temperature (the user can easily modify this assumption). Form factors are used to estimate the effect of thickness on drag, and a composite formula is used to include the effect of a partial run of laminar flow. Because the methods aren't described in detail in the text, details are provided here.

Laminar flow

The approach used is known as the Eckert Reference Temperature Method, and this particular version is the one given by F.M. White in *Viscous Fluid Flow*, McGraw-Hill, New York, 1974, pp. 589-590. In this method the incompressible skin friction formula is used, with the fluid properties chosen at a specified reference temperature, which includes both Mach number and wall temperature effects.

First, assumptions are made for the fluid properties:* Prandtl number, $Pr = 0.72$, Recovery factor, $r = Pr^{1/2}$, specific heat ratio, $\gamma = 1.4$, and edge temperature, $T_e = 390$ (°R). Then, for a given edge Mach number, M_e , and ratio of wall temperature to adiabatic wall temperature T_W/T_{AW} , compute:

$$\frac{T_W}{T_e} = \frac{T_W}{T_{AW}} \left(1 + r \frac{\gamma - 1}{2} M_e^2 \right).$$

Remember that

$$T_{AW} = T_e \left(1 + r \frac{\gamma - 1}{2} M_e^2 \right)$$

and then compute the reference temperature:

$$\frac{T^*}{T_e} \cong .5 + .039 M_e^2 + 0.5 \left(\frac{T_W}{T_e} \right)$$

* These values can be changed easily by modifying the source code.

The Chapman-Rubesin constant based on the reference temperature and Sutherland's viscosity law is then computed from:

$$C^* = \left(\frac{T^*}{T_e} \right)^{1/2} \left(\frac{1 + K/T_e}{T^*/T_e + K/T_e} \right)$$

where $K = 200^\circ\text{R}$ for air.

Finally, the local friction coefficient (τ_w/q) is found from the standard Blasius formula, with C^* added,

$$C_f = \frac{.664\sqrt{C^*}}{\sqrt{\text{Re}_x}}$$

and

$$C_F = 2C_f$$

which comes from

$$C_F = \frac{F}{qx} = \frac{1}{x} \int_{x'=0}^{x'=x} C_f(x') dx'$$

Recall that C_F accounts for one side of the plate only, so that if both sides are required for a drag estimate, then the skin friction coefficient, C_D , is twice C_F because the reference area is based on one side only, i.e., $S_{ref} \approx 1/2 S_{wet}$.

Note that the results are not sensitive to the value of edge temperature for low Mach numbers, and therefore, an exact specification of T_e is not required. This method is implemented in subroutine **lamcf**.

Turbulent flow

For turbulent flow the so-called van Driest II Method is employed. This method was selected based on the recommendation of E.J. Hopkins and M. Inouye, contained in "An Evaluation of Theories for Predicting Turbulent Skin Friction and Heat Transfer on Flat Plates at Supersonic and Hypersonic Mach Numbers," *AIAA J.*, Vol. 9, No. 6, June 1971, pp. 993-1003. The particular algorithm is taken from NASA TN D-6945, "Charts for Predicting Turbulent Skin Friction From the Van Driest Method (II)," also by E.J. Hopkins, and dated October 1972.

Again, assumptions are made for the fluid properties: turbulent flow recovery factor, $r = .88$, specific heat ratio, $\gamma = 1.4$, and edge temperature, $T_e = 222$ ($^\circ\text{K}$). Then, for a given edge Mach number, M_e , and ratio of wall temperature to adiabatic wall temperature T_w/T_{AW} the calculation is started by computing the following constants:

$$m = \frac{\gamma - 1}{2} M_e^2$$

$$F = \frac{T_w}{T_e} = \frac{T_w}{T_{AW}} \cdot \frac{T_{AW}}{T_e}$$

where

$$\frac{T_{AW}}{T_e} = 1 + rm$$

$$T_w = F \cdot T_e$$

$$A = \left(\frac{rm}{F} \right)^{1/2}$$

$$B = \frac{1 + rm - F}{F}$$

$$\alpha = \frac{2A^2 - B}{(4A^2 + B^2)^{1/2}}$$

$$\beta = \frac{B}{(4A^2 + B^2)^{1/2}}$$

$$F_c = \frac{rm}{(\sin^{-1} \alpha + \sin^{-1} \beta)^2} \quad M_e > 0.1$$

$$= \left(\frac{1 + \sqrt{F}}{2} \right)^2 \quad M_e \leq 0.1$$

and

$$F_\theta = \frac{\mu_e}{\mu_w} = \sqrt{\frac{1}{F}} \left(\frac{1 + \frac{122}{T_w} \times 10^{-5/T_w}}{1 + \frac{122}{T_e} \times 10^{-5/T_e}} \right)$$

which is the Keyes viscosity law.

Finally,

$$F_x = \frac{F_\theta}{F_c}$$

The analysis proceeds using barred quantities to denote “incompressible” variables, which are intermediate variables not used except to obtain the final results. Given the Reynolds number, Re_x , an iteration is used to obtain the final results. Proceed as follows, finding

$$\overline{\text{Re}}_x = F_x \text{Re}_x$$

now solve

$$\frac{.242}{\sqrt{\overline{C}_F}} = \log(\overline{\text{Re}}_x \overline{C}_F)$$

for $\overline{C}_F$.

Use as an initial guess

$$\overline{C}_F^0 = \frac{.074}{\overline{\text{Re}}_x^{.20}}.$$

Then, Newton's method is applied to the problem:

$$f(\overline{C}_F) = 0 \Rightarrow \overline{C}_F^{i+1} = \overline{C}_F^i - \frac{f}{f'}$$

which becomes for this equation:

$$\overline{C}_F^{i+1} = \overline{C}_F^i \left[1 + \frac{\left\{ .242 - \sqrt{\overline{C}_F^i} \log(\overline{\text{Re}}_x \overline{C}_F^i) \right\}}{\left\{ .121 + \sqrt{\overline{C}_F^i} / \ln 10 \right\}} \right]$$

Once this iteration is completed, and $\overline{C}_F$ is known,

$$C_F = \frac{\overline{C}_F}{F_c}$$

Note that this value applies to one side of a plate only, so it must be doubled if the friction on both sides is desired to account for the proper reference areas. Here again, the results are not sensitive to the value of edge temperature for low Mach numbers, and the default value should be adequate for most cases. This formula is implemented in routine **turbcf**.

Composite formula

When the flow is laminar and then transitions to turbulent, an estimate of the skin friction is available from a composite of the laminar and turbulent skin friction formulas using Schlichting's formula (see T. Cebeci and P. Bradshaw, *Momentum Transfer in Boundary Layers*, McGraw-Hill, New York, 1977, pp. 187). Given the transition position, x_c/L and Re_L , compute

$$\text{Re}_c = \left(\frac{x_c}{L} \right) \text{Re}_L$$

and compute the laminar skin friction based on Re_c and the turbulent skin friction twice, based on both Reynolds numbers and then find the value that includes both laminar and turbulent flow from:

$$C_F = C_{F_{TURB}}(Re_L) - \left(\frac{x_c}{L}\right) \left[C_{F_{TURB}}(Re_c) - C_{F_{LAM}}(Re_c) \right]$$

Several formulas are available, are all roughly equivalent, and have been evaluated extensively for incompressible flow. They are only approximate for compressible flow.

Form factors

To include the effects of thickness, it has been found that the skin friction formulas should be adjusted through the use of form factors. Two different factors are used in this code. For wing-like shapes,

$$FF = 1.0 + 1.8\left(\frac{t}{c}\right) + 50\left(\frac{t}{c}\right)^4$$

where t/c is the thickness ratio of of particular component. For bodies,

$$FF = 1.0 + 1.5\left(\frac{d}{l}\right)^{1.5} + 50\left(\frac{d}{l}\right)^3$$

where d/l is the ratio of diameter to length. This is the reciprocal of the fineness ratio.

Program Operation:

Running the program, you will be prompted for the name of an input data set, the maximum length is 15 characters. The output is sent to the screen, but can be sent to a file by changing the value of IWRIT to something other than 6 in the main program. The sample data case on the disk is F15.FRCTION.

INPUT

<u>Card</u>	<u>Field</u>	<u>Columns</u>	<u>Variable</u>	<u>Description</u>
1	1	1-60		Title Card
2	1	1-10	SREF	Full Scale reference Area
	2	11-20	SCALE	1./SCALE, i.e. 1/10 scale is input as 10.
	3	21-30	FNCOMP	number of component cards to be read in (15 max).
	4	31-41	FINMD	input mode: = 0.0, input Mach and altitude = 1.0, input Mach and Reynolds No. per unit length
3	1	1-16	COMP(i)	Component Name
	2	21-30	SWET(I)	Wetted Area (<i>i.e.</i> , top and bottom sides of the wing, and both left and right sides, the <i>total</i> area that is exposed to the air)

3	31-40	REFL	Reference Length
4	41-50	TC(I)	t/c for planar surf. or d/l (1/F) for body of revolution
5	51-60	FICODE	Component type clue = 0.: Planar surface = 1.: Body of revolution
6	61-70	FTRANS	Transition location = 0. : means boundary layer is all turbulent = 1. : " " " " " laminar. values between 0 and 1 approximate the value of the friction of the laminar/turbulent boundary layer at the specified length fraction of the component.

Note: card 3 is repeated NCOMP times

<u>Card</u>	<u>Field</u>	<u>Columns</u>	<u>Variable</u>	<u>Description</u>
4	1	1-10	XME	Mach number
	2	11-20	XINPUT	if FINMD = 0.0, this is the Altitude (in 1000 feet) if FINMD = 1.0, this is the Reynolds no. per unit length in millions

Note: Card 4 is repeated for each value of Mach and altitude desired. The program stops when either the end of the data is reached or a Mach number of zero is read.

Output: The input is echoed to allow for easy check of data and to keep all information together. Then the drag calculation for each M, h or $M, Re/L$ is made. First, the reference areas, lengths, thicknesses, form factors and the transition position are output. These values are fixed for each combination of Mach and Reynolds number. Next, for each case the Reynolds number of each component and the basic skin friction are found. Then the skin friction times the wetted area and the skin friction times the wetted area and form factor are found. Finally, the latter is divided by the reference area and the contribution to the total drag in terms of a drag coefficient for the particular component, CDCOMP, is then found. These columns are summed, and the bottom value under the CDCOMP column is the total skin friction and form drag coefficient. After all the conditions are computed, a summary of results is presented as a table at the end of the output.

Sample input for program FRICTION:

```

      F - 15  AIRCRAFT
608.      1.      7.      0.0
FUSELAGE      550.00      54.65      .05500      1.0      0.0
CANOPY      75.00      15.0      .12000      1.0      0.0
NACELLE      600.00      35.0      .04000      1.0      0.0
GLV/SPONSON      305.00      35.5      .117      1.0      0.0
OUTB'D WING      698.00      12.7      .05000      0.0      0.0
HORIZ. TAIL      222.00      8.3      .05000      0.0      0.0
TWIN V. T.      250.00      6.7      .0450      0.0      0.0
      0.200      35.000
      1.200      35.000
      2.000      35.000
      0.000      0.000

```

D-38 Applied Computational Aerodynamics

Sample output from program friction:

Enter name of data set:
f15frict.inp

FRICTION - Skin Friction and Form Drag Program
W.H. Mason, Department of Aerospace and Ocean Engineering
Virginia Tech, Blacksburg, VA 24060 email: mason@aoe.vt.edu
version: September 13, 1996

CASE TITLE: F - 15 AIRCRAFT

SREF = 608.00000 MODEL SCALE = 1.000 NO. OF COMPONENTS = 7
input mode = 0 (mode=0: input M,h; mode=1: input M, Re/L)

COMPONENT TITLE	SWET (FT2)	REFL(FT)	TC	ICODE	FRM	FCTR	FTRANS
FUSELAGE	550.0000	54.650	0.055	1	1.0205	0.0000	
CANOPY	75.0000	15.000	0.120	1	1.0744	0.0000	
NACELLE	600.0000	35.000	0.040	1	1.0124	0.0000	
GLV/SPONSON	305.0000	35.500	0.117	1	1.0712	0.0000	
OUTB'D WING	698.0000	12.700	0.050	0	1.0903	0.0000	
HORIZ. TAIL	222.0000	8.300	0.050	0	1.0903	0.0000	
TWIN V. T.	250.0000	6.700	0.045	0	1.0812	0.0000	

TOTAL SWET = 2700.0000

REYNOLDS NO./FT = 0.480E+06 Altitude = 35000.00 XME = 0.200

COMPONENT	RN	CF	CF*SWET	CF*SWET*FF	CDCOMP
FUSELAGE	0.262E+08	0.00251	1.38212	1.41047	0.00232
CANOPY	0.720E+07	0.00309	0.23164	0.24889	0.00041
NACELLE	0.168E+08	0.00269	1.61561	1.63573	0.00269
GLV/SPONSON	0.170E+08	0.00269	0.81944	0.87782	0.00144
OUTB'D WING	0.609E+07	0.00318	2.21681	2.41701	0.00398
HORIZ. TAIL	0.398E+07	0.00342	0.75829	0.82678	0.00136
TWIN V. T.	0.321E+07	0.00355	0.88656	0.95855	0.00158
		SUM =	7.91048	8.37525	0.01378

FRICTION DRAG: CDF = 0.01301

FORM DRAG: CDFORM = 0.00076

REYNOLDS NO./FT = 0.288E+07 Altitude = 35000.00 XME = 1.200

COMPONENT	RN	CF	CF*SWET	CF*SWET*FF	CDCOMP
FUSELAGE	0.157E+09	0.00175	0.96201	0.98175	0.00161
CANOPY	0.432E+08	0.00211	0.15826	0.17004	0.00028
NACELLE	0.101E+09	0.00186	1.11769	1.13160	0.00186
GLV/SPONSON	0.102E+09	0.00186	0.56700	0.60740	0.00100
OUTB'D WING	0.366E+08	0.00216	1.51055	1.64698	0.00271
HORIZ. TAIL	0.239E+08	0.00231	0.51314	0.55949	0.00092
TWIN V. T.	0.193E+08	0.00239	0.59777	0.64631	0.00106
		SUM =	5.42643	5.74356	0.00945

FRICTION DRAG: CDF = 0.00893

FORM DRAG: CDFORM = 0.00052

REYNOLDS NO./FT =0.480E+07 Altitude = 35000.00 XME = 2.000

COMPONENT	RN	CF	CF*SWET	CF*SWET*FF	CDCOMP
FUSELAGE	0.262E+09	0.00140	0.76912	0.78490	0.00129
CANOPY	0.720E+08	0.00169	0.12643	0.13585	0.00022
NACELLE	0.168E+09	0.00149	0.89337	0.90449	0.00149
GLV/SPONSON	0.170E+09	0.00149	0.45321	0.48550	0.00080
OUTB'D WING	0.609E+08	0.00173	1.20667	1.31564	0.00216
HORIZ. TAIL	0.398E+08	0.00185	0.40980	0.44681	0.00073
TWIN V. T.	0.321E+08	0.00191	0.47731	0.51607	0.00085
		SUM =	4.33591	4.58926	0.00755

FRICTION DRAG: CDF = 0.00713

FORM DRAG: CDFORM = 0.00042

SUMMARY

J	XME	Altitude	RE/FT	CDF	CDFORM	CDF+CDFORM
1	0.200	0.350E+05	0.480E+06	0.01301	0.00076	0.01378
2	1.200	0.350E+05	0.288E+07	0.00893	0.00052	0.00945
3	2.000	0.350E+05	0.480E+07	0.00713	0.00042	0.00755

END OF CASE

STOP

D.6 VLMpc

This manual describes the input for the pc version of John Lamar's vortex lattice program. This program is identical to the program given in reference 2. An advanced version described in reference 4 is also available. The input data sets differ slightly between the two versions.

The code is called **VLMpcv2.f** on the disk, and has been modified for WATFOR. This means that the output field length is limited to eighty columns. In WATFOR you may also need to invoke the NOCHECK option to prevent the program from halting because of undefined variables. The code is provided with two OPEN statements near the beginning of the main program:

```
OPEN(5,file=infile, status=old)
OPEN(6,file=outfil, status=new)
```

such that the input data is defined on the file infile, and the output is placed in file outfil. The user is prompted for the names of these files at the start of execution. Users should customize the code to fit their preferences. The disk also contains a sample input file, YF23.IN, and a sample output file, YF23.OUT.

The theory is described in references 1, 2 and 3, and the user's manual provided here is basically the instructions from references 1 and 2, with minor corrections and clarifications. Reference 4 describes the advanced version, VLM4.997.

References:

1. Margason, R.J., and Lamar, J.E., "Vortex-Lattice FORTRAN Program for Estimating Subsonic Aerodynamic Characteristics of Complex Planforms," NASA TN D-6142, Feb., 1971.
2. Lamar, J.E., and Gloss, B. B. "Subsonic Aerodynamic Characteristics of Interacting Lifting Surfaces with Separated Flow around Sharp Edges Predicted by a Vortex-Lattice Method," NASA TN D-7921, Sept., 1975.
3. Lamar, J.E., and Frink, N.T., "Experimental and Analytic Study of the Longitudinal Aerodynamic Characteristics of Analytically and Empirically Designed Strake-Wing Configurations at Subcritical Speeds, " NASA TP-1803, June 1981.
4. Lamar, J.E., and Herbert, H.E., "Production Version of the Extended NASA-Langley Vortex Lattice FORTRAN Computer Code," - Vol. I - User's Guide, NASA TM 83303, April 1982.

VLMpc User's Guide- (from references 1, 2, and 4)

This manual contains the output details for the pc version of the NASA-Langley Vortex Lattice Computer Program described in reference 2. The NASA - Langley Vortex Lattice FORTRAN Program (VLMpc) is designed to estimate the subsonic aerodynamic characteristics of up to two complex planforms. The concepts embodied in this program are mostly detailed in references 1 and 2.

MODELING THE CONFIGURATION

The configuration can be modeled with up to two planforms, all of which must extend to the plane of symmetry ($Y = 0.0$). The fuselage is represented by its planar projection; experience to date indicates that this produces acceptable global forces and moments for most wing-body-tail configurations.

Winglets can be modeled, but the dihedral angle must be less than 90.0 degrees and greater than -90.0 degrees. Both upper (positive dihedral) and lower (negative dihedral) winglets can be accounted for in this code. The program uses as its solution surface the chord plane which may

be inclined due to dihedral. Moreover, the only out of "X-Y plane" displacement specifically allowed for is dihedral. Local camber and twist is assumed to be small and can be represented by its slope projection to the local solution surface. The wind and body axes are assumed to be coincidental in the code.

RUNNING THE PROGRAM

INPUT DATA SETUP

The input data to VLM is organized into two distinct groups - group 1 defines the reference planform(s), and group 2 defines the details for the particular solution. An example input follows the description of the input and output. The individual details of the items in the deck layout are given in the following sections.

GROUP 1 DATA

This group of data defines the planform(s) projected into the X-Y plane, with all the coordinates being given for the left half of the configuration (negative y values!). The axis system is shown in Figure 1. The $Y = 0$ intercept coincides with the plane of symmetry and is positive to the right of this plane. The $X = 0$ intercept is taken to occur along the symmetry plane of the configuration; X is positive pointing into the wind.

Important tips for modeling configurations:

Good results require that a few common rules of thumb be used in selecting the planform break points. The number of line segments should be minimized. Breakpoints should line up streamwise on front and rear portions of each planform, and should line up between planforms. Streamwise tips should be used, and small spanwise distances should be avoided by making edges streamwise if they are actually very highly swept.

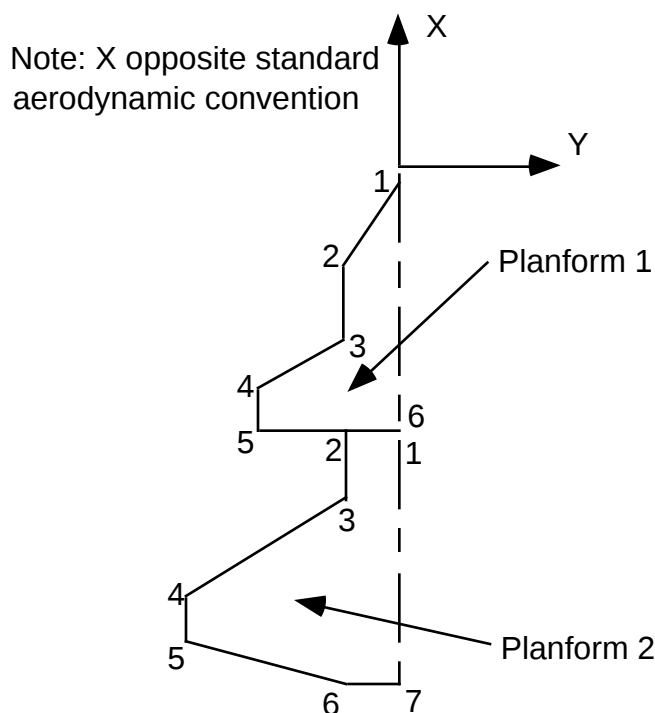


Figure 1. Definition of axis system for VLMpc.

It is important to note that each planform can only go out to a maximum y value once, and then return to the centerline. The program assumes that each planform is actually a wing. Most numerical input for the group 1 data uses an 8F10.6 format. The input is as follows:

0. (Cols. 1-80) Title Card
1. (Cols. 1-10) PLAN - Number of planforms for this configuration; PLAN can assume values of 1.0, or 2.0.
2. (Cols.11-20) TOTAL - Number of sets of group 2 data specified for this configuration.
3. (Cols.21-30) CREF - Reference chord of the configuration. This chord is used only to nondimensionalize the pitching-moment terms and must be greater than zero.
4. (Cols.31-40) SREF - Reference area; this is used only to nondimensionalize the computed output data such as lift and pitching moment and must always be greater than zero.
5. (Col. 41-50) CG - Center of gravity location with respect to the origin of the coordinate system. All moment computations are referenced to this location.

The data required to define the planform(s) is provided in the next set of group 1 cards as follows (the number of line segments is equal to the number of points minus one):

1. (Cols.1-10) AAN - Number of line segments used to define the left half of the planform (does not include the innermost streamwise). A maximum of 24 line segments may be used per planform, and each planform must extend to the plane of symmetry. *AAN is the number of defining point minus one.*
2. (Cols.11-20) XS - X location of the pivot; use 0.0 for a fixed planform.
3. (Cols.21-30) YS - Y location of the pivot; use 0.0 for a fixed planform.
4. (Cols.31-40) RTCDHT - Vertical distance of the particular planform being read in with respect to the reference planform root chord height; use 0. for the reference planform.

The rest of this set of data describes the breakpoints used to define the AAN line segments on this planform. The format is 4F9.4. There are (AAN+1) breakpoints and all data subsequently described are required on all except the last card of this set; the last card uses only the first two variables in the following list:

1. (Cols.1-9) XREG(I) - X location of the i th breakpoint. The first breakpoint is located at the most inboard location of the leading edge for the left-hand side of this planform. The other breakpoints are numbered around the planform perimeter in increasing order for each intersection of lines in a counterclockwise direction.
2. (Cols.10-18) YREG(I) - Y location of the i th breakpoint. Once the absolute value of Y starts to decrease, it cannot be increased.
3. (Cols.19-27) DIH(I) - Dihedral angle (degrees) in the Y-Z plane of the line from breakpoint of i to $i+1$, positive upward. Note that along a streamwise line, the dihedral angle is not defined, so use 0.0. for these lines. Note the sign of the dihedral angle is the same along the leading and trailing edges.
4. (Cols.28-36) AMCD - The move code; this number indicates whether the line s is on the movable panel of a variable-sweep wing. Use 1.0 for a fixed line (defaults to 1.0 if not set), or 2.0 for a movable line.

GROUP 2 DATA

There are four sections of group 2 data. Each section may be required or optional, depending on the previous input, and each may have one or more input cards (lines of input). Each section is described individually. Care must be taken to make sure the data is in the proper column.

Section one data (always required).

[1 Card - Format (8F5.2, F10.4,F5.1,F10.4,F5.1)]

1. (Cols.1-5) CONFIG - An arbitrary configuration designation of up to 4 digits.
2. (Cols.6-10) SCW - The number of chordwise horseshoe vortices to be used at a spanwise station for each planform. The maximum value for this variable is 20. If varying values of chordwise horseshoe vortices are desired due to either multiple planforms or large discontinuities in chord across the span, the user can input a value of 0. that will cause the program to expect user-supplied data at this point in the input stream. The data are in the form of a table that contains the number of chordwise horseshoe vortices from the tip to root, and is called TBLSCW(I). This SCW=0. option can only be used for planforms without dihedral and for coplanar configurations.
3. (Cols.11-15) VIC - The nominal number of spanwise stations at which chordwise horseshoe vortices will be located. This variable must not cause more than 50 spanwise stations to be used by the program in describing the left half of the configuration. In addition, the product of the stations spanwise and SCW cannot exceed 200. If SCW is 0., then the sum of the values in TBLSCW(I) cannot exceed 200. The use of variable VIC is discussed in references 1 and 2. VIC should always be greater than, or equal to, 10. so that the near-field drag or vortex flow forces on cambered configurations can be properly computed.
4. (Cols.16-20) MACH - Mach number; use a value other than 0.0 only if the Prandtl-Glauert compressibility correction factor is to be applied. The value used should be less than that of the critical Mach number.
5. (Cols.21-25) CLDES - Desired lift coefficient, CL_d . The number specified here is used to obtain the span load distribution at a particular lift coefficient. If the drag polar is required over a CL range from -0.1 to 1.0, use CLDES = 11.
6. (Cols.26-30) PTEST - C_{lp} indicator; if the damping-in-roll parameter is desired, use 1.0 for this quantity. Except for the Delta C_p and C_{lp} , all other aerodynamic data will be omitted. Use a 0. if C_{lp} is not required. The definition is the standard one, as in Etkin, with units of radians per second:

$$C_{lp} = \frac{\partial C_l}{\partial \left(\frac{pb}{2U_\infty} \right)}$$

7. (Cols.31-35) QTEST - C_{Lq} and C_{mq} indicator; if these stability derivatives are desired, use a 1.0 for this quantity. Except for Delta C_p , C_{Lq} , and C_{mq} , all other aerodynamic data will be omitted. It should be noted that both PTEST and QTEST cannot be set equal to 1. simultaneously for a particular configuration. Use 0. if C_{Lq} and C_{mq} are not required. The definition is the standard one, as in Etkin:

$$C_{m_q} = \frac{\partial C_m}{\partial \left(\frac{qc_{ref}}{2U_\infty} \right)}, \quad C_{L_q} = \frac{\partial C_L}{\partial \left(\frac{qc_{ref}}{2U_\infty} \right)}.$$

8. (Cols. 36-40) TWIST(1) - Twist code for the first planform. If this planform has no twist and/or camber, use a value of 0.; otherwise, specify a value of 1.
9. (Cols.41-50) SA(1) - Variable sweep angle for the first planform. Specify the leading edge sweep-angle (in degrees) for the first movable line adjacent to the fixed portion of the planform. For a fixed planform, this quantity may be omitted.
10. (Cols.51-55) TWIST(2) - same, for the second planform.
11. (Cols.56-65) SA(2) - same, for the second planform.
12. (Cols.66-70) ATPCOD - Set to 0., it will cause only linear aerodynamic results to be printed out. Set to 1., this will cause the program to print out the contributions to the lift, drag and moment from the separated flow around the leading/side edges.

Section two data is required when ATPCOD=1.* This section sets up the limits of integration used in the computations of the wing leading-edge and side-edge suction values. Normally these limits would be the wing root and the wing tip. However, other values could be used. Note: if section four data is used, this data may come after section four data - experiment if you try to use this combination.

[1 Card - Format (4F10.6)]

Card 1:

1. (Cols.1-10) YINNER(1) - Represents the Y inner for the first planform.
2. (Cols.11-20) YOUTER(1) - Represents the Y outer for the first planform.
3. (Cols.21-30) YINNER(2) - Represents the Y inner for the second planform.
4. (Cols. 31-40), YOUTER(2) - Represents the Y outer for the second planform.

Section three data is required when SCW=0. This section determines the number of span stations for each planform, and the number of chordwise control points along each span station. This option is rarely used.

[Multiple card sets per planform - Format (F5.1,n(/16F5.1))]

Card 1:

(Cols.1-5) STA - Number of spanwise stations of horseshoe vortices on the left half of the planform. This variable sets the number of TBLSCW values read in for that planform.

* Watch out about the order of input if both twist and vortex lift are used. Some students have reported problems with this. Actually, this is a somewhat rare calculation. Both twist and vortex lift should be run separately to the user's satisfaction before they are run together.

Cards 2-n:

(Cols. 1-5,6-10,etc) TBLSCW(I) - Number of horseshoe vortices at each spanwise station beginning at the station nearest the tip of the planform and proceeding toward the station nearest the root.

These sets of STA and TBLSCW(I) cards are repeated for each planform. The sum of all the STA values cannot exceed 100.

Section four data is required for any planform having a nonzero value for TWIST(I). This section determines the mean camber line slopes or angles of attack across the planform. Be careful here. Experience has shown that students find the proper input of this data to be very tricky.

[Multiple cards per planform - Format (8F10.6,n(8F10.6))]

(Cols.1-10,11-20,etc.) ALP - Local streamwise angles of attack, eg. camber or flap deflection, in radians. These are the values at the control point for each horseshoe vortex on the planform when the innermost streamwise edge of the reference planform has an angle of attack of 0. degrees. The volume of this data will usually require several input cards. For the first value on the first card, use the local angle of attack for the horseshoe vortex nearest the first planform leading edge at the tip; for the second value, use the angle of attack for the horseshoe vortex immediately behind in the chordwise direction. Continue in the same manner for the rest of the horseshoe vortices at the tip. *Begin a new card for the next inboard station* and input the data in the same chordwise manner. Repeat for all successive inboard spanwise stations on that planform. For each planform with twist/camber, start the data on a new card and specify the data from the tip and proceed chordwise and then inboard, as detailed above.

OUTPUT DATA

The printed results of this computer program appear in two parts: geometry data and aerodynamic data.

GEOMETRY DATA

The geometry data are described in the order that they are found on the printout.

The first group of the data describes the basic configuration: it states the numbers of lines used to describe each planform, the root chord height, pivot position, and then lists the breakpoints, sweep and dihedral angles, and move codes. These data are basically a listing of input data except that the sweep angle is computed from the input.

The second group of data describes the particular configuration for which the aerodynamic data are being computed. Included are the configuration designation, sweep position, a listing of the breakpoints of the planform (X,Y, and Z), the sweep and dihedral angles, and the move codes. The data are listed primarily for variable-sweep wings to provide a definition of the planform where the outer panel sweep is different from that of the reference planform. The number of horseshoe vortices are then described. In this code a maximum of 200 vortices can be used.

The third group of data presents a detailed description of the horseshoe vortices used to represent the configuration. These data are listed in two sets of five columns each describing one elemental panel of the configuration (see Figure 2) in the same order that the twist and/or camber angles of attack are to be provided.

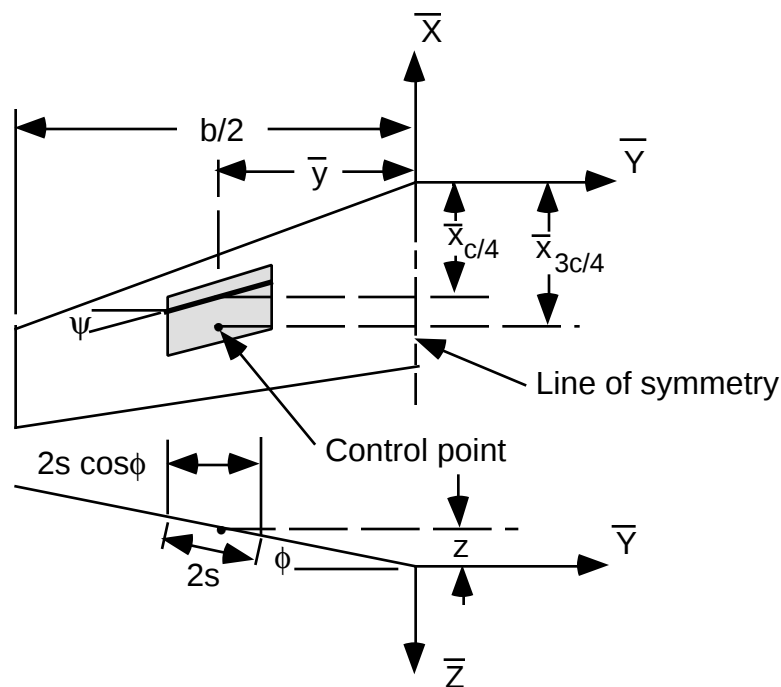


Figure 2. Nomenclature used to describe the geometry of an elemental panel.

The following items of data are presented for each elemental panel.

For set one:

1. $X_{C/4}$ - X location of quarter-chord at the horseshoe vortex midspan.
2. $X_{3C/4}$ - X location of three-quarter-chord at the horseshoe vortex midspan. This is the X location of the control point.
3. Y - Y location of the horseshoe vortex midspan.
4. Z - Z location of the horseshoe vortex midspan.
5. S - Semiwidth of horseshoe vortex.

Set two:

1. $X_{C/4}$ - X location of quarter-chord at the horseshoe vortex midspan. (same as set one)
2. C/4 SWEEP ANGLE - Sweep angle of the quarter-chord of the elemental panel and horseshoe vortex.
3. DIHEDRAL ANGLE - Dihedral angle of elemental panel.
4. LOCAL ALPHA IN RADIANS - Local angle of attack in radians at control point ($X @ X_{3C/4}, Y, Z$).

5. DELTA CP AT DESIRED CL - ΔC_p or Net C_p normal to the surface at dihedral for each elemental panel when the total lift is $C_{L,d}$. This is located across the panel as an average. It corresponds to the incremental lift associated with the bound vortex strength of the particular panel:

$$\Delta C_p \times \Delta c = \Delta L_i, \quad \text{where} \quad \Delta L_i = \rho U_\infty \Gamma_i$$

The fourth group of data presents the following geometric results:

1. REF.CHORD - Reference chord of the configuration.
2. C AVERAGE - Average chord, cav, true configuration area divided by true span.
3. TRUE AREA - True area computed from the configuration listed in second group of geometry data.
4. REFERENCE AREA - User input reference area.
5. B/2 - Maximum semispan of all planforms listed in second group of geometry data.
6. REF. AR - Reference aspect ratio computed from the reference planform area and true span.
7. TRUE AR - True aspect ratio computed from the true planform area and true span.
8. MACH NUMBER - Mach number.

AERODYNAMIC DATA

If PTEST = 1. or QTEST = 1. on the configuration card, then either C_{l_p} or C_{L_q} and C_{m_q} are computed and printed, followed by program termination. Otherwise, the aerodynamic data are described by at least two groups of results. The first is always present, but the second depends on what is requested on the configuration card. The following items of the first group of data are given in the order that they are found on the printout. Note that CL ALPHA, CL(TWIST), CM/CL, CMO, CDI/CL**2 are based on the specified reference dimensions. Many of the items that follow are for the complete configuration.

1. DESIRED CL - Desired lift coefficient, CL , d , specified in Input Data for complete configuration.
2. COMPUTED ALPHA - Angle of attack at which the desired lift is developed: $CL, d/(CL \text{ ALPHA}) + \text{ALPHA}$ at $CL=0$.
3. CL(WB) - That portion of desired lift coefficient developed by the planform with the maximum span when multiple planforms are specified. When one planform is specified, this is the desired lift coefficient. (If two planforms have the same span, and this value is equal to the maximum, the planform used here is the latter one read in).
4. CDI AT CL(WB) - Induced drag coefficient for lift coefficient in the previous item. When two or more planforms are specified, this is the induced drag coefficient of only the planform with the maximum span. This result is based on the far-field solution.
5. CDI/(CL(WB)**2) - Induced drag parameter computed from the two previous items.

6. $1/(\pi \cdot AR_{REF})$ - Induced drag parameter for an elliptic load distribution based on reference aspect ratio.
7. CL ALPHA - Lift-curve slope per radian, and per degree.
8. CL(TWIST) - Lift coefficient due to twist and/or camber at zero angle of attack (CL,tc).
9. ALPHA AT CL=0 - Angle of attack at zero lift in degrees; nonzero only when twist and/or camber is specified.
10. Y CP - Spanwise distance in fraction of semispan from root chord to center of pressure on the left wing panel.
11. CM/CL - Longitudinal stability parameter based on a moment center about the reference point. This is the negative of the static margin:

$$CM / CL_{Lamar} = \frac{\partial C_m}{\partial C_L}$$

and the value of C_{m_α} can be found from $C_{m_\alpha} = \left(\frac{\partial C_m}{\partial C_L} \right) \left(\frac{\partial C_L}{\partial \alpha} \right)$.

12. CMO - Pitching-moment coefficient at CL=0.

For each spanwise station, the following data are presented; from the left tip towards the root:

1. 2Y/B - Location of midpoint of each spanwise station in fraction of wing semispan.

The next two columns of data describe the additional (or angle of attack) wing loading at a lift coefficient of 1. (based on the total lift achieved and the true configuration area). The third column is the chord ratio result, and the other columns detail specific kinds of span loadings and local centers of pressure for the configuration.

2. SL COEF - span-load coefficient, $c_l c / C_L c_{av}$.
3. CL RATIO - Ratio of local lift to total lift, c_l / C_L .
4. C RATIO - Ratio of local chord to average chord, c / c_{av} .
5. LOAD DUE TO TWIST - Distribution of span-load coefficient due to twist and camber at 0° angle of attack for the configuration.
6. ADD. LOAD AT CL= - Distribution of additional span-load coefficient required to produce zero lift when combined with lift due to twist and camber. This distribution is computed at $C_{L,tc}$.
7. BASIC LOAD AT CL=0 - Basic span-load-coefficient distribution at zero lift coefficient. These data are the difference of the previous two columns of data.
8. SPAN LOAD AT DESIRED CL - Distribution of the combination of the basic span-load and additional span-load coefficients at the desired C_L .
9. AT CL DES - X LOCATION OF LOCAL CENT PR - The X location of the local center of pressure for the resulting span load at $C_{L,d}$ as a function of 2Y/b.

The other options available as group two aerodynamic data are accessed based on the values of CLDES and ATPCOD. For instance, with CLDES=11., and ATPCOD=0.0, the program will produce a drag polar, CDI at CL(WB) versus CL(WB), based on the linear aerodynamics in the middle of the first part of group one aerodynamic data. This, and other combinations, are given in the table below, along with their purposes:

Next, the induced drag, leading-edge thrust, and suction coefficient characteristics at each spanwise station are computed from a near-field solution for the total loading at CL,d and presented.

1. 2y/b - the spanwise location for these results
2. L.E. SWEEP ANGLE - Leading-edge sweep angle in degrees.
3. CDII C/2B - Nondimensional section induced-drag-coefficient term.
4. CT C/2B - Nondimensional section leading-edge thrust-coefficient term.
5. CS C/2B - Nondimensional section leading-edge suction in coefficient term.

Next, the total coefficients are given:

CDII/CL**2 - Total drag coefficient over (CL,d)**2.
 CT - Total leading-edge thrust coefficient.
 CS - Total leading-edge suction coefficient.

Additional printout is produced for vortex flows. In particular, Kp and Kv values, and respective centroids in both chordwise and spanwise directions, and the associated limits of integration for the leading- edge and side-edge values of Kv. (The item entitled "Sum of the positive side edge contributions" which appears here on the printout is indicative of the contribution to the side-edge forces for that particular planform which were oppositely-signed to those that contributed in a manner to increase Kv,se. The value of Kv,se does contain these positive contributions provided the sweep angle is positive. They should not be, and therefore are not added in for the planform with a swept forward leading edge). Furthermore, aerodynamic performance values for each planform and for the entire configuration will be listed over an angle of attack range by the use of the Polhamus Suction Analogy. The headings are explained below: *See the references for detailed explanations of these terms.*

KP	Kp
KVLE	Kv,le
KV SE	Kv,se
ALPHA	α
CN	CN,tot
CLP	CL,p
CLVLE	CL,vle
CLVSE	$Kv,se \sin\alpha \sin\alpha \cos \alpha$
CMP	pitching-moment coefficient due to CL,p
CMVLE	pitching-moment coefficient due to CL,vle
CMVSE	pitching-moment coefficient due to CL,vse
CM	total pitching moment
CD	$CL,tot \times \tan\alpha$
$CL^{**2}/(PI*AR)$	$(CL,tot)^2 / (PI*(Aspect Ratio))$

SAMPLE INPUT, - as developed by Bob Narducci to investigate the YF-23.

YF-23 Flaps Down

```

2.      1.      26.8917  950.0      0.0
6.0     0.      0.      0.
 37.80  0.0     0.0     1.
 22.73  -4.35   0.      1.
 14.69  -4.35   0.      1.
 00.11  -21.75  0.      1.
-03.24  -21.75  0.      1.
-14.96  -7.86   0.      1.
-14.96  0.      0.      0.
8.      0.      0.      0.
-14.96  0.      0.      1.
-14.96  -7.86   43.     1.
-22.00  -16.90  0.      1.
-24.51  -16.90  43.     1.
-29.50  -10.71  43.     1.
-27.02  -7.86   0.      1.
-28.36  -6.86   0.      1.
-25.68  -3.85   0.      1.
-29.20  0.
23. 6.  13.  .30  .53  0.  0.  0.  0.  1.  0.  0.
.1745  .1745  .1745  .1745  .1745  .1745
.1745  .1745  .1745  .1745  .1745  .1745
.1745  .1745  .1745  .1745  .1745  .1745
.1745  .1745  .1745  .1745  .1745  .1745
.1745  .1745  .1745  .1745  .1745  .1745
.1745  .1745  .1745  .1745  .1745  .1745
.1745  .1745  .1745  .1745  .1745  .1745
.0000  .0000  .0000  .0000  .0000  .0000
.0000  .0000  .0000  .0000  .0000  .0000
.0000  .0000  .0000  .0000  .0000  .0000
.0000  .0000  .0000  .0000  .0000  .0000
.0000  .0000  .0000  .0000  .0000  .0000
.0000  .0000  .0000  .0000  .0000  .0000

```

SAMPLE OUTPUT: The output is lengthy, but included here to help students check their codes.

This is what shows up on the screen:

```

enter name of data set:yf23.in
enter name of output file: yf23out.manual

all output is routed to disk file
computing may take quite some time

```

STOP

The output file yf23out.manual is:

vortex lattice aerodynamic computation program
nasa-lrc no. a2794 by j.e. lamar and b.b. gloss

modified for watfor77 with 72 column output

YF-23 Flaps Down

geometry data

first reference planform has 6 curves

center of gravity = 0.00000
root chord height = 0.00000
variable sweep pivot position x(s) = 0.00000 y(s) = 0.00000

break points for the reference planform

point	x ref	y ref	sweep angle	dihedral angle	move code
1	37.80000	0.00000	73.89906	0.00000	1
2	22.73000	-4.35000	90.00000	0.00000	1
3	14.69000	-4.35000	39.96069	0.00000	1
4	0.11000	-21.75000	90.00000	0.00000	1
5	-3.24000	-21.75000	-40.15675	0.00000	1
6	-14.96000	-7.86000	0.00000	0.00000	1
7	-14.96000	0.00000			

second reference planform has 8 curves

center of gravity = 0.00000
root chord height = 0.00000
variable sweep pivot position x(s) = 0.00000 y(s) = 0.00000

break points for the reference planform

point	x ref	y ref	sweep angle	dihedral angle	move code
1	-14.96000	0.00000	0.00000	0.00000	1
2	-14.96000	-7.86000	37.91007	43.00000	1
3	-22.00000	-16.90000	90.00000	0.00000	1
4	-24.51000	-16.90000	-38.87364	43.00000	1
5	-29.50000	-10.71000	41.02898	43.00000	1
6	-27.02000	-7.86000	-53.26718	0.00000	1
7	-28.36000	-6.86000	41.68077	0.00000	1
8	-25.68000	-3.85000	-42.43623	0.00000	1
9	-29.20000	0.00000			

1

configuration no. 23.

curve 1 is swept 73.89906 degrees on planform 1

curve 1 is swept 0.00000 degrees on planform 2

break points for this configuration

point	x	y	z	sweep angle	dihedral angle	move code
1	37.80000	0.00000	0.00000	73.89906	0.00000	1
2	24.46218	-3.85000	0.00000	73.89906	0.00000	1
3	22.73000	-4.35000	0.00000	90.00000	0.00000	1
4	14.69000	-4.35000	0.00000	39.96069	0.00000	1
5	12.58679	-6.86000	0.00000	39.96069	0.00000	1
6	9.36076	-10.71000	0.00000	39.96069	0.00000	1
7	4.17397	-16.90000	0.00000	39.96069	0.00000	1
8	0.11000	-21.75000	0.00000	90.00000	0.00000	1
9	-3.24000	-21.75000	0.00000	-40.15676	0.00000	1
10	-14.96000	-7.86000	0.00000	0.00000	0.00000	1
11	-14.96000	0.00000	0.00000			

second planform breakpoints

1	-14.96000	0.00000	0.00000	0.00000	0.00000	1
2	-14.96000	-4.35000	0.00000	0.00000	0.00000	1
3	-14.96000	-7.86000	0.00000	37.91007	43.00000	1
4	-22.00000	-16.90000	-8.42994	90.00000	0.00000	1
5	-24.51000	-16.90000	-8.42994	-38.87364	43.00000	1
6	-29.50000	-10.71000	-2.65767	41.02898	43.00000	1
7	-27.02000	-7.86000	0.00000	-53.26718	0.00000	1
8	-28.36000	-6.86000	0.00000	41.68077	0.00000	1
9	-25.68000	-3.85000	0.00000	-42.43624	0.00000	1
10	-29.20000	0.00000	0.00000			

168 horseshoe vortices used on the left half of the configuration

planform	total	spanwise
1	90	15
2	78	13

6. horseshoe vortices in each chordwise row

1

aerodynamic data

configuration no. 23.

static longitudinal aerodynamic coefficients are computed

panel no.	x c/4	x 3c/4	y	z	s
1	0.61276	0.21636	-20.91346	0.00000	0.83654
2	-0.18004	-0.57644	-20.91346	0.00000	0.83654
3	-0.97284	-1.36924	-20.91346	0.00000	0.83654
4	-1.76564	-2.16204	-20.91346	0.00000	0.83654
5	-2.55845	-2.95485	-20.91346	0.00000	0.83654
6	-3.35125	-3.74765	-20.91346	0.00000	0.83654
7	1.89745	1.26658	-19.24039	0.00000	0.83654

8	0.63571	0.00484	-19.24039	0.00000	0.83654
9	-0.62603	-1.25690	-19.24039	0.00000	0.83654
10	-1.88776	-2.51863	-19.24039	0.00000	0.83654
11	-3.14950	-3.78037	-19.24039	0.00000	0.83654
12	-4.41124	-5.04211	-19.24039	0.00000	0.83654
13	3.11717	2.26369	-17.65192	0.00000	0.75192
14	1.41021	0.55673	-17.65192	0.00000	0.75192
15	-0.29675	-1.15023	-17.65192	0.00000	0.75192
16	-2.00371	-2.85719	-17.65192	0.00000	0.75192
17	-3.71067	-4.56415	-17.65192	0.00000	0.75192
18	-5.41763	-6.27110	-17.65192	0.00000	0.75192
19	4.33688	3.26079	-16.06346	0.00000	0.83654
20	2.18470	1.10861	-16.06346	0.00000	0.83654
21	0.03253	-1.04356	-16.06346	0.00000	0.83654
22	-2.11965	-3.19574	-16.06346	0.00000	0.83654
23	-4.27183	-5.34792	-16.06346	0.00000	0.83654
24	-6.42401	-7.50010	-16.06346	0.00000	0.83654
25	5.62157	4.31101	-14.39038	0.00000	0.83654
26	3.00046	1.68990	-14.39038	0.00000	0.83654
27	0.37934	-0.93122	-14.39038	0.00000	0.83654
28	-2.24177	-3.55233	-14.39038	0.00000	0.83654
29	-4.86289	-6.17345	-14.39038	0.00000	0.83654
30	-7.48401	-8.79456	-14.39038	0.00000	0.83654
31	6.90626	5.36123	-12.71731	0.00000	0.83654
32	3.81621	2.27118	-12.71731	0.00000	0.83654
33	0.72616	-0.81887	-12.71731	0.00000	0.83654
34	-2.36390	-3.90892	-12.71731	0.00000	0.83654
35	-5.45395	-6.99897	-12.71731	0.00000	0.83654
36	-8.54400	-10.08903	-12.71731	0.00000	0.83654
37	7.99810	6.25380	-11.29539	0.00000	0.58538
38	4.50950	2.76521	-11.29539	0.00000	0.58538
39	1.02091	-0.72339	-11.29539	0.00000	0.58538
40	-2.46768	-4.21198	-11.29539	0.00000	0.58538
41	-5.95628	-7.70058	-11.29539	0.00000	0.58538
42	-9.44487	-11.18917	-11.29539	0.00000	0.58538
43	9.08994	7.14637	-9.87346	0.00000	0.83654
44	5.20280	3.25923	-9.87346	0.00000	0.83654
45	1.31566	-0.62790	-9.87346	0.00000	0.83654
46	-2.57147	-4.51504	-9.87346	0.00000	0.83654
47	-6.45861	-8.40218	-9.87346	0.00000	0.83654
48	-10.34575	-12.28931	-9.87346	0.00000	0.83654
49	10.18414	8.04087	-8.44846	0.00000	0.58846
50	5.89760	3.75432	-8.44846	0.00000	0.58846
51	1.61105	-0.53222	-8.44846	0.00000	0.58846
52	-2.67549	-4.81876	-8.44846	0.00000	0.58846
53	-6.96203	-9.10530	-8.44846	0.00000	0.58846
54	-11.24857	-13.39184	-8.44846	0.00000	0.58846
55	11.03750	8.77685	-7.36000	0.00000	0.50000
56	6.51620	4.25554	-7.36000	0.00000	0.50000
57	1.99489	-0.26576	-7.36000	0.00000	0.50000
58	-2.52641	-4.78706	-7.36000	0.00000	0.50000
59	-7.04772	-9.30837	-7.36000	0.00000	0.50000
60	-11.56902	-13.82967	-7.36000	0.00000	0.50000
61	12.11076	9.75678	-6.02346	0.00000	0.83654
62	7.40281	5.04883	-6.02346	0.00000	0.83654
63	2.69485	0.34087	-6.02346	0.00000	0.83654
64	-2.01311	-4.36709	-6.02346	0.00000	0.83654
65	-6.72107	-9.07505	-6.02346	0.00000	0.83654

66	-11.42903	-13.78301	-6.02346	0.00000	0.83654
67	13.11855	10.67694	-4.76846	0.00000	0.41846
68	8.23532	5.79371	-4.76846	0.00000	0.41846
69	3.35210	0.91049	-4.76846	0.00000	0.41846
70	-1.53113	-3.97274	-4.76846	0.00000	0.41846
71	-6.41435	-8.85597	-4.76846	0.00000	0.41846
72	-11.29758	-13.73919	-4.76846	0.00000	0.41846
73	21.98959	18.77658	-4.10000	0.00000	0.25000
74	15.56357	12.35056	-4.10000	0.00000	0.25000
75	9.13756	5.92455	-4.10000	0.00000	0.25000
76	2.71154	-0.50147	-4.10000	0.00000	0.25000
77	-3.71447	-6.92748	-4.10000	0.00000	0.25000
78	-10.14049	-13.35350	-4.10000	0.00000	0.25000
79	25.59691	22.07023	-3.01346	0.00000	0.83654
80	18.54354	15.01685	-3.01346	0.00000	0.83654
81	11.49016	7.96347	-3.01346	0.00000	0.83654
82	4.43679	0.91010	-3.01346	0.00000	0.83654
83	-2.61659	-6.14328	-3.01346	0.00000	0.83654
84	-9.66997	-13.19666	-3.01346	0.00000	0.83654
85	31.98795	27.90552	-1.08846	0.00000	1.08846
86	23.82309	19.74066	-1.08846	0.00000	1.08846
87	15.65823	11.57580	-1.08846	0.00000	1.08846
88	7.49337	3.41094	-1.08846	0.00000	1.08846
89	-0.67149	-4.75393	-1.08846	0.00000	1.08846
90	-8.83636	-12.91879	-1.08846	0.00000	1.08846

second planform horseshoe vortex descriptions

91	-21.66854	-21.95851	-16.28819	-7.85942	0.83654
92	-22.24848	-22.53845	-16.28819	-7.85942	0.83654
93	-22.82842	-23.11839	-16.28819	-7.85942	0.83654
94	-23.40836	-23.69833	-16.28819	-7.85942	0.83654
95	-23.98830	-24.27827	-16.28819	-7.85942	0.83654
96	-24.56824	-24.85822	-16.28819	-7.85942	0.83654
97	-20.79644	-21.24802	-15.06458	-6.71838	0.83654
98	-21.69960	-22.15118	-15.06458	-6.71838	0.83654
99	-22.60276	-23.05434	-15.06458	-6.71838	0.83654
100	-23.50591	-23.95749	-15.06458	-6.71838	0.83654
101	-24.40907	-24.86065	-15.06458	-6.71838	0.83654
102	-25.31223	-25.76381	-15.06458	-6.71838	0.83654
103	-19.92434	-20.53753	-13.84097	-5.57735	0.83654
104	-21.15072	-21.76390	-13.84097	-5.57735	0.83654
105	-22.37709	-22.99028	-13.84097	-5.57735	0.83654
106	-23.60347	-24.21665	-13.84097	-5.57735	0.83654
107	-24.82984	-25.44303	-13.84097	-5.57735	0.83654
108	-26.05622	-26.66940	-13.84097	-5.57735	0.83654
109	-19.05225	-19.82704	-12.61736	-4.43631	0.83654
110	-20.60184	-21.37664	-12.61736	-4.43631	0.83654
111	-22.15143	-22.92623	-12.61736	-4.43631	0.83654
112	-23.70102	-24.47582	-12.61736	-4.43631	0.83654
113	-25.25062	-26.02541	-12.61736	-4.43631	0.83654
114	-26.80021	-27.57500	-12.61736	-4.43631	0.83654
115	-18.15451	-19.09567	-11.35778	-3.26173	0.88572
116	-20.03682	-20.97798	-11.35778	-3.26173	0.88572
117	-21.91913	-22.86029	-11.35778	-3.26173	0.88572
118	-23.80145	-24.74260	-11.35778	-3.26173	0.88572
119	-25.68376	-26.62491	-11.35778	-3.26173	0.88572
120	-27.56607	-28.50722	-11.35778	-3.26173	0.88572

121	-17.21404	-18.23609	-10.09819	-2.08715	0.83654
122	-19.25814	-20.28019	-10.09819	-2.08715	0.83654
123	-21.30224	-22.32430	-10.09819	-2.08715	0.83654
124	-23.34634	-24.36839	-10.09819	-2.08715	0.83654
125	-25.39045	-26.41250	-10.09819	-2.08715	0.83654
126	-27.43455	-28.45660	-10.09819	-2.08715	0.83654
127	-16.09888	-17.11008	-8.67319	-0.75832	1.11190
128	-18.12127	-19.13247	-8.67319	-0.75832	1.11190
129	-20.14366	-21.15486	-8.67319	-0.75832	1.11190
130	-22.16605	-23.17725	-8.67319	-0.75832	1.11190
131	-24.18844	-25.19963	-8.67319	-0.75832	1.11190
132	-26.21083	-27.22202	-8.67319	-0.75832	1.11190
133	-15.49042	-16.55125	-7.36000	0.00000	0.50000
134	-17.61208	-18.67292	-7.36000	0.00000	0.50000
135	-19.73375	-20.79458	-7.36000	0.00000	0.50000
136	-21.85542	-22.91625	-7.36000	0.00000	0.50000
137	-23.97708	-25.03792	-7.36000	0.00000	0.50000
138	-26.09875	-27.15958	-7.36000	0.00000	0.50000
139	-15.48730	-16.54190	-6.02346	0.00000	0.83654
140	-17.59650	-18.65109	-6.02346	0.00000	0.83654
141	-19.70569	-20.76029	-6.02346	0.00000	0.83654
142	-21.81489	-22.86949	-6.02346	0.00000	0.83654
143	-23.92408	-24.97868	-6.02346	0.00000	0.83654
144	-26.03328	-27.08788	-6.02346	0.00000	0.83654
145	-15.44074	-16.40222	-4.76846	0.00000	0.41846
146	-17.36370	-18.32518	-4.76846	0.00000	0.41846
147	-19.28666	-20.24814	-4.76846	0.00000	0.41846
148	-21.20962	-22.17110	-4.76846	0.00000	0.41846
149	-23.13258	-24.09406	-4.76846	0.00000	0.41846
150	-25.05555	-26.01703	-4.76846	0.00000	0.41846
151	-15.41594	-16.32782	-4.10000	0.00000	0.25000
152	-17.23971	-18.15159	-4.10000	0.00000	0.25000
153	-19.06347	-19.97536	-4.10000	0.00000	0.25000
154	-20.88724	-21.79912	-4.10000	0.00000	0.25000
155	-22.71100	-23.62289	-4.10000	0.00000	0.25000
156	-24.53477	-25.44665	-4.10000	0.00000	0.25000
157	-15.43854	-16.39561	-3.01346	0.00000	0.83654
158	-17.35267	-18.30974	-3.01346	0.00000	0.83654
159	-19.26681	-20.22388	-3.01346	0.00000	0.83654
160	-21.18095	-22.13802	-3.01346	0.00000	0.83654
161	-23.09509	-24.05216	-3.01346	0.00000	0.83654
162	-25.00923	-25.96630	-3.01346	0.00000	0.83654
163	-15.51187	-16.61560	-1.08846	0.00000	1.08846
164	-17.71934	-18.82308	-1.08846	0.00000	1.08846
165	-19.92681	-21.03055	-1.08846	0.00000	1.08846
166	-22.13429	-23.23802	-1.08846	0.00000	1.08846
167	-24.34176	-25.44550	-1.08846	0.00000	1.08846
168	-26.54923	-27.65297	-1.08846	0.00000	1.08846
panel no.	x	c/4 sweep angle	dihedral angle	local alpha in rad	delta cp at cl=
1	0.61276	37.51921	0.00000	0.00000	1.93466
2	-0.18004	25.99276	0.00000	0.00000	0.80132
3	-0.97284	11.71110	0.00000	0.00000	0.44417
4	-1.76564	-4.17472	0.00000	0.00000	0.26877
5	-2.55845	-19.45708	0.00000	0.00000	0.16716

6	-3.35125	-32.35670	0.00000	0.00000	0.09160
7	1.89745	37.51921	0.00000	0.00000	1.76990
8	0.63571	25.99275	0.00000	0.00000	0.76368
9	-0.62603	11.71110	0.00000	0.00000	0.45451
10	-1.88776	-4.17472	0.00000	0.00000	0.28471
11	-3.14950	-19.45708	0.00000	0.00000	0.17716
12	-4.41124	-32.35670	0.00000	0.00000	0.09637
13	3.11717	37.51921	0.00000	0.00000	1.61448
14	1.41021	25.99276	0.00000	0.00000	0.70163
15	-0.29675	11.71110	0.00000	0.00000	0.42928
16	-2.00371	-4.17472	0.00000	0.00000	0.27496
17	-3.71067	-19.45708	0.00000	0.00000	0.17320
18	-5.41763	-32.35670	0.00000	0.00000	0.09524
19	4.33688	37.51921	0.00000	0.00000	1.47893
20	2.18470	25.99276	0.00000	0.00000	0.65076
21	0.03253	11.71110	0.00000	0.00000	0.40415
22	-2.11965	-4.17472	0.00000	0.00000	0.26327
23	-4.27183	-19.45708	0.00000	0.00000	0.16857
24	-6.42401	-32.35670	0.00000	0.00000	0.09444
25	5.62157	37.51921	0.00000	0.00000	1.35651
26	3.00046	25.99275	0.00000	0.00000	0.60063
27	0.37934	11.71110	0.00000	0.00000	0.37829
28	-2.24177	-4.17472	0.00000	0.00000	0.25096
29	-4.86289	-19.45708	0.00000	0.00000	0.16475
30	-7.48401	-32.35670	0.00000	0.00000	0.09603
31	6.90626	37.51921	0.00000	0.00000	1.25084
32	3.81621	25.99275	0.00000	0.00000	0.55670
33	0.72616	11.71110	0.00000	0.00000	0.35540
34	-2.36390	-4.17472	0.00000	0.00000	0.24065
35	-5.45395	-19.45708	0.00000	0.00000	0.16334
36	-8.54400	-32.35670	0.00000	0.00000	0.10177
37	7.99810	37.51921	0.00000	0.00000	1.17259
38	4.50950	25.99275	0.00000	0.00000	0.52388
39	1.02091	11.71110	0.00000	0.00000	0.33913
40	-2.46768	-4.17472	0.00000	0.00000	0.23444
41	-5.95628	-19.45708	0.00000	0.00000	0.16514
42	-9.44487	-32.35670	0.00000	0.00000	0.11344
43	9.08994	37.51922	0.00000	0.00000	1.10114
44	5.20280	25.99276	0.00000	0.00000	0.49372
45	1.31566	11.71110	0.00000	0.00000	0.32473
46	-2.57147	-4.17472	0.00000	0.00000	0.22932
47	-6.45861	-19.45708	0.00000	0.00000	0.16777
48	-10.34575	-32.35669	0.00000	0.00000	0.13011
49	10.18414	37.51921	0.00000	0.00000	1.03049
50	5.89760	25.99276	0.00000	0.00000	0.46250
51	1.61105	11.71110	0.00000	0.00000	0.31155
52	-2.67549	-4.17472	0.00000	0.00000	0.22487
53	-6.96203	-19.45708	0.00000	0.00000	0.17004
54	-11.24857	-32.35670	0.00000	0.00000	0.15708
55	11.03750	38.76506	0.00000	0.00000	0.99032
56	6.51620	33.55879	0.00000	0.00000	0.44531
57	1.99489	27.64136	0.00000	0.00000	0.30956
58	-2.52641	21.00937	0.00000	0.00000	0.22725
59	-7.04772	13.73368	0.00000	0.00000	0.17350
60	-11.56902	5.97943	0.00000	0.00000	0.18265
61	12.11076	38.76506	0.00000	0.00000	0.95236
62	7.40281	33.55879	0.00000	0.00000	0.43009
63	2.69485	27.64136	0.00000	0.00000	0.31413

64	-2.01311	21.00937	0.00000	0.00000	0.23344
65	-6.72107	13.73368	0.00000	0.00000	0.17482
66	-11.42903	5.97943	0.00000	0.00000	0.20034
67	13.11855	38.76506	0.00000	0.00000	0.88945
68	8.23532	33.55879	0.00000	0.00000	0.42661
69	3.35210	27.64136	0.00000	0.00000	0.33185
70	-1.53113	21.00937	0.00000	0.00000	0.23456
71	-6.41435	13.73368	0.00000	0.00000	0.15684
72	-11.29758	5.97943	0.00000	0.00000	0.21695
73	21.98959	73.23754	0.00000	0.00000	0.21302
74	15.56357	69.96743	0.00000	0.00000	0.38221
75	9.13756	65.21040	0.00000	0.00000	0.38601
76	2.71154	57.79776	0.00000	0.00000	0.29039
77	-3.71447	45.29755	0.00000	0.00000	0.25414
78	-10.14049	23.41484	0.00000	0.00000	0.19888
79	25.59691	73.23754	0.00000	0.00000	0.28844
80	18.54354	69.96743	0.00000	0.00000	0.19152
81	11.49016	65.21040	0.00000	0.00000	0.36631
82	4.43679	57.79776	0.00000	0.00000	0.30348
83	-2.61659	45.29755	0.00000	0.00000	0.23120
84	-9.66997	23.41484	0.00000	0.00000	0.20109
85	31.98795	73.23754	0.00000	0.00000	0.21732
86	23.82309	69.96743	0.00000	0.00000	0.16858
87	15.65823	65.21040	0.00000	0.00000	0.25630
88	7.49337	57.79776	0.00000	0.00000	0.30933
89	-0.67149	45.29755	0.00000	0.00000	0.23234
90	-8.83636	23.41482	0.00000	0.00000	0.19074

second planform horseshoe vortex descriptions

91	-21.66854	35.47837	43.00000	0.17450	2.07234
92	-22.24848	24.15973	43.00000	0.17450	0.86344
93	-22.82842	10.44929	43.00000	0.17450	0.50181
94	-23.40836	-4.55831	43.00000	0.17450	0.31432
95	-23.98830	-18.97688	43.00000	0.17450	0.19781
96	-24.56824	-31.30072	43.00000	0.17450	0.10867
97	-20.79644	35.47836	43.00000	0.17450	1.86622
98	-21.69960	24.15975	43.00000	0.17450	0.79905
99	-22.60276	10.44927	43.00000	0.17450	0.48369
100	-23.50591	-4.55835	43.00000	0.17450	0.31020
101	-24.40907	-18.97688	43.00000	0.17450	0.19537
102	-25.31223	-31.30072	43.00000	0.17450	0.10614
103	-19.92434	35.47837	43.00000	0.17450	1.62276
104	-21.15072	24.15975	43.00000	0.17450	0.69665
105	-22.37709	10.44928	43.00000	0.17450	0.42654
106	-23.60347	-4.55832	43.00000	0.17450	0.27531
107	-24.82984	-18.97687	43.00000	0.17450	0.17295
108	-26.05622	-31.30070	43.00000	0.17450	0.09307
109	-19.05225	35.47837	43.00000	0.17450	1.39650
110	-20.60184	24.15974	43.00000	0.17450	0.60027
111	-22.15143	10.44929	43.00000	0.17450	0.36950
112	-23.70102	-4.55834	43.00000	0.17450	0.23890
113	-25.25062	-18.97689	43.00000	0.17450	0.14921
114	-26.80021	-31.30070	43.00000	0.17450	0.07915
115	-18.15451	35.47837	43.00000	0.17450	1.17860
116	-20.03682	24.15975	43.00000	0.17450	0.50846
117	-21.91913	10.44929	43.00000	0.17450	0.31541
118	-23.80145	-4.55835	43.00000	0.17450	0.20462

D-58 Applied Computational Aerodynamics

119	-25.68376	-18.97689	43.00000	0.17450	0.12671
120	-27.56607	-31.30071	43.00000	0.17450	0.06503
121	-17.21404	38.04567	43.00000	0.17450	1.01268
122	-19.25814	38.58307	43.00000	0.17450	0.44608
123	-21.30224	39.11254	43.00000	0.17450	0.28734
124	-23.34634	39.63417	43.00000	0.17450	0.19345
125	-25.39045	40.14806	43.00000	0.17450	0.12395
126	-27.43455	40.65429	43.00000	0.17450	0.06535
127	-16.09888	38.04567	43.00000	0.17450	0.83667
128	-18.12127	38.58307	43.00000	0.17450	0.39566
129	-20.14366	39.11254	43.00000	0.17450	0.27313
130	-22.16605	39.63417	43.00000	0.17450	0.19673
131	-24.18844	40.14806	43.00000	0.17450	0.13605
132	-26.21083	40.65429	43.00000	0.17450	0.07985
133	-15.49042	-3.19570	0.00000	0.00000	0.56637
134	-17.61208	-15.59796	0.00000	0.00000	0.42340
135	-19.73375	-26.67953	0.00000	0.00000	0.31785
136	-21.85542	-35.97340	0.00000	0.00000	0.23703
137	-23.97708	-43.50610	0.00000	0.00000	0.16547
138	-26.09875	-49.53985	0.00000	0.00000	0.09394
139	-15.48730	2.12462	0.00000	0.00000	0.38372
140	-17.59650	10.50852	0.00000	0.00000	0.33354
141	-19.70569	18.46350	0.00000	0.00000	0.27610
142	-21.81489	25.74715	0.00000	0.00000	0.21449
143	-23.92408	32.23863	0.00000	0.00000	0.15167
144	-26.03328	37.92108	0.00000	0.00000	0.08438
145	-15.44074	2.12462	0.00000	0.00000	0.34990
146	-17.36370	10.50851	0.00000	0.00000	0.29853
147	-19.28666	18.46349	0.00000	0.00000	0.25906
148	-21.20962	25.74714	0.00000	0.00000	0.21134
149	-23.13258	32.23862	0.00000	0.00000	0.15849
150	-25.05555	37.92108	0.00000	0.00000	0.09723
151	-15.41594	2.12462	0.00000	0.00000	0.35024
152	-17.23971	10.50852	0.00000	0.00000	0.28116
153	-19.06347	18.46350	0.00000	0.00000	0.24782
154	-20.88724	25.74714	0.00000	0.00000	0.20712
155	-22.71100	32.23864	0.00000	0.00000	0.16006
156	-24.53477	37.92108	0.00000	0.00000	0.10421
157	-15.43854	-2.18164	0.00000	0.00000	0.32319
158	-17.35267	-10.78430	0.00000	0.00000	0.24905
159	-19.26681	-18.92465	0.00000	0.00000	0.22019
160	-21.18095	-26.34637	0.00000	0.00000	0.18542
161	-23.09509	-32.92787	0.00000	0.00000	0.14319
162	-25.00923	-38.65982	0.00000	0.00000	0.09029
163	-15.51187	-2.18164	0.00000	0.00000	0.29260
164	-17.71934	-10.78430	0.00000	0.00000	0.22016
165	-19.92681	-18.92465	0.00000	0.00000	0.19116
166	-22.13429	-26.34637	0.00000	0.00000	0.15602
167	-24.34176	-32.92787	0.00000	0.00000	0.11287
168	-26.54923	-38.65981	0.00000	0.00000	0.06313

ref. chord	c average	true area	reference area
26.89170	31.36179	1364.23767	950.00000

b/2	ref. ar	true ar	mach number
21.75000	1.99184	1.38704	0.30000

complete configuration

cl	computed alpha	lift cl(wb)	induced drag(far field solution) cdi at cl(wb)	cdi/(cl(wb)**2)
0.5300	7.6834	0.3851	0.0238	0.1608

complete configuration characteristics

cl per rad	alpha per deg	cl(twist) at cl=0	alpha at cl=0	y cp	cm/cl	cmo
3.11731	0.05441	0.11197	-2.05798	-0.42053	0.06834	-0.07080

additional loading
with cl based on s(true)

-at cl des-

stat	2y/b	sl coef	cl ratio	c ratio	load dueto twist	add. load at cl = 0.112	basic load at cl = 0	span load at cl desir	x loc of local cent of press
1	-0.962	0.310	2.045	0.152	0.003	0.024	-0.021	0.094	-0.162
2	-0.885	0.471	1.950	0.241	0.006	0.037	-0.031	0.143	0.575
3	-0.812	0.588	1.802	0.327	0.008	0.046	-0.038	0.179	1.273
4	-0.739	0.687	1.669	0.412	0.010	0.054	-0.044	0.210	1.949
5	-0.662	0.775	1.545	0.501	0.012	0.060	-0.048	0.238	2.630
6	-0.585	0.850	1.438	0.591	0.016	0.066	-0.051	0.263	3.257
7	-0.519	0.909	1.362	0.667	0.019	0.071	-0.052	0.284	3.709
8	-0.454	0.963	1.295	0.744	0.023	0.075	-0.052	0.303	4.081
9	-0.388	1.011	1.233	0.820	0.028	0.079	-0.051	0.322	4.316
10	-0.338	1.046	1.209	0.865	0.031	0.082	-0.050	0.336	4.526
11	-0.277	1.075	1.193	0.901	0.033	0.084	-0.051	0.346	5.045
12	-0.219	1.092	1.168	0.934	0.034	0.085	-0.052	0.351	5.530
13	-0.189	1.100	0.895	1.229	0.033	0.086	-0.053	0.353	6.950
14	-0.139	1.109	0.822	1.349	0.033	0.086	-0.053	0.356	8.812
15	-0.050	1.117	0.715	1.562	0.033	0.087	-0.054	0.358	11.245

contribution of the second planform to span load distribution

16	-0.749	0.116	1.047	0.111	0.041	0.009	0.032	0.075	-22.261
17	-0.693	0.164	0.948	0.173	0.061	0.013	0.048	0.108	-21.759
18	-0.636	0.189	0.806	0.235	0.073	0.015	0.059	0.129	-21.242
19	-0.580	0.200	0.676	0.296	0.082	0.016	0.066	0.140	-20.719
20	-0.522	0.200	0.555	0.360	0.086	0.016	0.070	0.144	-20.183
21	-0.464	0.187	0.479	0.391	0.084	0.015	0.070	0.139	-19.541
22	-0.399	0.164	0.423	0.387	0.076	0.013	0.063	0.124	-18.709
23	-0.338	0.174	0.429	0.406	0.071	0.014	0.058	0.122	-18.903
24	-0.277	0.148	0.366	0.404	0.054	0.012	0.043	0.097	-19.224
25	-0.219	0.133	0.363	0.368	0.045	0.010	0.035	0.084	-19.037
26	-0.189	0.128	0.366	0.349	0.041	0.010	0.031	0.079	-18.872
27	-0.139	0.124	0.339	0.366	0.038	0.010	0.028	0.074	-19.025
28	-0.050	0.126	0.297	0.422	0.036	0.010	0.027	0.073	-19.428

induced drag, leading edge thrust , suction coefficient characteristics
computed at the desired c_l from a near field solution

section coefficients					
l.e. sweep					
station	2y/b	angle	cdii c/2b	ct c/2b	cs c/2b
1	-0.96154	39.96069	0.00412	0.00085	0.00111
2	-0.88462	39.96069	0.00182	0.00507	0.00661
3	-0.81158	39.96069	0.00042	0.00821	0.01071
4	-0.73855	39.96069	0.00101	0.00912	0.01190
5	-0.66163	39.96069	0.00187	0.00962	0.01255
6	-0.58470	39.96069	0.00286	0.00985	0.01285
7	-0.51933	39.96069	0.00392	0.00978	0.01276
8	-0.45395	39.96069	0.00499	0.00967	0.01261
9	-0.38844	39.96069	0.00581	0.00975	0.01272
10	-0.33839	39.96069	0.00675	0.00940	0.01226
11	-0.27694	39.96069	0.00795	0.00872	0.01138
12	-0.21924	39.96069	0.00884	0.00813	0.01389
13	-0.18851	73.89906	0.00974	0.00733	0.02002
14	-0.13855	73.89906	0.01161	0.00557	0.02009
15	-0.05004	73.89906	0.02071	-0.00342	-0.01233

contribution of the second planform to the chord or drag force

16	-0.74888	37.91007	0.00645	0.00292	0.00370
17	-0.69262	37.91007	0.00477	0.00715	0.00906
18	-0.63637	37.91007	0.00430	0.00984	0.01247
19	-0.58011	37.91007	0.00588	0.00953	0.01208
20	-0.52220	37.91007	0.00755	0.00832	0.01054
21	-0.46428	37.91007	0.00901	0.00640	0.00812
22	-0.39877	37.91007	0.00956	0.00324	0.00400
23	-0.33839	0.00000	0.00665	0.00124	0.00128
24	-0.27694	0.00000	0.00449	0.00034	0.00034
25	-0.21924	0.00000	0.00408	0.00003	0.00003
26	-0.18851	0.00000	0.00389	-0.00001	-0.00001
27	-0.13855	0.00000	0.00361	0.00002	0.00002
28	-0.05004	0.00000	0.00331	0.00022	0.00022

total coefficients

cdii/ c_l^{**2} = 0.15439 ct= 0.04177 cs= 0.05673

1

end of file encountered after configuration 23.

E.1 FOILGEN

This program is used for airfoil geometry generation. For airfoils with analytically defined ordinates, this program produces airfoil definition data sets in the format required for PANELv2. This includes NACA 4-digit, 4-digit modified and 5-digit airfoils. In addition, the NACA 6 and 6A camber lines are available. The user can combine any combination of thickness and camber lines available within these shapes. This provides a wide range of airfoil definitions. The program runs interactively, and a sample terminal session is provided here to illustrate its use.

From terminal session:

```
NACA Airfoil Ordinate Generation
W.H. Mason, March 15, 1992

Thickness Distribution Options:

    1 - NACA 4 Digit Series
    2 - NACA Modified 4 Digit Series

Select 1 or 2 :2

Input Max Thickness,  T/C =.18

X/C Position of Max Thickness =.4

Input leading edge parameter:

Choose values from 0 to 9 -
(6 is the 4 Series value)  7

Leading Edge Radius, rle/C = 0.04859

Trailing Edge Angle is 31.60 degrees
[this is the TOTAL included angle]

Camber Distribution Options:

    1 - NACA 4 Digit Series
    2 - NACA 5 Digit Series
    3 - NACA 6 & 6A Series

Select 1,2 or 3: 3

Design Lift Coefficient = .2

Input X/C for constant loading, A = .8

6A-series camber line ? (Y/N):y

Choose output option :

    1 - Point by point
    2 - Distribution

Select 1 or 2:2

Select type of distribution:
```

- 1 - Even Spacing
- 2 - Full Cosine
(Concentrated at both LE & TE)
- 3 - Half Cosine
(Concentrated at LE)

Choose 1, 2, or 3 :1

Number of points in distribution,
(131 maximum) =21

I	X/C	YT/C	DYT/X	YC/C	DYC/C	XU/C(%)	YU/C(%)	XL/C(%)	YL/C(%)
1	0.0000	0.0000	99.9999	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	0.0500	0.0529	0.3774	0.0036	0.0543	4.7133	5.6407	5.2867	-4.9195
3	0.1000	0.0665	0.2020	0.0060	0.0412	9.7259	7.2420	10.2741	-6.0498
4	0.1500	0.0747	0.1343	0.0078	0.0331	14.7529	8.2487	15.2471	-6.6872
5	0.2000	0.0804	0.0969	0.0093	0.0269	19.7837	8.9702	20.2163	-7.1099
6	0.2500	0.0846	0.0706	0.0105	0.0217	24.8161	9.5086	25.1839	-7.4057
7	0.3000	0.0875	0.0478	0.0115	0.0172	29.8495	9.9018	30.1505	-7.6046
8	0.3500	0.0894	0.0249	0.0122	0.0130	34.8839	10.1599	35.1161	-7.7119
9	0.4000	0.0900	0.0000	0.0128	0.0090	39.9189	10.2786	40.0811	-7.7207
10	0.4500	0.0893	-0.0261	0.0131	0.0051	44.9543	10.2487	45.0457	-7.6202
11	0.5000	0.0874	-0.0518	0.0133	0.0012	49.9894	10.0698	50.0106	-7.4096
12	0.5500	0.0842	-0.0769	0.0133	-0.0028	55.0236	9.7439	54.9764	-7.0915
13	0.6000	0.0797	-0.1017	0.0130	-0.0071	60.0564	9.2725	59.9436	-6.6692
14	0.6500	0.0740	-0.1259	0.0125	-0.0118	65.0871	8.6561	64.9129	-6.1465
15	0.7000	0.0671	-0.1498	0.0118	-0.0172	70.1156	7.8941	69.8844	-5.5287
16	0.7500	0.0591	-0.1731	0.0108	-0.0241	75.1424	6.9835	74.8576	-4.8231
17	0.8000	0.0498	-0.1960	0.0093	-0.0361	80.1796	5.9136	79.8204	-4.0441
18	0.8500	0.0395	-0.2184	0.0072	-0.0469	85.1847	4.6629	84.8153	-3.2200
19	0.9000	0.0280	-0.2404	0.0047	-0.0480	90.1343	3.2641	89.8657	-2.3264
20	0.9500	0.0154	-0.2619	0.0023	-0.0480	95.0740	1.7693	94.9260	-1.3120
21	1.0000	0.0018	-0.2830	0.0000	0.0000	100.0000	0.1800	100.0000	-0.1800
I	X/C	YT/C	DYT/X	YC/C	DYC/C	XU/C(%)	YU/C(%)	XL/C(%)	YL/C(%)

send output to a file? (Y/N):

y

enter file name:

testout.txt

enter file title:

NACA 18% thick, xt=.4, I=7, 6A series cam, CLI = .2

Another case?

n

STOP

The disk file generated from the session shown above is:

NACA 18% thick, xt=.4, I=7, 6A series cam, CLI = .2

21.000000 21.000000

Upper Surface

0.000000 0.000000

0.047133 0.056407

0.097259 0.072420

E-4 Applied Computational Aerodynamics

0.147529	0.082487
0.197837	0.089702
0.248161	0.095086
0.298495	0.099018
0.348839	0.101599
0.399189	0.102786
0.449543	0.102487
0.499894	0.100698
0.550236	0.097439
0.600564	0.092725
0.650871	0.086561
0.701156	0.078941
0.751424	0.069835
0.801796	0.059136
0.851847	0.046629
0.901343	0.032641
0.950740	0.017693
1.000000	0.001800
Lower Surface	
0.000000	0.000000
0.052867	-0.049195
0.102741	-0.060498
0.152471	-0.066872
0.202163	-0.071099
0.251839	-0.074057
0.301505	-0.076046
0.351161	-0.077119
0.400811	-0.077207
0.450457	-0.076202
0.500106	-0.074096
0.549764	-0.070915
0.599436	-0.066692
0.649129	-0.061465
0.698844	-0.055287
0.748576	-0.048231
0.798204	-0.040441
0.848153	-0.032200
0.898657	-0.023264
0.949260	-0.013120
1.000000	-0.001800

E.2 LADSON

This is the NASA program that provides a reasonable approximation to the NACA 6 and 6A series airfoils. It was written by Charles Ladson and Cuyler Brooks (Ref E.2-1). Originally it ran on the NASA CDC computer. It has been ported to run on a personal computer (Macintosh). Only minor modifications were made to produce a program to generate a set of ordinates in the form required as standard input by the programs described in App. D.

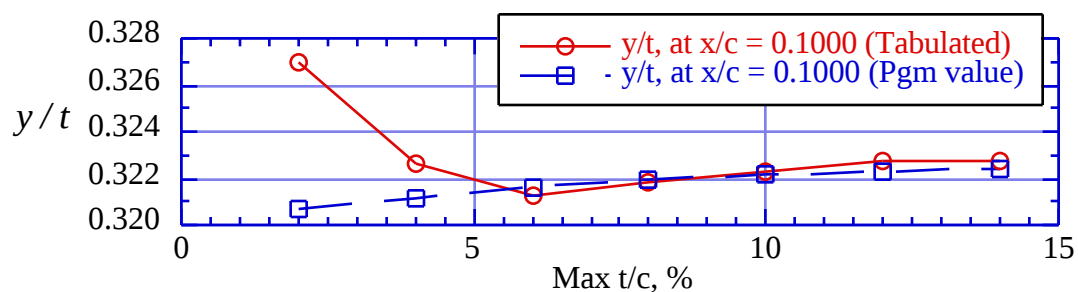
The program is only an approximation to the ordinates because there is no simple algebraic formula available to describe the thickness distribution. I spoke briefly to Charles Ladson some years ago, and he said that he thought it would be impossible to generate a more accurate program. When he was doing this work he investigated the availability of more detailed notes on these airfoils and discovered that all the records have been destroyed. The only information available is that contained in the actual NACA reports. However, this program is much more accurate than attempts to simulate the 6 and 6A series thickness envelope by using a modified NACA 4-digit airfoil formula. The program was developed to handle thicknesses from 6 to 15 percent.

Figure E.2-1 compares the program predictions with the official ordinates - which are given in Ref E.2-2, for 64-series airfoils. If the thickness distribution could be obtained by scaling a reference airfoil, each curve would be a straight flat line. Note especially that below thickness of around six percent the program deviates significantly from the tabulated values.

One other possible problem is the value at the trailing edge. Originally further processing was required to find the value. The program was modified to linearly extrapolate the values near the trailing edge to get the final values. This was the approach recommended by Ladson. This is done in the new routine added to generate the file of points, **stdout**. The user should check this approximation if the results appear to be in error at the trailing edge.

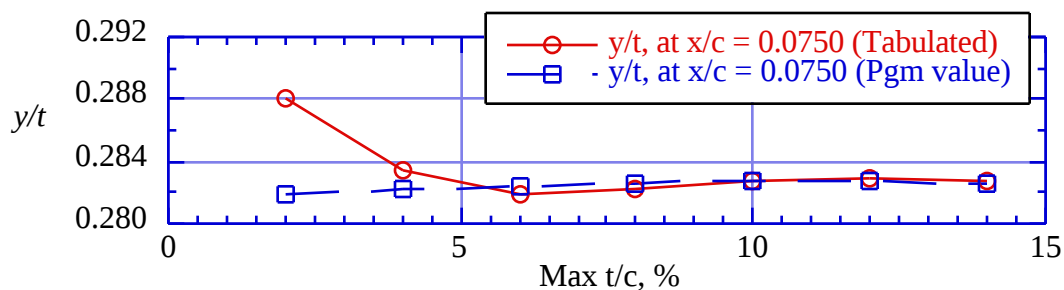
References

- E.2-1. Ladson, C.L., and Brooks, C.W., Jr., "Development of a Computer Program to Obtain Ordinates for NACA 6- and 6A-Series Airfoils," NASA TM X-3069, Sept., 1974.
- E.2-2. Patterson, E.W., and Braslow, A.L., "Ordinates and Theoretical Pressure Distribution Data for NACA 6- and 6A- Series Airfoil Sections with Thicknesses from 2 to 21 and from 2 to 15 Percent Chord Respectively," NASA R-84, 1961.

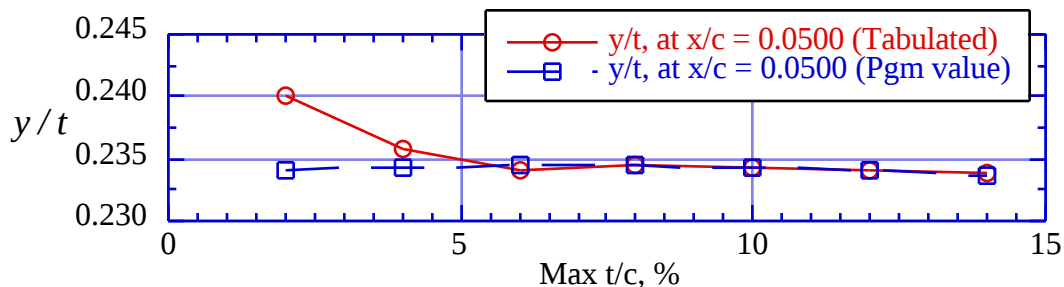


a. $x/c = 0.10$

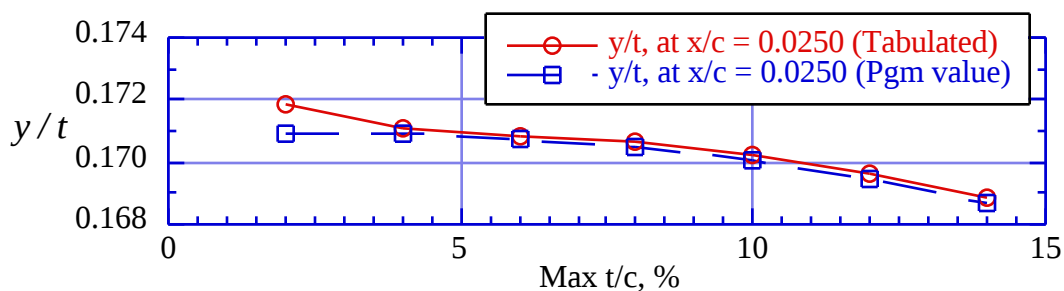
Fig. E.2-1 Comparison of tabulated and computed airfoil ordinate values.



b. $x/c = 0.075$



c. $x/c = 0.050$



d. $x/c = 0.025$

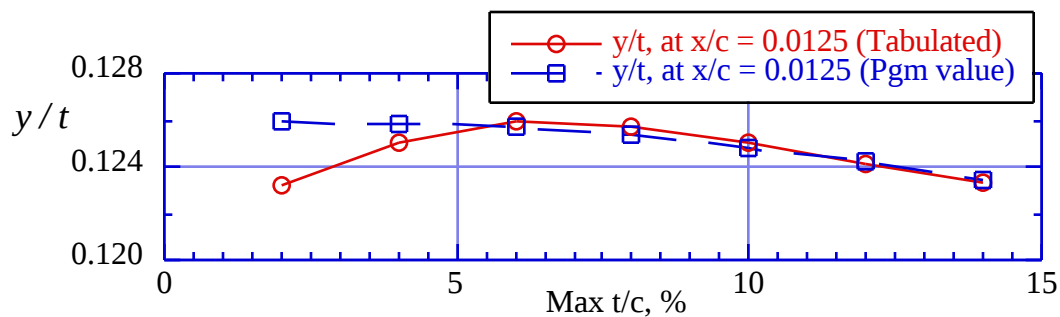
e. $x/c = 0.0125$

Fig. E.2-1 Comparison of tabulated and computed airfoil ordinate values. (continued)

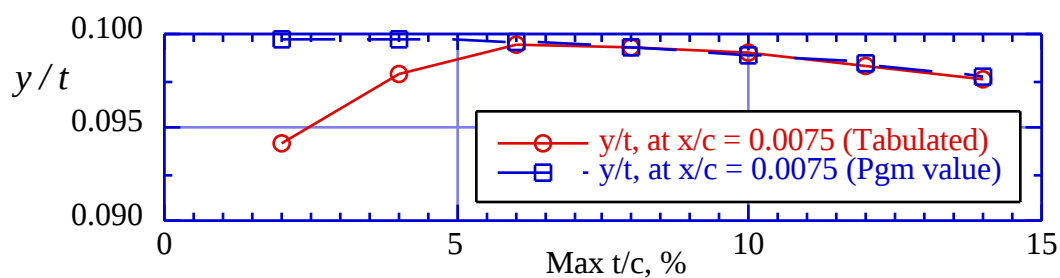
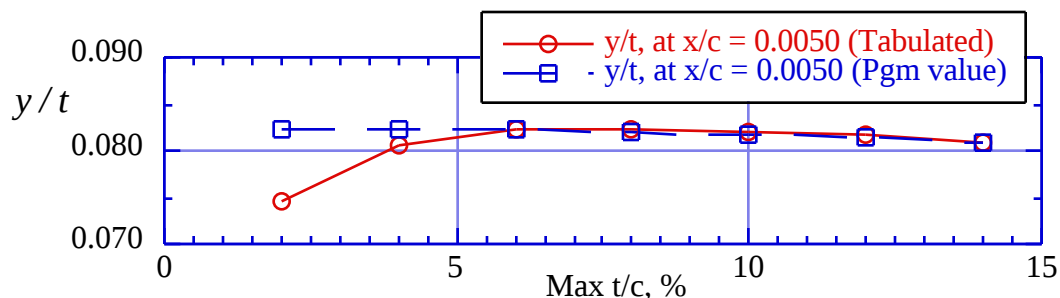
f. $x/c = 0.0075$ g. $x/c = 0.0050$

Fig. E.2-1 Comparison of tabulated and computed airfoil ordinate values. (concluded)

Input Description

The user first creates a data file as described below. Then, the program runs interactively. It queries the user for the name of the input data file. After the airfoil ordinates are found, the user is asked for the name of the output file containing the ordinates in standard format. The file names can be up to twenty characters long. Because the program was developed in the era of cards, it is critically important that the input be placed in the specified column.

E-8 Applied Computational Aerodynamics

<u>Card</u>	<u>Field</u>	<u>Variable</u>	<u>Description</u>
1.	2-80	TITLE	Case title card. Any values can be used, from columns 2 to 80

2. Airfoil and camber line series designations as follows:

NACA airfoil thickness family:	card designation	column
63-series	63	9,10
64-series	64	9,10
65-series	65	9,10
66-series	66	9,10
67-series	67	9,10
63A-series	63A	8,9,10
64A-series	64A	8,9,10
65A-series	65A	8,9,10

NACA Camber line	card designation	column
NACA 6-series	63	19,20
	64	19,20
	65	19,20
	66	19,20
NACA 6A-series	63A	18,19,20
	64A	18,19,20
	65A	18,19,20

Airfoil Parameter card (Note cards 3 to 6 are in floating point mode. Numbers are entered with a decimal point.

3.	1-10	TOC	Thickness to chord ratio of airfoil, <i>i.e.</i> , 0.120
	11-20	LER	Published leading-edge radius may be entered if desired (not used in program)
	21-30	CHD	model chord used for listing ordinates in dimensional units
	31-40	CLI	Design lift coefficient (<i>i.e.</i> , 0.20) set to 0.0 for a symmetrical airfoil
	41-50	A	mean line chordwise loading (use 0.8 for 6A-series airfoils)
	51-60	CMBNMR	number of mean lines to be summed, up to a max of nine (if only one, leave blank or insert 1.0)

and as required:

<u>Card</u>	<u>Field</u>	<u>Variable</u>	<u>Description</u>
4.	1-10	CLI	design lift for second mean line
	11-20	A	loading for second mean line
	21-30	CLI	design lift for third mean line
	31-40	A	loading for third mean line
	41-50	CLI	design lift for fourth mean line
	51-60	A	loading for fourth mean line
	61-70	CLI	design lift for fifth mean line
	71-80	A	loading for fifth mean line

<u>Card</u>	<u>Field</u>	<u>Variable</u>	<u>Description</u>
5.	1-10	CLI	design lift for sixth mean line
	11-20	A	loading for sixth mean line
	21-30	CLI	design lift for seventh mean line
	31-40	A	loading for seventh mean line
	41-50	CLI	design lift for eighth mean line
	51-60	A	loading for eighth mean line
	61-70	CLI	design lift for ninth mean line
	71-80	A	loading for ninth mean line
6.	1-10	CLI	design lift for tenth mean line
	11-20	A	loading for tenth mean line

Sample input:

```

NACA 65(1)213 A=0.5, CL=0.2
      65      65
0.130      0.00      1.0      0.2      0.5      1.0

```

Output

The program files also contain the sample output of the program. Because the program was written many years ago, it uses 133 column output, and doesn't fit on a normal page. The output file corresponds to the input data set given above and also available in the program files. This case should be verified before further use of the program.

Eighty values of the upper and lower surface are contained in the disk file. The following is the file generated from the sample input listed above. All numbers are output in 2F10.6 format.

```

NACA 65(1)213 A=0.5, CL=0.2
80.000000 80.000000
UPPER SURFACE
0.000000 0.000000
0.000294 0.004049
0.000862 0.005724
0.001472 0.007026
0.002106 0.008120
0.002756 0.009078
0.003416 0.009940
0.004095 0.010729
0.004801 0.011465
0.005750 0.012363
0.006706 0.013187
0.007668 0.013957
0.008635 0.014674
0.009605 0.015351
0.010578 0.015999
0.013509 0.017790
0.018419 0.020432
0.023349 0.022801
0.028292 0.024990
0.033243 0.027041
0.038202 0.028978
0.043167 0.030816
0.048138 0.032564
0.053113 0.034230
0.058093 0.035824
0.063077 0.037355
0.068064 0.038829
0.073053 0.040252
0.078045 0.041628

```

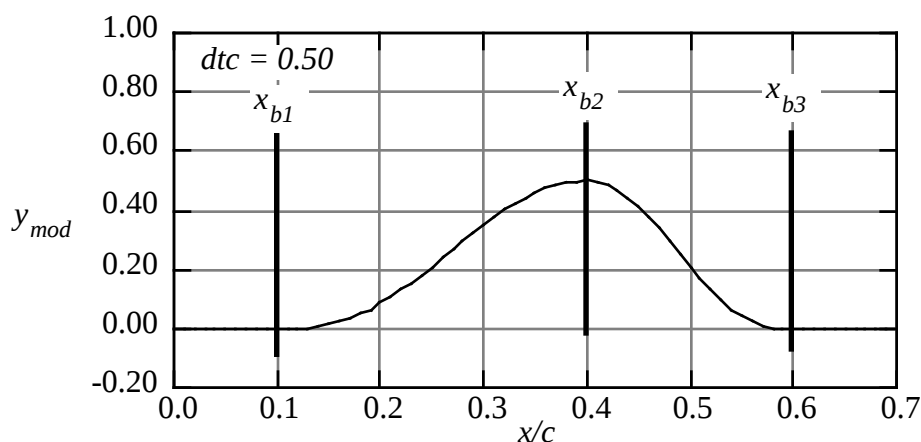
E-10 Applied Computational Aerodynamics

```
0.083040 0.042963
0.088036 0.044256
0.093035 0.045510
0.098035 0.046725
0.108042 0.049051
0.118054 0.051242
0.138096 0.055281
0.158154 0.058909
0.178228 0.062180
0.198314 0.065122
0.218410 0.067761
0.238517 0.070119
0.258632 0.072209
0.278755 0.074040
0.298885 0.075621
0.319022 0.076953
0.339166 0.078041
0.359316 0.078893
0.379473 0.079505
0.399636 0.079849
0.419807 0.079905
0.439987 0.079653
0.460179 0.079093
0.480389 0.078208
0.500661 0.076988
0.520914 0.075401
0.541078 0.073489
0.561197 0.071293
0.581282 0.068840
0.601339 0.066166
0.621370 0.063297
0.641379 0.060250
0.661367 0.057038
0.681337 0.053682
0.701290 0.050202
0.721229 0.046629
0.741156 0.042982
0.761071 0.039263
0.780978 0.035493
0.800877 0.031693
0.820771 0.027893
0.840662 0.024115
0.860553 0.020383
0.880445 0.016732
0.900341 0.013196
0.920244 0.009825
0.940156 0.006682
0.960082 0.003866
0.980027 0.001528
0.990009 0.000616
1.000000 -0.000015
LOWER SURFACE
0.000000 0.000000
0.001206 -0.003782
0.002138 -0.005234
0.003028 -0.006330
0.003894 -0.007228
0.004744 -0.008000
0.005584 -0.008681
0.006405 -0.009295
0.007199 -0.009859
0.008250 -0.010536
0.009294 -0.011145
0.010332 -0.011704
0.011365 -0.012216
0.012395 -0.012692
0.013422 -0.013143
0.016491 -0.014363
```

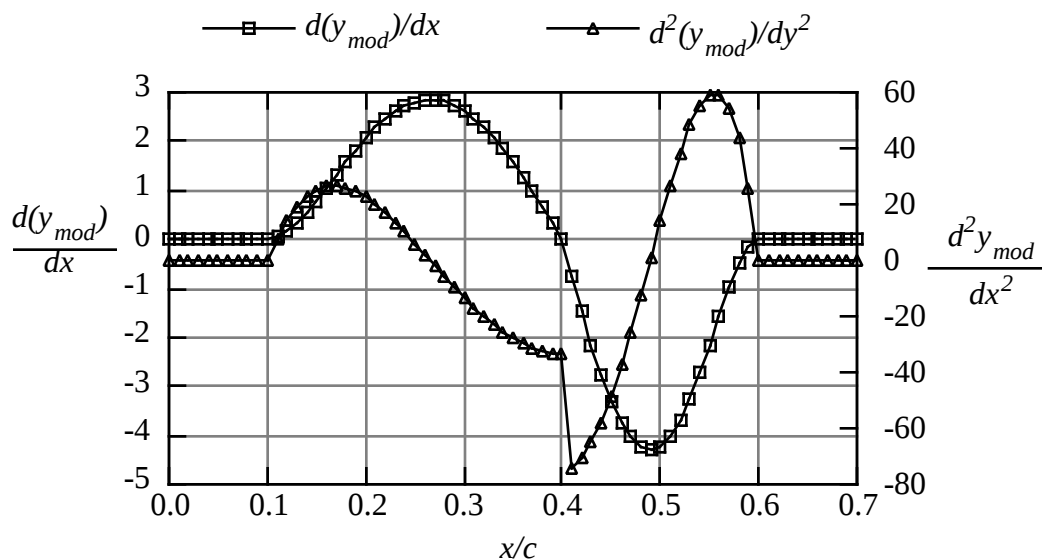
0.021581	-0.016110
0.026651	-0.017640
0.031708	-0.019033
0.036757	-0.020325
0.041798	-0.021536
0.046833	-0.022676
0.051862	-0.023751
0.056887	-0.024766
0.061907	-0.025730
0.066923	-0.026652
0.071936	-0.027533
0.076947	-0.028381
0.081955	-0.029197
0.086960	-0.029988
0.091964	-0.030750
0.096965	-0.031487
0.101964	-0.032198
0.111958	-0.033552
0.121946	-0.034818
0.141904	-0.037129
0.161846	-0.039182
0.181772	-0.041014
0.201686	-0.042644
0.221590	-0.044089
0.241483	-0.045366
0.261368	-0.046480
0.281245	-0.047440
0.301115	-0.048251
0.320978	-0.048913
0.340834	-0.049428
0.360684	-0.049806
0.380527	-0.050043
0.400364	-0.050111
0.420192	-0.049996
0.440012	-0.049680
0.459821	-0.049171
0.479611	-0.048463
0.499339	-0.047570
0.519085	-0.046504
0.538922	-0.045258
0.558802	-0.043842
0.578717	-0.042267
0.598660	-0.040559
0.618629	-0.038730
0.638620	-0.036789
0.658632	-0.034743
0.678662	-0.032603
0.698709	-0.030384
0.718770	-0.028110
0.738843	-0.025795
0.758928	-0.023432
0.779021	-0.021039
0.799122	-0.018629
0.819228	-0.016226
0.839337	-0.013846
0.859446	-0.011507
0.879554	-0.009237
0.899658	-0.007062
0.919755	-0.005023
0.939843	-0.003175
0.959916	-0.001603
0.979971	-0.000444
0.989990	-0.000089
1.000000	-0.000015

E.5 BUMP

This subroutine illustrates a means of making smooth changes to airfoil shapes. It is included in PANELv2. It is designed to place a “bump” on the airfoil contour. The shape change starts gradually with zero curvature at point x_{b1} . The bump is setup to be asymmetric about the bump midpoint, x_{b2} , and to blend back into the baseline shape with zero curvature at point x_{b3} . However, if an asymmetric bump is used, the curvature will be discontinuous at the bump maximum. The following plot defines the nomenclature, as well as plotting the output of the sample main program presented below.



The related slope and curvature are given in the next graph.



The equation of the bump is:

$$y_{mod} = -64 \left(\Delta \frac{t}{c} \right) x_d^3 (x_d - 1)^3$$

$$\frac{dy_{mod}}{dx_d} = -64 \left(\Delta \frac{t}{c} \right) 3x_d^2 (x_d - 1)^2 (2x_d - 1)$$

$$\frac{d^2 y_{mod}}{dx_d^2} = -64 \left(\Delta \frac{t}{c} \right) 6x_d (x_d - 1) (5x_d^2 - 5x_d + 1)$$

where

$$x_d = \frac{(x - x_1)}{2(x_2 - x_1)} \quad x_1 < x < x_2$$

or

$$x_d = \frac{(x + x_3 - 2x_2)}{2(x_3 - x_2)} \quad x_2 < x < x_3$$

This function is often called a “cubic bump” although it is clearly a sixth order polynomial.

The user should examine the subroutine to understand the transformation between the local variable x_d and the global variable x_{in} .

Listing of subroutine bump.f:

```

      subroutine bump(xb1,xb2,xb3,dtc,xmax,xin,ymod,ymodp,ymodpp)
      c
      c      so-called cubic bump function
      c
      c      used to make mods to aero surfaces
      c
      c      W.H. Mason, December 1989
      c
      c      xb1      - start of bump (dimensional)
      c      xb2      - location of maximum bump height (dimensional)
      c      xb3      - end of bump (dimensional)
      c
      c      dtc      - magnitude of bump
      c      xmax     - reference length of geometry
      c
      c      xin      - input location to get bump value
      c      ymod     - bump height
      c      ymodp    - first derivative of bump wrt xin
      c      ymodpp   - second derivative of bump wrt xin
      c
      x      = xin/xmax
      x1     = xb1/xmax
      x2     = xb2/xmax
      x3     = xb3/xmax

```

```

xd      = 0.0
dxddx  = 0.0
if ( x .ge. x1 .and. x .le. x2) then

    xd = (x - x1)/2.0/(x2 - x1)
    dxddx = 1./2./(x2 - x1)
endif

if ( x .gt. x2 .and. x .le. x3) then

    xd = (x + x3 - 2.0*x2)/2.0/(x3 - x2)
    dxddx = 1./2./(x3 - x2)
endif

ymod    = -64.*dtc*xd**3*(xd - 1.0)**3

ymodp   = -64.*dtc*3.0*dxddx*xd**2*
1      (xd - 1.0)**2*(2.0*xd - 1.0)
ymodpp  = -64.*dtc*6.0*dxddx**2*xd*(xd - 1.0)*
1      (5.0*xd**2 - 5.0*xd + 1.0)

return
end

```

This is a sample main program that can be used to check subroutine bump.

```

c
c   example of use of bump function
c   this is one way to modify an airfoil
c   w.h. mason, Feb. 12, 1994
c
c   set input parameters

xb1     = 0.1
xb2     = 0.4
xb3     = 0.6

dtc     = 0.50

xmax    = 1.0

write(6, 90) xb1,xb2,xb3,dtc
90 format(/3x,'bump example'//
1      3x,'xb1 = ',f7.4,3x,'xb2 = ',f7.4,3x,
2      'xb3 = ',f7.4/3x,'dtc = ',f7.4/
3      /4x,'i',7x,'x/c',7x,'delta y',4x,
4      'd(dy)/dx'3x,'d2(dy)/dy2')

do 10 i = 1,101
xc     = 0.01*(i-1)
call bump(xb1,xb2,xb3,dtc,xmax,xc,ymod,ymodp,ymodpp)
10 write(6,100) i,xc,ymod,ymodp, ymodpp

100 format(i5,4f12.5)

stop
end

```

sample output from the sample main program and subroutine bump.f

bump example

xb1 = 0.1000 xb2 = 0.4000 xb3 = 0.6000
dte = 0.5000

i	x/c	delta y	d(dy)/dx	d2(dy)/dy2
1	0.00000	0.00000	0.00000	0.00000
2	0.01000	0.00000	0.00000	0.00000
3	0.02000	0.00000	0.00000	0.00000
4	0.03000	0.00000	0.00000	0.00000
5	0.04000	0.00000	0.00000	0.00000
6	0.05000	0.00000	0.00000	0.00000
7	0.06000	0.00000	0.00000	0.00000
8	0.07000	0.00000	0.00000	0.00000
9	0.08000	0.00000	0.00000	0.00000
10	0.09000	0.00000	0.00000	0.00000
11	0.10000	0.00000	0.00000	0.00000
12	0.11000	0.00014	0.04154	8.02448
13	0.12000	0.00107	0.15505	14.41646
14	0.13000	0.00343	0.32490	19.31666
15	0.14000	0.00771	0.53686	22.86090
16	0.15000	0.01426	0.77803	25.18004
17	0.16000	0.02333	1.03680	26.40000
18	0.17000	0.03502	1.30277	26.64177
19	0.18000	0.04938	1.56676	26.02140
20	0.19000	0.06633	1.82070	24.65000
21	0.20000	0.08573	2.05761	22.63374
22	0.21000	0.10740	2.27156	20.07387
23	0.22000	0.13107	2.45760	17.06667
24	0.23000	0.15645	2.61171	13.70350
25	0.24000	0.18319	2.73077	10.07078
26	0.25000	0.21094	2.81250	6.25000
27	0.26000	0.23931	2.85540	2.31770
28	0.27000	0.26791	2.85872	-1.65453
29	0.28000	0.29635	2.82240	-5.60000
30	0.29000	0.32423	2.74702	-9.45699
31	0.30000	0.35117	2.63374	-13.16873
32	0.31000	0.37679	2.48430	-16.68333
33	0.32000	0.40074	2.30089	-19.95391
34	0.33000	0.42270	2.08618	-22.93848
35	0.34000	0.44237	1.84320	-25.60000
36	0.35000	0.45948	1.57536	-27.90638
37	0.36000	0.47380	1.28635	-29.83045
38	0.37000	0.48515	0.98010	-31.35000
39	0.38000	0.49336	0.66075	-32.44773
40	0.39000	0.49834	0.33259	-33.11131
41	0.40000	0.50000	0.00000	-33.33333
42	0.41000	0.49626	-0.74625	-73.87733
43	0.42000	0.48515	-1.47015	-70.53750
44	0.43000	0.46700	-2.14989	-65.06483
45	0.44000	0.44237	-2.76480	-57.60000
46	0.45000	0.41199	-3.29590	-48.33986
47	0.46000	0.37679	-3.72645	-37.53750
48	0.47000	0.33784	-4.04253	-25.50236
49	0.48000	0.29635	-4.23360	-12.60004
50	0.49000	0.25361	-4.29304	0.74764

51	0.50000	0.21094	-4.21875	14.06247
52	0.51000	0.16967	-4.01368	26.81011
53	0.52000	0.13107	-3.68640	38.39995
54	0.53000	0.09630	-3.25169	48.18510
55	0.54000	0.06633	-2.73105	55.46249
56	0.55000	0.04187	-2.15332	59.47264
57	0.56000	0.02333	-1.55520	59.39999
58	0.57000	0.01068	-0.98183	54.37266
59	0.58000	0.00343	-0.48735	43.46253
60	0.59000	0.00046	-0.13547	25.68523
61	0.60000	0.00000	0.00000	0.00000
62	0.61000	0.00000	0.00000	0.00000
63	0.62000	0.00000	0.00000	0.00000
64	0.63000	0.00000	0.00000	0.00000
65	0.64000	0.00000	0.00000	0.00000
66	0.65000	0.00000	0.00000	0.00000
67	0.66000	0.00000	0.00000	0.00000
68	0.67000	0.00000	0.00000	0.00000
69	0.68000	0.00000	0.00000	0.00000
70	0.69000	0.00000	0.00000	0.00000
71	0.70000	0.00000	0.00000	0.00000

(rest of the output deleted)

E.6 POSTp

This program reads the data set generated by **PANELv2**, and generates a data file containing the tables needed to make a boundary layer analysis. Although these tables are designed to be used by the boundary layer program **CBLv2**, it does not make the complete data set for **CBLv2**. The user has to construct the initial data input. The program is provided to automate the most tedious aspects of the input preparation.

The program reads the **PANELv2** output data file. Recall that the solution is given continuously starting at the lower surface trailing edge, moving forward around the leading edge, and then moving aft on the upper surface to the trailing edge. Using this data the arc length is calculated and the stagnation point found. If the stagnation point does not occur at an input point, the stagnation point value of the arc length is estimated, and a point is added. The user is then asked to name the output file. The output file is generated as a table of arc length and pressure coefficient values for the lower surface, followed by a table of arc length and pressure coefficient values for the upper surface.

As an example, and to verify the code, we give a sample input, the screen output and a listing of the disk data file. Note that the disk file contains an additional column, set to zero. This is the value of the surface heat flux for use in the boundary layer calculation. We assume that the wall is adiabatic, and the heat flux is zero. The arc length is normalized by the chord length, assumed to be unity. The output format of the tables is 3F10.6.

Sample input:

```
NACA 2412 with 5 deg flap at .75
Alpha    CL      cmc4    CD
5.0000   1.2116   -0.1172   -0.0015
```

```
98.0000000
```

X/C	Y/C	Cp	U/UE
1.0000000	-0.0218722	0.4457454	-0.7444828
0.9989193	-0.0218588	0.3699587	-0.7937514
0.9956822	-0.0218183	0.3248610	-0.8216684
0.9903033	-0.0217497	0.2959175	-0.8390962
0.9828073	-0.0216514	0.2764678	-0.8506070
0.9732280	-0.0215215	0.2633044	-0.8583097
0.9616088	-0.0213579	0.2546606	-0.8633304
0.9480022	-0.0211583	0.2494899	-0.8663198
0.9324694	-0.0209209	0.2471926	-0.8676447
0.9150801	-0.0206440	0.2474237	-0.8675116
0.8959120	-0.0203265	0.2500552	-0.8659936
0.8750503	-0.0199681	0.2552201	-0.8630063
0.8525876	-0.0195686	0.2634676	-0.8582147
0.8286229	-0.0191286	0.2764418	-0.8506222
0.8032618	-0.0186488	0.3003036	-0.8364786
0.7766151	-0.0181297	0.3571291	-0.8017923
0.7487994	-0.0176765	0.3026126	-0.8350973
0.7199356	-0.0196039	0.2852925	-0.8454037

0.6901487	-0.0215702	0.2705841	-0.8540585
0.6595672	-0.0235606	0.2596269	-0.8604494
0.6283228	-0.0255578	0.2511996	-0.8653325
0.5965492	-0.0275420	0.2446724	-0.8690959
0.5643821	-0.0294907	0.2397471	-0.8719248
0.5319584	-0.0313784	0.2363357	-0.8738789
0.4994153	-0.0331776	0.2345359	-0.8749080
0.4668901	-0.0348579	0.2347106	-0.8748082
0.4345195	-0.0363877	0.2381573	-0.8728360
0.4024391	-0.0377340	0.2431501	-0.8699712
0.3710229	-0.0389221	0.2475556	-0.8674355
0.3401839	-0.0399931	0.2525682	-0.8645414
0.3100253	-0.0409071	0.2589526	-0.8608411
0.2806665	-0.0416239	0.2673756	-0.8559348
0.2522234	-0.0421045	0.2785123	-0.8494043
0.2248088	-0.0423124	0.2930953	-0.8407763
0.1985313	-0.0422143	0.3119721	-0.8294745
0.1734958	-0.0417806	0.3361298	-0.8147823
0.1498030	-0.0409861	0.3667493	-0.7957705
0.1275494	-0.0398104	0.4052571	-0.7711958
0.1068267	-0.0382380	0.4533792	-0.7393381
0.0877222	-0.0362578	0.5131727	-0.6977301
0.0703175	-0.0338633	0.5869251	-0.6427090
0.0546887	-0.0310520	0.6766533	-0.5686358
0.0409059	-0.0278245	0.7823370	-0.4665437
0.0290324	-0.0241841	0.8964363	-0.3218132
0.0191246	-0.0201360	0.9877394	-0.1107275
0.0112310	-0.0156868	0.9593768	0.2015521
0.0053924	-0.0108432	0.5824928	0.6461480
0.0016409	-0.0056122	-0.4040008	1.1849054
0.0000000	0.0000000	-1.7449023	1.6567746
0.0005002	0.0058259	-2.6745226	1.9169044
0.0031628	0.0116941	-2.9049768	1.9761014
0.0079837	0.0175852	-2.7717383	1.9420964
0.0149496	0.0234709	-2.5445025	1.8826849
0.0240374	0.0293150	-2.3239744	1.8231770
0.0352145	0.0350744	-2.1347101	1.7705113
0.0484385	0.0407000	-1.9771560	1.7254437
0.0636571	0.0461390	-1.8458352	1.6869603
0.0808082	0.0513357	-1.7348001	1.6537231
0.0998199	0.0562337	-1.6390182	1.6245055
0.1206108	0.0607775	-1.5545312	1.5982901
0.1430902	0.0649138	-1.4783340	1.5742725
0.1671584	0.0685932	-1.4081639	1.5518260
0.1927072	0.0717714	-1.3423257	1.5304658
0.2196209	0.0744106	-1.2795517	1.5098184
0.2477766	0.0764795	-1.2188853	1.4895923
0.2770449	0.0779549	-1.1595500	1.4695407
0.3072913	0.0788213	-1.1008238	1.4494219
0.3383766	0.0790714	-1.0417638	1.4289030
0.3701580	0.0787058	-0.9795462	1.4069635
0.4024706	0.0777333	-0.9185968	1.3851342
0.4349543	0.0762537	-0.8643457	1.3654104
0.4677068	0.0743547	-0.8154098	1.3473715
0.5005847	0.0720664	-0.7705135	1.3306065
0.5334447	0.0694218	-0.7292154	1.3149964
0.5661440	0.0664560	-0.6915128	1.3005817
0.5985411	0.0632060	-0.6577775	1.2875471
0.6304962	0.0597102	-0.6291239	1.2763714
0.6618723	0.0560078	-0.6085705	1.2682943
0.6925348	0.0521391	-0.6073793	1.2678246
0.7223530	0.0481446	-0.6721267	1.2931074
0.7512006	0.0439604	-0.5213552	1.2334323
0.7789551	0.0374099	-0.4203068	1.1917663

E-20 Applied Computational Aerodynamics

0.8054997	0.0309641	-0.3440065	1.1593130
0.8307229	0.0246744	-0.2788006	1.1308407
0.8545192	0.0185923	-0.2196457	1.1043757
0.8767895	0.0127683	-0.1642191	1.0789899
0.8974414	0.0072517	-0.1111093	1.0540917
0.9163895	0.0020905	-0.0593664	1.0292553
0.9335560	-0.0026691	-0.0082251	1.0041041
0.9488706	-0.0069838	0.0429915	0.9782681
0.9622707	-0.0108130	0.0950488	0.9512892
0.9737022	-0.0141200	0.1489126	0.9225440
0.9831185	-0.0168725	0.2061138	0.8910029
0.9904819	-0.0190429	0.2693959	0.8547539
0.9957626	-0.0206093	0.3448462	0.8094157
0.9989396	-0.0215556	0.4457439	0.7444838

Output to screen:

PGM POSTP - POST PROCESS DATA FROM PGM. PANELv2

ECHO OF INPUT DATA:

Enter name of file to be read:
postp.test

Input data:

NACA 2412 with 5 deg flap at .75

Alpha	CL	cmc4	CD
5.00000	1.21160	-0.11720	-0.00150

98.0000000

	X/C	Y/C	Cp	U/UE
1	1.0000000	-0.0218722	0.4457454	-0.7444828
2	0.9989193	-0.0218588	0.3699587	-0.7937514
3	0.9956822	-0.0218183	0.3248610	-0.8216684
4	0.9903033	-0.0217497	0.2959175	-0.8390962
5	0.9828073	-0.0216514	0.2764678	-0.8506070
6	0.9732280	-0.0215215	0.2633044	-0.8583097
7	0.9616088	-0.0213579	0.2546606	-0.8633304
8	0.9480022	-0.0211583	0.2494899	-0.8663198
9	0.9324694	-0.0209209	0.2471926	-0.8676447
10	0.9150801	-0.0206440	0.2474237	-0.8675116
11	0.8959120	-0.0203265	0.2500552	-0.8659936
12	0.8750503	-0.0199681	0.2552201	-0.8630063
13	0.8525876	-0.0195686	0.2634676	-0.8582147
14	0.8286229	-0.0191286	0.2764418	-0.8506222
15	0.8032618	-0.0186488	0.3003036	-0.8364786
16	0.7766151	-0.0181297	0.3571291	-0.8017923
17	0.7487994	-0.0176765	0.3026126	-0.8350973
18	0.7199356	-0.0196039	0.2852925	-0.8454037
19	0.6901487	-0.0215702	0.2705841	-0.8540585
20	0.6595672	-0.0235606	0.2596269	-0.8604494
21	0.6283228	-0.0255578	0.2511996	-0.8653325
22	0.5965492	-0.0275420	0.2446724	-0.8690959
23	0.5643821	-0.0294907	0.2397471	-0.8719248
24	0.5319584	-0.0313784	0.2363357	-0.8738789
25	0.4994153	-0.0331776	0.2345359	-0.8749080
26	0.4668901	-0.0348579	0.2347106	-0.8748082
27	0.4345195	-0.0363877	0.2381573	-0.8728360
28	0.4024391	-0.0377340	0.2431501	-0.8699712
29	0.3710229	-0.0389221	0.2475556	-0.8674355
30	0.3401839	-0.0399931	0.2525682	-0.8645414
31	0.3100253	-0.0409071	0.2589526	-0.8608411
32	0.2806665	-0.0416239	0.2673756	-0.8559348
33	0.2522234	-0.0421045	0.2785123	-0.8494043
34	0.2248088	-0.0423124	0.2930953	-0.8407763

35	0.1985313	-0.0422143	0.3119721	-0.8294745
36	0.1734958	-0.0417806	0.3361298	-0.8147823
37	0.1498030	-0.0409861	0.3667493	-0.7957705
38	0.1275494	-0.0398104	0.4052571	-0.7711958
39	0.1068267	-0.0382380	0.4533792	-0.7393381
40	0.0877222	-0.0362578	0.5131727	-0.6977301
41	0.0703175	-0.0338633	0.5869251	-0.6427090
42	0.0546887	-0.0310520	0.6766533	-0.5686358
43	0.0409059	-0.0278245	0.7823370	-0.4665437
44	0.0290324	-0.0241841	0.8964363	-0.3218132
45	0.0191246	-0.0201360	0.9877394	-0.1107275
46	0.0112310	-0.0156868	0.9593768	0.2015521
47	0.0053924	-0.0108432	0.5824928	0.6461480
48	0.0016409	-0.0056122	-0.4040008	1.1849054
49	0.0000000	0.0000000	-1.7449023	1.6567746
50	0.0005002	0.0058259	-2.6745226	1.9169044
51	0.0031628	0.0116941	-2.9049768	1.9761014
52	0.0079837	0.0175852	-2.7717383	1.9420964
53	0.0149496	0.0234709	-2.5445025	1.8826849
54	0.0240374	0.0293150	-2.3239744	1.8231770
55	0.0352145	0.0350744	-2.1347101	1.7705113
56	0.0484385	0.0407000	-1.9771560	1.7254437
57	0.0636571	0.0461390	-1.8458352	1.6869603
58	0.0808082	0.0513357	-1.7348001	1.6537231
59	0.0998199	0.0562337	-1.6390182	1.6245055
60	0.1206108	0.0607775	-1.5545312	1.5982901
61	0.1430902	0.0649138	-1.4783340	1.5742725
62	0.1671584	0.0685932	-1.4081639	1.5518260
63	0.1927072	0.0717714	-1.3423257	1.5304658
64	0.2196209	0.0744106	-1.2795517	1.5098184
65	0.2477766	0.0764795	-1.2188853	1.4895923
66	0.2770449	0.0779549	-1.1595500	1.4695407
67	0.3072913	0.0788213	-1.1008238	1.4494219
68	0.3383766	0.0790714	-1.0417638	1.4289030
69	0.3701580	0.0787058	-0.9795462	1.4069635
70	0.4024706	0.0777333	-0.9185968	1.3851342
71	0.4349543	0.0762537	-0.8643457	1.3654104
72	0.4677068	0.0743547	-0.8154098	1.3473715
73	0.5005847	0.0720664	-0.7705135	1.3306065
74	0.5334447	0.0694218	-0.7292154	1.3149964
75	0.5661440	0.0664560	-0.6915128	1.3005817
76	0.5985411	0.0632060	-0.6577775	1.2875471
77	0.6304962	0.0597102	-0.6291239	1.2763714
78	0.6618723	0.0560078	-0.6085705	1.2682943
79	0.6925348	0.0521391	-0.6073793	1.2678246
80	0.7223530	0.0481446	-0.6721267	1.2931074
81	0.7512006	0.0439604	-0.5213552	1.2334323
82	0.7789551	0.0374099	-0.4203068	1.1917663
83	0.8054997	0.0309641	-0.3440065	1.1593130
84	0.8307229	0.0246744	-0.2788006	1.1308407
85	0.8545192	0.0185923	-0.2196457	1.1043757
86	0.8767895	0.0127683	-0.1642191	1.0789899
87	0.8974414	0.0072517	-0.1111093	1.0540917
88	0.9163895	0.0020905	-0.0593664	1.0292553
89	0.9335560	-0.0026691	-0.0082251	1.0041041
90	0.9488706	-0.0069838	0.0429915	0.9782681
91	0.9622707	-0.0108130	0.0950488	0.9512892
92	0.9737022	-0.0141200	0.1489126	0.9225440
93	0.9831185	-0.0168725	0.2061138	0.8910029
94	0.9904819	-0.0190429	0.2693959	0.8547539
95	0.9957626	-0.0206093	0.3448462	0.8094157
96	0.9989396	-0.0215556	0.4457439	0.7444838

STAGNATION PT. SEARCH

E-22 Applied Computational Aerodynamics

J	X/C	Y/C	SARC	UE/UINF	CP
40	0.087722	-0.036258	0.913194	-0.697730	0.513173
41	0.070317	-0.033863	0.930764	-0.642709	0.586925
42	0.054689	-0.031052	0.946645	-0.568636	0.676653
43	0.040906	-0.027825	0.960803	-0.466544	0.782337
44	0.029032	-0.024184	0.973225	-0.321813	0.896436
45	0.019125	-0.020136	0.983934	-0.110727	0.987739
46	0.011231	-0.015687	0.993005	0.201552	0.959377
47	0.005392	-0.010843	1.000607	0.646148	0.582493
48	0.001641	-0.005612	1.007069	1.184905	-0.404001
49	0.000000	0.000000	1.012945	1.656775	-1.744902
50	0.000500	0.005826	1.018822	1.916904	-2.674523

STAG PT: XSP= 0.016326 YSP=-0.018558 SSP=0.987150 JS=45 JLE=49

E1 = 0.0032163 E2 = 0.0058546
 ISTAGP = 1

OUTPUT OF POSTp RESULTS

send output to a file? (Y/N):
 y

enter file name:
 postp.out

ALPHA = 5.00000
 MACH NO. = 0.10000
 CL = 1.21160
 CMC4 = -0.11720
 CD = -0.00150
 No. of upper surface values in x/c, Cp table = 52
 No. of lower surface values in x/c, Cp table = 46

lower surface

J	X/C	Y/C	S/C	U/UINF	CP
1	0.016326	-0.018558	0.000000	0.000000	1.000000
2	0.019125	-0.020136	0.003216	-0.110727	0.987739
3	0.029032	-0.024184	0.013925	-0.321813	0.896436
4	0.040906	-0.027825	0.026347	-0.466544	0.782337
5	0.054689	-0.031052	0.040505	-0.568636	0.676653
6	0.070317	-0.033863	0.056386	-0.642709	0.586925
7	0.087722	-0.036258	0.073956	-0.697730	0.513173
8	0.106827	-0.038238	0.093164	-0.739338	0.453379
9	0.127549	-0.039810	0.113946	-0.771196	0.405257
10	0.149803	-0.040986	0.136232	-0.795771	0.366749
11	0.173496	-0.041781	0.159938	-0.814782	0.336130
12	0.198531	-0.042214	0.184977	-0.829475	0.311972
13	0.224809	-0.042312	0.211255	-0.840776	0.293095
14	0.252223	-0.042105	0.238671	-0.849404	0.278512
15	0.280667	-0.041624	0.267118	-0.855935	0.267376
16	0.310025	-0.040907	0.296486	-0.860841	0.258953
17	0.340184	-0.039993	0.326658	-0.864541	0.252568
18	0.371023	-0.038922	0.357516	-0.867436	0.247556
19	0.402439	-0.037734	0.388954	-0.869971	0.243150
20	0.434519	-0.036388	0.421063	-0.872836	0.238157
21	0.466890	-0.034858	0.453470	-0.874808	0.234711
22	0.499415	-0.033178	0.486039	-0.874908	0.234536
23	0.531958	-0.031378	0.518631	-0.873879	0.236336
24	0.564382	-0.029491	0.551110	-0.871925	0.239747
25	0.596549	-0.027542	0.583336	-0.869096	0.244672

26	0.628323	-0.025558	0.615171	-0.865332	0.251200
27	0.659567	-0.023561	0.646480	-0.860449	0.259627
28	0.690149	-0.021570	0.677126	-0.854059	0.270584
29	0.719936	-0.019604	0.706978	-0.845404	0.285293
30	0.748799	-0.017677	0.735910	-0.835097	0.302613
31	0.776615	-0.018130	0.763733	-0.801792	0.357129
32	0.803262	-0.018649	0.790385	-0.836479	0.300304
33	0.828623	-0.019129	0.815751	-0.850622	0.276442
34	0.852588	-0.019569	0.839719	-0.858215	0.263468
35	0.875050	-0.019968	0.862186	-0.863006	0.255220
36	0.895912	-0.020327	0.883050	-0.865994	0.250055
37	0.915080	-0.020644	0.902221	-0.867512	0.247424
38	0.932469	-0.020921	0.919613	-0.867645	0.247193
39	0.948002	-0.021158	0.935147	-0.866320	0.249490
40	0.961609	-0.021358	0.948755	-0.863330	0.254661
41	0.973228	-0.021521	0.960376	-0.858310	0.263304
42	0.982807	-0.021651	0.969956	-0.850607	0.276468
43	0.990303	-0.021750	0.977452	-0.839096	0.295918
44	0.995682	-0.021818	0.982832	-0.821668	0.324861
45	0.998919	-0.021859	0.986069	-0.793751	0.369959
46	1.000000	-0.021872	0.987150	-0.744483	0.445745

upper surface

J	X/C	Y/C	S/C	U/UINF	CP
1	0.016326	-0.018558	0.000000	0.000000	1.000000
2	0.011231	-0.015687	0.005855	0.201552	0.959377
3	0.005392	-0.010843	0.013457	0.646148	0.582493
4	0.001641	-0.005612	0.019919	1.184905	-0.404001
5	0.000000	0.000000	0.025795	1.656775	-1.744902
6	0.000500	0.005826	0.031672	1.916904	-2.674523
7	0.003163	0.011694	0.038141	1.976101	-2.904977
8	0.007984	0.017585	0.045770	1.942096	-2.771738
9	0.014950	0.023471	0.054900	1.882685	-2.544502
10	0.024037	0.029315	0.065711	1.823177	-2.323974
11	0.035214	0.035074	0.078289	1.770511	-2.134710
12	0.048439	0.040700	0.092662	1.725444	-1.977156
13	0.063657	0.046139	0.108826	1.686960	-1.845835
14	0.080808	0.051336	0.126748	1.653723	-1.734800
15	0.099820	0.056234	0.146382	1.624506	-1.639018
16	0.120611	0.060778	0.167665	1.598290	-1.554531
17	0.143090	0.064914	0.190523	1.574273	-1.478334
18	0.167158	0.068593	0.214871	1.551826	-1.408164
19	0.192707	0.071771	0.240618	1.530466	-1.342326
20	0.219621	0.074411	0.267661	1.509818	-1.279552
21	0.247777	0.076480	0.295894	1.489592	-1.218885
22	0.277045	0.077955	0.325200	1.469541	-1.159550
23	0.307291	0.078821	0.355459	1.449422	-1.100824
24	0.338377	0.079071	0.386546	1.428903	-1.041764
25	0.370158	0.078706	0.418330	1.406963	-0.979546
26	0.402471	0.077733	0.450657	1.385134	-0.918597
27	0.434954	0.076254	0.483175	1.365410	-0.864346
28	0.467707	0.074355	0.515983	1.347371	-0.815410
29	0.500585	0.072066	0.548940	1.330606	-0.770513
30	0.533445	0.069422	0.581907	1.314996	-0.729215
31	0.566144	0.066456	0.614740	1.300582	-0.691513
32	0.598541	0.063206	0.647300	1.287547	-0.657777
33	0.630496	0.059710	0.679446	1.276371	-0.629124
34	0.661872	0.056008	0.711040	1.268294	-0.608571
35	0.692535	0.052139	0.741946	1.267825	-0.607379
36	0.722353	0.048145	0.772030	1.293107	-0.672127
37	0.751201	0.043960	0.801185	1.233432	-0.521355
38	0.778955	0.037410	0.829706	1.191766	-0.420307
39	0.805500	0.030964	0.857022	1.159313	-0.344007

E-24 Applied Computational Aerodynamics

40	0.830723	0.024674	0.883018	1.130841	-0.278801
41	0.854519	0.018592	0.907579	1.104376	-0.219646
42	0.876790	0.012768	0.930598	1.078990	-0.164219
43	0.897441	0.007252	0.951975	1.054092	-0.111109
44	0.916390	0.002090	0.971613	1.029255	-0.059366
45	0.933556	-0.002669	0.989427	1.004104	-0.008225
46	0.948871	-0.006984	1.005338	0.978268	0.042992
47	0.962271	-0.010813	1.019274	0.951289	0.095049
48	0.973702	-0.014120	1.031175	0.922544	0.148913
49	0.983118	-0.016872	1.040985	0.891003	0.206114
50	0.990482	-0.019043	1.048661	0.854754	0.269396
51	0.995763	-0.020609	1.054170	0.809416	0.344846
52	0.998940	-0.021556	1.057485	0.744484	0.445744

STOP

Output disk data file:

NACA 2412 with 5 deg flap at .75

5.00000 0.10000 1.21160 -0.11720 -0.00150

lower surface		
s/c	Cp	dT/dy
0.000000	1.000000	0.000000
0.003216	0.987739	0.000000
0.013925	0.896436	0.000000
0.026347	0.782337	0.000000
0.040505	0.676653	0.000000
0.056386	0.586925	0.000000
0.073956	0.513173	0.000000
0.093164	0.453379	0.000000
0.113946	0.405257	0.000000
0.136232	0.366749	0.000000
0.159938	0.336130	0.000000
0.184977	0.311972	0.000000
0.211255	0.293095	0.000000
0.238671	0.278512	0.000000
0.267118	0.267376	0.000000
0.296486	0.258953	0.000000
0.326658	0.252568	0.000000
0.357516	0.247556	0.000000
0.388954	0.243150	0.000000
0.421063	0.238157	0.000000
0.453470	0.234711	0.000000
0.486039	0.234536	0.000000
0.518631	0.236336	0.000000
0.551110	0.239747	0.000000
0.583336	0.244672	0.000000
0.615171	0.251200	0.000000
0.646480	0.259627	0.000000
0.677126	0.270584	0.000000
0.706978	0.285293	0.000000
0.735910	0.302613	0.000000
0.763733	0.357129	0.000000
0.790385	0.300304	0.000000
0.815751	0.276442	0.000000
0.839719	0.263468	0.000000
0.862186	0.255220	0.000000
0.883050	0.250055	0.000000
0.902221	0.247424	0.000000
0.919613	0.247193	0.000000
0.935147	0.249490	0.000000
0.948755	0.254661	0.000000
0.960376	0.263304	0.000000
0.969956	0.276468	0.000000

0.977452	0.295918	0.000000
0.982832	0.324861	0.000000
0.986069	0.369959	0.000000
0.987150	0.445745	0.000000

upper	surface	
s/c	Cp	dT/dy
0.000000	1.000000	0.000000
0.005855	0.959377	0.000000
0.013457	0.582493	0.000000
0.019919	-0.404001	0.000000
0.025795	-1.744902	0.000000
0.031672	-2.674523	0.000000
0.038141	-2.904977	0.000000
0.045770	-2.771738	0.000000
0.054900	-2.544502	0.000000
0.065711	-2.323974	0.000000
0.078289	-2.134710	0.000000
0.092662	-1.977156	0.000000
0.108826	-1.845835	0.000000
0.126748	-1.734800	0.000000
0.146382	-1.639018	0.000000
0.167665	-1.554531	0.000000
0.190523	-1.478334	0.000000
0.214871	-1.408164	0.000000
0.240618	-1.342326	0.000000
0.267661	-1.279552	0.000000
0.295894	-1.218885	0.000000
0.325200	-1.159550	0.000000
0.355459	-1.100824	0.000000
0.386546	-1.041764	0.000000
0.418330	-0.979546	0.000000
0.450657	-0.918597	0.000000
0.483175	-0.864346	0.000000
0.515983	-0.815410	0.000000
0.548940	-0.770513	0.000000
0.581907	-0.729215	0.000000
0.614740	-0.691513	0.000000
0.647300	-0.657777	0.000000
0.679446	-0.629124	0.000000
0.711040	-0.608571	0.000000
0.741946	-0.607379	0.000000
0.772030	-0.672127	0.000000
0.801185	-0.521355	0.000000
0.829706	-0.420307	0.000000
0.857022	-0.344007	0.000000
0.883018	-0.278801	0.000000
0.907579	-0.219646	0.000000
0.930598	-0.164219	0.000000
0.951975	-0.111109	0.000000
0.971613	-0.059366	0.000000
0.989427	-0.008225	0.000000
1.005338	0.042992	0.000000
1.019274	0.095049	0.000000
1.031175	0.148913	0.000000
1.040985	0.206114	0.000000
1.048661	0.269396	0.000000
1.054170	0.344846	0.000000
1.057485	0.445744	0.000000

E.7 STDATM

This subroutine computes the 1976 standard atmosphere. It is used in program FRICTION. It covers an altitude range from sea level to 86 kilometers (282,152 ft.). The results are found in either English or metric units depending on the value of one of the input flags. The 1976 and 1962 standard atmospheres are identical for the first 51 kilometers above sea level.

Method of Computation

Given the geometric altitude Z_{in} (in dimensions of either meters or feet), convert to kilometers. The geopotential altitude H is then found from:

$$H = \frac{Z}{1 + \frac{Z}{r_0}}$$

where $r_0 = 6356.766$ kilometers (the radius of the Earth in kilometers) and $Z = C_1 Z_{in}$, where $C_1 = 0.001$ if Z_{in} is in meters, and $C_1 = 0.0003048$ if Z_{in} is in feet. The 1962 standard atmosphere used a much more complicated and slightly more accurate relationship.

The inverse relation is given by

$$Z = \frac{H}{1 - \frac{H}{r_0}}.$$

Once the geopotential altitude is found, the temperature is computed. The standard day temperature profile is defined by seven layers, where within each layer the temperature is found by the linear relation (T is given in degrees Kelvin):

$$T = T_{b_i} + L_{m_i}(H - H_{b_i})$$

and T_{b_i} , L_{m_i} and H_{b_i} are the values at the base of the particular layer. The following table defines these constants, as well as the ratio of pressure to sea level pressure, which is also needed.

i	H_{b_i} (Km)	T_{b_i} (°K)	L_{m_i} (°K/Km)	P/Psl	Z (ft.)
1	0.	288.15	-6.5	1.0	0
2	11.	216.65	0.0	2.2336×10^{-1}	36,152.
3	20.	216.65	+1.0	5.4032×10^{-2}	65,824
4	32.	228.65	+2.8	8.5666×10^{-3}	105,518
5	47.	270.65	0.0	1.0945×10^{-3}	155,348
6	51.	270.65	-2.8	6.6063×10^{-4}	168,676
7	71.	214.65	-2.0	3.9046×10^{-5}	235,571
-	82.	-	-	-	282,152

Once the temperature is determined, the pressure is computed using the hydrostatics equation and the perfect gas law. The resulting formulas are:

$$\frac{P}{P_{sl}} = \frac{P_b}{P_{sl}} \left(\frac{T_b}{T} \right)^{\frac{K}{L_m}} \quad L_m \neq 0$$

$$\frac{P}{P_{sl}} = \frac{P_b}{P_{sl}} e^{\frac{-K(H-H_b)}{T_b}} \quad L_m = 0$$

where $K = \frac{g_0 M_0}{R^*} = 34.163195$ in consistent units. The remaining fundamental property is the density, which is found using the equation of state as:

$$\frac{\rho}{\rho_{sl}} = \frac{P/P_{sl}}{T/T_{sl}} .$$

Additional parameters of interest in aerodynamics are:

i) The speed of sound

$$a = a_{sl} \sqrt{\frac{T}{T_{sl}}}$$

ii) The coefficient of viscosity, found from Sutherland's Law:

$$\mu = \frac{\beta \cdot T^{3/2}}{T + S}$$

where $S = 110.4^\circ\text{K}$ and β depends on the system of units and is defined below.

iii) The Reynolds number per unit length and Mach:

$$\frac{R_e}{M \cdot L} = \frac{\rho a}{\mu}$$

iv) The actual temperature, pressure and density:

$$T = T_{sl} \left(\frac{P}{P_{sl}} \right)$$

$$P = P_{sl} \left(\frac{P}{P_{sl}} \right)$$

$$\rho = \rho_{sl} \left(\frac{\rho}{\rho_{sl}} \right)$$

and v) the dynamic pressure normalized by the Mach number:

$$\frac{q}{M^2} = \frac{\gamma}{2} P = .7P$$

The sea level properties and other required constants are defined in the following table.

	<u>Metric</u>	<u>English</u>
T_{sl}	288.15 °K	518.67° R
P_{sl}	1.01325×10^5 N/m ²	2116.22 lb/ft ²
ρ_{sl}	1.2250 Kg/m ³	0.0023769 slugs/ft ³
a_{sl}	340.294 m/sec	1116.45 ft/sec
μ_{sl}	1.7894×10^{-5} Kg/m/sec	0.37373×10^{-6} slugs/ft/sec
β	1.458×10^{-6} Kg/m/sec/K ^{1/2}	3.0450963×10^{-8} slugs/ft/sec/K ^{1/2}

The ratio of specific heats, γ , is defined to be 1.40.

User instructions: the comments in the subroutine define the input and output argument list. If the maximum altitude is exceeded, the program returns a non zero value of the validity flag.

```
      subroutine stdatm(z,t,p,r,a,mu,ts,rr,pp,rm,qm,kd,kk)
C
C ***** 1976 STANDARD ATMOSPHERE SUBROUTINE *****
C
C   Mason's BASIC program, converted to FORTRAN - Sept. 1, 1989
C
C   kd -    = 0 - metric units
C           <> 0 - English units
C
C   kk -    0 - good return
C           1 - error: altitude out of table,
C               do not use output
C
C   z  - input altitude, in feet or meters (depending on kd)
C
C   output:
C
C           units:      metric      English
C   t  - temp.          deg K       deg R
C   p  - pressure       N/m^2       lb/ft^2
C   r  - density        Kg/m^3      slugs/ft^3
C   a  - speed of sound m/sec       ft/sec
C   mu - viscosity      Kg/m/sec    slug/ft/sec
C
C   ts - t/t at sea level
C   rr - rho/rho at sea level
C   pp - p/p at sea level
C
C   rm - Reynolds number per      Re/M/m      Re/M/ft
C         Mach per unit of length
C   qm - dynamic pressure/Mach^2  N/m^2       lb/ft^2
```

```

C
  real k, h, mu, ml
  KK = 0
  K = 34.163195
  C1 = 3.048E-04
  IF (KD .eq. 0) go to 1240
  TL = 518.67
  PL = 2116.22
  RL = .0023769
  AL = 1116.45
  ML = 3.7373E-07
  BT = 3.0450963E-08
  GO TO 1260

1240 TL = 288.15
  PL = 101325
  RL = 1.225
  C1 = .001
  AL = 340.294
  ML = 1.7894E-05
  BT = 1.458E-06

1260 H = C1 * Z / (1 + C1 * Z / 6356.766)
  IF (H .gt. 11.0) go to 1290
  T = 288.15 - 6.5 * H
  PP = (288.15 / T) ** ( - K / 6.5)
  GO TO 1420

1290 IF (H .gt. 20.0) go to 1310
  T = 216.65
  PP = .22336 * EXP ( - K * (H - 11) / 216.65)
  GO TO 1420

1310 IF (H .gt. 32.0) go to 1330
  T = 216.65 + (H - 20)
  PP = .054032 * (216.65 / T) ** K
  GO TO 1420

1330 IF (H .gt. 47.0) go to 1350
  T = 228.65 + 2.8 * (H - 32)
  PP = .0085666 * (228.65 / T) ** (K / 2.8)
  GO TO 1420

1350 IF( H .gt. 51.0) go to 1370
  T = 270.65
  PP = .0010945 * EXP ( - K * (H - 47) / 270.65)
  GO TO 1420

1370 IF (H .gt. 71.) go to 1390
  T = 270.65 - 2.8 * (H - 51)
  PP = .00066063 * (270.65 / T) ** ( - K / 2.8)
  GO TO 1420

1390 IF (H .gt. 84.852) THEN
      kk = 1
      write(6,200) H
      return
  END IF

```

```

      T = 214.65 - 2 * (H - 71)
      PP = 3.9046E-05 * (214.65 / T) ** ( - K / 2)

1420  RR = PP / (T / 288.15)
      MU = BT * T**1.5 / (T + 110.4)
      TS = T / 288.15
      A  = AL * SQRT (TS)
      T  = TL * TS
      R  = RL * RR
      P  = PL * PP
      RM = R * A / MU
      QM = .7 * P

200  format('    Out of Table in StdAtm- too high !'//
1    4x, 'H =',f12.3,' > 84.852 km'//)

      return
end

```

The following sample program and output can be used to validate your subroutine:

```

c    main program to check stdatm

c    loop is done twice to get output
c    suitable to include in text(80 col)

c    w.h. mason, Feb. 27, 1994

      real mu

      kd      = 1

      write(6,90)

      do 10 i = 1,21
      z      = 5000.*(i-1)

      call stdatm(z,t,p,r,a,mu,ts,rr,pp,rm,qm,kd,kk)

      if (kk .ne. 0) then
         write(6,120)
         stop
         endif

      write(6,100) z,t,p,r,a,mu
10  continue

      write(6,110)
      do 20 i = 1,21
      z      = 5000.*(i-1)

      call stdatm(z,t,p,r,a,mu,ts,rr,pp,rm,qm,kd,kk)

      if (kk .ne. 0) then
         write(6,160)
         stop
         endif

```

```

      write(6,120) z,ts,rr,pp,rm,qm
20  continue

90  format(/3x,'1976 Standard Atmosphere'//
      1      3x,'      alt      T      P      Rho',
      2      2x,'      a      Mu',
      4      /3x,'      (ft)      (deg R)      (psf)      (s/ft^3)',
      5      2x,'      (f/s)      (slugs/ft/sec)')
100 format(3x,f9.1,f8.2,f8.2,e12.4,f8.2,e12.4)
110 format(/3x,'1976 Standard Atmosphere'//
      1      3x,'      alt      T/Tsl      R/Rsl',
      2      2x,'      P/Psl      Re/M/ft      q/M^2',
      4      /3x,'      (ft)',34x,'(lb/ft^2)')
120 format(3x,f9.1,3f7.4,e10.3,f10.4)
160 format(/4x,'error in return code from stdatm - pgm stops')
      stop
      end

```

Sample output:

1976 Standard Atmosphere

alt (ft)	T (deg R)	P (psf)	Rho (s/ft ³)	a (f/s)	Mu (slugs/ft/sec)
0.0	518.67	2116.22	0.2377E-02	1116.45	0.3737E-06
5000.0	500.84	1760.88	0.2048E-02	1097.10	0.3637E-06
10000.0	483.03	1455.60	0.1756E-02	1077.40	0.3534E-06
15000.0	465.22	1194.79	0.1496E-02	1057.36	0.3430E-06
20000.0	447.42	973.28	0.1267E-02	1036.93	0.3324E-06
25000.0	429.62	786.34	0.1066E-02	1016.10	0.3217E-06
30000.0	411.84	629.67	0.8907E-03	994.85	0.3107E-06
35000.0	394.06	499.35	0.7382E-03	973.14	0.2995E-06
40000.0	389.97	393.13	0.5873E-03	968.08	0.2969E-06
45000.0	389.97	309.45	0.4623E-03	968.08	0.2969E-06
50000.0	389.97	243.61	0.3639E-03	968.08	0.2969E-06
55000.0	389.97	191.80	0.2865E-03	968.08	0.2969E-06
60000.0	389.97	151.03	0.2256E-03	968.08	0.2969E-06
65000.0	389.97	118.93	0.1777E-03	968.08	0.2969E-06
70000.0	392.25	93.73	0.1392E-03	970.90	0.2984E-06
75000.0	394.97	73.99	0.1091E-03	974.26	0.3001E-06
80000.0	397.69	58.51	0.8571E-04	977.62	0.3018E-06
85000.0	400.42	46.35	0.6743E-04	980.95	0.3035E-06
90000.0	403.14	36.78	0.5315E-04	984.28	0.3052E-06
95000.0	405.85	29.23	0.4196E-04	987.59	0.3070E-06
100000.0	408.57	23.27	0.3318E-04	990.90	0.3087E-06

1976 Standard Atmosphere

alt (ft)	T/Tsl	R/Rsl	P/Psl	Re/M/ft	q/M ² (lb/ft ²)
0.0	1.0000	1.0000	1.0000	0.710E+07	1481.3538
5000.0	0.9656	0.8617	0.8321	0.618E+07	1232.6129
10000.0	0.9313	0.7386	0.6878	0.535E+07	1018.9235
15000.0	0.8969	0.6295	0.5646	0.461E+07	836.3538
20000.0	0.8626	0.5332	0.4599	0.395E+07	681.2936
25000.0	0.8283	0.4486	0.3716	0.337E+07	550.4373
30000.0	0.7940	0.3747	0.2975	0.285E+07	440.7683

E-32 Applied Computational Aerodynamics

35000.0	0.7598	0.3106	0.2360	0.240E+07	349.5441
40000.0	0.7519	0.2471	0.1858	0.191E+07	275.1887
45000.0	0.7519	0.1945	0.1462	0.151E+07	216.6139
50000.0	0.7519	0.1531	0.1151	0.119E+07	170.5264
55000.0	0.7519	0.1205	0.0906	0.934E+06	134.2600
60000.0	0.7519	0.0949	0.0714	0.736E+06	105.7186
65000.0	0.7519	0.0747	0.0562	0.579E+06	83.2541
70000.0	0.7563	0.0586	0.0443	0.453E+06	65.6079
75000.0	0.7615	0.0459	0.0350	0.354E+06	51.7925
80000.0	0.7668	0.0361	0.0276	0.278E+06	40.9574
85000.0	0.7720	0.0284	0.0219	0.218E+06	32.4446
90000.0	0.7772	0.0224	0.0174	0.171E+06	25.7445
95000.0	0.7825	0.0177	0.0138	0.135E+06	20.4621
100000.0	0.7877	0.0140	0.0110	0.107E+06	16.2903

STOP