AIM

Association for Information and Image Management

1100 Wayne Avenue, Suite 1100
Silver Spring, Maryland 20910

301/587-8202

Centimeter

Inches

MANUFACTURED TO AIM STANDARDS
BY APPLIED IMAGE, INC.

1 of 1

A Proposed Framework for Computational Fluid Dynamics Code Calibration/Validation*

William L. Oberkampf**
Aerodynamics Dept., Org. 1554
Sandia National Laboratories, Albuquerque, NM 87185-0825

Abstract

The paper reviews the terminology and methodology that have been introduced during the last several years for building confidence in the predictions from Computational Fluid Dynamics (CFD) codes. Code validation terminology developed for nuclear reactor analyses and aerospace applications is reviewed and evaluated. Currently used terminology such as "calibrated code," "validated code," and a "validation experiment" is discussed along with the shortcomings and criticisms of these terms. A new framework is proposed for building confidence in CFD code predictions that overcomes some of the difficulties of past procedures and delineates the causes of uncertainty in CFD predictions. Building on previous work, new definitions of code verification and calibration are proposed. These definitions provide more specific requirements for the knowledge level of the flow physics involved and the solution accuracy of the given partial differential equations. As part of the proposed framework, categories are also proposed for flow physics research, flow modeling research, and the application of numerical predictions. The contributions of physical experiments, analytical solutions, and other numerical solutions are discussed, showing that each should be designed to achieve a distinctively separate purpose in building confidence in accuracy of CFD predictions. A number of examples are given for each approach to suggest methods for obtaining the highest value for CFD code quality assurance.

Introduction

In the past, flight vehicle design and development have been based primarily on wind tunnel and flight testing. Mathematical methods, primarily approximate analytical solutions, have also made important contributions to design and development, but these methods

* This work was performed at Sandia National Laboratories, which is operated by Martin Marietta Corp. for the U. S. Department of Energy under contract No. DE-AC04-94AL85000.

** Senior Member of Technical Staff, Associate Fellow AIAA

were commonly directed toward improving the understanding of the flow physics or developing approximate engineering solutions. Modern computational fluid dynamics (CFD) has evolved for roughly twenty years, the period of time since the appearance of significant computing hardware. During much of those twenty years, CFD has concentrated on the development of improved numerical algorithms and the solution of relatively simple research problems. More recently, a broader range of complex flow physics has been addressed along with grid generation techniques for complex and more realistic geometries. As a result, CFD has begun to make important contributions to actual (as opposed to paper) aircraft, missiles, reentry vehicles, gas turbines, compressors, and rocket engines.

When any numerical simulation tool begins to affect and influence commercial hardware, people with very different backgrounds and perspectives become involved and value judgements of the tool change dramatically. The value judgement of the tool moves from "Look what I can do!" to "How will this tool increase my profits?" The National Research Council committee chaired by Dr. Richard Bradley [1986], summarized it very well by describing the change from Stage IV, Learning to Use Effectively, to Stage V, Mature Capability, as "Most analysis done without supporting experimental comparisons." By this definition, most of us would agree that CFD is definitely not in Stage V.

There are many reasons why CFD has not yet progressed to Stage V (some would argue not even into Stage IV for some applications). Bradley [1986], and more recently Graves [1992] and Cosner [1994], discuss the technical, managerial, and business issues that must be addressed before CFD can have a significant impact on commercial fluid machinery products. The present paper discusses one of the key technical issues, that is, how is confidence built in CFD predictions? Of particular interest to the ground and flight test community is the question: How are experiments designed and conducted so that CFD achieves the maximum benefit from experiment? And the corollary: How do experiments benefit from CFD?

Some experimentalists may question this corollary.

However, consider simplified flow problems where CFD has recently obtained, or is presently capable of obtaining, very high accuracy solutions. For small, well-defined classes of problems, high-quality numerical solutions can now represent the flow physics more accurately than a wind tunnel can simulate free-flight conditions, or possibly even measure flow quantities. Examples of such flows are steady, laminar, supersonic, perfect gas, attached flow over simple bodies (e.g., Blottner, [1990] and Walker and Oberkampf [1992]) and steady, laminar, incompressible flow over simple geometries (e.g., Gresho et al [1993]). Although the classes of highly accurate numerical flow field simulations are very restricted at the present time, it is expected that the number and breadth of these classes of simulations to steadily increase in the coming years.

A clear parallel can be drawn between the state-of-the-art in CFD today and the status of numerical simulation of electrical analog circuits of the past. Until twenty or thirty years ago, the response of complex analogue circuits was determined by bread-board experiments. Since that time, the numerical simulation capability, in both hardware and software, for solving ordinary differential equations has completely changed the technology. Designers now do little or no bread-board experiments, but numerically simulate the response of circuits. Ordinary differential equation (ODE) codes are clearly in Stage V. Today the key issues in circuit simulation are related to characterization of electrical components as a function of voltage, current, temperature, humidity, electric field, etc. Progress to Stage V for ODE solvers has been extremely rapid for two reasons. First, the mathematical complexity of solving ODEs is generally miniscule relative to partial differential equations (PDEs). Second, the variety and complexity of fluid flow physics dwarf that of electrical circuits.

The issue of building confidence in the accuracy of a CFD prediction and formally defining terms such as "verification" and "validation" were initially done by the Standards Committee of the American Nuclear Society (ANS). This effort was initiated because of the extraordinary responsibility and liability placed on nuclear reactor operation and safety analyses. During the mid-1980s this committee studied the issue of verification and validation and the result of their work was published in "Guidelines for the Verification and Validation of Scientific and Engineering Computer Programs for the Nuclear Industry" [ANS, 1987]. The terms "verification" and "validation" were defined, but the definitions are very general and nebulous for prediction accuracy. The emphasis in the definitions was on computer programming consistency, record keeping, and "reasonableness" of solutions.

Nearly concurrent with the ANS Standards

Committee, the National Aeronautics and Space Administration (NASA) formed an ad hoc committee to discuss CFD validation issues and to review validation activities at the NASA research centers. Although the work of the ad hoc committee was not available for public dissemination, the essence of the ideas and terminology developed were published by Bradley [1988] and Marvin [1988]. This work went well beyond the ANS efforts in terms of code requirements and clarity of definitions. They defined code validation and calibration, in addition to types of CFD validation experiments. Although the new definitions for codes provided more detail to the meaning of terms, it has been found that the implementation and usage of these terms is still lacking. The present confusion, even disarray, toward the meaning of CFD code validation/calibration can be seen in hundreds of published CFD papers. It is all too common for a CFD paper to have the following structure: a description of the numerical method, "good" comparison with experimental results or another code, and then declaration of code "validation." This sad state of affairs is even more frustrating because most people believe in the revolutionary impact that CFD will achieve, but the present state is one of promise not fulfilling reality.

Martellucci [1990], Mehta [1990, 1991, 1993], Blottner [1990], Marvin and Holst [1990], Neumann [1990], and others have added new viewpoints to the CFD validation issue. Although a number of individuals are trying to use the NASA terminology, it is clear that the field is struggling with this issue. There seems to be more agreement concerning the NASA definitions for types of CFD validation experiments and how they contribute to code validation. The definitions concerning experiments provided clear recognition of the unique purposes of experiments, although most experimentalists have not yet embraced these ideas.

This paper proposes a new framework and perspective for the validation of CFD codes. First, a detailed, critical review of past work in this area is given, noting the strengths and weaknesses of past definitions and viewpoints. A new framework is proposed for building confidence in CFD code predictions that overcomes some of the difficulties of past procedures and delineates the causes of uncertainty in CFD predictions. Building on previous work, new definitions of code verification and calibration are proposed which provide more specific requirements for knowledge level of the flow physics involved and solution accuracy of the given partial differential equations. A distinction is drawn between fluid dynamics research and fluid dynamics prediction or application. An extensive discussion is then given concerning the most effective manner for physical experiments, analytical solutions, and independent numerical predictions to contribute in the proposed

framework. It is shown how CFD code verification and calibration experiments are distinctively different, and more difficult, than the more traditional "production" experiments.

Review of Existing Terminology

The ANS Standards Committee, Working Group ANS-10.4, first studied the issue of code validation. In 1987 they defined the code validation process as:

"Validation: The process of testing a computer program and evaluating the results to ensure compliance with specific requirements."

This definition of the validation process is so general that it seems to have very limited value. The key to the definition is the phrase "compliance with specific requirements." Because "specific requirements" are purposely not defined (to make the definition as generally applicable as possible) the definition is not useful in itself. The substance of the matter must be provided in the additional information.

The ANS Standards Committee also defined code verification process as [ANS, 1987]:

"Verification: The process of evaluating the products of a software development phase to provide assurance that they meet the requirements defined for them by the previous phase."

This definition is also so general that it seems to provide very limited value. Its value hinges on the definition of "requirements defined for them by the previous phase." Because those requirements are not given in the definition (as with the definition for validation), little value is added to understanding or to the construction of metrics. However, this definition gives a suggestion of requiring consistency of products (e.g., programming or predictions), from one phase to another.

One may ask why include these definitions in this paper if they are of such low utility. They were included and discussed for two reasons. First, it was felt that to review the history and status of CFD validation, the first published definitions of these terms must be presented. Second, if one accepts that the authors of these definitions (roughly 20 individuals from industry and government) were intelligent and innovative, why did they produce such a definition? The answer to this question is important because it demonstrates a distinct perspective toward the entire issue of code validation. This perspective argues that because of the extreme variety of fluid flows and accuracy prediction requirements, these requirements should be defined in a separate document for each applica-

tion; not in the definition of "validation." It should also be mentioned that the Institute of Electrical and Electronics Engineers (IEEE) [1984] and the International Organization for Standardization (ISO) [1993] share a similar perspective to ANS.

The NASA committee took a divergent view in defining the terms CFD code "calibration" and "validation." The term "calibration," as will be seen, is the less demanding of the two and was defined as follows [Bradley, 1988]:

"CFD Code Calibration: The comparison of CFD code results with experimental data for realistic geometries that are similar to the ones of design interest, made in order to provide a measure of the code's capability to predict specific parameters that are of importance to the design objectives without necessarily verifying that all the features of the flow are correctly modeled." (emphasis included in definition)

This definition provides much more precision to the meaning of the term and is quite demanding. The following are key elements of the definition:

- The code results must be compared with experimental data.
- The experimental data must be for "realistic" and "similar" geometries to those of interest.
- The code must have the capability to predict "specific" and "important" parameters.
- There may be "features" of the flow that are not predicted.

Mehta [1991,1993] provides several criticisms of this definition but the most fundamental is that the phrase "to provide a measure of the code's capability to predict" is ambiguous. He points out that the definition does not define what level of accuracy is required, that is, only a "measure" of capability is required. Although this is a valid criticism, the definition does give some guidance to the accuracy required by stating "specific" and "important" parameters. To keep the definition general, it is left to the user of the code to define what is "specific" and "important" for the particular application at hand.

Although the NASA definition provides important metrics to determine if a code is calibrated, there are some notable shortcomings. First, because the data may not exist, comparison with experimental data should not be required. An assumption that valid experimental data exists is unrealistic for many practical applications, such as a nuclear reactor during a failure mode; a hypersonic, airbreathing propulsion system at the proper total enthalpy; inside the fireball of a nuclear detonation; and

entry of a probe into a planetary atmosphere. However, an argument is not being made against comparison with experimental data; experimental data should be used *if possible*. Second, experimental data should not be restricted only to "realistic" and "similar" geometries; it should also include "realistic" and "similar" flow physics. Third, comparisons should also be made, if possible, with approximate analytical solutions. Comparisons with analytical solutions will improve confidence in computed results for certain ranges of parameters in the fluid dynamics. For example, code solutions can be compared with analytical solutions for special cases of flow physics or geometry and can also be compared with perturbation solutions for certain ranges of parameters. Fourth, comparisons should also be made, if possible, with other credible numerical solutions, such as comparisons with other CFD codes, other CFD formulations (e.g., finite volume vs finite element), and other mathematical approaches (e.g., Monte Carlo solutions and Chaos theory simulations). The key element in comparing code results with other results is the confidence one has in the validity of the data used for comparison.

The NASA committee defined CFD code "validation" as follows [Bradley, 1988]:

"CFD Code Validation: Detailed surface-and-flow-field comparisons with experimental data to verify the code's ability to accurately model the critical physics of the flow. Validation can occur only when the accuracy and limitations of the experimental data are known and thoroughly understood and when the accuracy and limitations of the code's numerical algorithms, grid-density effects, and physical basis are equally known and understood over a range of specified parameters." (emphasis included in definition)

This definition is the most demanding of all terminology offered. Not only does it require all of the elements of "calibration" mentioned above, it also requires the following:

- Detailed comparison of predicted and experimentally measured surface-and-flow-field data.
- Accurate modeling of the critical physics.
- Accuracy and limitations of the experimental data are thoroughly known and understood.
- Accuracy and limitation of the code's numerical algorithms and grid-density effects are thoroughly known and understood over a range of specified parameters.
- Accuracy and limitation of the code's physical models are thoroughly known and

understood over a range of specified parameters.

Since this definition was published, a number of authors have suggested modifications to it or criticized it directly. Martellucci [1990] has said, "In principle, validation is a continuous never ending process." Marvin and Holst [1990] state, "The development process... presents a dilemma because code validation is actually an open-ended process brought about by the ever-widening scope and accelerated pace of possible applications compared with the ability to provide experimental validation and by algorithm enhancements that are progressively being introduced." And finally, Neumann [1990] states, "Validation... is an ongoing process that parallels the development of CFD tools." In response to these suggested modifications, Mehta [1993] remarks, "Such declarations may be appropriate in all types of research, except programmatic research, or may be used for obtaining carte blanche to work on code validation . . . The outcome of an open-ended CFD validation process is of doubtful use in a design application."

In addition to these suggested modifications and criticisms, it is suggested that there are additional shortcomings with the NASA definition of code validation. Detailed surface-and-flow-field data may not exist, or even be possible to obtain (examples were given earlier). Also, accurate modeling of the critical physics may not be possible at the present time: Examples include boundary layer transition modeling, steam-air explosions, two-phase flow with combustion and turbulence, and noncontinuum and/or nonequilibrium reacting flows. With these criticisms and those expressed by other authors, it should be evident that it is impossible that a code can be validated, using the NASA definition, over a wide range of flow problems. The proposed framework discussed in the next section proposes a different approach to this dilemma.

Besides the ANS and NASA definitions, the IEEE has also defined terminology for software quality assurance. The IEEE Standard Dictionary of Electrical and Electronics Terms [1984] gives the definition for code verification as "Formal proof of program correctness". Although very brief, this definition brings unprecedented clarity to the meaning of the term, and it adds a new perspective to the issue. Specifically, this definition bluntly requires correctness or veracity of the prediction, without bringing in supporting topics such as "what" is being predicted or "how" it is done. If the wisdom of this perspective is not clear, it will hopefully become clear in the next section. The IEEE and ISO definitions for code validation are very similar to that of the ANS discussed earlier.

Boehm [1981] and Blottner [1990] use IEEE's definition of verification essentially unaltered and added its

mirror image to define validation. They define code verification as "Solving the governing equations right" and code validation as "Solving the right governing equations". These two succinct definitions clearly and pointedly segregate two essential elements in code quality assurance. Code verification deals only with the demonstration of numerical solution correctness or fidelity produced by a code. It does not necessarily have any relationship to nature, i.e., real fluid dynamics. Code validation deals with the much more encompassing issue of accurately simulating (i.e., numerically modeling) a real, physical phenomena. Some may argue that this definition for code verification is then only of academic interest, or provides no benefit to "real world" problems. However, one must realize that the realm of fluid dynamics is so vast that the complexity must be segregated into categories that are workable and understandable. These definitions divide the complexity into two fundamental ideas: First, the accuracy of solving a given set of partial differential equations, and second, the accuracy of the mathematical model of the physics. The proposed framework for code quality assurance presented in the next section builds upon these fundamental ideas.

Proposed Framework for Building Confidence in CFD Codes

To build a logical, consistent, and workable framework for the meaning of CFD quality assurance terminology, one must grasp that CFD has a more diverse foundation and realm of applications than analytical methods. Most fluid dynamics researchers and users of codes view CFD as an extension, or outgrowth, of traditional analytical methods in mathematical physics. Analytical methods, i.e., closed form, exact, or approximate solutions to the equations of interest, are built on well-defined, very reproducible, and rigorous methods of mathematical analysis. As is well known, the dominant shortcoming of analytical methods is that they address a much narrower range of flow physics and geometry than computational methods. The accuracy of analytical predictions fundamentally depends on the accuracy of the mathematical model of the physics; the rigor of the mathematical methods is rarely an issue. On the other hand, CFD prediction accuracy also depends on the equivalence of the discrete model to the continuum model *and* the accuracy of the solution to the discrete model. The accuracy of these components can be difficult to determine. Some of the more subtle examples are shock capturing, boundary conditions imposed at a finite distance from the region of interest, approximate factorization, and one-dimensional Riemann solvers applied to multidimensional flows. Delineating numerical errors is beyond the scope of this paper, but the important issue here is that these errors can be major, and they are entirely separate from flow modeling errors (Ferziger [1993], Anderson,

Tannehill and Pletcher [1984] and Roache [1993]). As a result, the proposed framework categorizes these errors separately from flow modeling errors.

The proposed framework divides fluid dynamics into research and prediction/application activities. This division clearly distinguishes between (1) activities directed toward seeking fundamental physical understanding or the mathematical representation of the physics, and (2) activities directed toward real world applications. As pointed out earlier, mathematical physics ("Look what I can do!") is a fundamentally different perspective than programmatic or business confidence in results ("How will this tool increase my profits?"). Fluid dynamics research is further divided into flow physics and flow modeling research. The proposed activities framework is similar to Mehta [1991] who refers to uncertainties inherent in CFD. He identifies the sources of uncertainties as computational, fluid dynamics, and human factors. Although it is beyond the scope of this paper to include the human factors perspective, Mehta's categories of fluid dynamics and computational uncertainties correspond to the present categories of flow physics/flow modeling and prediction/application, respectively. Fluid dynamics prediction/application activities are divided into code verification and calibration activities. (The definition of these terms will be given later in this paper.) The approach to each of these activities is categorized as either experimental, analytical, or numerical. Figure 1 graphically shows this framework.

Flow physics and flow modeling research are herein defined as:

Flow Physics Research: Research, either experimental, analytical, or numerical, that improves the fundamental understanding of flow physics.

Flow Modeling Research: Research, either experimental, analytical, or numerical, that develops, improves, or verifies mathematical models of flow physics.

These two activities are similar to NASA-defined categories of experimentation related to the development of CFD capability (Marvin [1988] and Bradley [1988]). The present framework, however, recognizes that research and prediction/application activities are also advanced by analytical and numerical solutions. Flow physics experiments, analytical solutions, or numerical simulations are characterized by very simple geometries and usually one type of complex flow physics. Flow modeling experiments, analytical solutions, or numerical simulations are characterized by simple geometries and one or possibly two types of complex flow physics. Examples of flow physics and flow modeling activities are

		FLUID DYNAMICS		ACTIVITIES	
		Fluid Dynamics Research		CFD Prediction/Application	
		Flow Physics	Flow Modeling	Code Verification	Code Calibration
		<u>Definition</u> Research, either experimental, analytical, or numerical, that improves the fundamental understanding of flow physics.	<u>Definition</u> Research, either experimental, analytical, or numerical, that develops, improves, or verifies mathematical models of flow physics.	<u>Definition</u> Activities, either experimental, analytical, or numerical, that build confidence in a code's ability to predict well known physics or a specified model of physics.	<u>Definition</u> Activities, either experimental, analytical, or numerical, that build confidence in a code's ability to predict flow variables when the physics is incompletely or poorly known.
Approach	Experimental	<u>Characterization</u> Flow physics experiments, analytical solutions, and numerical solutions are characterized by very simple geometries and commonly one type of complex flow physics.	<u>Characterization</u> Flow modeling experiments, analytical solutions, and numerical solutions are characterized by simple geometries and one or possibly two types of complex flow physics.	<u>Characterization</u> Code verification experiments, analytical solutions, and numerical solutions are characterized by simple or moderately complex geometries and well known or specified physics.	<u>Characterization</u> Code calibration experiments, analytical solutions, and numerical solutions are characterized by simple or moderately complex geometries and incompletely or poorly known physics.
	Analytical				
	Numerical				

Figure 1: Proposed Framework for Fluid Dynamics Activities and Approaches

shown in Figs. 2 and 3, respectively. Many other examples could be given for each element of the matrix, depending on one's interest and experience. The important aspect of the framework for research activities is that investigators recognize the distinct goal of each activity so that maximum benefit is derived. In addition, for the advancement of fluid dynamics research, analytical and numerical approaches can be as valuable as physical experiments.

Code verification and calibration are activities associated with building confidence in the prediction accuracy of numerical simulations. Although the simulations also improve the understanding of the fluid dynamics being analyzed, the understanding is for the purpose of hardware design or the reduction of uncertainties associated with a design. The contributing mathematical discipline of numerical analysis could be categorized separately, but it will be considered as part of code verification and calibration for the present discussion. Code verification is herein defined as:

Code Verification Activities: Activities, either experimental, analytical, or numerical, that build confidence in a code's ability to predict well-known physics or a specified mathematical model of physics.

The definition of code verification activities is based on Boehm's [1981] and Blotner's [1990] definition: "Solving the governing equations right". Code verification deals with the demonstration of numerical solution correctness and fidelity to a given continuum mathematics model, as opposed to a discrete model. The present definition goes one step further by requiring either well-known physics *or* a specified mathematical model. This connection is made because these activities are a precisely known subset of fluid dynamics prediction and application. In the literature this subset of problems is commonly called benchmark problems. Examples of well-known fluid physics are incompressible and (ideal gas) compressible laminar flows of a Newtonian fluid. Examples of flows with a specified mathematical model are inviscid flow and turbulent flow with an algebraic turbulence model. Code verification experiments, analytical solutions, or numerical simulations are characterized by simple or moderately complex geometries. Figure 4 shows examples of code verification activities.

As with flow physics and flow modeling activities, the unique contribution of experimental, analytical, and numerical approaches is recognized by this framework. Physical experiments for code verification *only* deal with well-known flow physics. In addition, code verification

FLUID PHYSICS RESEARCH	
PHYSICAL EXPERIMENTS	<u>Examples</u> Experiments designed to improve the fundamental understanding of: boundary layer transition; turbulence; high temperature gas chemistry; reacting (combusting) flow; two phase flow; liquid solidification and melting; non-continuum flows; etc.
ANALYTICAL SOLUTIONS	<u>Examples</u> 1) Mathematical existence and uniqueness theory to improve fundamental understanding of flow physics. 2) Application of hydrodynamic stability theory to improve understanding of boundary layer transition. 3) Application of chaos theory to improve understanding of turbulence.
NUMERICAL SOLUTIONS	<u>Examples</u> 1) Use of large eddy simulations to improve understanding of turbulence. 2) Use of direct numerical simulation to improve understanding of boundary layer transition. 3) Use of Maxwell's equations to improve understanding of non-continuum flows.

Figure 2: Examples of Flow Physics Research

experiments must also accurately measure all the boundary conditions required for the numerical simulation. Only the combination of an accurate mathematical model of the physics and accurately known boundary conditions is of value in code verification activities (See, for example, Armaly et al [1983] and Oberkampf and Aeschliman [1992]). Suppose one compares CFD predictions with accurate experimental measurements of a verification experiment. If the CFD predictions and the measurements disagree, then it can be unequivocally stated that a numerical simulation error exists. There can be no room for questioning the continuum mathematics, e.g., the Navier-Stokes equations, or pointing to a modeling issue, e.g., a turbulence model, as the culprit. The problem is the code; the numerical algorithm, lack of grid convergence, lack of iterative convergence, artificial viscosity, inaccurate numerical boundary conditions, programming error, etc.

Analytical solutions also provide significant value to code verification (Fig. 4). Compared to physical experiments, analytical verification covers a wider range of mathematical models. Analytical verification can be used for either well-known physics or specified mathematical models. Note that the specified mathematical model need not be physically real or physically accurate. At the present time, analytical verification seems to be distinctly out of favor with CFD proponents. There are a surprisingly large number of exact and approximate analytical solutions to the Euler and Navier-Stokes equations

FLUID MODELING RESEARCH	
PHYSICAL EXPERIMENTS	<u>Examples</u> Experiments designed to develop, improve, or verify mathematical models for: turbulence near boundary layer separation and reattachment; shock wave/turbulent boundary layer interaction; combustion/turbulence interaction; dispersed phase/turbulence interaction; liquid droplet break-up; etc.
ANALYTICAL SOLUTIONS	<u>Examples</u> 1) Development of more fundamental equations than the Navier-Stokes equations for rarefied flows. 2) Development of mathematical models to support Stokes assumption ($\lambda = -2/3 \mu$). 3) Development of mathematical models to support the Boussinesq approximation for turbulent flow.
NUMERICAL SOLUTIONS	<u>Examples</u> 1) Use of large eddy simulations to improve or verify $k-\epsilon$ and higher order turbulence models. 2) Use of direct numerical simulations to improve boundary layer stability models for transition. 3) Use of Maxwell's equations to improve direct simulation Monte Carlo models for rarefied flows.

Figure 3: Examples of Flow Modeling Research

(Karamcheti [1966], Panton [1984], Hayes and Probstein [1966], and Wang [1991]). For example, an exact solution is known for fully-developed, laminar, incompressible flow in a non-circular duct. Concerning approximate analytical solutions, highly accurate turbulent flow solutions using Prandtl's mixing length model are known for flow over a flat plate. One can only guess how often comparisons are made between analytical solutions and CFD predictions during the development of a code because these are not typically publishable results. However, one suspects it occurs rarely.

Independent numerical solutions can also aid the verification of a code (Fig. 4). Two types of code-to-code comparisons can be made. The first type comparison is between a code that numerically solves the given partial differential equations and a second code that solves a reduced form of the PDE resulting in an ordinary differential equation. This type code-to-code comparison can be extremely valuable, because the accuracy of the ODE solution is usually as good as an analytical, i.e., closed form, solution. Examples of these ODE solutions are: (1) fully developed laminar boundary layer on flat plate, (2) inviscid, supersonic flow over a sharp cone, and (3) incompressible laminar flow near a rotating disk of infinite diameter (Wang [1991] gives additional examples).

The second type of code-to-code comparison is between two codes that numerically solve the same partial differential equations. For example, if both codes use

CFD CODE VERIFICATION	
PHYSICAL EXPERIMENTS	<u>Examples</u> Experiments designed to build confidence in a code's ability to accurately predict: laminar, incompressible flow over a back-step; laminar, incompressible flow in a curved or turn-around duct; laminar, ideal gas, hypersonic flow over simple and complex vehicle geometries; etc.
	<u>Examples</u> 1) Inviscid, incompressible flow over an irregular object, in a curved duct, over a Joukowski airfoil (with and without circulation). 2) Fully developed, laminar, incompressible flow in a wide range of non-circular ducts. 3) Incompressible, turbulent flow (using Prandtl's mixing length model) with a specified pressure gradient.
	<u>Examples</u> 1) Ordinary differential equation solutions for: incompressible, laminar boundary layer on a flat plate; inviscid, ideal gas, supersonic flow over a sharp cone; incompressible, laminar flow near a rotating disk of infinite diameter; etc. 2) Separate partial differential equation solutions using dissimilar numerical techniques such as: finite volume vs. finite element techniques; finite difference vs. spectral methods; structured vs. unstructured grids.

Figure 4: Examples of Code Verification Activities

similar differencing schemes, the same structured grid, and similar solutions procedures, then the comparison adds limited quality assurance. However, if the codes use widely different numerical techniques, then significant value can be added. For example, if one code was a finite difference code with a structured grid and shock fitting, and the other code was a finite element code with an unstructured grid and shock capturing, then improved confidence can be gained by good agreement between the codes. If disagreement is noted between the code predictions, then a vigorous investigation should be undertaken to determine the cause and a means for its elimination. Admittedly, however, it can sometimes be very difficult to determine which code, if either, is correct.

Some argue that code-to-code comparisons are useless, because no comparison is made to the real world. To the contrary, code-to-code comparisons can be very valuable, but one must understand clearly what value is added to code quality assurance. Code-to-code comparisons in verification activities evaluate the accuracy of the numerical solution of the discrete formulation relative to the original continuum formulation, i.e., the PDE's. That is, are we "solving the governing equations right"? These types of code-to-code comparisons make no claim relative to the accuracy of the flow physics or necessarily to the "real world." As stressed earlier, validity of numerical simulations depends on the accuracy of the flow physics and the accuracy of the numerics.

CFD CODE CALIBRATION	
PHYSICAL EXPERIMENTS	<u>Examples</u> Experiments designed to build confidence in a code's ability to predict: reacting flow in liquid and solid rocket motors, hypersonic flows in chemical and thermodynamic non-equilibrium; rarefied flows; liquid droplet impact with solid boundaries; ionized flows; atomization of a liquid jet; etc.
	<u>Examples</u> 1) Approximate solutions for limiting cases of geometry which result in a decrease in dimensionality. 2) Approximate solutions for compressible, ideal gas, laminar and turbulent boundary layers. 3) Perturbation solutions for a wide range of ideal gas, laminar, compressible and incompressible flows.
	<u>Examples</u> 1) Similarity solutions of the given partial differential equations that yield ordinary differential equations. 2) Traditional numerical solution techniques (such as finite difference) compared with vortex methods, lattice gas methods, and chaos methods for turbulence.

Figure 5: Examples of Code Calibration Activities

Some say that code verification activities, whether comparison with experimental, analytical, or numerical results, are unnecessary. They note that several hundred CFD papers are published each year dealing with much more complex flow physics and geometries than suggested here in verification activities. They also note that commercial CFD codes show extraordinarily complex physics and geometries in their marketing material. Our response is threefold. First, because many published CFD results do not compare with experimental data, one asks, "How good is the color fluid dynamics?" For those that do compare, the criteria for "generally good," "good," and "very good" agreement with experiment are elusive. Any disagreement between computation and experiment in portions of the flow is commonly dismissed with statements such as "probably an experimental error" or "that region is of little interest." If you are a program manager with responsibility for flight vehicle crew safety or commercial product liability, these dismissals do not comfort you. Second, researchers conducting very careful and meticulous CFD studies are publishing results in the class of code verification defined in this paper. These type computations are usually called benchmark calculations. Example include two-dimensional, incompressible, laminar flow over a backstep (Gresho et al [1993]); two-dimensional, incompressible, unsteady laminar flow over a cylinder (Shimazaki et al [1993]); and two-dimensional, ideal gas, supersonic flow over a sphere (Blottner [1990]). Third, the author of this paper has both participated in and observed a number of "compute-offs" during the last ten

years. Each of these exercises has been a disappointment because what was initially hoped for was quite different from what was actually accomplished. Two examples of these types of activities are (1) comparisons of codes for laminar, ideal gas, hypersonic flow over a spherically blunted cone (Greene, [1993]), and (2), comparison of commercial codes on relatively simple benchmark problems (Freitas, [1993]).

Code calibration activities are herein defined as:

Code Calibration Activities: Activities, either experimental, analytical, or numerical, that build confidence in a code's ability to predict flow variables when the physics is incompletely or poorly known.

The definition of code calibration activities stresses prediction of real-world flow fields, that is, flow fields of particular interest in programmatic or design applications. The proposed definition of calibration is similar to Boehm [1981] and Blotner's [1990] definition of validation, "Solving the right governing equations," but theirs is a much stronger statement. Theirs implies that the mathematical model of the fluid physics is accurate. Our definition leaves room for inaccuracy so that a much broader range of physics can be included. Examples are numerous because most real-world applications include fluid physics that are *not* well known, such as reacting flow in liquid and solid rocket motors, hypersonic flow with gas chemistry, and atomization of a liquid jet. Code calibration experiments, analytical solutions, and numerical simulations are characterized by simple or moderately complex geometries and incompletely or poorly known physics. Examples of physical experiments that contribute to code calibration are shown in Fig. 5.

Analytical solutions can also contribute to code calibration (Fig. 5) but the number of analytical solutions is much smaller than for code verification. For code calibration, the analytical solution should approximately represent flow physics, but it need not model it exactly. Recall that code verification also includes specified mathematical models for flow physics that do not have to be accurate representation of real flow physics. As a result, analytical solutions for code calibration are actually a subset of analytical solutions for code verification.

The final category of code calibration is code-to-code comparisons (Fig. 5). As with analytical solutions, there is an overlap with code verification, for example, a comparison between two codes with approximate flow physics but different solution techniques. However, there is an additional type of code-to-code calibration that cannot be considered in code verification. This is the comparison of the predictions of two codes that have a fundamentally different mathematical model of the flow physics, but

both approximately represent reality. Examples of these are comparison of traditional finite difference solutions with predictions from vortex methods, lattice gas methods, and chaos methods for turbulence. These three methods do not solve the Navier-Stokes equations *per se*, but solve other models for fluid dynamics. Other numerical formulations for fluid dynamics may also be invented in the future, and they would reside in this element of the proposed framework.

The final topic to be addressed in this section is the proposed definitions for verified and calibrated codes. The definition of these terms has been delayed until now so that the structure and logic of the framework would be clear. The following definition is proposed for a verified code:

Verified code: A code that has demonstrated accurate solutions for specified variables to well-known physics or a particular mathematical model of physics, for a specified class of problems.

Two elements of this definition have not been discussed earlier in this section: "for specified variables" and "for a specified class of problems." The restriction to specified variables is made to call attention to the different levels flow detail and the corresponding difficulty for accurate prediction of these levels. This feature was also addressed in NASA's definition of code calibration. For example, if one considers predicting the external aerodynamics on an atmospheric flight vehicle, then the hierarchy, i.e., increasing levels of predictive difficulty, would be:

- Total body forces and moments
- Control surface forces and moments
- Surface pressure distributions
- Surface heat flux and shear stress
- Flow field distributions of pressure, temperature, and velocity components
- Flow field distributions of Reynolds stresses

What one chooses for "specified variables" will depend on the particular programmatic or design needs at the time. If one chooses the minimal levels of flow detail, for example, total body forces and moments, it is clear one cannot have much confidence in the code verification level achieved.

The second new element included in the definition of a verified code was "for a specified class of problems." This phrase was added to restrict the verification confidence range, and the difficulty of demonstrating verification, to a defined and manageable set of problems. If this restriction is not included, then the task of

verification is unending because of the extraordinary range of fluid dynamics complexity. Working in one area of fluid dynamics, particularly aerodynamics, one tends to forget the remarkable breadth of fluid flows that can exist. The impossibility of the task for a "general CFD code" has been recognized, as mentioned earlier.

How one might define the class of problems and how restrictive one might be is arbitrary. Some restrictions might be imposed concerning spatial dimensions and temporal nature, e.g., two-dimensional, steady flow. Restrictions on types and complexity of geometry would also be appropriate. How the classes of geometry are defined would be a difficult issue. Finally, restrictions on the flow physics would most certainly be imposed, for example:

- Inviscid or viscous flow
- Newtonian or non-Newtonian fluids
- Laminar or turbulent flow
- Ideal gas or gas chemistry
- Nonionized or ionized flow
- Non-reacting or reacting flow
- Thermal equilibrium or non-equilibrium
- Solid surfaces or free surfaces
- Single phase or multiphase flow

Summarizing, the specification of class of flow problems would include the following elements:

- Spatial dimensionality
- Temporal nature
- Geometry class
- Flow physics

The following definition is proposed for a calibrated code:

Calibrated code: A code that has demonstrated the required prediction accuracy for specified variables, for which the flow physics is incompletely or poorly known, for a specified class of problems, and the code has been verified for a related class of problems.

As can be seen, this definition is built on the framework already constructed and discussed at length. The key elements added in this definition, that is, beyond verification, are: first, only "required prediction accuracy" is required, not necessarily "high" accuracy; second, the flow physics is not well known or confidently described by existing mathematical models; and third, a calibrated code must also be verified for a similar or "related" class of problems. Calibration of a code is clearly most demanding. If one argues that it would be impossible for a code to achieve this level of predictive accuracy, be reminded of the two key caveats: first, only "required

prediction accuracy" is needed, however loose or stringent one chooses; and second, the class of problems can be "specified" as restrictive or broad as one chooses. These qualifications make it possible for a code to achieve this level of quality assurance.

If one attempts to define code validation as the next higher-level requirement from the proposed definition of calibration, the definition is seen to yield one of two results. First, if any of the caveats, such as "specified class of problems," is removed, then a clearly unachievable level of code quality would be required. Second, if one were to require a specified prediction accuracy for *all* significant variables of a flow field, then the definition would actually be a subset of the definition of calibration proposed above.

A final comment is made concerning the lack of a proposed definition for code validation. The term *calibrate* directly and bluntly implies restriction on accuracy (or, from the dictionary, "to determine the deviation from a standard so as to ascertain the proper correction factor"). Whereas the term *validate* implies a broad or unlimited range of accuracy. By not defining validation, we avoid this implication. Some CFD proponents, if not most, would argue that the term *calibration* is unduly harsh. However, it should be patently clear from this paper, and from most previous investigators dealing with this subject, that limitations are unavoidable. An analogy can be drawn showing the inappropriate use of the term *validation*. In ground test facilities (wind tunnels, engine test facilities, ballistic ranges, etc.), the term *validated facility* is never used because it is a misnomer -- facilities are calibrated. Concerning mathematical models of physics or CFD, it is traditional to use the word *validation* because of the breadth of physics typically encompassed. However, it is suggested that rethinking of this perspective is appropriate now that numerical simulations routinely take us to the limits of their applicability.

Summary and Conclusions

A review was given of past terminology and methodology related to building confidence in CFD code predictions. A detailed, critical evaluation of each of the previously offered definitions of code verification, calibration, and validation was given. Based on this previous work, a new framework was developed that presents a more comprehensive perspective on all of the elements that can contribute to building confidence in CFD code prediction accuracy. Flow physics research, flow physics modeling, and code prediction/application were delineated to clearly segregate the issues of mathematical model accuracy of the physics and numerical solution accuracy. Also integrated into the framework is the description of the appropriate roles that should be played

by physical experiments, analytical solutions, and independent numerical simulations. To clarify the proper role of each contributing element in the framework, a large number of examples were given. New definitions were proposed for code verification and calibration that eliminate most of the difficulties of past definitions. The proposed definitions suggest increasing levels of predictive difficulty and increasing breadth of problem classes to describe the flow physics and geometry. Alternative definitions of terminology by ANS and IEEE are also workable, but it is shown how they approach the issue from a different perspective. It is believed that the ANS and IEEE perspective and the present perspective are both usable and defensible, but the most important issue is that standardization of terminology in the aerospace community be adopted. It is hoped that the proposed framework and definitions will improve the understanding of the issues and will promote additional discussions and debate on this crucial topic.

Acknowledgements

The author thanks Daniel Aeschliman and Frederic Blotner of Sandia National Laboratories and Unmeel Mehta of NASA Ames Research Center for providing valuable discussions and comments during the preparation of this paper.

References

- American Nuclear Society, 1987, "Guidelines for the Verification and Validation of Scientific and Engineering Computer Programs for the Nuclear Industry," ANSI/ANS-10.4-1987.
- D. A. Anderson, J. C. Tannehill, and R. H. Pletcher, 1984, *Computational Fluid Mechanics and Heat Transfer*, Taylor & Francis.
- B. F. Armaly, F. Durst, J. C. F. Pereira, and B. Shoenung, 1983, "Experimental and Theoretical Investigation of Backward-Facing Step Flow," *J. of Fluid Mechanics*, Vol. 127, pp. 473-496.
- F. G. Blotner, 1990, "Accurate Navier-Stokes Results for the Hypersonic Flow Over a Spherical Nosedip," *J. of Spacecraft and Rockets*, Vol. 27, No. 2, pp. 113-122.
- B. W. Boehm, 1981, *Software Engineering Economics*, Prentice-Hall, p. 37.
- R. G. Bradley, 1986, "Current Capabilities and Future Directions in Computational Fluid Dynamics," National Research Council, National Academy Press, Washington, DC.
- _____, 1988, "CFD Validation Philosophy," AGARD Symposium on Validation of Computational Fluid Dynamics, Conf. Paper No. 1, Lisbon, Portugal.
- R. R. Cosner, 1994, "Issues in Aerospace Application of CFD Analysis," AIAA Paper No. 94-0464, AIAA 32th Aerospace Sciences Meeting, Reno, NV.
- J. H. Ferziger, 1993, "Estimation and Reduction of Numerical Error," FED-Vol. 158, Quantification of Uncertainty in Computational Fluid Dynamics, ASME Fluids Engineering Conf., Washington, DC., pp. 1-7.
- C. J. Freitas, 1993, editor, "The CFD Triathlon: Three Laminar Flow Simulations by Commercial CFD Codes," ASME Fluids Engineering Conf., Washington, DC.
- R. A. Graves, 1992, "Software Validation: The Bridge from R&D to Industrial Application," AIAA Paper No. 92-0587, AIAA 30th Aerospace Sciences Meeting, Reno, NV.
- M. J. Green, S. L. Lawrence, A. D. Dilley, R. W. Hawkins, M. A. Walker and W. L. Oberkampf, 1993, "Application of CFD to a Generic Hypersonic Flight Research Study," AIAA Paper No. 93-0312, AIAA 31st Aerospace Sciences Meeting, Reno, NV.
- J. W. Gresho, P. M., D. K. Gartling, J. R. Torczynski, K. A. Cliffe, K. H. Winters, T. J. Garratt, A. Spence, and J. W. Goodrich, 1993, "Is the Steady Viscous Incompressible Two-Dimensional Flow Over a Backward-Facing Step at $Re = 800$ Stable?" *Intl. J. Numerical Methods in Fluids*, Vol. 17, pp. 501-541.
- W. D. Hayes and R. F. Probstein, 1966, *Hypersonic Flow Theory; Vol. 1, Inviscid Flows*, Academic Press.
- Institute of Electrical and Electronics Engineers, 1984, F. Jay, editor, "IEEE Standard Dictionary of Electrical and Electronics Terms," ANSI/IEEE Std. 100-1984.
- International Organization for Standardization, 1993, "Quality management and quality assurance standards-Part 3: Guidelines for the application of ISO 9001 to the development, supply and maintenance of software," ISO 9000-3:1991(E).
- K. Karamcheti, 1966, *Principles of Ideal-Fluid Aerodynamics*, Wiley.
- A. Martellucci, 1990, "The Challenging Process of Validating CFD Codes," AIAA Paper No. 90-1402, AIAA 16th Aerodynamic Ground Testing Conf., Seattle, WA.

- J. G. Marvin, 1988, "Accuracy Requirements and Benchmark Experiments for CFD Validation," AGARD Symposium on Validation of Computational Fluid Dynamics, Conf. Paper No. 2, Lisbon, Portugal.
- ____ and T. L. Holst, 1990, "CFD Validation for Aerodynamic Flows: Challenge for the 90's," AIAA Paper No. 90-2995, AIAA 16th Aerodynamic Ground Testing Conf., Portland OR.
- U. B. Mehta, 1990, "Computational Requirements for Hypersonic Flight Performance Estimates," J. of Spacecraft and Rockets, Vol. 27, No. 2, pp. 103-112.
- ____, 1991, "Some Aspects of Uncertainty in Computational Fluid Dynamics Results," J. of Fluids Engineering, Vol. 113, pp. 538-543.
- ____, 1993, "Aerospace Plane Design Challenge: Credible Computations," J. of Aircraft, Vol. 30, No. 4, pp. 519-525.
- R. D. Neumann, 1990, "CFD Validation--The Interaction of Experimental Capabilities and Numerical Computations," AIAA Paper No. 90-3030, AIAA 16th Aerodynamic Ground Testing Conf., Portland, OR.
- W. L. Oberkampf and D. P. Aeschliman, 1992, "Joint Computational/Experimental Aerodynamics Research on a Hypersonic Vehicle: Part 1, Experimental Results," AIAA Journal, Vol. 30, No. 8, pp. 2000-2009.
- R. L. Panton, 1984, *Incompressible Flow*, Wiley.
- P. J. Roache, 1993, "A Method for Uniform Reporting of Grid Refinement Studies," FED-Vol. 158, Quantification of Uncertainty in Computational Fluid Dynamics, ASME Fluids Engineering Conf., Washington, DC., pp. 109-120.
- K. Shimazaki, Y. Himeno, and N. Baba, 1993, "An Attempt at Evaluating Numerical Errors in a 2-D Navier-Stokes Solver," FED-Vol. 158, Quantification of Uncertainty in Computational Fluid Dynamics, ASME Fluid Engineering Conf., Washington, DC., pp. 19-28.
- M. A. Walker and W. L. Oberkampf, 1992, "Joint Computational/Experimental Aerodynamics Research on a Hypersonic Vehicle: Part 2, Computational Results," AIAA Journal, Vol. 30, No. 8, pp. 2010-2016.
- C. Y. Wang, 1991, "Exact Solutions of the Steady-State Navier-Stokes Equations," *Annual Review of Fluid Mechanics*, Vol. 23, pp. 159-177.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DATE

FILMED

9 / 13 / 94

END