
1

An Overview of Numerical Simulation

Reactive flows include a broad range of phenomena, such as flames, detonations, chemical lasers, the earth’s atmosphere, stars and supernovae, and perhaps even the elementary particle interactions in the very early stages of the universe. There are striking physical differences among these flows, even though the general forms of the underlying equations are all quite similar. Therefore, considerations and procedures for constructing numerical models of these systems are also similar. The obvious and major differences are in the scales of the phenomena, the input data, the mathematical approximations that arise in representing different contributing physical processes, and the strength of the coupling among these processes.

For example, in flames and detonations, there is a close coupling among the chemical reactions, subsequent heat release, and the fluid dynamics, so that all of the processes must be considered simultaneously. In the earth’s upper atmosphere, which is a weakly ionized plasma in a background neutral wind, the chemical reactions among ionized gases and the fluid dynamics are weakly coupled. These reactions take place in the background provided by the neutral gas motions. The sun’s atmosphere is highly ionized, with reactions among photons, electrons, and ionized and neutral atomic species, all in the presence of strong electromagnetic fields. A Type Ia supernova creates the heavier elements in the periodic table through a series of strongly coupled thermonuclear reactions that occur in nuclear flames and detonations. The types of reactions, the major physical processes, and the degree and type of coupling among the processes vary substantially in these systems. Sometimes reactions are essentially decoupled from the fluid flow, sometimes radiation is important, and sometimes diffusive transport effects are important. These very different systems, however, are generally all dynamic and unsteady.

1–1. Some Comments on Terminology

The words *model*, *simulation*, *algorithm*, *numerical*, and *computational* are used repeatedly in this book. Even though these words seem easy enough to understand and use in ordinary conversation, their technical use is confused and imprecise. To different individuals and research communities, these words have different and sometimes overlapping meanings. To attack this technical “Tower of Babel” by proposing rigid definitions would

only add more confusion. Here, instead, we discuss some of the nuances of key terms used in this book, and then try to use the terms consistently.

What is generally meant by modeling is much broader than what is generally meant by simulation. A simulation attempts to imitate the dynamic behavior of a system and to predict the sequence of events that control that behavior. Modeling is generally used in a broader, more static sense: a model can represent a dynamic phenomenon in its entirety without specifying its evolution. A model can be a formula, a quantitative relation, a collection of empirical data, a set of equations, or even its equivalent analog circuit.

The objective of modeling is not necessarily to produce an exact copy of a system, but to reproduce certain salient features of the system. Because approximations are made to derive the model, it is understood that the model is an imperfect representation of the system. The result is that a model is invalid in some regimes and imprecise by some amount almost everywhere. A simulation, then, exercises a model or collection of models for a particular choice of physical parameters, initial conditions, and boundary conditions.

Throughout the literature, however, the use of the terms *modeling* and *simulation* overlap, and they are sometimes used interchangeably. The use of the words modeling and simulation has been further confused recently because several communities have adapted the terms to mean the use of very fast, data-driven models to create a virtual reality for interactive training or planning purposes. To combat the confusion caused by this degraded use of the terms, we usually hear the modified descriptions “detailed simulation,” “detailed modeling,” “high-fidelity simulation,” or “fundamental model” to mean the careful scientific and engineering computations that are the subject of this book.

The terms *numerical* and *computational* are often used interchangeably, although they have somewhat different meanings. Numerical analysis, generally a theoretical subject whose main practical application is to solve problems on computers, is not all computational. Computational work in simulations is not all numerical. For example, a reactive-flow computation may involve interactive systems programming, text manipulation, data management, graphics, and so on. Here we consider models and algorithms as computational when they describe how the problem is broken down, represented, manipulated, and stored in the computer. We use the word *numerical* to describe techniques, methods, and algorithms when they concern the calculation of numbers or the quantitative evaluation of formulas and equations.

An *algorithm* is a solution procedure used to implement a model. It is not a model, and it is not necessarily numerical. Not every model has an algorithm associated with it; there simply might not be an algorithm to implement a model. A numerical simulation does not necessarily solve the equations that make up the mathematical model directly. The words *advance* and *integrate* might be more appropriate within the context of a simulation than the word *solve*. As the general reactive-flow problem varies in time, we solve a set of time-dependent, reactive Navier-Stokes equations that have been converted into a set of discretized algebraic equations in the computational model to advance the overall mathematical model. This algebraic set of equations is derived from the original set of partial differential equations in the mathematical model. Algorithms are substituted for the various mathematical terms, for their interactions, and thus for the underlying physical phenomena.

1-2. A Classification of Models

One useful classification of models distinguishes among three levels: fundamental, phenomenological, and empirical. In this sequence of levels, the models are increasingly coarse and are solved correspondingly faster. Making appropriate choices for the level of models to use can produce very practical composite reactive-flow models for solving difficult problems.

A *fundamental model*, often called a *detailed model*, describes the properties or behavior of a system starting with as many basic physical assumptions, or *first principles*, as possible. The problems that such a model can treat are limited in scope because of the expense and complexity of the calculation. Thus the usual objective is to simulate one isolated type of process or interaction. Such models can, however, be extremely detailed and exact in what they do cover. What should be a fundamental or detailed model in a reactive-flow simulation is problem dependent. It is defined by what is currently possible to do for that problem, or what is actually needed to represent the behavior of the system being simulated.

Detailed models can provide constants or other information for use in more general but less rigorous calculations. For example, quantum mechanical calculations of the potential energy surface of a molecule often provide information on the chemical reaction pathways and reaction rates used in more phenomenological continuum representations of reacting fluids. The continuum model itself may be made up of submodels that themselves may be fundamental, phenomenological, or empirical. An important objective of detailed modeling is to develop computational models with well-understood ranges of validity and accuracy. In principle, the broader the range of validity, the more generally useful the model is and the more expensive it is to use.

Phenomenological models and *empirical models* must be used when the time or space scales of the physical processes are too disparate to resolve consistently in one calculation. Macroscopic, averaged models of the small-scale processes become the phenomenological models that appear in the governing equations. For example, in the detailed models of flames considered later in this book, the chemical rate constants, diffusive transport coefficients, and equations of state cannot be determined simultaneously with the convective flow. These three types of models represent atomic-scale processes that have been averaged over assumed Maxwellian distributions of particles to derive the continuous, fluid description. These quantities are phenomenologies representing processes that occur on a scale too small to be resolved in a detailed way in the fluid computation itself.

Often the exact mathematical forms of terms in the governing equations are not known. Alternatively, these terms, though known, may be too complex to evaluate. In these cases, a simpler, approximate form must be used. This approximate *phenomenological* form is usually motivated by physical considerations and contains input data obtained from fits to experimental data or more fundamental theories or simulations. Often global constraints and conservation conditions are used to derive the phenomenological model. Turbulence models are examples of phenomenologies in which complex interactions are modeled by equations with physically reasonable forms. These models are then calibrated by adjusting coefficients to fit data derived from experiments or from more specialized, detailed simulations. Other examples include an energy release model for a chemical system, and a

parametric model for soot formation. A phenomenological model is based on a formula or algorithm that represents our intuitive, qualitative understanding of a particular physical situation. Thus, it must be calibrated with a more basic theory or an experiment. Over time these phenomenologies often acquire the reputation of a fundamental model without the corresponding reliability or accuracy. The Arrhenius reaction rate formula for chemical reactions is a good example of this.

Empirical models are either direct fits of data to a given mathematical formula or are data used directly in tabular form. Because the data are usually derived from experiments, they include extraneous effects, such as measurement interference or equipment calibration errors. A caveat attending such models is that they can only be used for interpolation, not extrapolation. This is true with respect to both the physical parameter range over which the model is valid and the particular physical environment in which it is used. Chemical rate constants are usually empirical models. Other examples of empirical models are tables of the rate at which smoke diffuses through a chamber, the rate a fire spreads through a building, or the rate of chemical energy release in a turbulent flame.

In this book we are interested in numerical simulations of reactive flows in which fluid dynamics, chemistry, and a variety of diffusive transport processes play important, dynamic, and interacting roles. For example, consider a propagating laminar flame, for which we construct a detailed model that solves the compressible Navier-Stokes equations with chemical reactions. The fluid dynamics may be solved by a very accurate numerical method and can be considered a fundamental model of convective transport. The chemical reaction rate coefficients are either phenomenological models if they are taken from fundamental calculations that have been fit into an analytic form, or empirical models if they are fits to experimental data. The equation of state may be a table compiled from experiments. Diffusion constants could be estimates based on intelligent guesses, or they could come from experiments or theoretical studies. This detailed model of a flame would become more phenomenological and less detailed if the set of chemical rate equations were replaced by a simplified “reduced” set of equations, or if the radiative effects were approximated by a single loss term instead of solving the equations of radiative transport.

1–3. Statistical Averaging or Chunking

The reactive Navier-Stokes equations that we use to describe a flame do not solve for the behavior of individual particles. They assume that the medium is a continuous, macroscopic fluid. This continuum approximation incorporates statistical averaging processes performed over the small-scale particle dynamics that we know are present.

The process in which the small-scale phenomena are averaged to produce a global quantity or interaction law has been called *chunking*, which means that microscopic complexity becomes statistically simplified, or chunked at a macroscopic level (Hofstadter 1979). Chunking reduces the number of degrees of freedom needed to describe the behavior of a system. The chunked behavior of a system follows relatively simple laws that approximate the statistical behavior of the microscopically complicated physical system. For example, complete, detailed knowledge of quarks and chromodynamics is not necessary for understanding many aspects of the composition and dynamics of atoms and nuclei. The

detailed interactions of every individual molecule in a system are not needed to explain a propagating laminar flame.

As we concentrate attention on more macroscopic scales and interactions, further chunking is needed to obtain usable models. Nuclear physics, for all but extremely high-energy collisions, chunks the behavior of subelementary particles to create protons and neutrons. Physical chemistry chunks proton and neutron systems to produce the composite particle, the nucleus. Chunking at the level of atoms and molecules gives us the continuum fluid representation. Hofstadter describes the relations among various levels of chunking:

Although there is always some “leakage” between the levels of science, so that a chemist cannot afford to ignore lower level physics totally, or a biologist to ignore chemistry totally, there is almost no leakage from one level to a distant level. That is why people can have an intuitive understanding of other people without necessarily understanding the quark model, the structure of the nuclei, the nature of electron orbits, the chemical bond, the structure of proteins, the organelles in a cell, the methods of intercellular communication, the physiology of the various organs within the human body, or the complex interactions among organs. All that a person needs is a chunked model of how the highest level acts; and as we all know, such models are very realistic and successful.

A corollary is that chunked models no longer have the ability to predict exactly.

In short, in using chunked high level models, we sacrifice determinism for simplicity. . . . A chunked model defines a “space” within which behavior is expected to fall, and specifies probabilities of its falling in different parts of that space.

In a complex reactive flow, it is not possible to resolve all aspects of the microscopic particle interactions using a continuum approximation. It is, however, possible to include some of the macroscopic consequences of atomic and molecular phenomena in a fluid model. Chemical reactions, interspecies molecular diffusion, thermal conductivity, temperature-dependent enthalpies and equations of state are all examples of macroscopic models of microscopic processes that result from chunking the microscopic physics.

1–4. The Hierarchy of Levels of Simulation

In this book we are not concerned with elementary particles, but with a more macroscopic view of reactive media composed of molecules and atoms. The interactions of particles at this level result in the processes we want to simulate. Table 1.1 lists the hierarchy of mathematical models used to describe the behavior of systems involving many particles and interactions. Discussions of the derivations of these various approaches can be found in a number of texts (for example, Hirschfelder, Curtiss, and Bird [1954] and Bird [1994]). At the most basic level, these techniques provide very fundamental solutions of the Schrödinger equation describing accurate quantum mechanical interactions of particles, such as might lead to quantum molecular dynamics methods. More general approximations

Table 1.1. Levels of Models of Manybody Interactions

Equation	Solution Method
Schrödinger’s equation	Direct solution, density functional theory, (prescribed interparticle forces)
Newton’s law $f = ma$	Molecular dynamics (particle-based, prescribed interparticle forces)
Liouville equation equation for distribution function, $F(\mathbf{x}_i, \mathbf{v}_i, t), \quad i = 1, N_p$	Monte Carlo methods (statistical, particle-based methods)
Boltzmann equation $F(\mathbf{x}, \mathbf{v}, t)$ binary collisions (low density) good for gases	Direct simulation Monte Carlo Direct solution
Navier-Stokes equation $\rho(\mathbf{x}, t), \quad \mathbf{u}(\mathbf{x}, t)$ short mean free path	Direct solution: finite differences, finite volumes, spectral methods . . . (continuum flow methods)

replace individual particles by continuum fluid elements, as in the Navier-Stokes equations. In between these extremes, there are several levels of statistical and particle-based methods that can be necessary for accurate treatment of some reactive-flow problems.

Molecular dynamics (whether quantum mechanical or classical) is one of the more fundamental levels of this hierarchy. The interactions among the particles may be represented by as basic an interaction as Newton’s law in classical representations, or by extremely complex quantum mechanical potentials. The Liouville equation is the fundamental equation of statistical mechanics that describes the distribution function of a gas as a function of the $6N_p$ -dimensional phase space of N_p particles (Hirschfelder et al. 1954; Goldstein 1982). Monte Carlo approaches and the Boltzmann equation are both derived from the Liouville equation. Monte Carlo methods exploit the statistical nature of multiple collisions in physical systems and have often been used to describe microscopic systems. When particle densities are high enough that the system of particles can be considered as a continuum, the Navier-Stokes equations are generally used to describe the fluid. There are also various subsets or reduced forms of the reactive Navier-Stokes equations that are commonly used today. These are models that include averages of the behavior of the small continuum scales (for example, to describe turbulent flows, Chapter 12), averages of chemical reactions (to “reduce” the chemical models, Chapter 5), or averages over radiative transport processes (Chapter 13).

Some of the information in Table 1.1 is recast into quantitative form in Figure 1.1, which shows the physical regimes of validity of different types of reactive-flow representations. The regimes of validity are shown in this figure as functions of two important parameters, a characteristic length scale of interest in the system L , and the mean molecular spacing Δ , expressed as the cube root of the inverse of the fluid density. The vertical axis is L/d , a measure of the size of the region of interest, where d is the molecular diameter. The

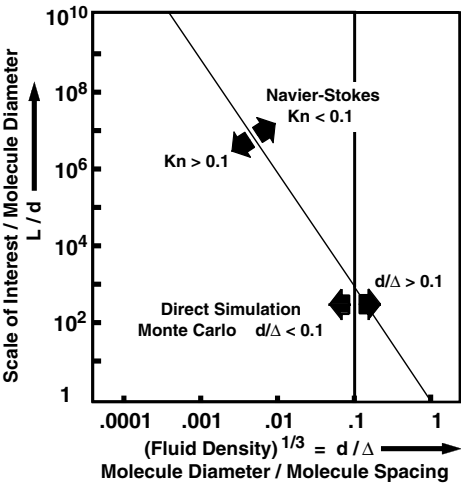


Figure 1.1. Regimes of validity of direct simulation Monte Carlo and Navier-Stokes equations, as a function of the characteristic length scale and mean molecular spacing of a system. In principle, molecular dynamics is valid throughout the entire domain.

horizontal axis shows the dilution of the fluid measured in units of the molecular diameter d divided by the mean spacing of the particles $\Delta \equiv N^{-1/3}$, where N is the number density. As the mean spacing between molecules decreases, the density, $(d/\Delta)^3$, increases. Expressed in this way, the density has a maximum value, at the right side of the figure, of about unity when the molecules are closely packed and the fluid is least dilute.

When the ratio of the mean molecular diameter, d , to the molecular spacing, Δ , is small, so that $d/\Delta \ll 1$ (for example, $d/\Delta < 0.1$), the fluid becomes *dilute*. This regime, which is on the left of the vertical line in the figure, is one in which the statistically based particle method, direct simulation Monte Carlo (DSMC) is an appropriate model (Bird 1994). DSMC is based on the same basic approximations as the Boltzmann equation, but unlike the Boltzmann equation, its regime of validity may be extended to systems where complicated chemistry and three-body collisions occur.

The *Knudsen number*, Kn , is another quantity that is useful in characterizing the regimes of a fluid. This is defined as $Kn \equiv \lambda/L$. Here, L is a characteristic scale length of the system, and λ is the *mean free path*, $\lambda \approx 0.1/Nd^2$, the average distance traveled by a typical molecule between collisions. When λ is small compared to L , and Kn is low enough, the medium behaves as a continuous fluid in which properties such as density, velocity, and energy are well defined at each point. In the collision-dominated regime, where $Kn < 0.1$, the Navier-Stokes model is generally valid.

Because of the different criteria for validity of the Navier-Stokes and DSMC models, both methods can be used in the triangular domain in the upper part of Figure 1.1 where $d/\Delta < 0.1$ and $Kn < 0.1$. When the diagonal line $Kn = 0.1$ is crossed from left to right, the Navier-Stokes equations become valid and are preferred to the DSMC model because of computational efficiency. Thus, this dividing line is usually the practical condition for applicability of DSMC. DSMC is used extensively for high-speed atmospheric reentry problems, where the gas is very dilute and often highly reactive.

Because molecular dynamics is based on a relatively fundamental model, it is valid in principle throughout the entire range of parameters shown. There are no *physical* reasons why it cannot be used for all ranges of densities and system sizes. However, the

computations are very expensive as the number of particles becomes large. In the small triangular regime in the lower right of the figure, neither the Navier-Stokes nor the DSMC model is valid. In this parameter regime, molecular dynamics is, therefore, the preferred model.

This book focuses on finite-volume methods for solving continuum reactive-flow problems. In this approach, time is divided into discrete intervals – *timesteps* – and space into discrete intervals – *computational cells* that define the *spatial grid* or *mesh*. Large sets of discrete variables are defined on the computational cells to approximate fluid variables such as density and velocity. This process puts the equations in a form suitable for numerical computation. We expect the corresponding numerical solutions to converge and become better representations of the continuous fluid variables as the size of the cells and timesteps becomes smaller and smaller. A continuum fluid simulation with cells smaller than a mean free path is subject to problems of physical interpretation even if the model does converge mathematically. Problems that arise due to limits on the attainable resolution are discussed in many parts of this book, but especially in Chapter 6.

Even if practical computational restrictions were removed and arbitrarily good resolution were possible, the continuous variables would still have to be represented in a computer by a discrete set of distinct real numbers with finite numerical precision. It is amusing to note that fluid equations were invented in the first place to simplify the discrete equations of individual particle dynamics. Now we find ourselves reformulating the continuum problem so that we can use digital computers to solve the equations for finite volumes of material. Nevertheless, the continuum set of equations is considerably smaller and simpler to solve than the original equations of particle dynamics.

1-5. The Growth of Computational Capability

Fifteen years ago, when the first edition of this book was written, a two-dimensional computation with 10^6 computational cells was considered very large. Steady-state approximations were the best that could be done for many systems. Today, three-dimensional computations with true dynamics have been performed with more than 10^9 cells. Complex, time-dependent simulations, that were considered an esoteric art confined to a few laboratories with large computers, can now be readily solved by most scientists and engineers on their desktop workstations. All of this reflects continuing exponential advances in computer memory and computational speed.

There have been corresponding changes in our expectations for computing reacting flows. One result of the general increase in computational resources is that proportionately fewer users are trying to develop their own numerical models. They are, instead, using complex, difficult-to-understand simulation programs that were initially written by others. One objective of this book, then, is to explain how to evaluate, use, and interpret the widespread and rapidly growing computational technology available for solving reactive-flow problems.

Figure 1.2 qualitatively illustrates these advances in computer speed and memory as a function of year (based on updating a lecture by Worlton [1988]). The essentially straight line increase of overall performance on the semilog scale means that computer power

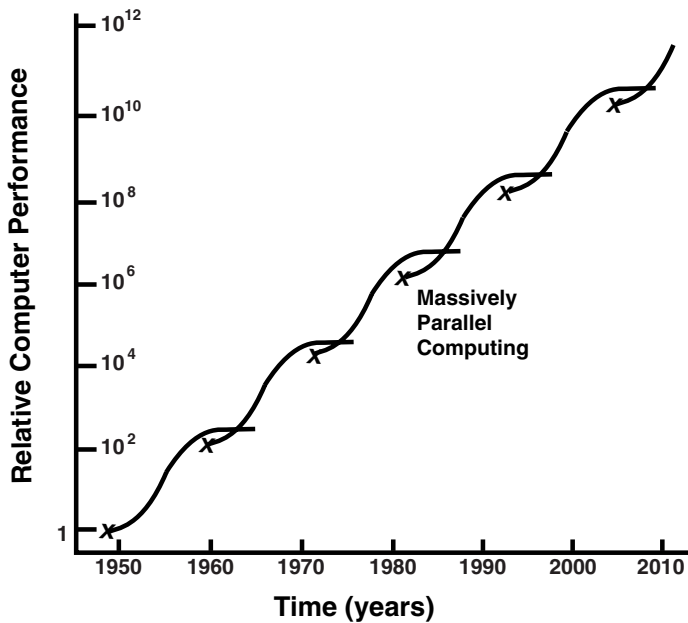


Figure 1.2. Schematic showing the exponential increase of computer power (speed and memory) as a function of time. The crosses indicate the introduction of a major technological breakthrough.

and memory are scaling exponentially. Projections based on current understanding of technology are that this rate can hold for at least another ten years. The figure shows that the exponential performance increase is really composed of a series of piecewise logistic curves, each representing the introduction of a new technology. The result of adding all of the curves is sustained exponential growth. For example, early breakthroughs have included introducing transistors to replace vacuum tubes and semiconductor memory to replace magnetic cores. More recently, we have changed from scalar to vector architectures, and from serial to parallel computing. Such a curve is typical of the growth of an enabling technology, such as high-performance computing. The time-dependent improvement of each component technology is necessarily short term, and saturates at some point. The leveling occurs because the particular technology that allowed the growth becomes too expensive and time-consuming to improve. One obvious feature of this curve is that to continue the growth, and not stagnate, new technologies must regularly be discovered.

Along with the development of high-end computing, there have been the corresponding improvements in personal computers and desktop workstations. As hardware and software become more sophisticated, computing improves across the board. Currently, most parallel supercomputers have about the same speed per processor because they are based on the same or similar chips. The performance increases are driven by microprocessor technology rather than research and development aimed at supercomputer architectures. The current design philosophy tends to use the same basic components (chips) for both supercomputers and powerful workstations. This philosophy may slow the overall pace of computer development, since the high-end systems are less and less the drivers of new

technology. Alternately, it could speed the pace as there is broad-based public demand for faster desktop computing. This is a curious interplay that has practical consequences for scientific computing.

The differences among high-end computers are in their interprocessor communications, memory structures, and architectures. The memory may be globally shared or distributed among the processors. Interprocessor communications may or may not scale linearly as a system expands from a few processors to many processors, and various problems can arise in sending data among processors. At another level, the different architectures cause problems in moving computer programs from one type of machine to another (the portability problem). The current goal is to obtain computational speeds of many teraflops (10^{12} floating-point operations per second), whereas computers delivering many gigaflops (billions of floating-point operations per second) are becoming common.

New advances in algorithms and computer architectures have made a significant contribution to computing reacting flows. Progress has also come through the straightforward application of various existing algorithms as well as from new procedures that combine the standard algorithms with specific features of the new architectures. Future advances could also come from simultaneously combining different kinds of computers, each of which is best suited for a different type of operation, an approach called heterogeneous computing. Another approach uses different sets of processors on the same computer to perform different kinds of operations.

1–6. The Tyranny of Numbers

A detailed numerical simulation model is a program that is executed, or “run,” on a computer. The final accuracy and detail we can expect from a numerical simulation of a reactive flow depends on the interplay of a number of factors, including the availability of high-speed computers, specialized numerical algorithms, and fast visualization of the output. The simulations we are concerned with “integrate,” or “advance,” the approximate time-evolution equations of the model from one time t to another time $t + \Delta t$, where Δt is the *timestep* and is generally smaller than the shortest characteristic times we need to resolve in the calculation. The advancing simulation thus defines the changing state of the system at a sequence of discrete times. The usual assumption is that the state of the system at any time during a timestep can be inferred by interpolating between the state of the system at the beginning and at the end of the timestep.

Typically, hundreds of thousands to millions of numbers are needed to specify the state of a system at any particular time. Consider a detonation propagating in a gas-phase reactive medium. If the problem is two-dimensional, roughly 200×200 discrete points in space are required to give about 1 percent spatial resolution of the flow field and of the chemical species present. If a detailed chemical rate scheme for a diluted hydrogen-oxygen mixture is used, approximately ten species have to be considered in addition to the total fluid density, momentum, and energy density variables. Thus, a minimum of 560,000 numbers are required to specify any state of this modest system at a given time. Realistic simulations are regularly performed for 10^7 variables at a few seconds per timestep, which gives thousands of timesteps per hour of computer time. A major practical problem is how to cope with and effectively interpret so many numbers. We cannot afford