FUTURE TRENDS IN NUMERICAL SIMULATIONS OF DETONATIONS

E. Oran, J. Boris

JOURNAL DE PHYSIQUE
Colloque C4, supplément au n°9, Tome 48, septembre 1987

FUTURE TRENDS IN NUMERICAL SIMULATIONS OF DETONATIONS

E.S. ORAN and J.P. BORIS

Laboratory for Computational Physics and Fluid Dynamics, Naval Research Laboratory, Washington, D.C. 20375, U.S.A.

Résumé

On discute les développements en cours et à venir pour des types de problèmes résolus, la sophistication des algorithmes et les nouveaux types d'ordinateurs qui vont grandement développer notre potentialité pour la simulation numérique du comportement des détonations. Trois voies principales de progrès s'ouvrent dans :

- L'application de l'architecture des nouveaux ordinateurs et du hardware associé qui font ressortir le traitement multiprocesseur, le parallélisme et le graphisme interactif,
- L'utilisation d'algorithmes numériques plus sophistiqués comprenant des algorithmes pour un maillage adaptatif des méthodes monotones d'éléments finis sur des maillages non structurés, les dynamiques des particules et Lagrangienne des fluides, et,
- La solution de problèmes physiques plus compliqués comprenant des espaces multidimensionnels avec des modèles chimiques plus sophistiqués et compliqués.

Abstract

We discuss current and future developments in scientific computation : the types of problems solved, the sophistication of algorithms, and the new types of computers that will greatly enhance our ability to simulate the behavior of detonations numerically. Three major types of advances are in :

- Application of new computer architectures and associated hardware that emphasizes multiprocessing, parallelism, and interactive graphics,
- Use of more sophisticated numerical algorithms, including algorithms for adaptive gridding, finite-element monotone methods on unstructured grids, and Lagrangian fluid and particle dynamics, and
- Solution of more complicated physical problems, including more spatial dimensions with more sophisticated or complete chemical models.

In this paper, we describe some recent developments and trends in scientific computing that will make a significant impact on simulating the properties and behavior of detonations. The three types of advances we discuss are advances in computer architectures, numerical methods, and in the complexity of the problems that we can solve. New supercomputer architectures allow vectorization and massively parallel computations. The availability of such computers makes it necessary to develop algorithms that are optimal on these machines, and to rethink how we write the software for them. New numerical algorithms, optimized for these computers, are now becoming available for both computational fluid dynamics and molecular dynamics. Finally, these advances in computers and numerical methods allow us to solve more complicated types of problems in two and three dimensions with more realistic boundary conditions and more complicated chemical models.

A major problem in calculating the macroscopic properties of detonations is in finding the input parameters that are needed. These include energy releasing reactions and associated parameters, equations of state, and thermophysical quantities such as heat capacities. In general, the more fundamental quantum mechanical and molecular dynamic models have not adequately bridged this gap when experimental data are not available. With the improvements in computers and numerical methods that are becoming available, we believe that microscopic simulations will be able to provide such data in the future.

1. New Computer Architectures

In the last several years, a number of different supercomputer architectures have emerged. These machines are rated by the maximum floating point operation count, usually in megaflops, the number of processors (an indication of the degree of parallelism available), and the type of vectorization permitted in the individual processors. Within this framework, we consider four categories of new computers:

1. *True Supercomputers.* These are very fast, powerful machines with operation counts of one half gigaflop or more. They include the CRAY X-MP, CRAY 2, Fujitsu, Hitachi, NEC, ETA, and perhaps the CYBER 205. They have vector capabilities (usually implemented by arithmetic pipelines) and some parallelism. In general, the degree of parallelism is not very high, generally two to four processors, and is often called "coarse-grained."
2. *New-Wave Supercomputers.* These have operation counts of 50 to 2500 megaflops and usually have more radical designs to exploit massive parallelism. They are, in principle, expandable. They promise to be competitive with the true supercomputers in the near future, although perhaps not as versatile for some time to come. These include, for example, the Connection Machine, Butterfly Machine, Navier-Stokes Computer (Princeton), the various hypercube machines, and systolic arrays.
3. *Super-minicomputers.* These have speeds of 10 to 50 megaflops. They are not massively parallel, but are designed in a more standard way than near supercomputers. These include the Convex, the SCS 40, the Alliant, and the Elexi machines. The performance is comparable to a current technology array processor with much better software and with interactive operating systems.
4. *Hybrid Machines.* These are combinations of the above. One example is the GAPS (Graphical and Array Processing System), constructed in our laboratory at NRL. It is somewhere in between a true supercomputer and the super-minicomputers. It was constructed from commerically available hardware and features the heirarchical parallelism of a number of separate processors with powerful vectorization in the individual array processors. GAPS is not quite as powerful (fast and multi-user) as a true supercomputers, features direct graphic

interaction with the evolving simulation, and is becoming a standard for bused AP (array processor) systems.

The existence of these machines and the speeds they offer now and will offer in the future cannot be ignored. For example, a Connection Machine, with approximately 64,000 processors, could be used effectively as an array processor with a CRAY. Then fluid dynamics calculations could be done efficiently on the CRAY as each node of the Connection Machine integrates an a set of ordinary differential equations representing the chemistry for a fluid element.

The Graphical and Array Processing System, (GAPS)

The GAPS system is a computer system put together at NRL from commercially available components, primarily for computational fluid dynamics and reactive flows. GAPS is an asynchronous, multi-tasking, parallel-processing system consisting of an APTEC 2400 I/O Computer connected to a VAX 11/780 with 12 megabytes of additional fast memory and about 3 gigabytes of online disc storage. GAPS contains several major computational components, including a Floating Point Systems 5305 array processor (maximum performance of 12 Megaflops) and two Numerix MARS 432 array processors (maximum performance of 30 Megaflops each). The three array processors can be programmed in Fortran and have a library of vectorized routines and synchronization software. Three high-resolution color graphics monitors, a Tectronix 4115B raster system, an Evans and Sutherland PS 300 vector system, and a Metheus Ω3600 are connected to the GAPS. The graphics systems have distributed and interactive processing capabilities such as zooming and rotating images.

The APTEC is a high-speed (24 megabyte/s) data bus used to move data among the VAX, the array processors, and other GAPS components through programmable data interchange adapters (DIAs). The ability of the VAX to control data flow is superb, but its ability to actually move the data is limited. In practice, a quarter megabyte/s on the VAX is the maximum information transfer rate it can sustain, and this severely limits the overall system performance. A very high-speed data switchyard is necessary behind the VAX to display high-resolution fluid dynamics simulations at the rate at which they are calculated. The APTEC, working at its full 24 megabyte/s potential, would barely keep up with the 12 megabyte/s speed required to show 12 frames per second of a 1024×1024 pixel array.

The current configuration of GAPS has 72 megaflops of array processors. The NRL Cray XMP 2/4 is rated as a 500 megaflop system at full use. When the two 30-megaflop array processors are processing a typical two-dimensional calculation, simultaneously pixelating the data arrays as it computes them, and sending every third or fourth frame to a Tektronix 4115B for monitoring, only about 7% of the APTEC data bus capacity is being used. When the two 30 megaflop array processors are used together on the benchmark, they give about 25% of the performance of a CRAY. The two systems together deliver computations at over one third the performance of a CRAY processor and use only 18% of the APTEC bus. The APTEC could support up to a dozen array processors and the system would profit greatly from higher-speed array processors.

An important part of our research has been to develop efficient algorithms that can take advantage of an asynchronous multitasking parallel architecture such as the GAPS. We have begun with algorithms that can take advantage of more parallelism than the pipeline and vector register architectures used in current supercomputers. The Reactive Flow Model (RFM), currently implemented on the GAPS, can be run for very long times on relatively high-resolution fluid simulations. Because the GAPS runs in the background without affecting the operation of the VAX, days or weeks of computer time can be devoted to a problem. Now the GAPS effectively yields an extra 40 hours of CRAY processor time every week. Results from GAPS simulations using the reactive flow model can be displayed as they are calculated using VOYEUR, a high-bandwidth

graphics package that can select data from the VAX, from the GAPS array processors, or from the CRAY.

2. New Numerical Algorithms

The latest algorithmic developments focus on more sophisticated adaptive gridding to maintain high-resolution at gradients, applications of finite-element methods with unstructured grids, very fast near-neighbors and constraint algorithms for molecular dynamics, more sophisticated methods for coupling physical processes with disparate time scales, and software especially optimized for the new computer architectures. Here we describe two new algorithms, the finite-element Flux-Corrected Transport algorithm for computational fluid dynamics and the Monotonic Lagrangian Grid for molecular dynamics.

Finite-Element Flux-Corrected Transport

The key to being able to simulate the properties of detonations is to use a nonlinear, monotone, positivity-preserving method such as Flux-Corrected Transport (FCT) (Boris, 1971; Boris and Book, 1976). A monotone method does not add unphysical maxima or minima to a computed profile, and a positivity-preserving method does not allow originally positive quantities to become unphysically negative. The algorithm is nonlinear because the numerical coefficients associated with the finite-differences at each timestep at each location depend on the value of that quantity at that location. These algorithms have been used extensively on quadrilaterial finite-difference grids and they can easily be used with nonorthogonal grids.

Finite-element techniques have been used for years to solve difficult, practical problems in structural engineering. Later they became interesting to mathematicians who formulated their properties in terms of broad classes of approximations. Triangles and tetrahedra are commonly used elements for finite-element grids in two and three dimensions, respectively. They are particularly useful because they can be used to describe irregular objects or strangely shaped internal boundaries and interfaces.

More recently, finite-element methods have been modified for a variety of problems in fluid dynamics and heat transfer, and most recently for time-dependent fluid dynamics. Their application to solutions of systems of continuity equations are of interest here. The recent book by Zienkiewicz and Morgan (1983) is a good introduction to the use of finite elements for solving partial differential equations. In particular, it is advantageous to combine the triangle-based, unstructured grid approach of a finite-element method with the accuracy of a nonlinear, monotone finite-volume method.

A major drawback of finite-element methods is that they can be very expensive. If time is advanced by an explicit finite-difference formula, it is necessary to solve a linear matrix problem at each step. When the time is advanced implicitly, there are additional problems. The cost in computer time for large matrix inversions at each timestep is often prohibitive. In multidimensions, finite-element methods require a great deal of computer storage. The cost of recomputing geometric and shape functions for each physical variable transported is often prohibitive. Thus these variables are usually stored rather than being recomputed, requiring dozens of quantities per node in two dimensions and many more in three dimensions.

There are compensating advantages to hybrid finite-element finite-volume methods. They are relatively straightforward to generalize to multidimensional problems in several dependent variables. In particular, the shape of the domains do not have to be regular, so the method is well suited to problems with irregular boundaries. The domain can also change in the course of time to allow improvements in resolution where needed or to allow for deformable structures in the flow.

We particularly want to mention two new approaches to solving time-dependent fluid dynamics problems. These are the *moving finite-element method* (see, for example, Miller and Miller, 1981; Gelinas et al., 1981; Djomehri et al., 1986), and the hybrid method developed by Löhner et al. (1985, 1986). Both of these are Galerkin methods using tent functions which are or can be made multidimensional. However, there are certain fundamental differences in the two approaches.

In the method of Löhner et al., time is advanced by an explicit second-order Lax-Wendroff discretization and has been extended to meshes that adapt in time. The degrees of freedom are the values of the dependent variables. The nodes are subdivided in front of advancing gradients and flow structures and recoarsened behind them according to rules based on gradients in the known properties of the evolving system. This is a rather straightforward method which combines the accuracy and flexible grid of finite elements with some of the convenient features of monotone finite-volume methods.

In the moving finite-element method, node positions are also treated as unknowns. This flexibility gives coupled ODEs for node positions in addition to the usual variables. Because the resulting equations are stiff and always implicit, the computational cost can be substantial. However, in addition to the value of the variables at the nodes, the method also produces "best" values for the positions of the nodes, given a certain number of nodes. Accordingly, nodes tend to migrate toward regions where there are abrupt changes in gradients of the variable values. Thus problems characterized by advancing surfaces may be more efficiently solved by such methods because the number of nodes can be minimized, even though a relatively large amount of computation is required for each node.

Finite-element methods have not been exploited for fluid calculations as fully as finite-difference and spectral methods. In general the number of operations per finite-element node substantially exceeds that per mesh point in finite-difference or finite-volume methods, so finite-element methods must be more accurate if they are to be competitive. The recent algorithms of Löhner et al. (1986), which incorporate the monotone FCT algorithm into a finite-element framework, are competitive with finite-difference methods. An example of a typical problem of a shock over irregular objects is shown in Figure 1. One drawback of this approach is the amount of computer memory per mesh point that it requires, a serious issue when memory is at a premium. However, these methods are accurate, relatively fast, and allow very flexible, adaptive gridding. Extensions to three dimensions are straightforward and potentially affordable with the advent of computers with large high-speed memories.

Monotonic Lagrangian Grid

A major part of a molecular dynamics calculation is the continual reevaluation of which points are near neighbors in a large set of nodes that often seem to be moving randomly. The expense arises because each node can potentially interact with any of $N-1$ other nodes in the system. This is called the N^2 problem because only a few of the N^2 interactions possible between pairs of nodes are potentially important.

The Monotonic Lagrangian Grid (MLG) is a new method designed to beat the N^2 problem (Boris, 1986; Lambrakos and Boris, 1987). The MLG is really a data structure for storing the positions and other data needed to describe N moving nodes. These N nodes could, for example, represent fluid elements, atomic particles, molecules, droplets, or dust particles that must be tracked in a multiphase detonation calculation. A node with three spatial coordinates has three indices in the MLG data arrays. The data relating to each node are stored in the memory locations indicated by these indices and ensure that nodes which are close to each other in real space are always near neighbors in the MLG data arrays. This simplifies and speeds most interaction

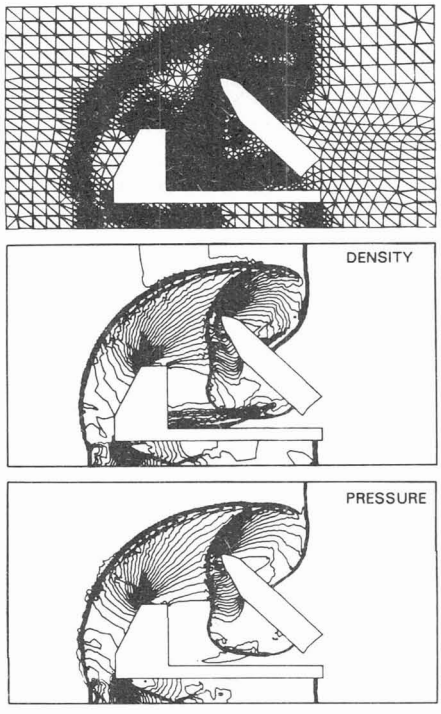

Figure 1 Finite-element FCT calculation of a shock over two irregularly shaped obstacles. (Figure is courtesy of R. Löhner.)

and correlation calculations. A computer program based on the MLG data structure does not need to check $N-1$ possible distances to find which nodes are close to a particular node. The indices of the neighboring nodes are automatically known because the MLG node indices vary monotonically in all directions with the Lagrangian node coordinates. The cost of the algorithm in practical calculations is dominated by the calculation of the interactions of nodes with their near neighbors, and the timing thus scales as N.

Figure 2 shows an example of a small two-dimensional MLG. Because spatial coordinates define a natural ordering of positions, it is always possible to associate two grid indices with a set of random locations in a two-dimensional space. The MLG shown is one of the many possible 4×4 MLGs linking 16 nodes distributed nonuniformly in a two-dimensional region. This freedom is illustrated in Figure 3, which shows three different MLGs passing through the same set of nodes.

The two-dimensional MLGs shown satisfy the following indexing constraints:

$$x(i,j) \leq x(i+1,j) \qquad i = 1,...,N-1; \qquad j = 1,...,N$$

$$y(i,j) \leq y(i,j+1) \qquad i = 1,...,N; \qquad j = 1,...,N-1 .$$

The 16 nodes are shown at their irregular spatial locations, but they are indexed regularly in the MLG by a monotonic mapping between the grid indices and the locations. Each grid line in each spatial direction is forced to be a monotone index mapping. Nodes stored in the data

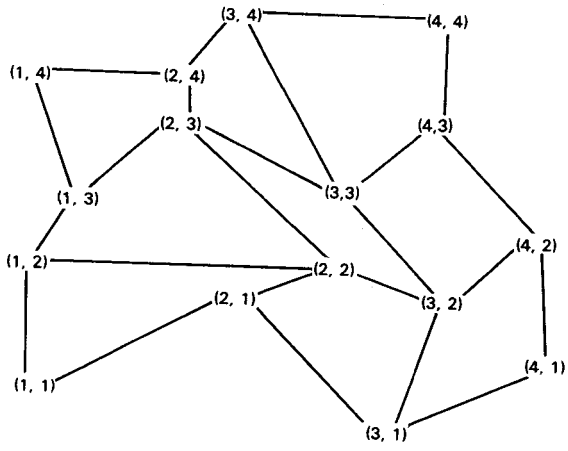

Figure 2 An example of a two-dimensional Monotonic Lagrangian Grid.

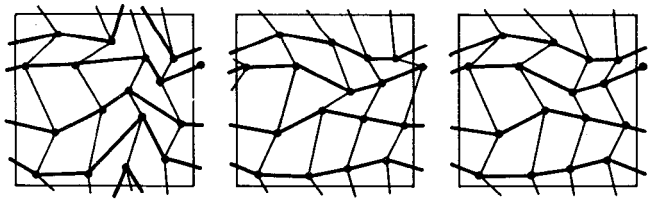

Figure 3 Three Monotonic Lagrangian Grids passing through the same set of nodes.

memory according to this prescription are at most two or three nodes away from the neighboring nodes which can affect them. Thus for gradient, derivative, or force calculations, only a fraction of the possible node-node interactions need to be considered. No search is necessary to locate near neighbors and the necessary logic and computation is ideal for parallel or multiprocessing methods using this data structure.

Using an MLG to index the positions and physical properties describing a node in computer memory, a Lagrangian neighborhood in the MLG can be determined by a maximum index offset, N_c, rather than a short-range cutoff radius R_c. Computations for the interactions of a particular node are only considered with those nodes whose MLG indices are offset from the index of the particular node by an amount less than or equal to N_c. Thus computations of interactions are only made between nodes located in a small contiguous portion of computer memory. Although this approach results in computing interactions for a few distant nodes, it provides a substantial reduction in computational cost. The computations can be vectorized efficiently because nodes that are close to each other are indexed through contiguous memory.

A construction algorithm that scales as $N \log N$ can be used to build an MLG from randomly located nodes. An MLG can always be found for arbitrary data although it is not necessarily unique. Further, when node motions in real space destroy some of the monotonicity conditions, another faster $N \log N$ algorithm exists to iteratively swap the node data between adjacent cells

to restore the MLG order. The swapping iterations continue until a new MLG is found that satisfies the monotonicity conditions and any additional constraints relating to special boundary conditions. The swapping iteration converges rapidly even for large deviations from MLG ordering. Swapping is a discrete, local, and reversible operation.

Using the MLG for molecular dynamics calculations offers large potential improvements in the time it takes to do a calculation. For example, to determine the structure of the current MLG, and hence the nearest neighbors, is negligible in a typical calculation. On the CRAY, swapping typically takes 3–5% of each timestep. In addition, the MLG is faster than other $N \log N$ algorithms because it is fully vectorizable and can make optimum use of parallel computers.

Lambrakos et al. (1987) have recently derived a new algorithm for use with the MLG that maintains constraints in a very efficient way, and so expands the types of calculations that can be done with the method. For example, when the intermolecular bonds have timescales that are are orders of magnitude faster than the intramolecular interactions, the intermolecular vibrations can be treated as rigid, and allowed to break according to a prescribed model.

3. More Complex Physical Problems

The costs and limitations encountered in developing and applying simulations are formidable. To deal with them calls for improved numerical algorithms and other software, and advantage taken of larger, faster, more accurate, and friendlier computers. Over the last few decades, improved algorithms have contributed to simulation capability at least as much as hardware development. Together, hardware and software advances have increased our capability by orders of magnitude. Early one-dimensional gas dynamics calculations were resolved with forty or fifty computational cells. Such problems can now be performed in three dimensions, about four or five orders of magnitude more computing, in the same time as the old one-dimensional calculations.

One way to deal with present resolution limitations and overall simulation costs is to isolate the expensive processes and to replace them in the simulation with simplified phenomenological models. The phenomenologies are ad hoc, physically reasonable, approximate models with parameters that are guesses, fits to experiments, or calibrated from more detailed but more restricted numerical models. For example, a large chemical reaction mechanism is composed of many chemical species and perhaps hundreds of reaction rates linking them. Integrating the ordinary differential equations gives the time history of the individual chemical species and bulk properties such as the temperature and pressure. Using this chemical mechanism in a multidimensional fluid dynamics model is possible in theory, but is extremely expensive in reality. An alternate approach is to use the detailed mechanism to calculate bulk properties, such as final temperatures and pressures as a function of a range of initial temperatures and pressures. This table of numbers forms the basis of a computationally inexpensive phenomenology for use in multidimensional fluid models. In general, the more accurate the underlying theoretical framework for the phenomenology, the smaller the tables of results needed and the broader the validity of the model.

There is always a tendency to overestimate how much a phenomenology can actually predict. Phenomena cannot really be predicted when the controlling physical processes are not resolved accurately enough in time, in space, or by the computational model. If one of the controlling physical or chemical processes in a simulation is being treated very accurately by a phenomenological model, the entire simulation might not be more accurate than the phenomenology, even if the other effects are treated by more detailed models. For example, representing an elementary chemical reaction rate with an Arrhenius form seems fundamental, but assumes that there is a Boltzmann distribution among the electronic and vibrational states of the molecule. However, a computational model based on Maxwellian distributions is questionable for slow molec-

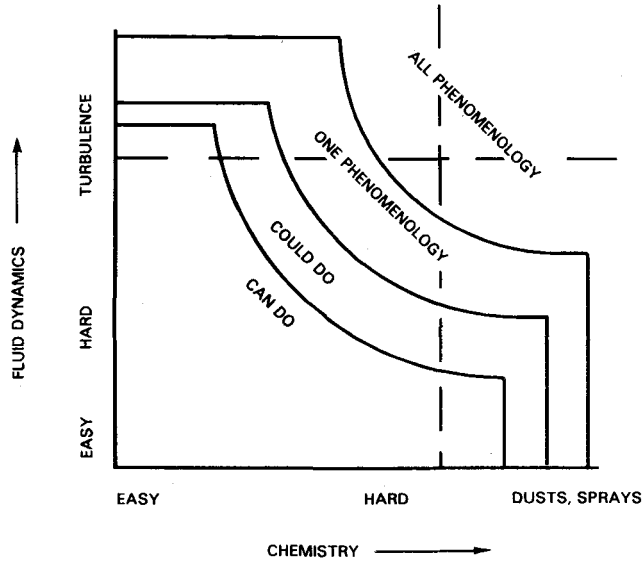

Figure 4. Diagram showing what can currently be calculated as more and more phenomenologies are added to the model.

ular equilibration in gases and high density reactions which occur in explosive materials because nonequilibrium reactions are not represented in this formalism.

Figure 4 shows schematically what levels of detailed simulations are practical for complex, multidimensional, reactive-flow problems in fuel-oxidizer systems. Two "dimensions" of difficulty are considered in the figure: the complex physics of chemical kinetics and local multiphase material effects along the horizontal axis, and the resolution-bound processes of fluid dynamics on the vertical axis. Each axis is labeled ranges in difficulty from EASY to HARD.

For example, on the horizontal axis an easy problem might involve several coupled chemical reactions among a few species. A hard problem might involve hundreds of chemical reactions among dozens of chemical species. Even more difficult problems have been indicated to the right of the vertical dashed line. In this diagram, the multiphase properties of dusts, sprays, droplets, and other heterogeneous phenomena are indicated as an extension of local chemical kinetics. These types of processes and interactions usually need a phenomenology to be included in a practical fluid simulation.

Along the vertical axis, an easy fluid dynamic problem might involve a one-dimensional shock calculation. A hard problem might involve transient multidimensional fluid dynamics where both divergence and curl components of the flow field interact. Detailed modeling of turbulence is a very hard fluid dynamic problem. Generally phenomenologies are used to represent the macroscopic fluid effects of turbulence. This situation is indicated high on the vertical axis, above the dashed horizontal line.

The region indicated CAN DO in the figure includes problems with either an easy fluid dynamic component or an easy chemistry component, so that the computational effort can be concentrated principally on the harder aspect of the problem. Today, detailed modeling of chemical and fluid processes is possible for certain problems with one difficult aspect, provided that the

other aspects are easy enough. The COULD DO region is an extension of the CAN DO region. If ample computational resources are available and the computations are directed at answering a few, specific, well-focused questions, the COULD DO region can be simulated. No new technology is needed, but ten to forty hours of supercomputer time are required, compared to the one half to two hours for a CAN DO calculation.

Hybrid systems such as GAPS are very useful for the COULD DO problems. Hundreds of hours of equivalent supercomputer time are available in these dedicated systems by running them around the clock. The cost effectiveness of this can be 10:1 over supercomputers once the initial (often substantial) reprogramming costs have been amortized by production calculations.

The next region is marked ONE PHENOMENOLOGY to indicate that the overall problem difficulty is so great that full resolution of all the important physics, fluid dynamics, and chemistry cannot be done simultaneously. At least one of the major aspects has to be treated phenomenologically. This phenomenological component may involve quantities such as induction times, energy release rates, or flame speeds which have to be interfaced with the rest of the simulation.

The outer region of Figure 4 is labeled ALL PHENOMENOLOGY. Both the fluid dynamics and the chemical kinetics of the problem are so difficult that detailed solution of neither is considered practical. Thus interacting phenomenologies must be constructed. Turbulent reacting flow models fall into this category. As a warning, note that if one phenomenology is a suspect representation, the interactions between two are at least doubly suspect. Global conservation laws help maintain the validity of the simulations, but detailed agreement with the real problem cannot be expected.

Consider, for example, building a simulation of the interacting shock wave structures at the front of a propagating gas phase detonation. We currently have detailed chemical kinetic mechanisms for some gases, such as hydrogen-oxygen mixtures and methane-oxygen mixtures, which are good enough to qualify as practical, calibrated reaction-kinetics models. Using the detailed chemical kinetics mechanism and a compressible fluid dynamics model, a three-dimensional calculation for hydrogen-oxygen would qualify as a COULD DO category, probably taking up the maximum amount of computer time because so much information would have to be transferred in and out of the computer memory in the course of the calculation. A two-dimensional hydrogen-oxygen simulation falls in the CAN DO category. An analogous two-dimensional calculation for a methane-oxygen system is a COULD DO calculation because the kinetics reaction mechanism for methane oxidation is more complicated and expensive to use. Using less detailed models of the chemistry allows better resolution of the fluid dynamics, one of the main trade-offs made in conducting reacting flow simulations, and moves the calculation into the ONE PHENOMENOLOGY category. In this example, the phenomenology is no more complicated for methane than for hydrogen and, once the phenomenology is developed, the two-dimensional calculations with methane as well as hydrogen oxidation are in the CAN DO range.

One advantage of the chemical kinetics phenomenologies over, for example, the fluid dynamics phenomenologies, is that we know, in principle, how to incorporate these processes into calculations. Though not all of the input chemical rates or specific heats necessary for particular reactions or species are known, we have some confidence in the equations and how they fit into an overall simulation model. This helps in developing calibrated phenomenologies and in evaluating analytic models with unknown input parameters. In turbulence, the modeler is in a much weaker position with respect to what is known and how to use that knowledge. The effects of turbulence are already in the conservation equations, but various limitations and costs keep us from being able to resolve all the active space scales in detailed simulations. This problem has been dealt with by proposing phenomenologies and fitting the constants in them to experiments and other calculations. Turbulence, dusts, and sprays are areas which are more than just difficult

to incorporate from first principles. New physical insights as well as numerical methods have to be devised before the complete spectrum of multiscale, multiphase reactive-flow phenomena can be simulated nearly as well as phenomena in laminar, gas-phase combustion.

With the improvements that are becoming available in computer hardware and software, we expect to see some changes in this picture. First, it will be easier to do calculations routinely with more spatial dimensions. It will also be possible to include larger, more detailed chemical reaction schemes. When we find that the real limitation is the lack of input data, it will be possible to do the more fundamental molecular dynamics or quantum chemical calculations routinely to estimate the input data. A better bridge will exist between microscopic and macroscopic models.

There is a lot of current activity now in molecular dynamics. Until now, there have been some enlightening calculations, but very little that has been really quantitative in the area of detonations. With the advent of new types of computers and new algorithms such as the MLG, this picture should change dramatically in the next five years.

Acknowledgements

This work was sponsorted by the Office of Naval Research through the Naval Research Laboratory.

References

Boris, J.P., 1971, A Fluid Transport Algorithm that Works, in *Computing as a Language of Physics*, International Atomic Energy Agency, Vienna, 171–189.

Boris, J.P., 1986, A Vectorized "Near Neighbors" Algorithm of Order N Using a Monotonic Logical Grid, *J. Comp. Phys.* 66: 1–20.

Boris, J.P., and D.L. Book, 1976, Solution of the Continuity Equation by the Method of Flux-Corrected Transport, *Methods in Computational Physics*, 16: 85–129.

Djomehri, M.J., S.K. Doss, R.J. Gelinas, and K. Miller, 1986, Applications of the Moving Finite Element Method for Systems in 2-D, *J. Comp. Phys.*, submitted.

Gelinas, R.J., S.K. Doss, and K. Miller, 1981, The Moving Finite Element Method: Applications to General Partial Differential Equations with Multiple Large Gradients, *J. Comp. Phys.* 40: 202–249.

Lambrakos, S., and J.P. Boris, 1987, Geometric Properties of the Monotonic Lagrangian Grid Algorithm for Near Neighbors Calculations, *J. Comp. Phys.*, to appear.

Lambrakos, S., J.P. Boris, E.S. Oran, I. Chandrasekhar, and M. Nagumo, A Constraint Algorithm for Maintaining Rigid Bonds in Molecular Dynamics Simulations of Large Molecules, submitted to *J. Comp. Phys.*

Löhner, R., K. Morgan, and O.C. Zienkiewicz, 1985, An Adaptive Finite Element Procedure for Compressible High Speed Flows, *Comp. Meth. Appl. Mech. Eng.*, 51: 441–465.

Löhner, R., K. Morgan, M. Vahdati, J.P. Boris, and D.L. Book, 1986, FEM-FCT: Combining Unstructured Grids with High Resolution, *J. Comp. Phys.*, submitted.

Miller, K., and R. Miller, 1981, Moving Finite Elements, Part I., *SIAM J. of Num. Anal.* 18: 1019–1032.

Zienkiewicz, O.C., and K. Morgan, and 1983, *Finite Elements and Approximation*, John Wiley, New York.

Discussion et Synthèse - M. PEYRARD

This day on microscopic approach has been largely devoted to energy transfer processes (energy exchanges between different degrees of freedom) rather than chemical approach. Both perfect and imperfect materials have been considered.

Perfect materials

In his study of initiation mechanism, R. BARDO has shown that, at high pressure, the rate determining step is the phonon-vibron energy transfers rather than the chemical reaction half lives.
This is confirmed by the experimental investigations presented by A. ZAREMBOWITCH who found rather slow relaxation rates at low temperature. Although they are strongly temperature dependent, the studies conclude also that energy transfers between different degrees of freedom are likely to be the limiting processes.
In the classical limit, these energy transfers can be investigated by molecular dynamics. In this approach the chemistry is simply included by interatomic potentials that allow molecular dissociations on recombinations with energy change but the numerical simulations treat rather completely the energy exchanges between the different degrees of freedom. In particular no hypothesis on thermal equilibrium is made. With realistic potentials, small systems of 10^3-10^4 atoms or molecules are found to be capable of producing thermodynamic and transport properties, as well as chemical reaction rates in endothermic and exothermic reactions, in complete accord with macroscopic experiments.
In the calculations presented by M. PEYRARD and D. TSAI the chemical dissociation is assumed to be exothermic. Although the models are different some common features appear in the results :

- the shock front is narrow on a molecular scale,
- thermal equilibrium is not achieved in the shock front.

As suggested by D. TSAI, it would be interesting to compare the results given by molecular dynamics and the hydrodynamics approach. This could be done by selecting a given model system with simple chemistry and to use MD approach to produce the macroscopic equilibrium input data that are needed for the macroscopic hydrodynamics calculations. By varying the systems parameters to change the chemical reaction rates with respect to the energy transfers rate, the effect of assuming equilibrium when one or the other dominates could be estimated by observing the difference between the MD and hydrodynamics results.

Imperfect materials

Since energy transfers may be strongly affected by inhomogeneities, their role at the microscopic level has been investigated by different authors. S. COFFEY has studied the energy dissipated in the phonon field radiated by moving dislocations in shear bands. In his calculations the dislocation is described by a sum of plane waves (lattice phonons).
A. KARO has considered the case of a single molecule or mass defect embedded in a host lattice and D. TSAI has treated the case of vacancies.
Although the mechanisms may be different, all these investigations conclude that defects should help the energy transfers that finally lead to chemical reactions.
The details of the impact phenomena during lattice collisions (which occur in shock initiations) have been studied by J. TASI. Due to adhesive forces between the lattices, on clean lattices, at the arrival time, a large negative surface velocity of the impacted lattice is found. Surface impurities are needed to obtain a positive velocity (compression) at the time of impact.