1/2
2-6-95 JSD

UCRL-ID-122548

# White Paper: A Vision for a Computing Initiative for MFE

R.H. Cohen
J.A. Crotinger
D.E. Baldwin

**Revised Version**
**January 22, 1996**

Lawrence Livermore National Laboratory



Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under Contract W-7405-ENG-48.

# A Comprehensive Computing Initiative for MFE

**Ronald H. Cohen, James A. Crotinger,**
Lawrence Livermore National Laboratory
**David. E. Baldwin,** General Atomics

December 15, 1995
(Revised January 22, 1996)

## Introduction

The scientific base of magnetic fusion research comprises three capabilities: experimental research, theoretical understanding and computational modeling, with modeling providing the necessary link between the other two. The past decade has seen dramatic advances in all three areas, particularly in the first two where the larger emphasis has been placed. As a result, the MFE experimental program during this period has continued to advance performance despite the absence of major new experimental facilities, a feat which can be attributed at least in part to increased understanding enabled by the theory and computing efforts.

In parallel have been near-revolutionary advances in computer hardware and software. These have been continuously exploited by the MFE program, but the promise of further dramatic increases in computing capability offers an opportunity for a qualitative increase in the role of modeling within MFE.

The U.S. now faces a budget climate that will preclude the construction of major new MFE facilities and limit MFE experimental operations. The situation is rather analogous to the one experienced by the DOE Defense Programs (DP), in which continued viability of the nuclear stockpile must be ensured despite the prohibition of underground experimental tests. DP is meeting this challenge, in part, by launching the Accelerated Strategic Computing Initiative (ASCI) to bring advanced algorithms and new hardware to bear on the problems of science-based stockpile stewardship (SBSS). ASCI has as its goal the establishment of a "virtual testing" capability, and it is expected to drive scientific software and hardware development through the next decade.

We argue that a similar effort is warranted for the MFE program, that is, an initiative aimed at developing a comprehensive simulation capability for MFE, with the goal of enabling "virtual experiments." It would play a role for MFE analogous to that played by present-day and future (ASCI) codes for nuclear weapons design and by LASNEX for ICF, and provide a powerful augmentation to constrained experimental programs.

Developing a comprehensive simulation capability could provide an organizing theme for a restructured science-based MFE program. The code would become a central vehicle for integrating the accumulating science base. In time, it would lead to a fundamental shift in the relationship between computing and experimentation within MFE. Currently, as in the past, experimental facilities are regarded as the primary vehicle for exploration. Theory and computing are invoked to confirm experiments through analysis and understanding of their results, to provide the rationale for some of the new experiments on existing facilities, and to contribute to the design bases for conservatively extrapolated new experimental facilities. In the context we propose, the relationship would ultimately be reversed: computer simulation would become a primary vehicle for exploration, with experiments providing the necessary confirmatory evidence (or guidance for code improvements). This shift would allow much more aggressive steps to be taken in the experimental program, potentially saving the program significant resources.

## Why is Now the Time to Start?

The concept of a comprehensive simulation capability for MFE is not new and has been proposed before. The question always has been "When is the understanding base sufficiently well founded to start developing a large-scale integration?" Given that development might take several years, we argue that the time is now. At a minimum, four reasons support this contention:

- the MFE theoretical base is maturing very well;
- developing a simulation capability for MFE would provide a focus and a deliverable for the program over the next decade;
- we now have a demonstration of at least one prototype for such an endeavor; and, finally,
- starting now would permit MFE to leverage off of the much larger DP investment in ASCI.

The physics understanding and the implementing computational packages in such areas as magnetohydrodynamic (MHD) equilibrium and stability and some kinds of heating and non-inductive current drive are sufficiently advanced that they could be incorporated in mature form immediately. Edge physics packages are less well developed, and turbulent transport and 3-D resistive MHD even less so. However, even in these least developed areas, there is growing confidence that we are solving the right equations and that with advances in hardware and algorithms we will be able to deliver good solutions. We should, therefore, assume that these efforts will be successful and begin taking steps to ensure that we can take timely advantage of the results. The experience with LASNEX and the fledgling but growing experience of the MFE community with prototype efforts such as LLNL's CORSICA (see below) indicate that extremely useful codes can be generated today even in the absence of fully matured physics packages in all areas. Early versions of the new code would contain the best prevailing packages. Later, because its architecture will be made sufficiently flexible, improved packages can be included as they are developed. Thus, the new code should be viewed as an evolving entity that is constantly being modernized, much as is the case with LASNEX.

The second reason deals with the role such an initiative could play in a restructured MFE program and in providing the program meaningful deliverables as it enters a period of focus on the science base for fusion. While "science" has been the content of the program since its outset, the accumulated science base has not yet been integrated into the kind of predictive capability that we now propose. One benefit of having such a capability lies in the renewed interest in alternatives to the tokamak. In the absence of significant funds for new experimental facilities candidate alternate concepts will need to be screened as far as possible via theoretical and computational means.

The third reason is that there is now at least one demonstration of a prototype, CORSICA. The project has demonstrated the feasibility of coupling together relevant disparate-scale physics modules to make a comprehensive simulation, and the first released version is in routine use today by ITER and DIII-D scientists.

Finally, being funded at a much higher level than will be possible for an MFE initiative, ASCI provides a real opportunity for the MFE program. Many of the algorithms, software and hardware advances made by ASCI could directly benefit the MFE effort, provided the coupling and information flow between the two activities were appropriately structured.

## The Vision

In brief, our vision is as follows: code modules describing all important aspects of toroidal magnetic fusion physics (and, later, engineering) will be assembled into a common programmable framework. This framework will allow these modules to communicate with each other, either rapidly via a shared-memory database (preferred), or more slowly via shared self-describing disk files or interprocess communication. Tight coupling of the modules as in the shared-memory database approach allows simultaneous and self-consistent solution of all of the equations being solved. The CORSICA project has demonstrated that coupling algorithms can be developed to make such a tight coupling work efficiently. A comprehensive simulation of a tokamak or an alternate configuration (or a partial simulation of one of these devices) is created by combining the appropriate modules for each case.

We emphasize that the framework should be interactively programmable and extensible. A programmable system allows the user to experiment with individual modules or combinations of modules, without recompilation, and to perform tasks that were not envisioned by the module authors. Furthermore, interactive extensibility allows the code to be designed with a layered "onion-skin" structure: a menu of modules of varying complexity can be available for modeling particular physical processes. This menu would provide quick-running options, executable on work stations, for parameter surveys and other fast-look applications, and more comprehensive slower-running options, requiring supercomputers or massively parallel platforms, for more detailed studies. This layered structure ensures that the code will be usable at an earlier date and be continuously usable, even as more complete layers are being developed. Furthermore, the menu can also include competing modules, perhaps developed with different physical approximations or different numerical techniques. This would allow module developers and users to test and compare modules in a common environment, and to easily experiment with the effect that the models have in a fully self-consistent simulation.

Important aspects of such an endeavor are that it be accessible to the broad community of experimentalists and theorists, that the codes be sufficiently robust and the user interface be sufficiently intuitive that non-developers will be able to easily use it, that the constituent modules be validated against experiments, other computational models, and theories, and that both the code framework and project management structure encourage participation from code developers around the community. These aspects would be assured by having integrated teams of computational physicists, theorists, computer scientists, and experimentalists, by incorporating user- and developer-friendly tools in the framework, and by having project managers committed to these goals.

## A Prototype

The MFE Program at Lawrence Livermore National Laboratory has been developing CORSICA, a prototype for such a coupled suite, under Laboratory-Directed Research and Development (LDRD) funding. The project has developed efficient coupling algorithms and successfully applied these algorithms to several of the important coupling problems that must be addressed in a comprehensive code. These include: (1) coupling a module describing the macroscopic transport of heat, particles and current to a module that solves for the magnetic geometry (MHD equilibrium) in response to these profiles and to currents in external circuits; (2) the coupling of macroscopic transport to calculations of the microscopic turbulence that drives the transport; and (3) the coupling of transport in the core and edge regions, where the characteristic time and space scales are vastly disparate and the basic dimensionality also differs. The CORSICA project has also demonstrated the utility of structuring the code as a suite of modules connected by a programmable shell: The developers find that they can do considerable algorithm development at the interactive shell level, and that they can easily add or substitute modules. The users (including experimentalists) find that they can easily set up problems, define new diagnostics, and even define new classes of numerical experiments, without direct involvement of the development team. As a "comprehensive tokamak simulation," the project is far from complete; nevertheless, the released version (core transport plus axisymmetric MHD and circuit equations) is being frequently used by members of the DIII-D experimental team and by the ITER designers, was an active contributor to the Tokamak Physics Experiment

(TPX) design effort, and is being utilized for studies of the spheromak, an alternative to the tokamak. Throughout, the CORSICA effort has been managed as a project with time lines, milestones, etc. We believe that this should be the case for the proposed initiative, as well, in order to assure a clearly focused effort, timely development of deliverables, and a high degree of accountability for the investment.

## Relationship to Numerical Tokamak Project

The Numerical Tokamak Project (NTP) was formed by a consortium of Laboratories and universities in 1992, with the long term vision of developing "a physics-based model of an entire large fusion device." It was recognized that the most difficult aspect of such an endeavor is the calculation of the plasma turbulence that governs the confinement of particles and heat. Hence the NTP is focusing exclusively on this problem under partial sponsorship of the High Performance Computing and Communications Program. The initiative being proposed here has as its objective the achievement of the NTP's long-term vision. It assumes future success of the NTP's turbulence mission, by developing a structure into which the NTP output can be inserted along with models for the rest of the device. Until such a time as the NTP results are available in appropriate form, the new code can progress by utilizing provisional models of the turbulent fluxes, it would offer a series of increasingly realistic modeling tools as short and intermediate-term deliverables.

## Leveraging ASCI

There is an opportunity to obtain significant leverage from the Defense Programs investment in ASCI, particularly if a major portion of the responsibility for the MFE initiative were to be given to LLNL, as LLNL is one of the three major ASCI participants (with a particularly relevant ASCI program) as well as a significant participant in MFE. This leverage would come in several ways: (1) the computer hardware and associated computer science developed by and for ASCI will facilitate the MFE initiative. (2) the ASCI program at LLNL will have two components which have considerable intellectual overlap with needs of the MFE initiative. The high-energy density (HED3D) component is envisioned as a closely coupled suite of complicated codes with a programmable framework, much as we propose for the MFE initiative; and there are a number of physics ingredients, particularly in the intended application of HED3D to inertial-confinement fusion problems, which are common with MFE. The ASCI turbulence component will have spin-off benefit to the calculation of plasma turbulence and, even more directly, to the calculation of neutral-particle turbulence in the divertor region. Finally, (3) the ASCI and MFE initiatives would share some common personnel (technical and management), a common Laboratory heritage in large code construction, and a common commitment to project-oriented science.

## Appendix 1: Historical background

Theory and computing has been an important component of the MFE program throughout its near-half-century history. Much effort has gone into writing a large number of codes covering most aspects of magnetic-confinement physics. For the most part these codes have been stand-alone entities aimed at a particular piece of the physics.

The closest the tokamak community has come to "comprehensive modeling" has been "1 1/2 D Transport codes", which combine a one-dimensional core transport model with a two-dimensional MHD equilibrium calculation. Codes of this nature have been available for many years. The transport models employ analytic expressions for transport coefficients based on empirical data or simple analytic models. As predictive tools for performance of scaled-up machines, these codes have been viewed with suspicion, because of lack of confidence in the underlying transport coefficients.

About five years ago, John Dawson introduced the notion of a "numerical tokamak" – a computational model, based on first-principles physics, of an entire magnetic-fusion device, capable of self-consistently evolving the density, temperature and current profiles. Dawson proposed to implement this concept via a single large particle simulation code running on a future massively parallel computer. In particular, since the first-principles equations contain both the turbulent and long-time-scale (*i.e.,* density, temperature and current confinement time-scale) physics, this implementation would encompass both, though at considerable computational cost. The feasibility of this implementation on computers likely to be available in the next decade is doubtful. But a group at LLNL proposed that Dawson's vision could be realized by coupling separate calculations of the turbulence and the resultant transport effects, each being computed on their own natural time and space scales. In a similar fashion the evolving magnetic configuration, plasma edge and divertor physics, particle and heating sources, etc., could be added. This proposal became the CORSICA project. It built upon LLNL experience with the LASNEX code, the LLNL MFE program's experience with several of the individual codes that were to become modules in the suite, and their earlier experience with MERTH, a comprehensive simulation project for tandem mirrors. The MERTH program spawned Basis, a code development system that provides the programmable computational framework for both LASNEX and CORSICA.

At the same time, a consortium of the major fusion laboratories and university programs organized the national Numerical Tokamak Project, aimed at what is probably the most difficult part of tokamak physics computation, namely the simulation of plasma turbulence. This consortium competed for and won the designation as a Grand Challenge problem in the High Performance Computing and Communications program. In the intervening years the consortium has made excellent progress in producing simulations for experimentally relevant parameters that reproduce qualitative and quantitative aspects of tokamak turbulent transport and that facilitate our understanding of the underlying phenomena, though a number of important challenges remain. The codes developed by the NTP make full use of the most advanced computing platforms now available; the remaining challenges will be met through a combination of hardware improvements and algorithmic advancements.

## Appendix 2: Examples of magnetic-fusion physics problems which require tight coupling of modules

There are many instances where it is beneficial to tie together calculations embodied in separate codes. There are numerous examples where a loose coupling, for example by reading and writing common disk files, will suffice. This is adequate if one code post-processes the result of another, or if only occasional two-way communication is required. However there are also a number of problems in magnetic-fusion physics where a much tighter communication is required; these occur in problems where one in effect needs to solve the equations in multiple codes simultaneously as opposed to sequentially. Examples are the following:

Efficient modeling of plasma control and shaping systems and calculation of volt-second consumption requires tight coupling of core transport, MHD equilibrium, and external circuits.

Modeling of plasma evolution through a "soft beta limit" requires tight coupling of core transport, MHD equilibrium, and MHD stability (with enhanced transport coefficients related to the degree of instability).

Proper treatment of core gas-puff fueling requires self-consistent coupling of core transport to 2-D (at least)plasma edge transport and neutrals packages in order to quantitatively assess the attenuation of neutrals in the edge.

Quantitative assessment of radiative divertor detachment requires coupling of core and edge transport with good impurity transport and radiation packages, in order to account for the radiation inside the last closed flux surface as well as contamination of the core.

Modeling of disruption effects in a reactor requires coupling core transport, 2-D edge plasma transport, MHD, external circuits, radiation transport, plasma-wall interactions, and neutral transport, at least.

Quantitative modeling of the L-H transition and its effects on core and SOL properites will require a self-consistent core-edge-SOL coupling. If the currently popular paradigm (Diamond et al) of suppression of edge turbulence by sheared $\mathbf{E} \times \mathbf{B}$ flow coupled with turbulent generation of flows is correct, then the narrowing of the SOL that accompanies the transition to H mode will impact core and edge transport and the transition itself. Likewise, conditions in the core affect the other regions. Similarly, modeling of ELMs and their effects requires close coupling of the core, the edge, and an MHD instabililty model.

Calculation of turbulence with self-consistent profiles for the driving equilibrium fields (temperatures, density, flow velocity, ...): Particularly if the turbulence is non-local (e.g. because of correlation lengths non-negligible compared to equilibrium scale lengths, or coherent structures which make sizeable radial excursions), simple parameterizations of the turbulent fluxes in terms of equilibrium quantities may not be possible, and a numerical approach becomes necessary. It is best done with coupled turbulence and transport codes, because of the large disparity in equilibrium and fluctuation time scales. Such coupled simulations should be useful in addressing such fundamenetal questions as how gyrobohm-like turbulence could give Bohm-like transport scaling in a tokamak.

## Appendix 3: Status and needs of ingredients for a comprehensive simulation

The ingredients required for a comprehensive simulation suite include codes for free-boundary ideal magnetohydrodynamic (MHD) equilibrium (including interaction with coils and external circuits), plasma turbulence, core (one-dimensional, averaged over flux surface) transport, edge (two-dimensional) transport, ideal MHD stability, non-ideal, nonlinear time-dependent 3-D MHD, alpha particle confinement (orbits and instability effects), neutral beam deposition and other fueling processes, heating and current drive, neutral gas interaction with the edge plasma, plasma-surface interactions, and impurity radiation (as well as impurity transport, which may or may not be part of the main-plasma core and edge transport packages). Also required is the shell in which these will be tied together, as well as the coupling algorithms and a scripting feature which allows one to program the shell and its interaction with the modules.

A complete status report on these codes would constitute a review of the entire fusion computations program. We highlight here several modules which are central to the proposed project.

Ideal MHD equilibrium: although there are many MHD equilibrium codes in the community, most of these are "fixed boundary" (specified outer flux surface shape); there are only a few which are set up to interface with coils and external circuits. These include LLNL's TEQ, the DINA code from Triniti, Russia and Princeton's TSC. Future development required is minimal for coupling to core transport, but additional work is required to accommodate currents in the boundary plasma in a way that is consistent with the 2-D edge transport models.

Core transport: Many choices exist. One was developed for the CORSICA project which already includes the possibility of obtaining its transport fluxes from another code, such as a turbulence code, and its time-stepping accommodates CORSICA iterative schemes for coupling to turbulence. Further development required for the codes themselves is minimal, though key pieces of physics input -- such as turbulent transport -- remain the subjects of major research efforts.

Core turbulent fluxes: This is the focus area of the national Numerical Tokamak Project. Two main computational lines have emerged: gyrokinetic codes (PPPL, LLNL, UCLA), which follow particles in self-consistent (usually electrostatic) fields but average over the fast gyro motion of the particles, and gyrofluid codes (PPPL, U. Texas, GA, NERSC, ORNL), which follow multi-species fluid equations in which models for kinetic effects have been incorporated. Both have made major strides in the past five years, to the point where simulations for realistic parameters of large tokamaks are now routinely done and compared with experiment. These large-tokamak simulations are done mainly with "flux-tube codes," which follow the local neighborhood of a field line around the torus. Some progress has been made in parameterizing the results of these simulations, offering the hope of a simple way of incorporating the results into transport codes. However, there are indications that such parameterizations may not be adequate under all circumstances. (For example, the turbulence appears some times to be non-local, responding to remote changes in background profiles more rapidly than local transport models would predict.) Hence, a capability for direct coupling of transport and turbulence codes is a desirable option. For this purpose a global turbulence code is highly desirable, both because it allows for non-local turbulence effects and because it makes more efficient use of the computational grid than multiple copies of a local code. With the exception of a special-purpose code at ORNL, there are no existing global gyrofluid codes. There are global gyrokinetic codes (PPPL, UCLA), but they are more expensive to run than a comparable fluid code would be, and the adequacy of the resolution in present versions has been questioned. Hence, a global turbulence code, preferably gyrofluid, remains an important unfulfilled need. Other needs include improved treatments of collisions, fluctuating magnetic fields, and kinetic electrons.

Edge transport: the two principal codes in the U.S. are UEDGE (LLNL/INEL/MIT/ORNL) and B2/B2.5/B3 (NYU/PPPL and European collaborators). These are both 2-D fluid codes, with similar physics; both are the result of a substantial development effort. These codes are heavily employed in modeling edge and divertor performance in tokamaks. The underlying fluid approximation is at best marginally satisfied, making incorporation of kinetic effects a high priority for future work. Another area requiring future attention is incorporation of improved models of edge turbulent transport; this subject is much less well developed for the boundary plasma than for the core.

Three-dimensional non-ideal magnetohydrodynamics (MHD) to model the evolution during a disruption, sawtooth crash, ELM, etc. A 3-D resistive code has been developed over the past several years by Park at PPPL, and has been extended to include fast alphas from a gyrokinetic code. Also, OFE has launched an MHD computing initiative (NIMROD) aimed at developing a new 3-D non-ideal MHD code with equilibrium flows and arbitrary cross-section shapes.

Other modules that are less central to the basic concept of a "numerical tokamak" but still important for a comprehensive simulation include neutral beam deposition, other fueling, R.F. heating and current drive of various types, orbital loss of and instabilities driven by alphas or other energetic particles, neutral gas penetration, radiation transport, atomic physics and plasma-surface interactions.

Coupling algorithms: If two codes are to be tightly coupled with frequent two-way flow of information, then it is essential to find ways of exchanging this information that does not slow down progress of the codes. For example, if one of the codes is a diffusion equation solver, which depends on fluxes generated by another code, then the flux provided must be at the advanced time step in order to not impose a time step constraint on the diffusion code. Such coupling algorithms have been formulated and demonstrated, as part of the CORSICA project, for coupling core transport to axisymmetric MHD and external circuits, core transport to turbulence, and core transport (1-D) to edge transport (2-D). The limits of applicability of these algorithms are still being explored, and optimization and improvement remain outstanding challenges.

Programmable shell: We propose that the shell which binds the modules be fitted with an interpreted scripting language. This allows the user or developer to set up "on the fly" complex problems that use the compiled modules as building blocks. A fairly trivial example is that the programmable shell can be used to instantaneously add a new diagnostic. A less trivial example is that one can construct, without recompilation, an optimization procedure that optimizes over the output of one or more of the modules. CORSICA was written under Basis, an LLNL-developed code framework, which was the best available at the time the CORSICA project began. Basis continues to serve us well; However, some newer products are becoming available that are more compatible with object-oriented programming languages and methodology, and with massively parallel and distributed computing. These newer options should be explored.

## Appendix 4: Synopsis of Proposed Project

The objective of the project is to assemble a suite of codes from which a set can be selected to simulate a shot in an entire magnetic fusion device, or alternatively a portion of device operation can be simulated (potentially, in more detail) by selecting a smaller set. Some of the codes in the smaller set may not be ones that one would run in the larger set. For example, one might want to study turbulent transport using a gyrokinetic code coupled to an MHD equilibrium code, in a year where it is not (yet) feasible to use the gyrokinetic code as part of a comprehensive simulation. Thus the project must proceed in several directions simultaneously:

1. Continue the development and refinement of the coupled prototype suite begun under the CORSICA project. This includes additions to CORSICA core-edge coupling to include evolving MHD, impurities, and the rotation profile, extending core-turbulence coupling to global gyrofluid and gyrokinetic codes with a full set of coupled variables coupled to toroidal transport, and implementing edge turbulence models. The project may need to assume responsibility for developing the global turbulence codes if these are not forthcoming from the Numerical Tokamak consortium.

2. Add modules to the suite to make it (a) "comprehensive" as well as (b) "layered". Item (a) includes 3-D resistive MHD, R.F. heating and current drive, coupling to dynamic neutrals models, etc., as well as modules needed to describe specific alternate concepts. Some of these, such as 3-D resistive MHD, are themselves ongoing major computational physics projects. Item (b) includes, for example, adding a menu of transport models ranging from simple phenomenological ones, through models like the IFS-PPPL model or GA's quasilinear model that require running a linear stability code, to full turbulence simulations. One might also

include under (b) interpolation from a look-up table that summarizes the results of simulations.

3. Add modules that might never be used in the comprehensive suite but that might be used, in conjunction with modules from the comprehensive suite, to study in more detail a piece of overall device performance; for example, gyrokinetic codes coupled to MHD for turbulence studies, or a Fokker-Planck code coupled to electron-cyclotron or lower-hybrid ray tracing to study current drive.

4. Support module improvements, particularly with regard to robustness and portability. Physics improvements to individual modules are the responsibility of the core MFE theory/computations program, which should be expanded to support this effort.

5. Establish and maintain a program of ongoing validation for individual modules and the coupled combinations, with validation to be provided by bench-marking with experiments, theory, and other codes.

6. Adapt and implement a more modern application framework as a replacement for Basis, with the object of providing increased portability, increased user and developer friendliness, increased compatibility with new programming paradigms such as object-oriented programming, and increased ease of accommodating massively parallel and distributed applications. This should include development and upgrading of a graphical user interface.

7. Add "advisor" features to the suite to monitor initial conditions and the progress of the simulation and warn users about potential pitfalls---for example, the suite is entering into a regime where it needs to rely on a model outside its established regime of validity---and suggest alternatives.

8. Foster development of physics modules and new computational algorithms that will add important functionality to the suite or improve the existing functionality.

9. Provide support for users and developers. User support should include collaborations to set up and interpret applications, as well as diagnostic development.

10. Internal documentation: provide users' manuals in hard-copy and on-line form.

11. External documentation: write articles on algorithm and scientific research results for publication in the appropriate journals; prepare periodic progress reports and meeting presentations.

## Acknowledgments:

We thank L.D. Pearlstein and K.I. Thomassen for useful discussions. This work was performed under the auspices of U.S. Department of Energy Contract W-7405-ENG-48 and Grant DE-FG03-89ER51114.

*Technical Information Department* · Lawrence Livermore National Laboratory
University of California · Livermore, California 94551