

Numerics of Partial Differential Equations

A Proposal to the Applied Mathematical Sciences Research Program
Office of Advanced Scientific Computing Research
Office of Science
U.S. Department of Energy

Mathematics and Computer Science Division
Argonne National Laboratory
9700 South Cass Avenue
Argonne, IL 60439-4844
gropp@mcs.anl.gov

August 2000 – August 2006

The Mathematics and Computer Science (MCS) Division of Argonne National Laboratory has a small effort in the applied mathematics of the numerical solution of partial differential equations (PDEs). The goal of our work is to provide solid mathematical foundations for the use of PDEs in applications of interest to the Office of Science in the Department of Energy (DOE). The focus of the part of this effort that is supported by the applied mathematics program is the development of cutting-edge algorithms, software, and PDE models, as well as tools for their analysis. Specifically, our work is organized in this report as follows:

- Chapter 1: Algorithms and Software
 - Project 1: Implicit Solvers for Nonlinear PDEs
 - * Mathematical Aspects of Multigrid Methods
 - * Adaptive Polyalgorithmic Solvers
 - Project 2: Eigenvalue Computations for Computational Chemistry
 - Project 3: Automatic Generation of Discretizations and Error Estimators
- Chapter 2: Modeling and Applications
 - Project 1: Dynamics of Micromagnetics
 - Project 2: Evolution of Structure in Interacting Particles
 - Project 3: Dynamic Systems and Computational Biology

Our approach is to identify the needs of important DOE applications, develop the mathematics needed by those classes of applications, design algorithms for high-performance computers based on the best mathematics, and provide high-quality implementations of those algorithms as software that is usable by computational scientists. This approach provides several benefits:

- Quality algorithms are available to wide ranges of applications. Computational scientists can make use of state-of-the-art methods without needing to become experts in, for example, algorithms for solving nonlinear systems of equations.
- Experience with real applications often exposes new needs and challenges. An important part of our approach involves learning from the successes and failures of new methods in “real world” applications. These in turn drive the development of new mathematics and algorithms.

As a result of this approach, our work has been very influential. For example, many researchers, both within DOE and in the larger research community, make use of the PETSc software library [6] and the algorithms contained within it. Application domains include aerodynamics, acoustics, biology, biomechanics, econometrics, electromagnetics, fusion, materials science, multiphase flow, nanotechnology, optimization, reactive transport, and numerical algorithm analysis [14]. In turn, our work on mathematics and the algorithms expressing that mathematics has been inspired by the results obtained (both in terms of performance and correctness) with the software. In fact, much of the work proposed is in response to the general needs of state-of-the-art computational science applications, as determined by experience with the results, including the software, of the work done under this project.

We note that there is a great deal of interaction with the other applied mathematics research in the MCS Division, including optimization, computation of derivatives (automatic differentiation), and computational fluid dynamics. In addition, some of this research complements and extends several projects funded through the DOE’s Scientific Discovery through Advanced Computing (SciDAC) Initiative, namely, the TOPS (Terascale Optimal PDE Simulations) Center and two Basic Energy

Sciences (BES) projects: ASCTKD (Advanced Software for the Calculation of Thermochemistry, Kinetics and Dynamics) and TCDSECR (Theoretical Chemical Dynamics Studies for Elementary Combustion Reactions). The applied mathematical work discussed in this proposal is distinguished from the SciDAC work by its longer-term nature and focus on fundamental mathematical understanding of the issues arising in PDE simulations. The various chapters and projects of this proposal are also interrelated; for example, the solver and computational eigenvalue work forms part of the PETSc software package, while the dynamics of micromagnetics project uses the PETSc solvers for various Poisson problems that appear during the simulation process.

The applied mathematics research under this grant has been funded at a level of 3.0 FTEs: one full FTE for Dr. Leaf's work on Modeling and Applications and two FTEs spread among the other PIs, including support for one postdoctoral researcher, two visiting faculty members, and several students. The rest of the PIs' funding is provided by SciDAC grants and various small sources. We request funding for the proposed work at a level of 5.5 FTEs: another full FTE for the Modeling and Applications work (we have an ideal candidate who has previously worked with Dr. Leaf and the PETSc team), and the second to cover Dr. Knepley as an Assistant Scientist. The final .5 FTE would enable the PIs to focus more on the mathematical issues in their work than they have been able to do in recent years.

Chapter I

Algorithms and Software

The applied mathematics program in MCS has been providing analysis, algorithms, and software for the implicit solution of PDEs since 1991. Much of this work is represented in the software package PETSc (the Portable, Extensible Toolkit for Scientific computation) [6].

1 Previous Work

1.1 Executive Summary of Previous Work

The goal of this work is both to provide and to use solid mathematical foundations for the numerical solution of PDEs. The particular emphasis of our group is the interface between *algebraic* computations (sparse matrix linear solvers, nonlinear solvers, eigensolvers and element stiffness computations) and the underlying grids, discretizations, and PDEs that give rise to the required algebraic computations. Purely algebraic approaches that essentially ignore the mathematical structure of the underlying scientific problems are fundamentally limited; on the other hand, the algebraic mathematical tools, analysis, and software that have been developed by the research community are so powerful and mature that the solution process needs to be cognizant of them. Thus, we have decided to focus our efforts on the combination of the lower-level discrete algebraic properties with the higher-level continuous operators as a means to achieve breakthroughs in numerical methods for PDEs.

Through our numerous software users [?, 14], we have access to detailed technical information on “what works” and “what doesn’t work” and “what is needed” from a broad range of real scientific applications of interest to the DOE’s Office of Science. Obviously we cannot address all PDE-related issues in all of these problems, but we are able to detect trends and commonalities in the needs and then use this information to guide us to specific research problems. Currently our research focuses on three primary areas, which are discussed in detail in Section 1.2:

- developing more robust algebraic nonlinear solvers, which should enable scientific application users to handle their difficult nonlinear problems without obtaining an advanced degree and ten years’ experience in numerical analysis themselves;
- designing robust eigensolvers that leverage not only algebraic properties but also the underlying mathematical structure arising from the underlying physical systems (a current project involves computational chemistry); and
- automating the tedious process of discretization and error estimators. We note that very few PDE simulations actually provide error estimators, not only because of mathematical issues, but also since this must currently be done on a “problem by problem” basis.

Because users of our PDE software have active needs for all of these mathematical advances, we receive immediate feedback on the progress we are making and also on unforeseen limitations that may (and often do) arise.

1.2 Summary of Previous Work

1.2.1 Implicit Solvers for Nonlinear PDEs

The solution of large-scale nonlinear systems that arise from discretized PDEs is a key facet of many scientific simulations of interest to the DOE's Office of Science, including areas such as plasma physics, computational fluid dynamics, geophysics, and radiation transport. Such simulations may be either steady state (e.g., boundary value problems) or dynamically evolving in time (e.g., initial value problems), and the unknowns may be either scalar (such as heat conduction) or a system of coupled equations (such as the Navier-Stokes equations). Our research has focused on *implicit* solution techniques, which update all (or most) variables in a single global linear or nonlinear solve. Since they propagate information throughout the global problem domain at each iteration, implicit methods are often more effective for large-scale problems than explicit techniques. Work within the PETSc library prior to the initiation of the current grant (that is, before August 2000) included the development of a suite of parallel implicit nonlinear solvers, with emphasis on inexact Newton-based methods with line search, trust region, and pseudo-transient techniques to extend their radius of convergence. These nonlinear solvers can employ a suite of Krylov iterative methods and preconditioners to solve the resulting linearized systems.

While these numerical methods and software have been successfully and scalably employed in a broad range of scientific applications [14], our recent research, during the period of the current grant (that is, since August 2000), has focused on tackling difficulties that have arisen when solving particularly challenging nonlinear PDEs. We are addressing problems that incorporate strong nonlinearities through features such as multiscale, multirate, and multiphysics modeling, are extremely large (e.g., hundreds of millions of unknowns), and can run on massively parallel computer architectures, including anticipated ultra-parallel machines with up to 100,000 processes. The following subsections discuss our two main areas of recent work for implicit nonlinear PDEs, namely, (1) the mathematical aspects of (non)multigrid methods and (2) adaptive polyalgorithmic solvers. Earlier work on the scalable and high-performance implementation of linear and nonlinear iterative methods was completed and published during this period [?, ?, ?, ?].

Mathematical Aspects of Multigrid Methods

Motivated by the difficulties that some PETSc users have encountered when solving coupled PDEs and related problems in PDE constrained optimization, we have decided to tackle the disconnect between three common approaches: (1) using Newton's method with linear multigrid, (2) using nonlinear multigrid (Full approximation schemes (FAS) [?]), and (3) using traditional schemes involving operator splitting. We have developed a common mathematical approach to describing the schemes and are developing a common software framework that employs this mathematical framework. The use of automatic differentiation makes this approach a possibility, since the previous requirement of manually providing derivative information for subproblems is totally impractical.

Until the recent past, only a relatively small, exclusive group worked directly on simulations with complicated physical modeling and numerical techniques. They had the luxury of developing a rich physical understanding of the model and its numerical properties. The numerical approach (splitting, solvers, etc.) was selected based on their experiences over many years with the models. Today, with the increasingly rapid pace with which models and simulations are developed, the models are often not well understood before large-scale computation is undertaken; in addition,

the code writers may not understand the details of the model well enough to know what numerical techniques even make sense. Common symptoms of this are comments like “I coded the application but my solver doesn’t converge.” Getting a handle on why the “solver doesn’t converge” is very difficult when all one has is a monolithic application code: Is the model correct and well defined? Is the discretization stable? Is the discretization coded correctly. Are the nonlinear systems even solvable? Are the linear systems singular? Or is it simply that the algebraic systems are correctly defined and just very difficult to solve? To help answer these questions, we are developing a system to easily (and somewhat automatically) try out the various “numerical modeling” approaches that an expert would apply. We briefly outline the approach (assuming the solution of steady-state problems only to simplify the notation).

Consider the nonlinear set of algebraic equations $F(u) = 0$, generally arising from a finite difference, finite element, or finite volume discretization of some coupled set of PDEs. Let $J(u)$ denote the Jacobian of $F()$ at the point u . Simple Newton’s method may be written as

$$u \leftarrow u - J^{-1}(u)F(u).$$

Let \bar{u} denote u with all components frozen at the beginning of the Newton step. Now Newton-SOR (Newton with a single SOR cycle as an approximate linear solver) may be written as

$$u_i \leftarrow u_i - J_{ii}^{-1}(\bar{u})[F_i(\bar{u}) - \sum_{j=0}^{j<i} J_{ij}(\bar{u})(\bar{u}_j - u_j)].$$

Meanwhile, SOR-Newton (a nonlinear SOR with a single step of Newton used to solve each one-dimensional subproblem) can be written as

$$u_i \leftarrow u_i - J_{ii}^{-1}(u)F_i(u).$$

We can write a SOR-Newton with “frozen” Jacobian values as

$$u_i \leftarrow u_i - J_{ii}^{-1}(\bar{u})F_i(u).$$

By using a Taylor series expansion for $F_i(u) \approx F_i(\bar{u}) + \sum_j A_{ij}(\bar{u})(u_j - \bar{u}_j)$ and noting that $u_j = \bar{u}_j$ for $j \geq i$ one obtains, remarkably, for the frozen SOR-Newton, exactly the Newton-SOR given above. Thus we see that Newton-SOR and SOR-Newton represent a spectrum of nonlinear algebraic solvers that depend only on how recent the information in the Jacobian has been updated based on updated u values. This result trivially extends to “block-smoothers” where the i subscript denotes not just a single vector/matrix component but a set of them. For example, with a “point-block” smoother, u_i denotes all values at a single point in the grid (for multi-component PDEs). For operator-splitting based solvers, u_i may indicate all field variables of a particular type (e.g., pressure). For ADI-like schemes they represent points on a line or plane.

Taken alone, the above observation of the Newton-SOR/SOR-Newton algorithm is not of much practical value, but it turns out to extend straightforwardly to the Newton-Multigrid and Multigrid-Newton methods. We demonstrate this by outlining the control structure for both cases. (We will adopt the terminology MG-Newton, to contrast with Newton-MG, rather than the standard terminology of FAS.)

Newton-Multigrid:

Loop over Newton steps

 Restrict u to all coarser levels and compute Jacobian on each level

 (Linear) Smooth over all grid points

 Restrict residual

 Call linear multigrid on coarser level

 Interpolate correction and add to solution

 (Linear) Smooth over all grid points

Multigrid Newton:

(Nonlinear) Smooth over all grid points (using 1 dimensional Newton)

Restrict “residual” and u

 Call nonlinear multigrid on coarser level

 Interpolate correction and add to solution

(Nonlinear) Smooth over all grid points (using 1 dimensional Newton)

Since these two algorithms are virtually identical in the abstract, they should be virtually identical in the software (obtaining reuse of code for different situations is extremely important in high-performance computing, since the software is so expensive to write and maintain). Again, the differences are only in when various pieces of the Jacobian are computed. What makes this use of commonality practical is *automatic differentiation*; if one had to provide the Jacobian “fragments” required for the spectrum of algorithms manually it would be impractical to consider a unified software framework, just writing and debugging the Jacobian computations would be overwhelming. Using the ADIC [17] automatic differentiation package for C, we have developed a prototype implementation of what we call MG-Newton-MG for structured grids in parallel using vector element based smoothing, where the i represents a single component. This approach enables the application user to write the same code and then switch at *runtime* between the Newton-MG and MG-Newton methods. No other software provides even close to this much flexibility.

Operator splitting and ADI-like nonlinear solvers and linear preconditioners can also be analyzed in the same manner, only rather than solving very small dimensional subproblems, the subproblems correspond to, for example, single components (in multicomponent PDEs) over the entire grid. This corresponds, in the linear case, to the *subspace decomposition* [?] or *Schwarz analysis* [22] splitting in domain decomposition, though rather than being associated with subdomains they are associated with the various fields. Again, the key point is that for nonlinear problems, any particular algorithm relates to how recently the Jacobian values have been computed.

Adaptive Polyalgorithmic Solvers

The solution of large-scale, nonlinear PDE-based simulations using implicit and semi-implicit methods typically depends on the performance of sparse linear solvers, which may be invoked at each nonlinear iteration. In collaboration with Padma Raghavan (who served as the Maria Goeppert Mayer visiting scholar at Argonne during 2001-2002) and her doctoral student Sanjukta Bhowmick of Penn State University, we are exploring polyalgorithmic, multi-method solvers in such simulations to potentially improve the execution time and reliability of linear system solution. We are

considering *composite solvers*, which provide reliable linear solution by using a sequence of preconditioned iterative methods on a given system until convergence is achieved. We are also exploring *adaptive solvers*, where the solution method is selected dynamically to match the attributes of linear systems as they change during the course of a simulation.

Many fundamental problems in scientific computing tend to have several competing solution methods. There are often two broad classes of methods: direct, and iterative, with many algorithms within each class. The performance of a specific algorithm often depends on the numerical properties of the problem instance. The choice of a particular algorithm could depend on many factors such as its computational cost, its memory requirements, the likelihood that it computes a solution without failure, and the level of scalability of a parallel implementation. It is therefore possible to view each method as reflecting a certain trade off between several metrics of performance and reliability. Even with a very limited set of metrics (for example, the time to compute a solution, and the probability of failure), it is often neither possible nor practical to predict *a priori* which algorithm will perform best for a given suite of problems. For a difficult problem instance, an expensive but reliable method might be required, while a simpler problem instance could be solved easily with a faster method with potentially poor reliability. We observe that even a single simulation could produce problem instances with varying attributes, such as the degree of nonlinearity and conditioning of the operator. Furthermore, significant variations in attributes are natural when considering problems across different application domains. This situation has motivated us (and other researchers) to explore the potential benefits of multi-method solver schemes for improving the performance of PDE-based simulations [?, ?].

Ern et al. have investigated the performance of several preconditioned Krylov iterations and stationary methods in nonlinear elliptic PDEs [?]. Barrett et al. have studied polyalgorithms for multiprocessor implementations in a formulation where different types of Krylov methods are applied in parallel to the same problem [?]. Our work on multi-method solvers for PDE-based applications focuses on formulating general-purpose metrics that can be used in different ways to increase reliability and reduce execution times. We are also developing software infrastructure to facilitate this research, with emphasis on design issues that arise in component-based implementations.

The first type of polyalgorithmic solver we consider is a *composite*, which comprises a sequence of several different base algorithms. The failure of one method results in the invocation of the next method in the sequence until the given problem is solved successfully. A composite solver could fail if all underlying methods fail on the problem, but in practice this situation is rare even with as few as three or four methods. The cumulative reliability of the composite is thus much higher than any of its constituent methods; this reliability is independent of the ordering of the methods in the sequence. However, the ordering can significantly affect total execution time. Our work has focused on developing a combinatorial scheme for construction of an optimal composite in terms of simple observable metrics such as execution time and mean failure rates. Preliminary numerical experiments in several nonlinear PDE-based applications that use the PETSc library, including flow in a driven cavity [13] and compressible Euler flow over an ONERA M6 wing [16], demonstrate their potential in improving the overall time to solution [9, 10], though many issues remain to be explored, as discussed in Section 2.

The second type of multimethod scheme that we are investigating is an *adaptive* solver. Such a solver once again comprises several base methods, but only one base method is applied to a given

linear system, as opposed to a sequence of solvers in the composite method. We have developed a framework for adaptively selecting the linear solvers that are at the heart of the pseudo-transient Newton-Krylov algorithms [?] that are used to solve test problems of flow in a driven cavity and compressible Euler flow over an ONERA M6 wing. The numerical properties of the resulting linear systems can change during the course of nonlinear iterations [?, 19], and our adaptive solver attempts to select a base method for each linear system that best matches its numerical attributes. Our approach combines more robust (but more costly) methods when needed in a particularly challenging phase of solution, with faster (though less powerful) methods in other phases. Once again, our focus is on dynamic method selection using metrics such as normalized time per iteration and both linear and nonlinear convergence rates, with a goal of decreasing overall time to solution. Preliminary work has demonstrated how simulation time can be reduced significantly by adaptively selecting linear solvers to fit the evolving numeric properties of the linear systems that arise in such simulations, though as discussed in Section 2, many outstanding questions remain to be addressed.

1.2.2 Eigenvalue Computations for Computational Chemistry

The scalable computation of eigenvalues and eigenvectors with large sparse matrices is becoming increasingly important for Office of Science applications, for example in accelerator design [?], computational chemistry (density-functional-based tight-binding (DFTB) and MultiReference Configuration Interaction (MRCI)) [?], and fusion stability studies. These matrices arise from discretizations on a grid or on a hierarchy of approximation models in a basis expansion. The standard numerical approaches as typified in the software package ARPACK [?] are a good starting point but require improvements for better scaling to large numbers of processors and large sizes, better numerics for tight clustering and multiple eigenvalues, and tighter integration with linear system solvers (for example, with shift and invert).

Our mathematical work has focused on three aspects: (1) the unique computations required for DFTB, (2) the Subspace Projection Approximate Method (SPAM), which uses multilevel preconditioners for the Davidson iterative method (particularly suited to electronic structure calculations), and (3) a general abstract software framework for all computational approaches, but with efficiency for specialization (much as PETSc provides a framework for linear solvers).

The mathematical core of DFTB and related electronic structure methods is the generalized real symmetric eigenvalue problem. More than 90 percent of the computational time in current implementations of DFTB is spent on computing eigenvalues. The size of the matrix is the number of basis functions in the system (four times the number of atoms for an sp minimal basis set). Our goal is to be able to handle on the order of 100,000 atoms. The DFTB application requires approximately 60% of the eigenvalues and their associated orthogonal eigenvectors in high accuracy. Although recent years of research and development have resulted in many sparse iterative eigensolvers, none meets the robustness and accuracy requirements of the DFTB applications. Thus far, only the eigensolvers within LAPACK and ScaLAPACK can be used. As these techniques are based on the dense QR algorithm, they require storage of entire matrices and heavy inter-processor communications on parallel machines, resulting in strong degradation of performance with increasing system size. Note that the matrix size 100,000 is not large by today's high-performance computing standards, but the sheer number of eigenvalues and eigenvectors that are needed (very accurately) and the number of distinct eigen problems that need to be solved (in the thousands)

make this a difficult and interesting computation challenge. We note that today a DFTB with a matrix size of 1,000 is considered large.

The Lanczos algorithm is well established for computing a small set of extreme eigensolutions of sparse symmetric matrices. We use ARPACK to implement Lanczos iterations with a shift-and-invert spectral transformation over a distributed portion of the requested eigen spectrum. The spectral transformation enhances convergence on a desired portion of the spectrum and provides reliability through mapping congregated eigenvalues to well-separated values. Although ARPACK is relatively mature and reliable, it does not support the block version of the Lanczos algorithm, which is useful for efficiently handling eigenvalues of high multiplicity, and it provides no shift selection scheme for computing large portions of the spectrum.

We use small error tolerances for computing multiple eigenvalues to ensure the correctness of computed eigensolutions (larger tolerances on non-block Lanczos iterations likely lead to extraneous eigenvalues [?]). Thus far, the small error tolerance does not appear to be a concern as long as the shifts and other Lanczos iteration parameters are carefully chosen, and it helps to produce eigensolutions with higher accuracy.

We have initiated development of: (1) the strategy for choosing shifts, (2) verification of the trust intervals (the interval in which the number of eigenvalues expected agrees with the number actually computed) and the technique for automatically recovering missing eigenvectors of the multiple eigenvalues, (3) bookkeeping of computed eigensolutions, and (4) task management in a parallel implementation.

The code developed in the first phase of this project was successfully tested on Argonne's Linux cluster, Jazz, and the following data was collected: On one processor,

1. the code solves larger size problems than LAPACK;
2. the code is slightly faster than LAPACK in most cases ($N=1,000, \dots, 5,000$ were tested), with best performance on larger systems;
3. the code gives more accurate eigenvalues and eigenvectors than LAPACK in general, largely because the shift-invert improves the numerical conditioning of the eigen problem; and
4. the code gives satisfying solutions for eigenvalues with separation on the order of 10^{-5} .

On parallel processors, the code has been run with up to 32 processors and matrix sizes from 400 to 8,000. Speedups are seen for up to 8 processors. Overall, the algorithm and code design is relatively simple, setting a good foundation for code robustness.

In the area of multilevel matrix approximations for Davidson's Method, during the past three years we have worked with colleagues in BES and related university collaborators on several related sub-projects. The work involves a common thrust of solving linear algebra problems with applications driven toward high-performance computation.

In 2001 we developed the Subspace Projection Approximate Method (SPAM), which extends the Davidson iterative method for eigenvalue calculations; initial results are discussed in [?]. This approach focuses on problems where the matrix-vector products are difficult and expensive (such problems arise in many applications areas). We construct a sequence of approximating matrices and generate expansion vectors at each approximation level. Our goal is to use approximations that lead to relatively inexpensive matrix-vector products. Thus, while the total number of matrix-vector products may become large, the number of expensive matrix-vector products with the "true" matrix

is greatly reduced. For the problems we have and plan to study (self-consistent field, configuration interaction, molecular vibration analysis, and cumulative reaction probability), the matrix-vector product with the Hamiltonian matrix is the dominant computational cost.

The software developed for SPAM is freely available; there have been nearly 100 downloads thus far, and we continue to see several downloads per week. Highlights of recent work are discussed in [?] by our postdoctoral appointee, Y. Zhou.

In a related subproject we been developing methods for calculating time-independent cumulative reaction probabilities (CRP). The calculation of these coefficients is fundamental to the accurate simulation of chemical phenomena from combustion to smog. These coefficients can be obtained from exact quantum CRP by methods that are computationally intensive and rarely applied to problems with more than three degrees of freedom (DOF). Transition state theory (TST) rate constant methods in effect approximate the CRP. Thus, exact CRP methods provide a calibration for TST-based methods. We are developing parallel codes to carry out exact CRP calculations on problems of six to ten DOF or larger on massively parallel processors. The solution of CRP problems involves iteratively obtaining the eigenvalues of the probability operator by the evaluation of two Green's functions in each eigenvalue iteration. The Green's function evaluations themselves involve the iterative solution of linear systems. We have developed a parallel code by using PETSc to address this central step in the method. We use GMRES to resolve the Green's functions and a Lanczos method to resolve the eigenvalues. This code has been tested on a model potential energy surface that has the property of being expandable to any number of dimensions but has an analytic CRP. Thus, even for large-scale problems we can benchmark against analytic results. Tests on this model show that the code is scalable to a hundred or more processors with time to solutions at NERSC measured at about ten minutes per eigenvalue for seven DOF problems. However, the solution of the Green's function linear systems is a central bottleneck in the calculation. At present we have used only diagonal preconditioning for the Green's functions. Our current and future focus is directed at the role of effective preconditioners for this class of problems. To investigate this direction, we also have developed a single processor code to examine banded preconditioners, and it is giving insight into to the challenges of preconditioners that are more global than the simple diagonal method. Future work will examine global preconditioners such as SPAM that can be applied for striped, blocked, or banded systems as well as Poirier methods (optimal block preconditioning). In this regard we have established a collaboration with W. Poirier under the DOE/MICS Young Investigator Program ("Efficient Iterative Linear Solvers and Eigensolvers for Terascale Computing"). During this past summer, we held a week long meeting between W. Poirier, his students, T. Carrington (Univ. of Montreal) and Argonne computational and chemical scientists targeting the theory and applications of iterative methods for electronic structure calculations and the role of preconditioning techniques in quantum chemistry. These discussions have established a foundation for new directions in the applications of our work with SPAM and the joint development of optimal block preconditioners complimenting W. Poirier's work.

We were fortunately able to host a faculty visitor, Jose Román (Polytechnic University of Valencia, Spain), who had designed a prototype set of interfaces for eigenvalue computations. During his time at Argonne, we worked with Jose to refine the mathematical design to increase generality and allow for more problem-specific tuning. Some of the needs for problem-specific tuning were brought out by our work in the eigenvalue computations for DFTB to handle the eigenvalue clustering. The Scalable Library for Eigenvalue Problem Computations (SLEPc) is now available for

download [1]. As the first parallel, general purpose eigenvalue solution package designed using object-oriented techniques, SLEPc has much more flexibility than current packages. In addition, SLEPc provides interfaces to ARPACK, BLZPACK, PLANZO, TRLAN, thus allowing users the power of most state of the art eigenvalue software available.

SLEPc is designed around two main classes, the Eigenvalue Problem Solver (EPS), which handles all state related to the particular problem one is solving (the type of eigenvalue problem, the tolerances, the operators (matrices), the orthogonalization approach), and the Spectral Transformation (ST) class, which encapsulates the functionality required for acceleration techniques based on the transformation of the spectrum (e.g., shift and invert). SLEPc employs PETSc for all of its sparse linear algebra computations and thus can use virtually any linear solver technology, either provided directly within PETSc or available via PETSc interfaces to external software packages. Another benefit of building on PETSc infrastructure is the very small size of SLEPc and the ease at which new methods may be added.

Eigenvalue problems arise in a large number of disciplines of science and engineering. They constitute a basic tool in modeling and simulation. As modern technology advances, the need for solutions of larger systems with extreme clustering eigenvalues is anticipated. The research and development proposed here will leverage advances in computational mathematics, numerical algorithms, and software packages.

1.2.3 Automatic Generation of Discretizations and Error Estimators

The goal of this work is to raise the level of abstraction at the interface between application scientists and the numerical simulation framework when discretizing PDEs. This will have the joint benefits of increasing productivity, elevating the potential for optimization, and reducing the likelihood of errors in common calculations. Ideally, a PDE simulation could be constructed from a specification of a weak form of the governing equations, the domain geometry, and the discretization method. The user will still have the freedom to alter any part of the constructed simulation, as we will employ the same layered interface design used for PETSc. These components of the simulation will be representable using the Mesh and Expression components developed for our next generation software, PETSc-3. We are also collaborating with the MCS Optimization group to extend the Expression components in order to represent variational *inequalities* as well.

However, it is abundantly clear, especially for scientific components, that a model for component verification will be necessary in order to assure the user that the underlying simulation has been correctly constructed. That is, not only should a component report the “result” of its computation, it also should report information about the quality of the result, i.e. “error bars”. For components that solve PDEs, the first step along this path is to institute generic *a posteriori* error estimates that are reported with the same regularity as the approximate solution and its associated residual.

Code generation will be an integral part of the proposed work. The starting point for our numerical code generation will be the automatic generation of finite element integration routines. These routines, which form the kernel of many large-scale PDE applications, are often free of branches and have a simple flow of execution. Such characteristics make these routines excellent candidates for generation. In fact, since element matrix routines arise from automatic differentiation, we can generate optimal code based upon several metrics using the ADIC-2 [17] framework. The

integration routines can be constructed on the basis of the mesh geometry and basis function expressions, which are represented symbolically. After integration, the assembly of discrete functions and operators over the domain is a matter of bookkeeping based upon the mesh, the distribution of unknowns on an element for the given discretization, and the boundary conditions. Boundary conditions are also represented symbolically as expressions, and therefore code can be automatically generating for their evaluation as well.

We adopt the finite element model for error estimation, as it is theoretically sound [3, 4, 5, ?, ?]. This requires a representation of the dual problem, whose solution is used to recover accurate error estimates for arbitrary functionals over the domain [7]. The dual is represented and solved exactly as the original PDE. This approach is not as burdensome as it may seem, since this same information is required to perform sensitivity analysis and much of it may be automated.

The estimated error can be used not only to verify the solution, but also to adaptively control the accuracy of the computation. Such adaptivity is often accomplished through mesh refinement but could also involve changing the element basis or the integration parameters. This technology is a necessary first step toward verifiable components for the solution of PDEs.

An excellent illustration of the proposed new technology is provided by a prototype Poisson problem example in our next generation software, PETSc-3. This example combines nearly every aspect of the new infrastructure to render a provably good solution on a complicated domain with minimal user effort and great flexibility. We represent many of the key equations symbolically, as an abstract syntax tree (AST), which allows complex manipulations, usually using the visitor pattern. For instance, differentiation, variation, integration, interpretation, and code generation are all accomplished with this methodology. The problem is solved by using an unstructured mesh, which is adapted using *a posteriori* error estimates. A full account of this example is given at [18].

We will briefly explain the mathematics behind the above prototype. The total discretization error in a given mesh element τ can be split into two components, the locally produced truncation error and the transported error, often called a pollution error in mathematical literature,

$$e_\tau = e_\tau^{loc} + e_\tau^{tran}. \quad (1)$$

In order to control e_τ^{loc} it is sufficient to know the residual ρ_τ on a mesh element τ , but in order to control e_τ^{tran} we need to know the influence of the residual at another mesh element $\rho_{\tau'}$ on the error at τ . In the continuous case, this relationship is governed by the Green's function.

We seek an *a posteriori* error estimate scheme that has the following properties:

- it detects a local truncation error;
- it detects propagated error; and
- it designs 'economical' meshes, i.e., given N elements, it gives the best distribution of elements on the domain.
- We would also like to achieve *a posteriori* accuracy control of physical quantities in which we are interested.

A scheme that satisfies all of the above properties involves the computation of local residuals and the solution of the dual problem. Loosely speaking, using the local residual we will be able to control the truncation error, and using the solution of the dual problem we will be able to control

error propagation. We are also able to represent the error in various physical quantities, or in fact arbitrary functionals, in terms of local residuals and solution of the dual problem.

The solution z to the dual problem satisfies

$$\begin{aligned} -\Delta z &= e \text{ in } \Omega, \\ z &= 0 \text{ on } \partial\Omega, \end{aligned} \quad (2)$$

where $e = u - u_h$ is the error in the Galerkin solution u_h . The corresponding variational form is given by

$$\langle \nabla z_h, \nabla \chi \rangle = \langle e, \chi \rangle \quad \forall \chi \in S_h. \quad (3)$$

The solution z can be thought as a generalized Green's function, in that it relates the residual in other mesh elements with the error in a given element. The error e in the L_2 norm can be expressed as

$$\begin{aligned} \|e\|^2 &= \langle e, e \rangle \\ &= \langle e, -\Delta z \rangle \\ &= \langle \nabla e, \nabla z \rangle \\ &= \langle \nabla e, \nabla(z - \chi) \rangle. \end{aligned} \quad (4)$$

The last step in the above derivation utilizes a key property of the Galerkin approximation,

$$\begin{aligned} \langle \nabla e, \nabla \chi \rangle &= \langle \nabla(u - u_h), \nabla \chi \rangle \\ &= \langle f, \chi \rangle - \langle \nabla u_h, \nabla \chi \rangle \\ &= 0, \quad \forall \chi \in S_h, \end{aligned} \quad (5)$$

which might be called the *orthogonality of the error*, since e is orthogonal to all functions in the test space.

By cell-wise integration of the expression (4) by parts, we obtain

$$\begin{aligned} \|e\|^2 &= \sum_{\tau \in T_h} \langle \nabla e, \nabla(z - \chi) \rangle_{\tau} \\ &= \sum_{\tau \in T_h} \langle f + \Delta u_h, z - \chi \rangle_{\tau} - \\ &\quad \langle n \cdot \nabla u_h, z - \chi \rangle_{\partial\tau}. \end{aligned} \quad (6)$$

When we sum over all elements, the boundary integrals give two contributions from each element edge, computed in opposite directions. Suppose $\tau_1, \tau_2 \in T_h$ share a common edge, $e_1 = e_2 \in \tau_1 \cap \tau_2$, with corresponding normals n_1, n_2 . The contribution from that edge is

$$\int_{e_1} \nabla u_h|_{\tau_1} \cdot n_1(z - \chi) ds + \int_{e_2} \nabla u_h|_{\tau_2} \cdot n_2(z - \chi) ds.$$

Using that $n_1 = -n_2$ and the fact that e_1 and e_2 have opposite orientations, the above contribution is equal to

$$\begin{aligned} &\int_{e_1} \nabla u_h|_{\tau_1} \cdot n_1(z - \chi) ds + \int_{e_1} \nabla u_h|_{\tau_2} \cdot n_1(z - \chi) ds \\ &= \int_{e_1} (\nabla u_h|_{\tau_1} + \nabla u_h|_{\tau_2}) \cdot n_1(z - \chi) ds \\ &= \int_{e_1} [\nabla u_h] \cdot n_1(z - \chi) ds, \end{aligned} \quad (7)$$

where $[\nabla u_h] = \nabla u_h|_{\tau_1} + \nabla u_h|_{\tau_2}$ denotes the jump in ∇u_h across the edge e_1 in the direction of the normal n_1 .

Thus, from (6) the error in the L_2 norm can be expressed as

$$\|e\|^2 = \sum_{\tau \in T_h} \left\{ \langle f + \Delta u_h, z - \chi \rangle_{\tau} - \frac{1}{2} \langle [n \cdot \nabla u_h], z - \chi \rangle_{\partial\tau} \right\}, \quad (8)$$

where $[\nabla u_h \cdot n]$ denotes the jump in the gradient of u_h across boundary in the direction of outer normal n . Thus, on each cell the residual splits onto two parts, the *equation residual*, $f + \Delta u_h$, and the *flux residual*, $[n \cdot \nabla u_h]$. Both these residuals are multiplied by the weighting function $z - \chi$, which provides quantitative information about the impact of the cell residual on the global error $\|e\|$.

To make the expression (8) practical, we need a “good” approximation to $z - \chi$. One obvious choice is to find a Galerkin approximation \hat{z}_h to z by using P_{k+1} elements and take χ to be restriction of \hat{z}_h on P_k elements. However, this strategy is computationally intensive. An alternative method finds a Galerkin approximation, z_h , by using P_k elements, and then uses local information from z_h to obtain a higher order interpolation $i_h^+ z_h$, so that the weighting function $z - \chi$ becomes $i_h^+ z_h - z_h$. Although the latter method lacks a good mathematical foundation, it is much more cost efficient. The method appears to work well in many practical cases [7]. Moreover, in many cases, such as linear and semi-linear second order PDE’s, the stiffness matrix from the original solution may be reused for the calculation.

Often in applications we would like the error in some derived quantity involving u rather than just the solution error. For simplicity, we assume that this quantity can be represented by a linear functional, $J(u)$. Let z be the solution of the corresponding adjoint problem

$$\langle \nabla \varphi, \nabla z \rangle = J(\varphi), \quad \forall \varphi \in H_0^1(\Omega), \quad (9)$$

and z_h be a finite element approximation given by

$$\langle \nabla \varphi_h, \nabla z_h \rangle = J(\varphi_h), \quad \forall \varphi_h \in S_h. \quad (10)$$

Using the orthogonality of the error (5) and the orthogonality of the dual problem

$$\langle \nabla \varphi_h, \nabla z - \nabla z_h \rangle = 0, \quad (11)$$

we have

$$J(e) = \langle \nabla \varphi, \nabla z \rangle - \langle \nabla \varphi_h, \nabla z_h \rangle = \langle \nabla e, \nabla (z - \chi) \rangle, \quad \forall \chi \in S_h. \quad (12)$$

Integrating cell-wise by parts, we arrive at functional error representation

$$J(e) = \sum_{\tau \in T_h} \left\{ \langle f + \Delta u_h, z - \chi \rangle_{\tau} - \frac{1}{2} \langle [n \cdot \nabla u_h], z - \chi \rangle_{\partial\tau} \right\}. \quad (13)$$

By the Riesz representation theorem, the error functional $J(e)$ can be represented as $\langle e, \psi \rangle$, for some $\psi \in H^{-1}(\Omega)$. Thus, the error functional can be viewed as a projection of the error e in the direction of ψ . Some widely used choices for ψ are:

- $\psi = 1_{\omega}/|\omega|$, to estimate the average over $\omega \subset \Omega$, where 1_{ω} is the characteristic function of ω and $|\omega|$ is its volume.

- $\psi = \delta_p$, to obtain the error at a point p , where δ_p is the Dirac measure at point p . In practice we can take $\psi = 1_{B_p(\epsilon)}/|B_p(\epsilon)|$, for ϵ sufficiently small, where $B_p(\epsilon)$ is a Euclidean ball of radius ϵ centered at p .
- $\psi = \delta_c$, to estimate the average on a curve c .
- $\psi = 1_\omega e/\|e\|_\omega$, to estimate the error in the L_2 norm over $\omega \subset \Omega$. In practice to obtain e , it is common to approximate u by higher order elements or use a finer mesh.
- ψ that is weakly defined by

$$\langle v, \psi \rangle = -\langle \nabla v, \nabla u_h \rangle - \langle v, f \rangle, \forall v \in H_0^1(\Omega).$$

to estimate the energy norm $\|\nabla e\|$. Indeed with this choice of ψ

$$\begin{aligned} \langle e, \psi \rangle &= -\langle \nabla e, \nabla u_h \rangle - \langle e, f \rangle \\ &= -\langle \nabla e, \nabla u_h \rangle + \langle \nabla e, \nabla u \rangle \\ &= \langle \nabla e, \nabla e \rangle. \end{aligned} \tag{14}$$

For more complicated examples we refer to [7].

1.3 Highlights

Highlights of recent research include the following:

- We developed a common framework for MG-Newton (FAS) and Newton-MG algorithms; we tested and publicly distributed a prototype implementation for structured meshes.
- We conducted preliminary research on adaptive and composite polyalgorithmic solvers for nonlinear PDEs, with emphasis on the dynamic selection of Krylov methods and preconditioners within Newton methods.
- We developed and implemented a general abstract software framework for large-scale eigensolvers, called SLEPc (the Scalable Library for Eigenvalue Problem Computations).
- We developed and implemented a Subspace Projection and Approximation Method (SPAM), for scalable (order N work) eigenvalue computations.
- We developed a prototype algorithm for eigenvalue computations arising from density-functional-based tight-binding (DFTB) problems in computational chemistry; this algorithm is faster than and more accurate than LAPACK for our model problems.

1.4 Program Personnel

Staffing levels are given in FTEs; FY03 is the current level, and FY04 is the level requested for proposed work. Dr. Knepley would be promoted to Assistant Scientist in FY04.

Principal Investigators

	FY03	FY04
William Gropp	0	0.25
Lois Curfman McInnes	0.38	0.5
Mike Minkoff	0.3	0.5
Barry Smith	0.5	0.5

Postdoctoral Scholars

	FY03	FY04
Matthew Knepley	0.3	1

Senior Scientific Programmers

	FY03	FY04
Satish Balay	0.5	0.75

Other Staff:

Investigators

- Hong Zhang

Visiting Faculty

- Padma Raghavan (Penn State University)
- Jose Román (Polytechnic University of Valencia, Spain)

Summer Graduate Students

- Sanjukta Bhowmick (Penn State University)
- Paulo Goldfeld (Courant Institute, New York University)
- Ernesto Prudencio (University of Colorado at Boulder)
- Oliver Rheinbach (University of Essen, Germany)

2 Proposed Work

2.1 Executive Summary of Proposed Work

As explained in Section 1.1, the overarching goal of our research in algorithms and software for PDEs is both to provide and to use solid mathematical foundations for the numerical solution of PDEs, with emphasis on the interface between *algebraic* computations and the models that give rise to them. Our proposed future work is motivated by ongoing interactions with computational scientists studying topics such as plasma physics, computational fluid dynamics, geophysics, computational chemistry, and radiation transport [14], with particular emphasis on collaborations with that have recently arisen with fusion researchers, A. Bhattacharjee of the University of Iowa, and S. Jardin of the Princeton Plasma Physics Laboratory. We are also working with computational chemists A. Wagner and R. Shepard at ANL. Additional motivating applications involve the geodynamics research of R. Katz and M. Spiegelman of Columbia University [?] and work by Argonne’s computational biology group [?, ?] on metabolic dynamics.

Proposed future research focuses on three primary areas, which are discussed in detail in Section 2.2:

- robust algebraic nonlinear solvers,
- eigenvalue computations, and
- discretizations and error estimators.

2.2 Outline of Proposed Work

2.2.1 Implicit Solvers for Nonlinear PDEs

Our proposed research in implicit solvers for nonlinear PDEs addresses the ever expanding challenges created by current trends in scientific simulations and computer architectures. As memory capacities and processor speeds continue to increase, application scientists are developing higher-fidelity models that incorporate more multiphysics, multiscale, and multirate features. We are thus faced with systems that incorporate a rich variety of time scales and nonlinearities and are orders of magnitude larger than previously considered. In addition, as the DOE Office of Science is beginning to invest in research in ultra-parallel computing with up to 100,000 processes, we must rethink our approaches to implicit methods for PDEs on such machines. Our proposed research in polyalgorithmic solvers and multigrid methods for nonlinear PDEs tackles these issues with emphasis on incorporating asynchronous behavior and dynamic adaptivity.

Mathematical Aspects of Multigrid Methods

As noted in the discussion of previous work in Section 1.2, we have developed a mathematical/algorithmic/software framework for the spectrum of MG-Newton-MG nonlinear PDE solver algorithms. The software that we currently distribute publicly supports finite difference discretizations implemented in C/C++ on structured meshes. The main features of our work proposed over the next three years are as follows:

- (years 1-3) developing a framework for the convergence theory of MG-Newton-MG methods and performing systematic numerical studies for several large-scale applications to develop an understanding of convergence behaviors of the various hybrid algorithms,

- (years 2-3) studying the ultra-parallel scaling of MG-Newton methods as compared to Newton-MG and hybrid techniques,
- (year 1) extending this approach to finite element discretizations,
- (year 1) supporting block structured discretizations, and
- (year 2) enabling application code function evaluations in Fortran and possibly other languages.

Newton-Krylov-Multigrid methods are fundamentally limited by the frequent global reductions called for in the Krylov methods (vector inner products and possibly norms). Although some improvement can be obtained by reorganizing the calculations to combine reductions, when using on the order of 100,000 processes, any global reductions in core computational kernels are problematic. MG-Newton and Newton-MG methods, as well as their various hybrids, are natural candidates that offer the potential for better scaling. In particular, the MG-Newton method can require *no* synchronous global reductions.

The largest parallel runs made with PETSc have been on the ASCI Red machine [2] with slightly more than 6000 processes for an unstructured mesh CFD application; they show surprisingly good parallel efficiency – above 70 percent in going from 128 processes to over 6000. But, even on the low latency ASCI Red interconnect, the inner products and norms required by Newton and GMRES are beginning to dominate the computation, limiting scalability. We propose a two-pronged approach to evaluating MG-Newton and hybrid methods on ultra-parallel systems. First, we will perform extensive experimental studies on ultra-parallel system simulators using our coupled PDE model codes, in particular involving magnetohydrodynamics [21], computational fluid dynamics [2], and Maxwell’s equations. Second, we will develop analytic computational models [15] for MG-Newton-MG and include our “best guess” for parameters from upcoming systems, including IBM BG/L and BG/P. The computational models can then be compared with the results from the experimental studies.

Most finite difference codes compute a *node-based* discretization, that is, the residual evaluation is computed one vector component at a time. For example, $F_i(u) = u_{i+1} - 2u_i + u_{i-1} - dx^2 f(x_i)$ corresponds to a discretization of a one-dimensional elliptic problem. This node-based calculation makes implementations of MG-Newton methods straightforward since the application code explicitly provides the one (or low) dimensional functions, $F_i(\cdot)$. In contrast, finite element codes almost always employ *element-based* calculations. Here, $F_i(u)$ is never explicitly provided by the user, but rather portions of the function are computed on all elements that share a particular node, and the results are added together. Thus, MG-Newton is not trivial to implement in a finite element context. We intend to explore two approaches. The first corresponds to replacing the classical element-based computation of finite element residuals with a nodal-based approach. The second corresponds to an “element” based MG-Newton, much as AMGe [?] is an element based linear multigrid family of algorithms.

Although extremely well developed convergence theory exists for linear multigrid [?, ?], and some basic theory exists for FAS (nonlinear multigrid), there is not yet a complete understanding of when or why FAS performs better than Newton-MG or vis-versa. Difficulties in providing general-purpose FAS libraries have also slowed down the common use of FAS in applications, despite anecdotal evidence that it can be far superior to Newton-MG. With our ability to compute the Jacobian information automatically and on-the-fly, we intend to develop some basic understanding

of the issues that determine when one approach may be more efficient than the other. In addition, our work on adaptive polyalgorithmic solvers makes us ideally suited to develop adaptive MG-Newton-MG schemes and code that will determine, during the calculation, what portion of the Jacobians to compute and when.

Adaptive Polyalgorithmic Solvers

As discussed in Section 1.2, our work thus far in adaptive polyalgorithmic solvers has focused on preliminary exploration of composite and adaptive methods for several PDE-based applications. This initial research has taught us what metrics and techniques can be effective for dynamic algorithm selection and composition, as well as what features are needed in a software framework to manage such adaptivity. While this work is a solid starting point, many issues remain to be addressed in our quest to develop an algorithmic and software framework for adaptive, multimethod solvers. We will continue to build intuition by exploring composite and adaptive solvers in additional nonlinear PDE-based applications that are under development by our collaborators and already incorporate the nonlinear solvers within PETSc. Using this knowledge, we will develop heuristics and theory for polyalgorithms in preconditioned Newton-Krylov methods for particular problem classes. We also propose to develop software infrastructure that is sufficiently general to coordinate polyalgorithmic adaptivity in areas beyond our initial focus on preconditioned Krylov schemes within Newton-type techniques, such as discretization, Jacobian evaluation, and the hybrid MG-Newton-MG methods discussed above. Particular tasks include:

- (year 1) exploring composite and adaptive approaches in MHD and CFD applications,
- (year 2) developing heuristics and theory for particular classes of problems, and
- (years 2-3) building an algorithmic and software framework to manage polyalgorithmic adaptivity for nonlinear PDEs.

2.2.2 Eigenvalue Computations for Computational Chemistry

We propose to focus on two special features of the DFTB eigen problems:

- (year 1) clustering of eigenvalues,
- (years 1-2) repeated eigensolutions.

In DFTB, all N eigenvalues are located in a fixed interval $[-2.0, 1.5]$, with matrix dimension N ranging from several hundred up to hundreds of thousands. Therefore, the average eigenvalue separation is $O(1/N)$, which becomes very small as the system size N increases. In addition, extreme clustering of the eigenvalues and clustering of clusters are present. For example, a mixed_SiO2_P0 matrix of size $N = 5,524$ has two large clusters, each containing more than 100 eigenvalues. From mathematical theory, the reliability and efficiency of any eigensolvers heavily depend on the eigenvalue separations. Furthermore, for clustered eigenvalues, the computational efficiency is highly sensitive to the shift selection. A shift needs to be close to the cluster; otherwise, a large number of Lanczos runs occurs. However, the shift must be kept away from the cluster when computing eigenvalues that are in the neighborhood of a computed cluster, in order to avoid repeated computation of the cluster. The good news from our preliminary investigation is that the clustered eigenvalues can be obtained with satisfactory accuracy and in reasonable efficiency if an “optimal” shift is applied.

How to compute clustered eigensolutions is the immediate challenge in developing a massively parallel efficient eigen solver for the DFTB applications. Handling clustered eigensolutions as large as we anticipate from the DFTB application has not been explored in either mathematical theory or numerical techniques. We propose to investigate how to do the following: detect clusters; estimate their locations and sizes; determine criteria for moving toward clusters that are yet to be computed and away from those that have been computed already; determine parameters and their values for restarting the Lanczos iteration after a cluster has been detected; achieve a parallel load balance in the presence of clustered eigenvalues; and determine the parallel task redistribution when a cluster lies between two processors as well as maintain the orthogonality of eigenvectors.

For DFTB, the eigenvalue/eigenvector pairs are required repeatedly for thousands of time steps. Fortunately, these pairs generally evolve smoothly in time, so theoretically information from one time step can be used to bootstrap the solution at the next time step. Unfortunately, none of the current software packages and algorithmic approaches provides a convenient way to leverage the previous solution. We will develop such an approach and provide analysis for the quality of such “initial guesses”. For DFTB, the general solution is non-trivial since eigenvalues do occasionally shift between clusters. We will study this phenomenon to determine how fast convergence can be maintained even with these jumps.

In collaboration with Larry Curtis, Peter Zapol, and Michael Sternberg of the Material Science Division at Argonne, we will carry out the proposed tasks at three levels:

DFTB application modeling level: We will predict the location or size of the cluster—use *a priori* information to aid runtime decision making; this capability will be useful in maintaining a good parallel load distribution.

Mathematical analysis level: We will develop an improved approach for cluster detection (currently, we use a heuristic number as a lower bound on the number of Lanczos runs for the detection); determine how much to move toward or away from a cluster based on spectral transformation information; use mathematical parameter estimates for restarting Lanczos runs; and replace a k -sum of rank-1 eigenvector projections by a single rank- k eigen space projection for the clustered eigensolutions.

Software development level: We will design and test heuristic parameters and schemes and provide reliable backup in the case that an optimized approach fails (solver robustness), thereby using a polyalgorithmic approach. We will build the optimal schemes as objects for code reuse, portability, and update. The code developed in this proposed research would also serve as an interface to external sparse eigen packages being used for repeatedly computing large number of dense and highly accurate symmetric eigensolutions on massively parallel computers.

In the area of multilevel matrix approximations for Davidson’s Method, we plan to extend the Subspace Projected Approximation Method (SPAM) to other areas of linear algebra and applications. We plan to extend our work from the standard eigenvalue problem to

- (years 1-2) the solution of linear equations,
- (year 2) the solution generalized eigenvalue problems, and
- (year 3) the solution of nonlinear tensor equations.

In developing the mathematical foundations for the SPAM approach, we are seeking to develop dynamic error control to determine when to optimally switch from one approximation level to the

next level. We have been developing an approach to obtaining error bounds focused particularly on the case of separated eigenvalues. We are actively seeking to apply our experience to other areas of application. In particular, we have begun a collaboration with J. Muckerman and H. G. Yu at BNL in another area of computational chemistry, quantum molecular vibrational analysis.

In the area of cumulative reaction probabilities (CRP), we are developing parallel codes to carry out exact CRP calculations on problems of six to ten DOF or larger on massively parallel processors. A major cost of computation is in the solution of two Green's function linear systems. Current versions of the software use only diagonal preconditioning for the Green's functions. We plan to explore global preconditioners for this problem

- (year 1) to develop multilevel SPAM-based preconditioners, and
- (years 1-2) to develop Optimal Separable Basis methods.

Using the SPAM approach for linear systems, we will explore the use of multilevel preconditioning. There are several potential approaches to generating such preconditioners. For example, since the CRP method is based upon the use of a spectral method – DVR (discrete variable representation), we will explore the use of multiple grids of points for generating the approximation. Another approach to developing a preconditioner is based upon the Optimal Separable Basis (OSB) method developed by Poirier. We will work collaboratively with his group to design a general approach; we will also explore the properties and applicability of the use of the basis for the CRP, as well as investigate potential applicability to other areas. We plan to utilize our software for sequential banded and parallel diagonal preconditioning to test preconditioning approaches.

2.2.3 Automatic Generation of Discretizations and Error Estimators

We have produced full prototype PDE simulations using the the next generation of our software framework, PETSc-3, for both the Poisson and Bratu problems. These incorporate both *a posteriori* error estimation and adaptive mesh refinement based upon these estimates. However, there are still several pieces missing from the framework. We will generate code expression graphs, which currently are just interpreted (generally interpreted languages are too slow to be in the inner kernels of numerical computations). The discretization support will be extended to handle mixed discretizations and multicomponent fields. More advanced discretizations, such as Discontinuous Galerkin, will be added. We will also explore formal optimization methods for adaptive mesh refinement using error estimates in the cost functional, which will augment our current *error-balancing* approach. Our error estimators are currently time-independent. They will be extended to the time-dependent case, building on work in this area for sensitivity analysis. We will also need to formulate the dual problem in the hyperbolic case in order to handle simulations of MHD.

We are also involved in several significant collaborations that help to direct and enrich this research. The MCS Computational Biology Group, the Columbia University Geodynamics Group under Marc Spiegelman, and the MCS Optimization Group are all current users of the prototype PETSc-3 framework. Our contributions to these individual efforts are outlined below. Specific tasks from this work include:

- (year 1) code generations for expression graphs,
- (years 1-2) mixed discretization support,
- (year 3) advanced discretization support, e.g., discontinuous Galerkin,
- (years 2-3) formal optimization methods for mesh adaptation, and

- (years 1-3) error estimators for time-dependent and hyperbolic problems.

We have already developed a prototype SIDL component, in collaboration with Dmitry Karpeyev of the MCS Computational Biology Group, which represents the metabolic network as a specialization of a prototype mesh component now used in PDE simulations [?]. The symbolic expression components [?] allow the corresponding system of ODEs to be represented as an object hierarchy. This allows direct interaction with automatic differentiation tools [17], and eventually generation of optimal evaluation code [20]. This will be integrated with a multi-parameter continuation tool being developed by the MCS Computational Biology Group in order to produce a complete solution surface in parameter space. Specific tasks include:

- (year 1) generating a systems of ODEs directly from metabolic data and
- (years 2-3) incorporating multi-parameter continuation into the solution process.

The expression components also provide a gateway for interaction with the parallel optimization components provided by the Toolkit for Advanced Optimization (TAO) [?, 8] and Veltisto [11]. In the upcoming version of TAO, the expression components that are used to represent PDEs in PETSc-3 will be extended to represent variational inequalities, which can then be automatically solved by TAO in direct analogy with the current PDE case. Furthermore, since PDEs are already representable in the same system, we will immediately be able to leverage the Veltisto technology for PDE constrained optimization. In addition, the symbolic representation of VIs will facilitate complex categorization operations, such as the identification of convex problems. Specific tasks include:

- (years 2-3) extending the expression component to represent Variational Inequalities and
- (years 2-3) extending PETSc-3 to handle PDE-constrained problems.

We have been successful in solving simple two-dimensional geophysical subduction problems at moderate scales. However, some of the key constraints on subduction zone volcanism are inherently three-dimensional, and thus we must develop more scalable solver technology for 3D problems. Any multigrid preconditioner must handle the strongly variable viscosity field and the singular behavior of the flow near the interface of the two plates. Ultimately, we will need to couple the full 3D flow simulation with models of melting and reactive porous flow of up-welling magma. Using an expression representation of the discretization will allow autogeneration of interpolation and restriction matrices on unstructured and semistructured grids. Moreover, automatic differentiation of the expressions will allow easy evaluation of forms such as the second invariant tensor, even in the presence of a complicated viscosity function.

In the subduction simulation, an introduction of finite elements will allow good error estimates, especially in the troublesome corner region of the problem domain, where adaption will be important. Furthermore, since we can assess the error in an arbitrary functional, adaptivity can also be tied to large gradients in the viscosity, thereby alleviating problems in preconditioning the linear systems. Specific tasks include

- (years 1-2) use error estimation to adapt mesh in order to resolve large viscosity variation, and
- (year 2) use expressions to represent the second invariant tensor.

2.3 Impact of Proposed Work

The proposed work in PDE algorithms and software will:

- Enable accurate density-functional-based tight-binding (DFTB) electronic structures simulations for up to 100,000 atoms (currently 1000 atoms is considered large).
- Provide a one-stop shop (SLEPc) for advanced mathematical methods for eigenvalue problems for sparse matrices arising from PDE simulations.
- Allow application users to easily select both MG-Newton and Newton-MG methods within the same application codes.
- Provide a flexible framework for discretizing and computing error estimators from finite element simulators with a solid mathematical basis.
- Enable explanation of the large-scale patterns in volcanism (for example, the distances between volcanoes along faults) by solving the fully implicit Navier-Stokes with strongly variable viscosity in three dimensions and coupling with the geochemistry.
- Enable the solution of large-scale “exact” cumulative reaction probabilities problems to validate the accuracy of reaction rate coefficients.

3 Publications

The work funded under this project has generated 7 journal papers, 1 book chapter, 10 proceedings, and 3 technical reports. The software has been extensively downloaded. In addition, there have been 9 invited presentations, 9 contributed presentations, and 8 tutorials.

3.1 Journal Publications

The Subspace Projected Approximate Matrix (SPAM) Modification of the Davidson Method, R. Shepard, A. F. Wagner, J. L. Tilson, and M. Minkoff, *Journal of Computational Physics*, 172 (2) (2001) pp. 472–514.

High Performance Parallel Implicit CFD, W. D. Gropp, D. K. Kaushik, D. E. Keyes, and B. F. Smith, *Journal of Parallel Computing*, 27 (2001) pp. 337–362.

Parallel Components for PDEs and Optimization: Some Issues and Experiences, B. Norris, S. Balay, S. Benson, L. Freitag, P. Hovland, L. C. McInnes, and B. Smith, *Parallel Computing*, 28 (12) (2002) pp. 1811–1831.

Parallel Simulation of Compressible Flow Using Automatic Differentiation and PETSc, P. D. Hovland and L. C. McInnes, *Parallel Computing*, 27 (2001) pp. 503–519.

A Case Study in the Performance and Scalability of Optimization Software, S. Benson, L. C. McInnes, and J. J. Moré, *ACM Transactions on Mathematical Software*, 27 (3) (2001) pp. 361–376.

Globalized Newton-Krylov-Schwarz Algorithms and Software for Parallel Implicit CFD, W. D. Gropp, D. E. Keyes, L. C. McInnes, and M. D. Tidriri, *Int. Journal on High Performance Computing Applications*, 14 (2000) pp. 102–136.

Interpolating moving least squares methods for fitting potential energy surfaces: Detailed analysis of one-dimensional applications, G. G. Maisuradze, D. L. Thompson, A. F. Wagner, and M. Minkoff, to appear in the *J. of Chemical Physics* (also Argonne National Laboratory preprint ANL/MCS-P1083-0803).

3.2 Book Chapters

Software for the Scalable Solution of PDEs, S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith, chapter 21, pp. 621–647, *Sourcebook of Parallel Computing*, Morgan Kaufmann, 2003.

3.3 Proceedings

The Role of Multi-Method Linear Solvers in PDE-based Simulations, S. Bhowmick, L. C. McInnes, B. Norris, and P. Raghavan, *Proceedings of the 2003 International Conference on Computational Science and Its Applications, ICCSA 2003*, Montreal, Canada May 18–21, 2003. *Lecture Notes in Computer Science 2677*, editors V. Kumar, M. L. Gavrilova, C. J. K. Tan, and P. L’Ecuyer, pp. 828–839, 2003.

Adaptive Sparse Linear Solvers for Implicit CFD Using Newton-Krylov Algorithms, L. C. McInnes, B. Norris, S. Bhowmick, and P. Raghavan. *Proceedings of the Second MIT Conference on Computational Fluid and Solid Mechanics*, Massachusetts Institute of Technology, Boston, USA, June 17-20, 2003.

Integrating Automatic Differentiation with Object-Oriented Toolkits for High-Performance Scientific Computing, J. Abate, S. Benson, L. Grignon, P. Hovland, L. C. McInnes, and B. Norris. In George Corliss, Christèle Faure, Andreas Griewank, Laurent Hascoët, and Uwe Naumann, editors, *Automatic Differentiation of Algorithms: From Simulation to Optimization*, Springer, New York, 2002. ANL/MCS-P820-0500.

Using Automatic Differentiation for Second-order Matrix-free Methods in PDE-constrained Optimization, D. E. Keyes, P. D. Hovland, L. C. McInnes, and W. Samyono. In George Corliss, Christèle Faure, Andreas Griewank, Laurent Hascoët, and Uwe Naumann, editors, *Automatic Differentiation of Algorithms: From Simulation to Optimization*. Springer, New York, 2002.

Challenges and Opportunities in using Automatic Differentiation with Object-Oriented Toolkits for Scientific Computing, P. Hovland, S. Lee, L. C. McInnes, B. Norris, and B. Smith, Proceedings of the Workshop on PDE-Constrained Optimization, Santa Fe, 2001.

Making Automatic Differentiation Truly Automatic: Coupling PETSc with ADIC, P. Hovland, B. Norris, and B. Smith, Proceedings of ICCS2002, Amsterdam, 2002.

Latency, Bandwidth, and Concurrent Issue Limitations in High-Performance CFD, W. D. Gropp, D. K. Kaushik, D. E. Keyes, and B. F. Smith, Proceedings of the First M.I.T. Conference on Computational Fluid and Solid Mechanics, Cambridge, MA, June, 2001.

PETSc and Overture: Lessons Learned Developing an Interface between Components, K. R. Buschelman, W. D. Gropp, L. C. McInnes, and B. F. Smith, Proceedings of the International Federation for Information Processing Working Conference on Software Architectures for Scientific Computing, Kluwer, 2000, pp. 57-68.

Understanding the Parallel Scalability of An Implicit Unstructured Mesh CFD Code, W. D. Gropp, D. K. Kaushik, D. E. Keyes, and B. F. Smith, Proceedings of the 7th International Conference on High Performance Computing (HiPC '2000), Bangalore, India, December, 2000, pp. 395-404.

Performance Modeling and Tuning of an Unstructured Mesh CFD Application, W. D. Gropp, D. K. Kaushik, D. E. Keyes, and B. F. Smith, Proceedings of SC2000, IEEE Computer Society, 2000.

3.4 Technical Reports

Eigenvalue Computation from the Optimization Perspective: On Jacobi-Davidson, IIGD, RQI and Newton Updates, Y. Zhou, submitted to Numerical Linear Algebra with Applications, Argonne National Laboratory preprint ANL/MCS-P1074-0803.

Faster PDE-based Simulations Using Robust Composite Linear Solvers, S. Bhowmick, P. Raghavan, L. C. McInnes, and B. Norris, Argonne National Laboratory preprint ANL/MCS-P993-0902, 2002, to appear in Future Generation Computer Systems.

PETSc 2.0 Users Manual, S. Balay, K. Buschelman, W. D. Gropp, D. Kaushik, M. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang, MCS Technical Report, ANL-95-11, Revision 2.1.6, August 2003.

3.5 Software Projects

As part of our approach to develop mathematical algorithms for use in large-scale PDE-based applications, we make available high-quality implementations within the PETSc toolkit, which

is freely available via <http://www.mcs.anl.gov/petsc>. We made seven public software releases during the duration of this grant (Sept. 2000, Apr. 2001, Dec. 2001, Apr. 2002, May 2002, Jan. 2003, Aug. 2003). PETSc has several hundred users, including those at other DOE laboratories (LANL, LBNL, LLNL, ORNL, PNNL, and PPPL), Boeing, NASA, industry, and universities. The software is downloaded over 250 times per month and generates on average 4 email requests, reports, and communications per day. There is also an email list to which over three hundred PETSc users have subscribed. A partial list of publications arising from the use of our software is available via [14]. We are aware of more than fifty publications that have appeared during the years 2000-2003, in a broad range of application domains, including aerodynamics, acoustics, biology, biomechanics, econometrics, electromagnetics, fusion, materials science, multiphase flow, nanotechnology, optimization, reactive transport, and numerical algorithm analysis.

3.6 Invited Presentations

Issues in Newton-Krylov Methods for Large-Scale Scientific Simulations, L. C. McInnes, invited minisymposium presentation, SIAM Conference on Applied Linear Algebra, Williamsburg, VA, July 15-19, 2003.

Issues in High-Performance Numerical Library Design, L. C. McInnes, invited seminar, NASA Goddard Space Flight Center, Greenbelt, MD, June 20, 2003.

High-Performance Numerical Components, L. C. McInnes, invited seminar, National Institute of Standards and Technology, Gaithersburg, MD, November 5, 2002.

High-Performance Components for PDEs and Optimization, L. C. McInnes, invited colloquium, Computer Science and Engineering Department, Penn State University, College Park, PA, October 17, 2002.

Tradeoffs in High-Performance Numerical Library Design, L. C. McInnes, invited speaker at the Conference on High Speed Computing, Salishan Lodge, Gleneden Beach, Oregon, April 22-25, 2002.

Experiences with Parallel Numerical Software Interoperability, W. D. Gropp (speaker) and L. C. McInnes, invited presentation at the First CASC/ISCR Workshop on Object-Oriented and Component Technology for Scientific Computing, Livermore, CA, July 2001.

Multigrid for Constrained Optimization, B. Smith, invited presentation at the Fred Howes Memorial Conference, March 2001.

Parallel Software for Large-Scale Optimization, S. Benson, L. C. McInnes (speaker), and J. J. Moré, invited minisymposium presentation, SIAM CSE Conference, Washington, D.C., Sept. 22-25, 2000.

Matrix-free Newton-Krylov Methods Using a Hybrid Automatic Differentiation-Finite Difference Strategy, P. Hovland and L. C. McInnes (speaker), invited presentation at the Workshop on Solution Methods for Large-Scale Nonlinear Problems, Center for Applied Scientific Computing, Livermore, CA, July 26, 2000.

3.7 Other Presentations

PETSc Tutorials:

Workshop on the ACTS Toolkit at NERSC, August 2003, W. Gropp.

15th Annual Domain Decomposition Meeting, July 17-18, 2003 at Freie Universität Berlin (FUB), W. Gropp.

Workshop on the ACTS Toolkit at NERSC, September 5th, 2002, B. Smith.

3-day tutorial as part of the Parallel Computing Workshop, Center for Computational Science and Engineering, Peking University, Beijing, China, July 1-August 2, 2002, W. Gropp.

Workshop on the ACTS Toolkit at NERSC, October 12, 2001, L. C. McInnes and S. Balay.

Use of Object Oriented Solvers, seminar in the Illinois Institute of Technology Computer Science Department, April 2001, B. Smith.

Nonlinear Solvers and PETSc, seminar at the Purdue University Computer Science Department, March 2001, B. Smith.

2-day PETSc tutorial on the Access Grid, October 17-18, 2000, B. Smith and S. Balay.

2-day tutorial on PETSc, including its support for domain decomposition and multigrid, October 7-8, 2000, in Lyon, France, W. Gropp.

Workshop on the ACTS Toolkit, at NERSC, September 29, 2000, L.C. McInnes and S. Balay.

Contributed Presentations:

The Role of Multi-Method Linear Solvers in PDE-based Simulations, S. Bhowmick (speaker), L. C. McInnes, B. Norris, and P. Raghavan, International Conference on Computational Science and Its Applications, ICCSA 2003, Montreal, Canada May 18–21, 2003.

Adaptive Sparse Linear Solvers for Implicit CFD Using Newton-Krylov Algorithms, L. C. McInnes, B. Norris (speaker), S. Bhowmick, and P. Raghavan. Second MIT Conference on Computational Fluid and Solid Mechanics, Cambridge, USA, June 17-20, 2003.

Single Precision Incomplete LU Factorization for Incompressible Fluid Flow Applications on Pentium III Processors in PETSc, K. Buschelman (presenter), W. Gropp, and B. Smith, poster presented at the 2001 International Conference On Preconditioning Techniques For Large Sparse Matrix Problems In Industrial Applications, 2001.

Latency, Bandwidth, and Concurrent Issue Limitations in High Performance CFD, D. Kaushik (speaker), D. Keyes, W. Gropp, and B. Smith, First MIT Conference on Computational Fluid and Solid Mechanics, Cambridge, USA, June 14, 2001.

Performance Monitoring and Steering of PETSc-FUN3D Using the Asynchronous Memory Snooper (AMS), D. Keyes, W. Gropp (speaker), D. Kaushik, I. Ba, and B. Smith, Tenth SIAM Conference on Parallel Processing for Scientific Computing, Portsmouth, USA, March 13, 2001.

Algorithms and Data Structures for High Performance in PETSc, D. Kaushik, D. Keyes, W. Gropp, B. Smith, and K. Buschelman (speaker), Tenth SIAM Conference on Parallel Processing for Scientific Computing, Portsmouth, USA, March 12, 2001.

Understanding the Performance of a Hybrid (Distributed/Shared Memory) Programming Model, D. Kaushik (speaker), D. Keyes, W. Gropp, and B. Smith, 7th International Conference on High Performance Computing (HiPC '2000), Bangalore, India, December 20, 2000.

Performance Modeling and Tuning of an Unstructured Mesh CFD Application, D. Kaushik, D. Keyes, W. Gropp (speaker), and B. Smith, SC2000, Dallas, USA, November 8, 2000.

Achieving High Sustained Performance in an Unstructured Mesh CFD Application, D. Kaushik (speaker), D. Keyes, W. Gropp, and B. Smith, ACTS Toolkit Workshop 2000, National Energy Research Scientific Computing Center (NERSC), Berkeley, USA, September 29, 2000.

Chapter II

Modeling and Applications

4 Previous Work

4.1 Executive Summary of Previous Work

An important part of our research in PDEs is the formulation and analysis of mathematical models of physical systems. Our recent focus has been on four areas in the dynamics of micromagnetics: (1) pinning and depinning, (2) magnetic spin, (3) micromagnetic particle interaction, and (4) conservative integration schemes. Uniting all of these projects is the development of techniques (including mathematical) that apply broadly to a variety of applications.

To test the usability and robustness of these techniques, we carry out much of our modeling work in collaboration with application scientists who are solving large-scale problems. These collaborations often lead to new research directions (see, e.g., Section 4.3) in scalable numerical algorithms and software and to the use of state-of-the-art numerical and parallel algorithms in those applications.

4.2 Summary

4.2.1 Dynamics of Micromagnetics

We focus on the study of the dynamics of complex systems. The systems arise primarily from magnetic materials in the nanoscale range. A thorough study of transient phenomena is essential in understanding the fundamental physics and its consequent technological developments. Each of our investigations encompasses modeling, algorithm development, simulations, and analysis. We will briefly describe the results of four studies completed in the past three years.

Pinning and Depinning

In collaboration with the Magnetic Films Group in Argonne's Materials Science Division, we have undertaken a comprehensive investigation of microstructural effects on the performance of complex magnetic systems at high temperatures and over long periods of time. Specifically, we considered the process of domain wall movement in disordered composite magnetic materials. The details of pinning and depinning are basic in the design of storage devices at all scales. Pinning is also of theoretical interest as an example of a critical phenomenon associated with glassy/creep dynamics.

We modeled the domain wall as an elastic membrane in the presence of a randomly distributed pinning potential while being driven by a sub-critical force. Earlier studies considered the dynamics induced by supercritical force. These studies were limited in the sense that they could not follow the pinning process in its later stages. Thus, for example, they were limited in estimating pinning rates and the nature (topology) of the pinned state. However, sub-critical forces imply creep dynamics characterized by extremely low activity as the critical force is approached from below. In particular, ergodicity is not applicable, necessitating the use of statistical sampling and serious computations.

In this study, [?], we successfully identified three dynamical regimes, as well as an estimate for the critical exponent for the decay of the activity.

In addition, we considered the dynamics of superconducting vortices in type-II materials. A continuum description of the equilibrium state of a superconductor can be based on a complex-valued order parameter and a real-valued vector potential determined by the critical points of an energy potential. Thus, in principle, a state can be determined by minimizing the energy. However, there are many minima, not all of which are physical. Therefore, we explored an alternative approach that follows a dynamical system to equilibrium. The system is modeled as an over-damped system driven by the generalized force generated by the functional gradient of the energy. The functional is the Ginzburg-Landau energy, and the resulting dynamics is governed by the time-dependent Ginzburg-Landau (TDGL) equations.

The advantage of this approach is that the resulting state is physical. In superconductivity, the solutions of the GL equations that are of interest are the vortex solutions. These are singular solutions characterized by a phase change of 2π in the order parameter in any closed circuit surrounding a vortex. Thus, the vortex state is the result of a phase transition, and any dynamics describing the evolution toward this state will be characterized by very slow approach to equilibrium—the so-called critical slowing down associated with phase transitions.

Moreover, the use of dynamics has another significant advantage over a static description: It enables the study of vortex pinning/depinning in the presence of external currents. It is the pinning of vortices in a disordered material that is the basis of *practical* superconductivity.

Thus, because of the intrinsic slowing-down aspect of the dynamics, it is imperative that algorithms designed for the TDGL equations strive to reach the equilibrium state in the shortest (CPU) time. Of course, any numerical procedure must preserve the gauge-invariance of the continuum equations. In the course of our studies of vortex dynamics for type-II superconductors, we have developed several algorithms for integrating the TDGL equations in two and three dimensions. These were compared in terms of the cost in CPU time to achieve equilibrium.

In general terms, the TDGL equations can be described as a nonlinear system of complex-valued PDEs, second order in space and first order in time. (The real vector potential was described in terms of a unit complex variable, the so-called link variable, in order to maintain gauge invariance in the numerical approximations. Spatial discretizations were based on staggered finite differences. Domain decomposition was used as a means of introducing parallelism. The four algorithms considered were based on the extent of implicitness used in the solution procedure, They ranged from fully explicit and semi-implicit to linearly implicit and nonlinearly implicit. The semi-implicit version arose by observing that the spatial variation of the vector potential is driven by a $\nabla \times \nabla$ operator, which could be approximately factored in a preprocessing step and used in an ADI procedure of the Douglas-Gunn type. The linearly implicit version also treated the spatial operator for the order parameter in an implicit manner. In this case, however, although the gradient operator was directly coupled to the vector potential, use of link variables permitted the use of operator splitting at each time step.

Before describing the nonlinearly implicit version, we note that the nonlinearities in the TDGL equations arise from several sources. First, the vector potential is coupled to the order parameter via the superconducting current density that is an additive term. Because of its structure, no attempt is made to force implicitness here. The equation for the order parameter was coupled to the potential

via the spatial operator, this was treated as mentioned above. Finally, the order parameter equation has a local nonlinearity from the condensation term, which is responsible for the instability that initiates the phase transition to the vortex state. Unless this condensation nonlinearity is treated, a fully implicit version could not be achieved in the sense that there would always exist a stability limit for the time step. We addressed this situation by observing that for type-II superconductors, we can solve the local condensation dynamics analytically, with higher-order approximations in other cases. This analytic expression is then incorporated into the time-stepping algorithm. In two dimensions, these algorithms were compared in detail, assessing stability, accuracy, storage, scalability with number of processors, and wall clock time, [?]. Among the conclusions, we note the following. The wall-clock time for the fully nonlinear version was one-sixth that of the explicit version; moreover, it showed nearly linear speedup. Another observation of interest is the following. Although the dynamics of each version evolve to the same vortex state, the path, in phase space, to this state will be different. However, the (physical) time to reach this equilibrium state is essentially the same for the four versions. This may indicate that the discrete dynamics for any one of these versions will approximate the transient path implied by the continuum equations. This is of importance in simulating pinning/depinning behavior where the transient behavior is of interest.

Magnetic Spin

In another application of dynamical systems in materials science, we considered the effects of microstructures (hard/soft grains) and temperature on the performance of nanoscale magnetic particles. Like the first project, this effort involves theory, analysis, experiments, and simulations; also like the first project, it combines the experimental and theoretical expertise of the Magnetic Films Group at Argonne with our extensive computational expertise in such related areas as vortex and fluid dynamics.

In micromagnetic calculations, the dynamics of the magnetic spins are described by the Landau-Lifshitz-Gilbert equations, which are essentially the equations of motion for an overdamped spinning top subject to a driving force (thus, inertial effects are neglected). The driving force is the local effective magnetic field. This field is derived as a functional gradient of an energy density. The energies considered include exchange, anisotropy, demagnetization, and the external energy. A fundamental constraint on the dynamics of the spins is that the magnetization (spin length) remain constant at all times.

We developed a time integration scheme that maintains this constancy at every time step. This is an intrinsic part of the algorithm, as opposed to a procedure that would require a projection onto the sphere at each step. The essence of the process is as follows.

At each time step and each spatial location, the spin vector is expressed in a local orthogonal basis/coordinate system determined by the local effective field. In this basis, and over the next time interval, the spin motion is decomposed into a composite motion consisting of a damping motion toward the local effective field, and a precessional motion around this local field. The damping motion is nonlinear but can be solved analytically under the assumption of a constant effective field over a single timestep interval. The precessional motion is linear and is solved analytically. Thus, the only approximation is isolated to the assumption that the local effective field is constant in time over the time step. Since the algorithm changes only the spin direction, the spin magnitude is unchanged. We used this time integration scheme as the basis for one-, two-, and three-dimensional

micromagnetic descriptions of spin dynamics in nanoparticle systems.

We also used large-scale simulations to investigate reversal processes for so-called spring magnets, [?]. Spring-type permanent magnets are composite materials with nanoscale soft and hard magnetic phases exchange-coupled at the interfaces. Their performance potentially exceeds that of the strongest known magnetic material by at least a factor of two and would greatly enhance the energy efficiency of any device based on permanent magnets.

We modeled these magnets as a bilayer of soft and hard magnetic material in one dimension. The hysteresis loops were compared with several experiments and, in addition, provided explanations for some experimental observations that were inherently limited to surface measurements, [?].

Micromagnetic Particle Interactions

Current processing techniques have made it possible to fabricate nanoscale magnetic particles while maintaining control of their shape and thickness. Because of their size, these particles are of single domain type; thus, their magnetic properties differ considerably from those of larger particles. Controlling these properties is of theoretical as well as technological interest. For example, the size of the read head of a storage device is finally limited by spin-wave noise. This noise is the manifestation of excited magnetic normal modes determined by the shape and material properties of the particle. Hence, determining the structure of these modes is of basic importance.

Unfortunately, experimental techniques are limited in resolution as well as being confined to measurements of indirect surface effects. At the same time, analytical techniques are limited to extreme parameter ranges such as those, for example, where exchange coupling is the dominant effect. Thus, in order to study the effects of all types of micromagnetic interactions, there is need for numerical simulation techniques capable of resolving the normal modes. Each mode is characterized by a spatial distribution of magnetization and the precession frequency, and represents an elementary form of undamped motion.

The task of achieving numerical normal mode resolution capability has two requirements. First, a three-dimensional transient simulation of the spin dynamics is needed. Two aspects are paramount here: (1) the dynamics must be followed over a long period of time in order to obtain the necessary frequency resolution of each normal mode; and (2) the dynamics must be essentially undamped in order to distinguish the weakly excited modes. Since our time integrator decomposes the motion into independent precession and damping components, the latter can be easily turned off, and the resulting scheme naturally forms a basis for simulation of long-time undamped spin dynamics.

The second requirement of the simulation is that the nanoparticles be adequately resolved spatially. Computationally this is a severe requirement, especially in three dimensions, since the magnetic state dynamics is inherently nonlocal in character because of the presence of the demagnetizing field: the time-derivative of the magnetization state at any point directly depends on the state at any other point of the entire sample. The demagnetizing field arises through Maxwell's equations for the field induced by a distribution of magnetic spins, and the computation of this field consumes at least 80 percent of the CPU time in any micromagnetic dynamics calculation and has to be done at every time step. Thus, an efficient solution for this field is a major design consideration for spin dynamics. The first consideration is whether to use a Green's function formulation, which directly expresses the field as a convolution over the material sample, or a potential formulation, in which

the field is derived from a scalar potential function, satisfying a PDE over the entire physical space. We used the potential approach as it lends itself to a variety of numerical PDE solution techniques, for which extensive software tools, such as PETSc, and expertise have been developed at MCS over the past decade. In addition, the potential approach derives directly from Maxwell’s equations and is intrinsically more accurate than the integral approach, which typically uses a dipole approximation to the Green’s function kernel regardless of the distances between points. With the potential formulation, two numerical solution approaches are possible. The viability of each depends on the number of computational degrees of freedom necessary for adequate approximation to the field. Since the demagnetizing field is defined everywhere, one approach is to solve a Poisson equation over a large computational domain encompassing the particle and its surroundings, subject to an interface condition at the particle boundary arising from a singular change in magnetization there. Since the computational domain approximates all of the physical space, its size and hence the number of computational degrees of freedom will depend on the rate of decay of the field, which must vanish at the boundary of the computational domain, representing the infinitely remote part of space. In 2D the rate of decay is slow (logarithmic), while in 3D it may be fast enough for the potential approach to be computationally viable. In the problem described above, the amount of memory required, M , is modest, scaling linearly with N , the measure of the size of the problem, which is related to the size of the computational domain, or the spatial resolution accuracy: $M = O(N)$. The CPU time requirement T , on the other hand, is directly related to the computational complexity of the PDE solver used and constitutes the main limiting computational factor. Assuming that performance comparable to that of the fast Fourier transform (FFT) can be achieved, the scaling of T with the problem size is $T = O(N * \log(N))$.

An alternative to solving one PDE on a large domain is to recast the problem as a pair of Poisson problems defined only on the particle, but coupled through an integral over the boundary of the particle. The estimates of computational requirements for the solution of the two PDEs are similar to those of a single PDE, with $n \ll N$ measuring the smaller problems size, instead of N : $M(PDE) = O(n)$, $T(PDE) = O(n * \log(n))$. The memory required to store the discretized boundary integral kernel, $M(BI)$, scales as $O(k^2)$, where $k = n^{(d-1)/d}$ measures the number of degrees of freedom on the particle boundary depending on the dimension, d . Application of the discretized kernel to the boundary data of size k results in a time requirement of $T(BI) = O(k^2)$. The share of the boundary integral computation in the complexity of the total problem depends on the dimension parameter d . In 2D, $k = n^{1/2}$, so that $M(BI) = O(n) = M(PDE)$, and $T(BI) = O(n) < O(n * \log(n)) = T(PDE)$, so that the boundary integral computation is negligible compared to the PDE computation. In 3D ($d = 3$), $k = n^{2/3}$, so that $M(BI) = O(n^{4/3}) > M(PDE)$, and $T(BI) = O(n^{4/3}) > O(n * \log(n)) = T(PDE)$, and the boundary integral computation dominates both the memory and time complexity of the problem, reflecting in part the complexity of the convolution-based integral formulation, albeit in a domain of lower dimension—the particle boundary.

The boundary integral memory requirement, $M(BI)$, is the hard limiting factor determining the bounds on the size of simulations that can be attempted. Therefore, it is natural to try to trade off memory complexity for time complexity. This can be done by utilizing multipole expansion of the boundary integral kernel. Ordinarily the multipole method is employed to reduce the time spent evaluating integrals with potential-type kernels by using a hierarchical decomposition of the space containing the charges. However, in contrast to the impressive speedup that the multipole

method delivers in 2D, in three dimensions the benefits of using the multipole method are much less dramatic. The explanation has to do with significant computational complexity associated with use of spherical harmonics expansions and their shifting and lumping.

Another complication, also present in the 2D version, comes from the necessity to manage hierarchical data structures such as oct-trees. Instead of using the complete version of the multipole algorithm, we partition the boundary of the particle into fixed subdomains of computational points that are too close for the remote multipole expansions (in terms of inverse powers of the distance between the points) to be valid. Direct interactions between the points within a subdomain are computed the same way as in the full discretization of the kernel. Subdomains now interact as single entities: potential due to a given “charge” subdomain evaluated at the center of a “field” subdomain using the remote multipole expansion if the subdomains are well separated, and direct computation otherwise. The potential at the center of a subdomain is then “distributed” to the individual points using local power series expansions. The coefficients of the remote expansions are evaluated each time using precomputed coefficients of Legendre polynomials and their derivatives of fixed order determined by prescribed accuracy. With this approach only a small number of Legendre coefficients need to be stored along with the direct interaction coefficients, whose number is controlled by the prescribed separation distance between subdomains. This achieves memory savings at the cost of more expensive potential computations between remote subdomains, mitigated by the “lumping” of charges and relatively fast evaluation of remote expansions.

As currently implemented, the geometry of particles is rectangular with a uniform discretizing collocation mesh, which enables us to use FFT-based solvers for all Poisson problems involved. We have developed and implemented versions of the simulation code based on the various treatments of the demagnetization discussed above, including discretization of the convolution integral, a single PDE over a large ambient space (2D only), and the “hybrid” approach of two Poisson’s problems related by a boundary integral, which can be discretized directly or using multipole expansions, [?]. Recall that there are two tasks involved in determining the structure of magnetic normal modes. The first is the development of long-time undamped dynamics as discussed above. The second is the analysis of the time series for the magnetic spins. The object is to identify and describe the lower frequency modes. These are most physically relevant. The procedure is based on identifying the spin spectra and then obtaining the spatial distribution of common modes. Employing the 3D simulations and analysis described above, we investigated the structure of some low frequency magnetic modes for a thin rectangular polycrystalline iron particle. Up to this point only the bulk mode had been identified experimentally. Various analytical approximations to this bulk mode had been made (neglecting finite size effects) in the either the limit of exchange or field dominated cases. We were able to identify this bulk mode for finite size particles, accounting for both exchange and demagnetization. In addition, the frequency/applied-field dependence was mapped. We also detected and described the spatial structure of an “end” mode. This mode, being the lowest frequency mode we detected, is expected to be particularly relevant in the study of the mechanisms of magnetic reversal in nanoparticles.

Conservative Integration Schemes

In numerical dynamical simulations an important question is whether the discrete integration scheme preserves conservation properties of the simulated continuous system. Conservation of

magnetic spin length by the Landau-Lifshitz-Gilbert (LLG) equations is an example of conservation properties preserved in the discrete LLG system. Other examples of conservation properties include conservation of energy in the undamped LLG and preservation of spatial momentum in the isotropic case. The undamped LLG is a Hamiltonian system with respect to the vector product Poisson bracket, and by the well-known theorem of Hamiltonian dynamics, the Poisson bracket tensor is conserved along the orbits of the system. The conservation of the Poisson bracket implies energy conservation of the continuous system as well as other conservation properties characterizing Hamiltonian systems. Therefore, it is natural to identify discrete analogs of Poisson bracket preservation and connection to discrete conservation laws, whose preservation is important in studies of undamped LLG, such as the normal mode analysis.

In development of conservative integrators we have focused on preservation of the symplectic form, the dual of the Poisson bracket, and its PDE analogs. The theory of discrete symplectic integration provides compelling evidence that integrators preserving the symplectic form possess superior conservation properties as well as reflect the Hamiltonian character of dynamics, such as the incompressibility of the phase flow [?]. In the infinite-dimensional PDE setting there is frequently an alternative structure called the multisymplectic form that better reflects the local nature of the problem and the equal treatment of space and time by allowing to recast the system in a covariant form with a new covariant Hamiltonian. Local conservation laws naturally follow from the multisymplectic formulation. The multisymplectic form of the system is naturally derived from a variational principle applied to the associated Lagrangian function, when it exists.

We have developed methods for derivation of space-time discretizations of Hamiltonian PDEs that admit a Lagrangian formulation. These discretizations possess a natural analog of the multisymplectic form and its conservation property, and exhibit superior preservation of conservation laws as compared to standard discretizations. The methodology has been applied to the 1D nonlinear Schrodinger equation [?], which is related to the undamped LLG in the anisotropic case. In an attempt to extend this theory to the LLG and seek a Lagrangian form of the equation, we have discovered a topological obstruction to the existence of a globally defined Lagrangian density deriving from the fact that the LLG field takes values in a topologically nontrivial manifold – the sphere. However, since the multisymplectic form has purely local character, as well as the variational principle itself, we were able to derive a local Lagrangian density for a reduced version of LLG as well as a local multisymplectic form of the system along with local conservation laws. This lays the foundation for development of the theory of discretization of such locally defined multisymplectic PDEs as well as its extensions to nonlocal and damped systems of LLG type.

4.3 Highlights

- We developed a new model of the domain wall in disordered composite magnetic materials that enabled us to study the pinning process in later stages than previously possible.
- Our use of dynamics enabled the study of vortex pinning and depinning in the presence of external currents.
- We provided explanations for experimental observations of magnetic reversal.
- We identified the bulk mode for finite-sized particles, accounting for both exchange and demagnetization. This was the first such identification outside of experiment.
- We derived a local Lagrangian density for a reduced version of the Landau-Lifshitz-Gilbert

equations, thus laying the foundation for development of the theory of discretization of locally defined multisymplectic PDEs.

4.4 Program Personnel

Staffing levels are in FTEs; FY03 is the current level and FY04 is the level requested. Dr. Kaper is on leave at the National Science Foundation (FY02-03). A candidate for the new hire in FY04 has been identified.

Principal Investigators

	FY03	FY04
Hans Kaper	(on leave)	
Gary Leaf	1	1
New Hire		1

Other Staff:

Visiting Faculty

- Juan Restrepo (University of Arizona)

Graduate Students

- Dmitry Karpeyev (Old Dominion University)

5 Proposed Work

5.1 Executive Summary

We propose to continue our work in the dynamics of micromagnetics at roughly the same level of effort and to expand our work in two new areas, which we have just recently begun to explore:

- evolution of structure in interacting (nano-size and larger) particles and
- analysis and simulation of dynamics in studies of the cell clock, in collaboration with the MCS Computational Biology Group.

We have done preliminary and related work in the latter two areas and thus have a solid foundation to make rapid progress with a small increment in our resources.

5.2 Outline of Proposed Work

5.2.1 Dynamics of Micromagnetics

When fabricated, nanosized particles in general are essentially oval cylinders in shape, which could be approximated as elliptic cylinders. The shape impacts the magnetic properties of the particle. This impact is particularly important in the study of reversal processes that are determined by the nature of the magnetic normal modes. To explore the influence of shape on the reversal mechanisms,

- (year 1) we propose to extend the current 3D micromagnetic dynamics capability to non-rectangular geometries.

For this work, we will maintain the present time integration scheme described earlier. Our primary reason is that this scheme has minimal dissipation, which enables long-time undamped dynamic simulations that are necessary for normal mode analysis. Maintaining this scheme will, however, affect the general formulation of the discrete model.

A natural approach for domains with curved boundaries is the use of isoparametric finite elements. The time integrator is a collocation scheme in that it tracks the dynamics of spin vectors at particular spatial locations. Thus, our use of finite elements is not in the context of a weak formulation of the dynamics. Rather, finite elements are used in the construction of an effective field that drives the spin dynamics. Recall that the dynamics is that of an inertia-less spinning top driven by a torque generated by a functional gradient of an energy. This torque is the effective field.

Two approaches present themselves for the use of finite elements in generating an approximation to the effective field. The first approach would derive the effective as the gradient of the energy in the continuum model. One then approximates this field by projecting into a finite element subspace. This approach has serious disadvantages. For example, since we follow the dynamics at specified spatial locations, the field has to be collocated. The exchange field, which is one of the components of the effective field, is generated by a second-order differential operator. Such a collocation would entail the use of higher-order elements with at least C^2 continuity and would thus require quintic Hermite-type elements. Aside from the general impracticality of using such elements in multiple dimensions, one would have to derive and track the dynamics for the derivative degrees of freedom intrinsic to Hermite basis.

We therefore prefer another approach based on first discretizing the energy functional on a finite element basis and then differentiating the resulting multivariate function in order to obtain an

effective field. This approach allows us to use Lagrange-type elements of lower order. We observe that with a Lagrange interpolatory basis, the degrees of freedom are function values (spins) whose dynamics can then be tracked with the time integrator. Also, the effective field, being an intensive variable, is the derivative of the energy density, which is the actual quantity that is approximated in the above procedure.

- (years 1-2) We plan to implement this approach.

Besides the issue of the overall formulation of the numerical model, other issues arise with this more general geometry. These include mesh generation and solution methods for the demagnetizing field. With elliptical cylinders, it is natural to mesh the base by means of triangulation. For this purpose,

- (year 1) we intend to use the Triangle [?, ?] package.

As noted earlier, the demagnetization field calculation consumes at least 80 percent of the CPU time in any simulation. Consequently it is imperative to use the most efficient solvers available. We will continue to use a formulation of this field as a gradient of a scalar potential rather than using an integral dipolar approximation. Because of the greater complexity, in the case of nonrectangular geometry, we will use a formulation based on a single large domain rather than the splitting combined with the use of a surface integral as in the current 3D version. With this approach, one must solve a Poisson equation defined over two regions subject to an interface condition. With this geometry, fast multigrid solvers will be used rather than the FFTs currently employed. This approach has been examined in the case of two dimensions, but with rectangular geometry.

To implement the computation of the demagnetization field efficiently and in a parallelizable manner, we will port the code to the PETSc framework. As discussed elsewhere in this document, PETSc represents a flexible yet comprehensive environment for the development of scalable parallel PDE codes. Using PETSc will allow us to employ its extensive collection of parallel linear and nonlinear solvers, as well as interfaces to mesh partitioning and manipulation tools, debugging and profiling capabilities. Using the Scientific Interface Definition Language (SIDL) [12, ?] interoperability environment will allow us to naturally incorporate legacy code, such as the spin-length preserving integrator, in a fully object-oriented manner, and enable stepwise refinement and tuning of the implementation.

In continuing the development of the theory and implementation of conservative schemes,

- (years 1-3) we intend to extend the technique of variational multisymplectic discretizations to the full LLG system.

In order to do this, several tasks have to be accomplished. First,

- (year 1) finite element bases have to be developed that are invariant under a change of coordinates on a sphere, with rational function being a prime candidate.

These bases will deliver accurate and stable discretizations of the highly nonlinear local LLG Lagrangian (action) functional. Second,

- (year 1) discretization of Lagrangians of fields with values in Lie groups and their reductions will be explored to obtain invariant multisymplectic schemes for LLG.

It is well known that the target space of LLG fields – the 2-sphere – is a homogeneous space of Lie group $SO(3)$. In particular, it is a coadjoint orbit of $SO(3)$ on the dual of its Lie algebra $so(3)$. Multivaluedness of the undamped LLG Lagrangian is rooted in the fact that $SO(3)$ is not simply

connected, and single-valuedness is expected to be restored by passing to the universal cover of $SO(3)$ —the three-dimensional spinor group $Spin(3) = SU(2)$. The LLG action is obtained from the $SU(2)$ Lagrangian by reduction to the factor space. In this we can rely on an extensive theory of numerical integration of ODEs with values in Lie groups and their homogeneous spaces. Third,

- (year 2) the formalism of multisymplectic PDEs must be extended to integrodifferential Hamiltonian systems, such as the undamped LLG with the global demagnetizing field.

Fourth,

- (year 3) LLG damping has a particular form of the double Poisson bracket, which can be cast as the natural analog of the Rayleigh dissipation function on the $so(3)$ Lie algebra.

Exploiting the variational form of this sort of a dissipative system, we will develop the dissipative extension to the conservative multisymplectic form and its discretizations, preserving the dissipation density and rates.

5.2.2 Evolution of Structure in Interacting Particles

The simulation of molecular self-assembly seems well suited to the introduction of fictitious domain (FD) techniques. The complicated evolution of the interior domain geometry makes moving mesh methods impractical. However, we need not make all the traditional trade offs at once. The FD method is an excellent way to avoid moving the main domain mesh, although it does not necessarily follow that this mesh must be structured. It is possible that a well-adapted unstructured background mesh could be more efficient than a structured counterpart with equivalent accuracy. Strongly varying accuracy requirements across the domain could favor this strategy [?, ?]. Moreover, should we base our mesh adaptation on *a posteriori* error estimates, we will need a new formulation that respects the discretization of both the solution and Lagrange multiplier. The precise condition on the multiplier discretization in general for convergence is still unclear, but should be calculable for this individual problem. Using our error estimation technology, we can monitor the error in the constraint being weakly enforced by FD by defining the error functional J to be the constraint operator. Moreover, we may then control this error by adaptively refining the background mesh, which is not currently possible in existing implementations.

The constituent particles of the simulation are perhaps best modeled as elastic bodies, rather than rigid particles. The FD method may be extended by using *distributed* Lagrange multipliers to implement this constraint, which amounts to the imposition of the Navier equation inside the particle. Thus, very general stress tensors may be used to describe the particle itself without changing the solution strategy. The long range forces between particles may be handled efficiently by a Fast Multipole Method solver, which will be coupled to the FD simulation directly rather than iterating separately.

A large particle simulation could be carried out using an unstructured mesh that would be regenerated whenever it became unduly deformed. Here, the FD method provides an important advantage over moving meshes. In order to handle the contact problem for the particles, we propose to implicitly calculate not only the particle velocities but also the positions. With a moving mesh, implicit determination of particle positions would necessitate mesh movement and operator reconstruction at every Newton iterate, whereas the FD method would maintain the same background mesh.

Particular tasks for this research on the evolution of structure in interacting particles include:

- (year 1) introducing the unstructured FD method, which allows explicit control of the error introduced,
- (years 2-3) introducing complex particles, such as elastic bodies, and
- (years 2-3) using implicit particle positions in order to efficiently handle the contact problem.

5.2.3 Dynamic Systems and Computational Biology

An ongoing program led by the MCS Computational Biology Group focuses on reconstruction and analysis of metabolic dynamics of several sequenced organisms, in particular several species of cyanobacteria [?, ?]. An important feature of metabolism of many organisms is the presence of stable biochemical oscillations of long period, the *cell clock*. The exact mechanism of this clock, which has eluded researchers for quite some time, can be understood by analyzing ODE models of cell metabolism, whose structure is determined by the cells' *metabolic network*. We intend to utilize our expertise in analysis and simulation of dynamics in collaboration with the MCS Computational Biology Group in studies of the cell clock.

While ODE models arising in studies of cell biochemistry tend to have very high dimension, significant time-scale separation that is naturally present in these systems allows the reduction to a low-dimensional slow manifold of the system. Capitalizing on earlier work, we will develop numerical methods for the automatic computation of slow manifolds and bifurcation analysis of the reduced systems.

As the exact structure of metabolic dynamical systems is not always known, we will develop numerical reconstruction tools based on optimization of the discrete structure of regulatory interactions among the chemical species of the system, as well as continuous parameters of such interactions.

This research focuses on the following tasks:

- (years 1-2) isolating the invariant manifold for “slow” variables in metabolic reactions, and
- (years 1-3) developing a hybrid discrete/continuous optimization process in order to discover a stable metabolic clock mechanism.

5.3 Impact of Proposed Work

The proposed work in PDE modeling and applications will:

- Enable mathematical analysis of the self-assembly problem, thus answering the question of whether we can predict/guide structure formation by adjusting fluid parameters/flow. (This is a complicated control problem.) The correlation functions (Green functions) ought to provide the basis for the cost function.
- Enable modeling of complex magnetic storage devices.
- Lead to the discovery of a chemical dynamical mechanism for the metabolic cell-clock. This would revolutionize thinking on a century old problem. Flexible, conservative integration schemes are crucial for simulating such long-time behavior.

Direct numerical simulation of the molecular self-assembly mechanism will enable a more precise, general, and useful mathematical analysis of the self-assembly process. Phenomenological scaling laws, parameter sensitivities, and large-scale correlations could be derived from computer

experiments. Current laboratory experiments in this area suffer from common problems: high cost, incomplete data, and difficulty in varying experimental parameters. An excellent example of a successful collaboration of this type, between theory and computer experiment, is chronicled in the book by Joseph [?]. We expect similar advances based upon simulations of interacting particles.

The ability to model and predict the effect of microstructures on the performance of complex magnetic systems at high temperatures and over long periods of time is required for the design of storage devices. As these devices get smaller, an even more detailed understanding is required.

Analysis of the cell-clock mechanism for cyanobacteria will require the harmonious interaction of large-scale numerical simulations, dynamical systems technology, and methods for optimization and parameter continuation. The problem itself has not as yet been completely expressed in mathematical terms, relying on physical intuition and evaluation by the “eye”. A solid mathematical formulation would be of enormous benefit to the computational biology community.

6 Publications

The work funded under this project has generated 8 journal papers and 1 technical report.

6.1 Journal Publications

- *Transient Dynamics of Pinning of Domain Walls*, G. K. Leaf, S. Obukhov, S. Scheidl, V. M. Vinokur, *J. Magnetism and Magnetic Materials*, v 241 (2002) pp. 118–123.
- *Implicit Integration of the Time-Dependent Ginzburg-Landau Equations of Superconductivity*, D. O. Gunter, H. G. Kaper, and G. K. Leaf, *SIAM J. Sci. Comp.* v 23 (2002) pp. 1944–1959.
- *The Frozen-Field Approximation and the Ginzburg-Landau Equations of Superconductivity*, H. G. Kaper and H. Nordborg, *J. Eng. Math.* 39(1–4) (2001), pp. 221–40.
- *Waves in Ferromagnetic Media*, T. Colin, C. Galusinski, H. G. Kaper, *Commun. Partial Differ. Equations* 27(7–9) (2002), pp. 1625–58.
- *Asymptotic Analysis of Two Reduction Methods for Systems of Chemical Reactions*, H. G. Kaper and T. J. Kaper, *Physica D* 165(1–2) (2002), pp. 66–93.
- *Noise Effect on Wave-Generated Transport Induced by Ideal Waves*, J. M. Restrepo and G. K. Leaf, *J. of Physical Oceanography*, v 32, (2002), pp. 2334–2349.
- *Hysteresis in Layered Spring Magnets*, J. S. Jiang, H. G. Kaper, and G. K. Leaf, *Discrete and Continuous Dynamical Systems, Series B*, v 1 (2001) pp. 219–232.
- *Rotational Hysteresis of Exchange-Spring Magnets*, J. S. Jiang, S. D. Bader, H. G. Kaper, G. K. Leaf, *J. Phys. D Appl. Phys.* v 35 (2002) pp. 2339–2343.

6.2 Technical Reports

- *Normal Modes of Spin Excitation in Magnetic Nanoparticles*, M. Grimsditch, G. K. Leaf, H. G. Kaper, D. A. Karpeev, and R. E. Camley, Submitted to *Phys. Rev. B*.

References

- [1] slepc. Scalable Eigenvalue Solvers Package.
- [2] W. K. Anderson, W. D. Gropp, D. K. Kaushik D. E. Keyes, and B. F. Smith. Achieving high sustained performance in an unstructured mesh CFD application. In *Proceedings of SC 99*, 1999. also available as Argonne preprint ANL/MCS-P776-0899.
- [3] I. Babuška and A.D. Miller. The post-processing approach in the finite element method, i: Calculations of displacements, stresses, and other higher derivatives of the displacements. *Int. J. Numer. Meth. Eng.*, 20:1085–1109, 1984.
- [4] I. Babuška and A.D. Miller. The post-processing approach in the finite element method, ii: The calculation of stress intensity factors. *Int. J. Numer. Meth. Eng.*, 20:1111–1129, 1984.
- [5] I. Babuška and A.D. Miller. The post-processing approach in the finite element method, iii: A posteriori error estimation and adaptive mesh refinement. *Int. J. Numer. Meth. Eng.*, 20:2311–2324, 1984.
- [6] Satish Balay, Kris Buschelman, William D. Gropp, Dinesh Kaushik, Matt Knepley, Lois Curfman McInnes, Barry F. Smith, and Hong Zhang. PETSc 2.0 users manual. Technical Report ANL-95/11 - Revision 2.1.5, Argonne National Laboratory, November 2002.
- [7] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica*, 10:1–102, 2001.
- [8] Steve Benson, Lois Curfman McInnes, and Jorge Moré. GPCG: A case study in the performance and scalability of optimization algorithms. Technical Report ANL/MCS-P768-0799, Mathematics and Computer Science Division, Argonne National Laboratory, 1999.
- [9] S. Bhowmick, L. McInnes, B. Norris, and P. Raghavan. The role of multi-method linear solvers in pde-based simulations. In *2003 International Conference on Computational Science and its Applications, ICCSA 2003*, 2003. Submitted.
- [10] S. Bhowmick, P. Raghavan, L. McInnes, and B. Norris. Faster pde-based simulations using robust composite linear solvers. Technical Report ANL/MCS-P993-0902, ANL, 2002. Submitted to Future Generation Computer Systems.
- [11] George Biros. Veltisto Web page. <http://www.cs.nyu.edu/~{ }biros/veltisto>.
- [12] Andrew Cleary, Scott Kohn, Steven Smith, and Brent Smolinski. Language interoperability mechanisms for high-performance scientific applications. In *Proceedings of the SIAM Workshop on Object Oriented Methods for Inter-operable Scientific and Engineering Computing*, pages 30–39. SIAM, 1999.
- [13] Todd S. Coffey, C. T. Kelly, and David E. Keyes. Pseudo-transient continuation and differential-algebraic equations. Technical report, ODU, 2002. Submitted to the special issue of the SIAM Journal of Scientific Computing dedicated to papers from the 2002 Copper Mountain Conference on Iterative Methods.
- [14] B. Smith et al. Scientific applications using PETSc. <http://www.mcs.anl.gov/petsc/petsc-2/publications>.
- [15] W. D. Gropp, D. K. Kaushik, D. E. Keyes, and B. F. Smith. Toward realistic performance bounds for implicit CFD codes. In A. Ecer et al., editor, *Proceedings of Parallel CFD'99*. Elsevier, 1999.
- [16] William D. Gropp, David E. Keyes, Lois Curfman McInnes, and M. D. Tidriri. Globalized Newton-Krylov-Schwarz algorithms and software for parallel implicit CFD. *Int. J. High*

- Performance Computing Applications*, 14:102–136, 2000. also available as Argonne preprint ANL/MCS-P788-0100.
- [17] Paul Hovland, Uwe Naumann, and Boyana Norris. ADIC. <http://www.mcs.anl.gov/autodiff>.
 - [18] Dmitry Leykekhman, Nesy Tania, and Matthew G. Knepley. The Poisson problem. <http://www.mcs.anl.gov/petsc/petsc-3/poisson.html>.
 - [19] L. McInnes, B. Norris, S. Bhowmick, and P. Raghavan. Adaptive sparse linear solvers for implicit cfd using newton-krylov algorithms. In *Proceedings of the Second MIT Conference on Computational Fluid and Solid Mechanics*, 2003. To appear.
 - [20] Uwe Naumann. Optimal accumulation of Jacobian matrices by elimination methods on the dual computational graph. *to appear in Mathematical Programming*, 2002. Preprint ANL/MCS-P944-0402, Argonne National Laboratory, April 2002.
 - [21] Amitava Bhattacharjee (PI). Center for magnetic reconnection studies. <http://www.physics.uiowa.edu/cmrs>.
 - [22] Barry F. Smith, Petter Bjørstad, and William D. Gropp. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press, 1996.

Curriculum Vitae of Satish Balay

Mathematics and Computer Science Division

Argonne National Laboratory, Argonne, IL 60439-4844

<http://www.mcs.anl.gov/~balay> balay@mcs.anl.gov

Education

M.S. in Computer Science, Old Dominion University, 1995.

B.Tech. in Electronics and Communications Engineering, Jawaharlal Nehru Technological University, 1993.

Professional Experience

Sr. Software Developer, Mathematics and Computer Science Division, Argonne National Laboratory, 2003 – present.

Sr. Scientific Programmer, Mathematics and Computer Science Division, Argonne National Laboratory, 1999 – 2003.

Scientific Programmer, Mathematics and Computer Science Division, Argonne National Laboratory, 1997 – 1999.

Scientific Associate, Mathematics and Computer Science Division, Argonne National Laboratory, 1995 – 1997.

Selected Publications Related to the Proposed Research

1. *Efficient Management of Parallelism in Object-Oriented Numerical Software Libraries*, S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith, Modern Software Tools in Scientific Computing, E. Arge, A. M. Bruaset and H. P. Langtangen, eds., Birkhauser Press, 1997, pp. 163–202.
2. *PETSc Solver Technology for Enhanced Oil Recovery Simulators*, S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith, Section 5.3 in the New Generation Framework for Petroleum Reservoir Simulation First Annual Report, June 1996, pp. 101–123.
3. *A Microkernel Design for Component-based Parallel Numerical Software Systems*, Satish Balay and William D. Gropp and Lois Curfman McInnes and Barry F. Smith, "Proceedings of the SIAM Workshop on Object Oriented Methods for Inter-operable Scientific and Engineering Computing", October, 1998.
4. *A Fully Implicit Parallel EOS Compositional Simulator for Large Scale Reservoir Simulation*, P. Wang, S. Balay, K. Sepehrnoori, J. Wheeler, J. Abate, B. Smith, G.A. Pope, "SPE 51885 Proceedings of the 1999 Reservoir Simulation Symposium", Society of Petroleum Engineers, February, 1999.
5. *Parallel Components for PDEs and Optimization: Some Issues and Experiences*, B. Norris, S. Balay, S. Benson, L. Freitag, P. Hovland, L. McInnes, and B. Smith, available as Argonne National Laboratory preprint ANL/MCS-P932-0202, February, 2002, submitted to a special issue of Parallel Computing
6. *Software for the Scalable Solution of Partial Differential Equations*, Satish Balay, William D. Gropp, Lois Curfman McInnes, Barry F. Smith, "Sourcebook of Parallel Computing", Morgan Kaufmann Publishers, 2003, pp. 621–648
7. *PETSc 2.0 Users Manual*, S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith, Tech. Rep. ANL-95/11 - Revision 2.1.6, Argonne National Laboratory, Aug, 2003.

Honors

Pacesetter Award by Argonne National Laboratory, for work on developing the New Generation Oil Reservoir Simulator, 1998

Collaborators within the past 4 years

S. Benson, ANL; K. Buschelman, ANL; W. Gropp, ANL; D. Kaushik, ANL; M. Knepley, ANL; L. C. McInnes, ANL; B. Norris, ANL; B. Smith, ANL; H. Zhang, ANL.

Curriculum Vitae of William D. Gropp

Mathematics and Computer Science Division

Argonne National Laboratory, Argonne, IL 60439-4844

<http://www.mcs.anl.gov/~gropp> gropp@mcs.anl.gov

Education

Ph.D. in Computer Science, Stanford University, 1982.

M.S. in Physics, University of Washington, 1978.

B.S. in Mathematics, Case Western Reserve University, 1977.

Professional Experience

Associate Division Director, Mathematics and Computer Science Division, Argonne National Laboratory, 2000 –.

Senior Computer Scientist, Mathematics and Computer Science Division, Argonne National Laboratory, 1996 –.

Computer Scientist, Mathematics and Computer Science Division, Argonne National Laboratory, 1990 – 1996.

Associate Professor, Computer Science Department, Yale University, 1988 – 1990.

Assistant Professor, Computer Science Department, Yale University, 1982 – 1988.

Selected Publications Related to the Proposed Research

1. *Efficient Management of Parallelism in Object-Oriented Numerical Software Libraries*, S. Balay, W.D. Gropp, L. C. McInnes, and B. F. Smith, *Modern Software Tools in Scientific Computing*, E. Arge, A.M. Bruaset and H.P. Langtangen, Ed., Birkhauser Press, 1997, pp. 163–202, invited.
2. *Achieving High Sustained Performance in an Unstructured Mesh CFD Application*, W. Anderson, W. Gropp, D. Kaushik, D. Keyes, B. Smith, *Proceedings of SC '99*, 1999, winner of a 1999 Gordon Bell Prize.
3. *Sowing MPICH: A case study in the dissemination of a portable environment for parallel scientific computing*, W. Gropp and E. Lusk, *International Journal of Supercomputer Applications and High Performance Computing*, 11(2) 103-114, Summer, 1997.
4. **Using MPI: Portable Programming with the Message-Passing Interface**, W. Gropp, E. Lusk, and A. Skjellum, MIT Press, Cambridge, MA (1994). 2nd edition, 1999.
5. **MPI The Complete Reference: Volume 2: The MPI-2 Extensions**, W. Gropp, S. Huss-Lederman, A. Lumsdaine, E. Lusk, B. Nitzberg, W. Saphir, and M. Snir, MIT Press, Cambridge, MA (1998).
6. **Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations**, B.F. Smith, P.E. Bjorstad, and W.D. Gropp, Cambridge University Press, 1996.
7. *Towards Realistic Performance Bounds for Implicit CFD Codes*, W.D. Gropp, D.K. Kaushik, D.E. Keyes, and B.F. Smith, *Proceedings of Parallel CFD'99*, pp. 241–248, 1999.
8. *A Microkernel Design for Component-based Parallel Numerical Software Systems*, S. Balay, W. Gropp, L. C. McInnes, and B. Smith, *Proceedings of the SIAM Workshop on Object Oriented Methods for Interoperable Scientific and Engineering Computing*, SIAM, 1999. pp. 58-67.

9. *A high-performance, portable implementation of the MPI message passing interface standard*, W. Gropp, E. Lusk, N. Doss, and A. Skjellum, *Parallel Computing*, 22(6) 789-828, September, 1996.
10. *A high-performance MPI implementation on a shared-memory vector supercomputer*, W. Gropp and E. Lusk, *Parallel Computing* 22(11) 1513-1526, January, 1997.

Honors

SC'99 Gordon Bell prize in the special category for the paper *Achieving High Sustained Performance on an Unstructure Mesh CFD Application*, by Kyle Anderson, William Gropp, Dinesh Kaushik, David Keyes and Barry Smith.
Beale-Orchard-Hays Honorable Mention, 2000, with Jorge Moré.

Collaborators and Co-Editors within the past 4 years

(not at Argonne National Laboratory, not in above publication list, and not involved in the present proposal): Prof. X.-C. Cai, Prof. E. Jessup, University of Colorado-Boulder; Dr. A. Geist, Oak Ridge National Laboratory; Prof. Robert Rosner, Dr. Andrea Malagoli, University of Chicago; Dr. David Keyes, ODU; Dr. David P Young, Boeing, Dr. Philip Dickens, Illinois Institute of Technology; Dr. Thomas Sterling, CalTech; James Cownie, Etnus Inc.

Postdoctoral Scholars Sponsored in past 5 years

Lois Curfman McInnes, Rajeev Thakur, Robb Ross.
Total postdoctoral scholars sponsored, 3.

Doctoral Advisor

Professor Joseph Oliger, Stanford University.

Curriculum Vitae of Hans G. Kaper

Mathematics and Computer Science Division

Argonne National Laboratory, Argonne, IL 60439-4844

<http://www.mcs.anl.gov/~kaper> kaper@mcs.anl.gov

Education

Ph.D. in Mathematics and Physical Sciences (cum laude), Rijksuniversiteit Groningen, Netherlands, 1965.

M.S. in Mathematics and Physical Sciences, Rijksuniversiteit Groningen, Netherlands, 1960.

B.S. in Mathematics, Rijksuniversiteit Groningen, Netherlands, 1957.

Professional Experience

Associate Professor 1967–69; Assistant Professor, 1965–67 – Rijksuniversiteit Groningen.

Mathematician, 1969–82; MCS Division Director, 1987–91; Senior Mathematician, 1982–present
– Argonne National Laboratory.

Program Director for Applied Mathematics, National Science Foundation, 2001–present.

Selected Publications Related to the Proposed Research

1. *Implicit Integration of the Time-Dependent Ginzburg-Landau Equations of Superconductivity*, D. O. Gunter, H. G. Kaper, G. K. Leaf, SIAM J. Sci. Comput. 23(6) (2002), 1943–58.
2. *The Frozen-Field Approximation and the Ginzburg-Landau Equations of Superconductivity*, H. G. Kaper and H. Nordborg, J. Eng. Math. 39(1–4) (2001), 221–40.
3. *Waves in Ferromagnetic Media*, T. Colin, C. Galusinski, H. G. Kaper, Commun. Partial Differ. Equations 27(7–9) (2002), 1625–58.
4. *Asymptotic Analysis of Two Reduction Methods for Systems of Chemical Reactions*, H. G. Kaper and T. J. Kaper, Physica D 165(1–2) (2002), 66–93.
5. *Asymptotic-Numerical Study of Supersensitivity for Generalized Burgers Equations*, M. Garbey and H. G. Kaper, SIAM J. Sci. Comput. 22(1) (2000), 368–85.

Honors and Professional Societies

Corresponding Member – Koninklijke Nederlandse Academie van Wetenschappen (Royal Netherlands Academy of Arts and Sciences)

Foreign Assignment, Argonne National Laboratory, 1976–77

NATO Science Fellowship, 1966–67

EURATOM Research Fellowship, 1961

American Mathematical Society (AMS)

Society for Industrial and Applied Mathematics (SIAM)

Wiskundig Genootschap (Amsterdam)

International Computer Music Association (ICMA)

International Community for Auditory Display (ICAD)

Selected Other Activities

Associate Editor

Integral Equations and Operator Theory

Transport Theory and Statistical Physics

J. of Engineering Mathematics

Editor-at-Large – Applied Mathematics, Marcel Dekker, Inc.

Reviewer – NSF, DOE

Referee – Various professional journals

Curriculum Vitae of Dmitry Karpeev

Mathematics and Computer Science Division

Argonne National Laboratory, Argonne, IL 60439-4844

<http://www.mcs.anl.gov/~karpeev> karpeev@mcs.anl.gov

Education

Ph.D. in Computer Science, Old Dominion University, 2002

B.S. *summa cum laude* in Applied Mathematics, Old Dominion University, 1996.

Professional Experience

Postdoctoral Researcher, Science Division, Argonne National Laboratory, Mathematics and Computer, 2002 – present.

Givens Research Fellow, Argonne National Laboratory, Mathematics and Computer Science Division, 2000 – 2001.

Systems programmer, Advanced Distributed Simulation, Computer Science Department, Old Dominion University and BMH Associates, Norfolk, VA.

Selected Publications Related to the Proposed Research

1. *Geometric Integrators for the Nonlinear Schrödinger Equation*, A. L. Islas, D. A. Karpeev, and C. M. Schober, *J. Comp. Phys.* **173** 116-148 (2001).
2. *Symplectic Integrators for Discrete Nonlinear Schrödinger Systems*, D. A. Karpeev, and C. M. Schober, *Mathematics and Computers in Simulation*, **56(2)** 145-156 (2001).
3. *Geometric Integrators for Hamiltonian PDEs*, D. A. Karpeev, PhD Thesis, Computer Science, Old Dominion University, 2002.
4. *Normal Modes of Spin Excitations in Nanoparticles* M. Grimsditch, G. K. Leaf, H. G. Kaper, D. A. Karpeev, R. E. Camley, *Phys. Rev. B*, submitted.

Honor and Professional Societies (member since)

American Mathematical Society (1998), Society for Industrial and Applied Mathematics (1997)

Collaborators within the past 4 years

S. Balay, Argonne National Laboratory, ANL; H. Kaper, ANL; M. Knepley, ANL; G. Leaf, ANL; E. Selkov, ANL; C. Schober, University of Central Florida;

Doctoral Advisor

David E. Keyes, Old Dominion University

Curriculum Vitae of Matthew G. Knepley

Mathematics and Computer Science Division

Argonne National Laboratory, Argonne, IL 60439-4844

<http://www.mcs.anl.gov/~knepley> knepley@mcs.anl.gov

Education

Purdue University, Jan 1997 – Dec 2000

Ph. D. in Computer Science, Dec 2000

University of Minnesota, Sept 1995 – Dec 1996

M.S. in Computer Science, Dec 1996

University of Chicago, Sept 1994 – May 1995

Physics graduate student

Case Western Reserve University, Cleveland, OH, Aug 1990 – May 1994

B. S. *summa cum laude* in Mathematical Physics, May 1994

Professional Experience

Postdoctoral Researcher, Argonne National Laboratory, Mathematics and Computer Science Division, 2001 – present.

Research Scientist, Akamai Technologies Inc., Distributed Data Collection, 2000 – 2001.

Selected Publications Related to the Proposed Research

1. *Design of Large Scale Parallel Simulations*, Matthew G. Knepley, Ahmed H. Sameh, and Vivek Sarin, Proceedings of Parallel Computational Fluid Dynamics '99, David Keyes, et.al., Ed., Elsevier North-Holland, 1999.
2. *Algorithm Development for Large Scale Computing*, Matthew G. Knepley and Vivek Sarin, Proceedings of the SIAM Workshop on Object-Oriented Methods for Inter-operable Scientific and Engineering Computing, October 1998.
3. *Parallel Simulation of Particulate Flows*, Matthew G. Knepley, Ahmed H. Sameh, and Vivek Sarin, in Solving Irregularly Structured Problems in Parallel, LNCS 1457, pp. 226–237, 1998.
4. *Parallel Building Blocks for Finite Element Simulations*, Matthew G. Knepley and Denis Vanderstraeten, Proceedings of Parallel Computational Fluid Dynamics '97, R. Emerson, et.al., Ed., Elsevier North-Holland, pp. 281–287, 1998.
5. *PETSc 2.0 Users Manual*, S. Balay, K. Buschelman, W. D. Gropp, D. Kaushik, M. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang, MCS Technical Report, ANL-95-11, Revision 2.1.5, January 2003, (see <http://www.mcs.anl.gov/petsc>).

Honor and Professional Societies (member since)

Sigma Xi (1993), Phi Beta Kappa (1994), Golden Key (1994), Upsilon Pi Epsilon (2000)

Selected Other Activities

Reviewer for TOMS.

Collaborators within the past 4 years

S. Balay, Argonne National Laboratory (ANL); V. Eijkhout, University of Tennessee; W. Gropp, ANL; D. Keyes, Old Dominion University; J. Moré, ANL; V. Sarin, Texas A&M University; B. Smith, ANL; S. Teng, Boston University

Doctoral Advisor

Ahmed H. Sameh, Purdue University

Curriculum Vitae of Gary K. Leaf

Mathematics and Computer Science Division

Argonne National Laboratory, Argonne, IL 60439-4844

<http://www.mcs.anl.gov/~leaf> leaf@mcs.anl.gov

Education

University of Illinois

Ph.D., Mathematics, 1961

M.S., Mathematics, 1959

B.S., Physics, 1957

Professional Experience

Postgraduate National Science Foundation Fellow, University of Chicago, Mathematics Department, 1961 – 1963.

Mathematician, Mathematics and Computer Science Division, Argonne National Laboratory, 1963 – present.

Selected Publications Related to the Proposed Research

1. *Noise Effects on Wave-Generated Transport Induced by Ideal Waves*, J. M. Restrepo and G. K. Leaf, *J. Phys. Oceanogr.* 32 (2002), 2334–2349.
2. *Numerical Simulations of Driven Vortex Systems*, G. W. Crabtree, D. O. Gunter, H. G. Kaper, A. E. Koshelev, G. K. Leaf, V. M. Vinokur, *Phys. Rev. B* 61(2) (2000) 1446–55.
3. *Rotational Hysteresis of Exchange-Spring Magnets*, J. Jiang, S. Bader, H. Kaper, G. Leaf, R. Shull, A. Shapiro, V. Gornakov, V. Nikitenko, C. Platt, A. Berkowitz, S. David, E. Fullerton, *J. Phys. D* 35(19) (2002), 2339–343.
4. *Applicability of D.C. Relaxation Techniques to Multi-Step Reactions*, Z. Nagy, N. C. Hung, K. C. Liddell, M. Minkoff, G. K. Leaf, *Electroanal. Chem.* 421 (1997), 33–44.
5. *Implicit Integration of the Time-Dependent Ginzburg-Landau Equations of Superconductivity*, D. O. Gunter, H. G. Kaper, G. K. Leaf, *SIAM J. Sci. Comput.* 23(6) (2002) 1943–58.

Curriculum Vitae of Lois Curfman McInnes

Mathematics and Computer Science Division

Argonne National Laboratory, Argonne, IL 60439-4844

<http://www.mcs.anl.gov/~mcinnes> mcinnes@mcs.anl.gov

Education

University of Virginia, Sept 1988 – May 1993

Ph. D. in Applied Mathematics, May 1993

Masters of Applied Mathematics, Jan 1990

Muhlenberg College, Allentown, PA, Sept 1984 – May 1988

B. S. *summa cum laude* in Mathematics and Physics, May 1988

Professional Experience

Software Engineer, Mathematics and Computer Science Division, Argonne National Laboratory, 2001 – present.

Assistant Computer Scientist, Mathematics and Computer Science Division, Argonne National Laboratory, 1997 – 2001.

Fellow, Computation Institute, University of Chicago and Argonne National Laboratory, December 1999 – present.

Postdoctoral Fellow, Argonne National Laboratory and Old Dominion University, Computer Science Department of ODU, February 1996 – May 1997.

Department of Energy Distinguished Postdoctoral Fellow, Mathematics and Computer Science Division, Argonne National Laboratory, February 1993 – February 1996.

Selected Publications Related to the Proposed Research

1. *Efficient Management of Parallelism in Object-Oriented Numerical Software Libraries*, S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith, *Modern Software Tools in Scientific Computing*, E. Arge, A.M. Bruaset and H.P. Langtangen, Ed., Birkhauser Press, 1997, pp. 163–202, invited.
2. *Parallel Components for PDEs and Optimization: Some Issues and Experiences*, B. Norris, S. Balay, S. Benson, L. Freitag, P. Hovland, L. C. McInnes, and B. Smith, *Parallel Computing*, volume 28:12, 2002, pp. 1811–1831.
3. *Globalized Newton-Krylov-Schwarz Algorithms and Software for Parallel Implicit CFD*, W.D. Gropp, D.E. Keyes, L. C. McInnes, and M.D. Tidriri, *Int. Journal on High Performance Computing Applications*, volume 14, 2000, pp. 102–136.
4. *Toward a Common Component Architecture for High-Performance Scientific Computing*, R. Armstrong, D. Gannon, A. Geist, K. Keahey, S. Kohn, L. C. McInnes, S. Parker, and B. Smolinski, *Proceedings of the 1999 High-Performance Distributed Computing Conference*, pp. 115–124.
5. *A Quality-of-Service Architecture for High-Performance Numerical Components*, P. Hovland, K. Keahey, L. C. McInnes, B. Norris, L.F. Diachin, and P. Raghavan, to appear in the *Proceedings of the Workshop on QoS in Component-Based Software Engineering*, Toulouse, France, June 20, 2003.

6. *A Case Study in the Performance and Scalability of Optimization Software*, S. Benson, L. C. McInnes, and J.J. Moré, ACM Transactions on Mathematical Software, volume 27, number 3, 2001, pp. 361–376.
7. *A Microkernel Design for Component-based Parallel Numerical Software Systems*, S. Balay, W.D. Gropp, L. C. McInnes, and B.F. Smith, Proceedings of the SIAM Workshop on Object Oriented Methods for Inter-operable Scientific and Engineering Computing, Oct 21-23, 1998, pp. 60–69.
8. *PETSc 2.0 Users Manual*, S. Balay, K. Buschelman, W. D. Gropp, D. Kaushik, M. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang, MCS Technical Report, ANL-95-11, Revision 2.1.5, January 2003, (see <http://www.mcs.anl.gov/petsc>).
9. *Toolkit for Advanced Optimization (TAO) Users Manual*, S. Benson, L. C. McInnes, J. Moré, and J. Sarich, MCS Technical Report, ANL/MCS-TM-242, Revision 1.5, January 2003, (see <http://www.mcs.anl.gov/tao>).
10. *The Role of Multi-Method Linear Solvers in PDE-Based Simulations*, S. Bhowmick, L. C. McInnes, B. Norris, and P. Raghavan, Lecture notes in Computer Science 2677, Editors V. Kumar, M.L. Gavrilova C.J.K. Tan, and P. L’Ecuyer, 2003, pp. 828–839.

Honor and Professional Societies (member since)

Phi Beta Kappa (1987), Omicron Delta Kappa (1987), SIAM (1990), ACM (1999)

Selected Other Activities

SIAM Activity Group on Supercomputing, Vice Chair (2000-2002).

Workshops Co-Chair, Grace Hopper Celebration of Women in Computing Conference (2000, 2002).

BOFs Chair, SC2003 Conference.

Program Committee, 2004 SIAM Parallel Processing Conference, February 2004.

Reviewer for SIAM J. Matrix Anal., SIAM J. on Sci. Comput., Int. J. High Performance Computing Applications, J. Parallel and Distributed Computing Practices, NSF and DOE proposals.

Adjunct professor for Northwestern University course Engineering Analysis I, focusing on introductory linear algebra and programming, fall quarter 1999.

Collaborators within the past 4 years

R. Armstrong, Sandia National Laboratories (SNL); S. Balay, Argonne National Laboratory (ANL); S. Benson, ANL; L.F. Diachin, SNL; D. Gannon, Indiana University; W. Gropp, ANL; P. Hovland, ANL; K. Keahey, ANL; D. Keyes, Old Dominion University; S. Kohn, Lawrence Livermore National Laboratory (LLNL); J. Moré, ANL; B. Norris, ANL; S. Parker, University of Utah; P. Raghavan, Penn State University; B. Smith, ANL; B. Smolinski, LLNL.

Doctoral Advisor

Professor James Ortega, University of Virginia

Curriculum Vitae of Michael Minkoff

Mathematics and Computer Science Division

Argonne National Laboratory, Argonne, IL 60439-4844

<http://www.mcs.anl.gov/~minkoff> minkoff@mcs.anl.gov

Education

Ph.D. in Computer Science, University of Wisconsin, Sept. 1968 – Jan. 1973.

M.S.E. in Aerospace and Mechanical Sciences, Princeton University, Sept. 1966 – June, 1968.

B.S. in Mathematics, University of Wisconsin, Sept. 1962 – June 1966.

Professional Experience

Computer Scientist, Mathematics and Computer Science Division, Argonne National Laboratory, 1980 –.

Assistant Computer Scientist, Mathematics and Computer Science Division, Argonne National Laboratory, 1973 – 1980.

Assistant Professor, Mathematical Sciences Department, Northern Illinois University, 1973 – 1975.

Adjunct Professor, Mathematical Sciences Department, Northern Illinois University, 1985-1992.

Adjunct Professor, Computer Science Dept., Harper College, 1989 – 1992.

Selected Publications and Presentations Related to the Proposed Research

1. *The Subspace Projected Approximate Matrix (SPAM) Modification of the Davidson Method*, R. Shepard, A. F. Wagner, J. L. Tilson, and M. Minkoff, *Journal of Computational Physics*, Vol. 172, No. 2, 2001, pp. 472-514.
2. *Interpolating Moving Least Squares Methods for Fitting Potential Energy Surfaces: Detailed Analysis of One-Dimensional Applications*, G. G. Maisuradze, D. L. Thompson, A. F. Wagner, and M. Minkoff, *Journal of Chemical Physics*, accepted for publication.
3. *High-Performance Computational Chemistry: Hartree-Fock Electronic Structure Calculations on Massively Parallel Processors*, J. L. Tilson, M. Minkoff, A. F. Wagner, R. Shepard, P. Sutton, R. J. Harrison, R. A. Kendall, and A. T. Wong, *International Journal Of High Performance Computing Applications*, Vol. 13, 1999, pp. 291-302.
4. *Large-Scale Methods for Calculating Exact Cumulative Reaction Probabilities*, M. Minkoff, A. F. Wagner, R. Shepard, and Y. Zhou, *SIAM Computational Science and Engineering Conference*, San Diego, CA, February 10-13, 2003.
5. *Application and Extension of the Subspace Projected Approximate Matrix (SPAM) Modification of the Davidson Method*, M. Minkoff, and R. Shepard, 2002 Copper Mountain Conference on Iterative Methods, Copper Mountain, CO, March 25-29, 2002.

Honors

SIAM Visiting Lecturer, 1990-present

Listed in Who's Who in America, 2001-2003

Honor and Professional Societies (member since)

SIAM (1973)

Phi Kappa Phi (1972)

Sigma Xi (1975)

Collaborators within the past 4 years

R. J. Harrison, University of Tennessee R. A. Kendall, DOE-AMES R. Kulak, Argonne National Laboratory (ANL) G. G. Maisuradze, Oklahoma State University (OSU) Z. Nagy, ANL B. Ruscic, ANL D. A. Scherson, CMU R. Shepard, ANL; P. Sutton, North Central College D. Thompson, OSU J. Tilson, State University of New York – Buffalo (SUNY-Buffalo); G. Von Laszewski, ANL A. F. Wagner, ANL; A. T. Wong, NERSC

Curriculum Vitae of Barry F. Smith

Mathematics and Computer Science Division

Argonne National Laboratory, Argonne, IL 60439-4844

<http://www.mcs.anl.gov/~bsmith> bsmith@mcs.anl.gov

Education

Ph.D. in Mathematics, Courant Institute of Mathematical Sciences, New York University, Sept. 1986 – Aug. 1990.

B.S. in Mathematics, Yale University, Sept 1982 – May 1986.

Professional Experience

Computer Scientist, Mathematics and Computer Science Division, Argonne National Laboratory, 1995 –.

Assistant Computer Scientist, Mathematics and Computer Science Division, Argonne National Laboratory, 1994 – 1995.

Assistant Professor, Mathematics Department, UCLA, 1992 – 1994.

Wilkinson Post-doctoral Fellow, Mathematics and Computer Science Division, Argonne National Laboratory, 1990 – 1992.

Selected Publications Related to the Proposed Research

1. *Efficient Management of Parallelism in Object-Oriented Numerical Software Libraries*, S. Balay, W.D. Gropp, L. C. McInnes, and B. F. Smith, *Modern Software Tools in Scientific Computing*, E. Arge, A.M. Bruaset and H.P. Langtangen, Ed., Birkhauser Press, 1997, pp. 163–202, invited.
2. *PETSc 2.0 Users Manual*, S. Balay, K. Buschelman, W.D. Gropp, D. Kaushik, M. Knepley, L. C. McInnes, B.F. Smith, and H. Zhang, MCS Technical Report, ANL-95-11, Revision 2.1.5, January 2003, (see <http://www.mcs.anl.gov/petsc>).
3. *Achieving High Sustained Performance in an Unstructured Mesh CFD Application*, W. Anderson, W. Gropp, D. Kaushik, D. Keyes, and B. Smith, *Proceedings of SC '99*, 1999, winner of a 1999 Gordon Bell Prize.
4. *PETSc and Overture: Lessons Learned Developing an Interface between Components*, K. Buschelman, W. Gropp, L. C. McInnes, and B. Smith, *Proceedings of the International Federation for Information Processing Working Conference on Software Architectures for Scientific Computing*, Kluwer Publishing, 2000.
5. *Remote Access to Mathematical Software*, E. Dolan, P. Hovland, J. Moré, B. Norris, and B. Smith, *Proceedings of the Internet Accessible Mathematical Computation (IAMC) Workshop 2001*, part of the International Symposium on Symbolic and Algebraic Computation (ISSAC) 2001, London, ON, Canada.
6. *Parallel Domain Decomposition Software*, W. Gropp and B. Smith, *Domain-Based Parallelism and Problem Decomposition Methods in Computational Science and Engineering*, editors D. Keyes, Y. Saad and D. Truhlar, SIAM, Philadelphia, PA, 1994, pp. 97–106.
7. *A Microkernel Design for Component-based Parallel Numerical Software Systems*, S. Balay, W. Gropp, L. C. McInnes, and B. Smith, *Proceedings of the SIAM Workshop on Object Oriented Methods for Interoperable Scientific and Engineering Computing*, SIAM, 1999, pp. 58-67.

8. *Developing a Derivative-Enhanced Object-Oriented Toolkit for Scientific Computations*, P. Hovland, B. Norris, and B. Smith, Proceedings of the SIAM Workshop on Object Oriented Methods for Interoperable Scientific and Engineering Computing, SIAM, 1999. pp. 129-137.
9. *Parallel Components for PDEs and Optimization: Some Issues and Experiences*, B. Norris, S. Balay, S. Benson, L. Freitag, P. Hovland, L. McInnes, and B. Smith, Parallel Computing, 28 (12) (2002) pp. 1811-1831.
10. *Making Automatic Differentiation Truly Automatic: Coupling PETSc with ADIC*, P. Hovland, B. Norris, and B. Smith, Proceedings of ICCS2002, Amsterdam, 2002.

Honors

First Prize, Student Paper Competition, Copper Mountain Conference on Iterative Methods, April 1990.

1991–1992 Wilkinson Fellow, post-doctoral fellowship at Argonne National Laboratory in numerical analysis/numerical computing.

Second Prize, Fifth Leslie Fox Prize Meeting, June 1991, international prize in numerical analysis offered every two years.

Co-winner, 1993 Householder Prize for best dissertation in numerical linear algebra during the previous three years.

SC'99 Gordon Bell prize in the special category for the paper Achieving High Sustained Performance on an Unstructure Mesh CFD Application, by Kyle Anderson, William Gropp, Dinesh Kaushik, David Keyes and Barry Smith.

Professional Societies (member since)

SIAM (1990), AMS (1990)

Collaborators within the past 4 years

S. Balay, Argonne National Laboratory (ANL); K. Buschelman, ANL; F. Dobrian, Old Dominion University (ODU); V. Eijkhout, University of Tennessee; K. Germaschewski, University of Iowa; W. Gropp, ANL; S. Jardin, Princeton Plasma Physics Laboratory; D. Keyes, ODU; D. Kaushik, ANL; M. Knepley, ANL; L. C. McInnes, ANL; H. Zhang, ANL.

Doctoral Advisor

Professor Olof Widlund, New York University

Curriculum Vitae of Hong Zhang

Mathematics and Computer Science Division

Argonne National Laboratory, Argonne, IL 60439-4844

<http://www.mcs.anl.gov/~hzhang> hzhang@mcs.anl.gov

Education

Ph.D. in Applied Mathematics, Michigan State University, 1985 – Aug. 1989.

M.S. in Applied Mathematics, Michigan State University, 1985.

B.S. in Mathematics, Beijing Normal University, China

Professional Experience

Senior Scientific Programmer, Mathematics and Computer Science Division, Argonne National Laboratory, 2001 –

Research Professor, Department of Computer Science, Illinois Institute of Technology, 2000 –

Visiting Scientist, Mathematics and Computer Science Division, Argonne National Laboratory, 1999 – 2000.

Associate Professor, Department of Mathematics, Louisiana State University, 1997 – 2000

NSF VPW Visiting Professor, Department of Mathematics, Louisiana State University, 1996 – 1997

Associate Professor, Department of Mathematical Sciences, Clemson University, 1996- 1997.

Visiting Scientist, Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, 9/1992-8/1994, 7-8/1995, 7-8/1996.

Assistant Professor, Department of Mathematical Sciences, Clemson University, 1989- 1996.

Selected Publications Related to the Proposed Research

1. *Pseudozeros of Multivariate Polynomials*, J. W. Hoffman, J. J. Madden and H. Zhang, *Mathematics of Computation*, Vol. 72, No. 242, 2003, pp.975-1002.
2. *On the Multigrid Waveform Relaxation Method*, S. Ta'asan and H. Zhang, *SIAM J. Sci. Comput.*, Vol. 16, No. 5, 1995, pp.1092-1104.
3. *Using Parallel Banded Linear System Solvers in Generalized Eigenvalue Problems*, H. Zhang and W. F. Moss, *Parallel Computing*, Vol. 20, No. 8, August 1994, pp.1089-1105.
4. *Efficient Tridiagonal Solvers on Multicomputers*, X. Sun, H. Zhang and L. M. Ni, *IEEE Trans. On Computers*, Vol. 41, No. 3, March 1992, pp.286-296.
5. *Parallel Homotopy Algorithm for the Symmetric Eigenvalue Problem*, T. Y. Li, H. Zhang and X. Sun, *SIAM J. Sci. Stat. Comput.*, Vol. 12, No. 3, 1991, pp.469-487.
6. *On the Accuracy of the Parallel Diagonal Dominant Algorithm*, *Parallel Computing*, H. Zhang, 17, 1991, pp. 265-272.

Honors

Acknowledgement from the Alpha Lambda Delta Freshman Honor Society for Superior Instruction of Freshman Students, Fall, 1998.

Award for Faculty Excellence, Clemson University, 1996.

Collaborators within the past 4 years

S. Balay, Argonne National Laboratory (ANL); B. Smith, ANL.