

Final Report:
Symposium on Adaptive Methods for
Partial Differential Equations

Michael Pernice

Center for High Performance Computing
pernice@chpc.utah.edu

Christopher R. Johnson

Department of Computer Science
crj@cs.utah.edu

Philip J. Smith

Department of Chemical and Fuels Engineering
smith@opus.utah.edu

Aaron Fogelson

Department of Mathematics
fogelson@math.utah.edu

submitted December 10, 1998

US Department of Energy Contract DE-FG03-98ER25345

DISCLAIMER

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

DISCLAIMER

**Portions of this document may be illegible
in electronic image products. Images are
produced from the best available original
document.**

1 Overview

Complex physical phenomena often include features that span a wide range of spatial and temporal scales. Accurate simulation of such phenomena can be difficult to obtain, and computations that are under-resolved can even exhibit spurious features. While it is possible to resolve small scale features by increasing the number of grid points, global grid refinement can quickly lead to problems that are intractable, even on the largest available computing facilities. These constraints are particularly severe for three dimensional problems that involve complex physics. One way to achieve the needed resolution is to refine the computational mesh locally, in only those regions where enhanced resolution is required.

Adaptive solution methods concentrate computational effort in regions where it is most needed. These methods have been successfully applied to a wide variety of problems in computational science and engineering. Adaptive methods can be difficult to implement, prompting the development of tools and environments to facilitate their use. To ensure that the results of their efforts are useful, algorithm and tool developers must maintain close communication with application specialists. Conversely it remains difficult for application specialists who are unfamiliar with the methods to evaluate the trade-offs between the benefits of enhanced local resolution and the effort needed to implement an adaptive solution method.

These issues were discussed in a Symposium on Adaptive Methods for Partial Differential Equations that was hosted by the University of Utah June 22-24, 1998. Through the generous support of the Department of Energy Office of Energy Research, the Center for High Performance Computing, and the Center for the Simulation of Accidental Fires and Explosions, the symposium brought together an international group of producers and consumers of adaptive solution technology in a focused environment. The overall themes of the symposium were algorithms, tools, and applications. The symposium featured 20 presentations on methods for both Cartesian and triangulated grids and on the demanding applications being undertaken by partners in DoE's Accelerated Strategic Computing Initiative (ASCI). Scheduling conflicts and the added participation of the ASCI Alliance partners warranted a change in the planned conference dates and an increase in its length. The symposium also provided extensive opportunities for informal discussions on the current state of the field.

Two classes of adaptive solution methods were addressed at the symposium. *Structured adaptive methods* use locally regular rectangular grid patches to create refinement regions. By representing refinement regions

as unions of rectangular patches, structured methods incur a very low cost per gridpoint. This approach is quite suitable for models that employ finite difference and finite volume discretization, which are most frequently used on regular structured grids. Structured adaptive methods lend themselves readily to re-use of existing single-grid models, and simplify some grid generation issues. However the representation of refinement regions as a union of rectangular regions may produce added resolution in places where it is not needed.

Unstructured adaptive methods add grid points where further resolution is deemed necessary. This approach is most suitable for models that use finite volume or finite element discretization on triangulated grids, and is effective at increasing resolution precisely where it is needed. Unstructured adaptive methods must be implemented carefully to maintain the quality of the finite element grid in order to avoid generating poorly conditioned systems of linear equations which may be difficult to solve accurately. However unstructured adaptive methods can readily be incorporated into existing finite element solvers with modest enhancements to data structures.

While the idea of concentrating computational effort where it is needed through local grid refinement sounds attractive, realizing this objective in automatic, efficient, and robust software presents significant analytic, conceptual, and logistical challenges. Automatically refining a grid to resolve fine scale features of a solution requires reliable estimation of local errors. Presentations and discussion at the symposium revealed that analytically-based error estimators are known only for a small class of problems, and that even within this class a poorly chosen discretization will fool the best estimator. Local grid refinement requires the capability to both add and delete mesh points, which means that data structures must be dynamic and flexible, and that memory utilization must be carefully managed. Efficient parallel implementation of static, non-adaptive methods is difficult enough, and the additional considerations needed to address the dynamic nature of an adaptive computation (namely dynamic load balancing, scheduling data exchanges, and distributed data structures) further complicates matters. Finally, locally refined grids can be organized in a hierarchy, and efficient algorithms for solving problems on such grids exploit this hierarchical description and are multilevel in nature. Several different software environments that provide these capabilities were described. While still evolving, these frameworks reflect an interesting trend in general scientific and engineering software, away from procedure-based libraries and towards object-based toolkits.

Overall, the symposium revealed that, while significant progress in the

development and use of adaptive methods has occurred, much work remains to be done. By all accounts, the symposium successfully provided a useful forum for presentation and discussion of the state of the art. An online proceedings is being prepared and will shortly be available at the conference web site at

<http://www.chpc.utah.edu/~pernice/adaptive/symposium.html>.

2 Symposium Attendees

There were 57 registered attendees at the symposium, representing four national laboratories, five different nations, a dozen universities, the ASCI Alliance Partners, and faculty, postdoctoral researchers and graduate students from the University of Utah.

M. Ainsworth*	University of Leicester	ain@le.ac.uk
M. Berzins*	The University of Leeds	martin@scs.leeds.ac.uk
M. Bockelie	Reaction Engineering International	bockelie@reaction-eng.com
D. Balsara	University of Illinois	u10956@ncsa.uiuc.edu
K. Bogar	University of Utah	bogar@math.utah.edu
K. Devine*	Sandia National Laboratories	kddevin@sandia.gov
S. Domino	University of Utah	stefan@summit.che.utah.edu
A. Dubey	University of Chicago	dubey@oddjob.uchicago.edu
M. Dumett	University of Utah	dumett@math.utah.edu
T. Dupont	University of Chicago	dupont@cs.uchicago.edu
D. Eyre	University of Utah	eyre@math.utah.edu
J. Flaherty*	Rensselaer Polytechnic Institute	flaherje@cs.rpi.edu
A. Fogelson†	University of Utah	fogelson@math.utah.edu
L. Freitag Diachin*	Argonne National Lab	freitag@mcs.anl.gov
B. Fryxell	NASA/Goddard	fryxell@neutrino.gsfc.nasa.gov
W. Henshaw*	Los Alamos National Lab	henshaw@lanl.gov
R. Hornung*	Lawrence Livermore National Lab	hornung@llnl.gov
C. Johnson†	University of Utah	crj@cs.utah.edu
J.-G. Kim	University of Utah	jonggyung@eng.utah.edu
J. Keener	University of Utah	keener@math.utah.edu
R. Klepfer	University of Utah	rklepfer@cs.utah.edu
S. Kohn*	Lawrence Livermore National Lab	kohn1@llnl.gov
S. Kumar	University of Utah	skumar@talash.crsim.utah.edu
Y.-S. Lee	University of Utah	ylee@math.utah.edu
Z. Li*	North Carolina State University	zhilin@pams.ncsu.edu
Y. Livnat	University of Utah	ylivnat@cs.utah.edu

A. Malagoli*	University of Chicago	a-malagoli@uchicago.edu
D. Meiron*	California Institute of Technology	dim@cco.caltech.edu
G. Miller	University of Chicago	gh-miller@uchicago.edu
G. Miguez-Macho	University of Utah	gmiguez@atmos.met.utah.edu
C. Montgomery	Reaction Engineering International	montgomery@reaction-eng.com
W. Mitchell*	NIST/Gaithersburg	william.mitchell@nist.gov
Q. Nie	University of Chicago	qingnie@cs.uchicago.edu
E. Newren	University of Utah	newren@math.utah.edu
D.-P. Nguyen	University of Utah	Diem-Phuong.Nguyen@m.cc.utah.edu
M. E. Ong*	Lawrence Livermore National Lab	ong4@llnl.gov
M. Parashar	Rutgers University	parashar@caip.rutgers.edu
M. Pernice†	University of Utah	pernice@chpc.utah.edu
D. Quinlan*	Los Alamos National Lab	dquinlan@lanl.gov
J. Quirk*	California Institute of Technology	jjq@galcit.caltech.edu
J. Raquepas	University of Utah	raquepas@math.utah.edu
A. Roma*	Universidade de São Paulo	roma@ime.usp.br
U. Rüde*	Universität Augsburg	ruede@math.uni-augsburg.de
T. Shaw	University of Utah	shaw@math.utah.edu
M. Shephard*	Rensselaer Polytechnic Institute	shephard@scorec.rpi.edu
P.-P. Sloan	University of Utah	ppsloan@cs.utah.edu
P. Smith†	University of Utah	smith@opus.utah.edu
J. Spinti	University of Utah	spinti@loki.che.utah.edu
D. Swensen	University of Utah	swensen@reaction-eng.com
M. Tocci	Worcester Polytechnic Institute	mdtocci@wpi.edu
H. Walker	Worcester Polytechnic Institute	walker@wpi.edu
M. Wang	University of Utah	mwang@atmos.met.utah.edu
N.-T. Wang	University of Utah	nwang@math.utah.edu
D. Weinstein	University of Utah	dweinste@cs.utah.edu
W. K. Yee	University of Utah	wing.yee@m.cc.utah.edu
C.-S. Yi	University of Utah	yi@math.utah.edu
H. Yu	University of Utah	haoyu@math.utah.edu

3 Format and Content

The symposium consisted of 17 invited presentations on adaptive solution technology and three presentations on applications. There were no parallel sessions, and ample breaks gave participants numerous opportunities to in-

*Invited Speaker.

†Co-organizer.

teract outside the sessions. The symposium schedule is included in §3.1 and abstracts of the invited presentations appear in §3.5.

In addition the symposium featured several discussion periods during which conference attendees exchanged views on a variety of specific topic areas: strengths and weaknesses of structured and unstructured approaches, error estimation and strategies for automatic grid refinement, and toolkit support and requirements of applications. These discussions are summarized in §§3.2, 3.3 and 3.4.

3.1 Symposium Schedule

Monday, June 22 1998

9:00 – 9:30 Welcoming remarks

David W. Pershing

Dean of Engineering, University of Utah

Julio Facelli

Director, Center for High Performance Computing

Michael Pernice

Conference Co-organizer

9:30 – 10:00 **Joe Flaherty**

Adaptive and Parallel Computational Tools for Solving PDEs

10:00 – 10:30 **James Quirk**

AMRITA – Animal, Vegetable or Mineral?

10:30 – 11:00 **Break**

11:00 – 11:30 **Andrea Malagoli**

Chicago Alliance Center presentation

11:30 – 12:00 **Martin Berzins**

Parallel Unstructured Tetrahedral Mesh Algorithms for Transient Compressible and Reacting Flow Problems Using Domain Decomposition

12:00 – 1:30 **Lunch**

1:30 – 2:00 **Dan Meiron**

CalTech Alliance Center presentation

2:00 – 2:30 **Zhilin Li**

Introduction to the Immersed Interface Method and Recent Progress

2:30 – 3:00 **Mark Shephard**
Parallel Mesh Generation Using a Distributed Octree
3:00 – 3:30 **Break**
3:30 – 4:00 **Philip J. Smith**
Utah Alliance Center presentation
4:00 – 5:00 **Group Discussion: Strengths and Weaknesses of Structured and Unstructured Approaches**

Tuesday, June 23

9:00 – 9:30 **Scott Kohn**
SAMRAI: A Framework for Structured AMR
9:30 – 10:00 **Mark Ainsworth**
A posteriori Error Estimation for Finite Element Approximations
10:00 – 10:30 **Alexandre Roma**
An Adaptive Immersed Boundary Method: Implementation Issues, Applications, and Work in Progress
10:30 – 11:00 **Break**
11:00 – 11:30 **William Henshaw**
Overture: A Framework for Solving PDEs on Moving Geometries with Overlapping Grids
11:30 – 12:00 **Karen Devine**
Dynamic Load-Balancing Tools for Adaptive Computation
12:00 – 1:30 **Lunch**
1:30 – 2:00 **Ulrich Rüde**
FAMe – Fully Adaptive Multigrid (with Extrapolation
2:00 – 2:30 **Chris Johnson**
Computational Inverse Problems and Adaptive Methods
2:30 – 3:00 **Break**
3:30 – 4:30 **Group Discussion: Error Estimation and Strategies for Automatic Grid Refinement**
5:30 – 8:00 **Conference Banquet**
Santa Fe Restaurant

Wednesday, June 24

9:00 – 9:30 **Dan Quinlan**

Object-Oriented Parallel AMR

9:30 – 10:00	Maria Elizabeth Ong <i>Hierarchical Basis Methods</i>
10:00 – 10:30	William Mitchell <i>Full Domain Partitions to Reduce Communication in Parallel PDE Solvers</i>
10:30 – 11:00	Break
11:00 – 11:30	Rich Hornung <i>SAMRAI: Structured AMR Applications</i>
11:30 – 12:00	Lori Freitag <i>Unstructured Mesh Improvement Strategies</i>
12:00 – 1:30	Lunch
1:30 – 3:00	Group Discussion: Toolkit Support and Requirements of Applications
3:00	Symposium Adjourns

3.2 Strengths and Weaknesses of Structured and Unstructured Approaches

It is difficult to make a direct comparison of the costs of structured and unstructured approaches, since they are both dynamic in nature. While it is possible to discuss storage overhead in terms of bytes per element in an unstructured approach or bytes per refinement patch for a structured approach, the actual cost can only be determined at run-time and is highly problem-dependent. Different refinement criteria and regridding heuristics can lead to different meshes that are meant to resolve the same solution features.

Though some effort was made the conference participants were unable to characterize applications that favored one approach over the other. Applications involving complex geometry seem to favor unstructured approaches. However structured methods are making progress toward automating mesh generation. New approaches such as overset grids and embedded boundary methods seem promising but require significantly more work for them to be fully automatic. To date no systematic comparison of structured and unstructured methods applied to realistic problems has been conducted. The conference participants agreed that more experience is needed working with the most complex problems before the strengths and weaknesses of the two approaches could be fully assessed.

Few participants have worked with both approaches; the effort required

to do so remains daunting. This is a reflection of the broad range of expertise that must be brought to bear in order to achieve the benefits of local mesh refinement. Discussion revealed a communication gap in easily acquiring knowledge of advances in each approach. Application developers learned of advances in using Godunov methods on unstructured grids. The problem of dynamically partitioning unstructured grids seems largely solved. New approaches such as hierarchical grid generation and aggregation techniques have advanced the application of multigrid solution techniques on unstructured grids. The creation of an AMRnet was suggested but seemed to draw a lukewarm response. Options such as an MGnet-subnet or creation of an AMR resource page will be explored. One existing site that was identified was the Adaptive Mesh Refinement for Astrophysics site at <http://www.camk.edu.pl/~tomek/AMRA/amr.html>. If there is sufficient interest resources to support the effort will be sought.

3.3 Error Estimation and Strategies for Automatic Grid Refinement

Though progress is being made, reliable and robust error estimation is not available for most problems of interest. While the study of a posteriori error estimation for elliptic problems is relatively mature, the analysis of hyperbolic and parabolic problems is still evolving. Reliability remains a problem even when such estimates are available, since an inappropriate discretization will break any error estimator. In the best cases there are some heuristics that can be used to assist the estimators, but these are highly problem-dependent. Analysis of discretization errors for complex problems of interest (e.g. reacting flow) is lacking, but the complexity of the governing equations makes this extremely difficult. The advances that are needed require a multidisciplinary effort, and interactions among mathematicians, computer scientists, and engineers should be further encouraged.

The lack of reliable error estimators for complex problems has implications for model validation. Solutions using several different methods should be obtained and results should be compared. Significant disagreement may indicate inappropriate models. For multiphysics problems validation should include the coupling between sub-models as well as the individual component models.

r-refinement (nodal relocation) was identified as a useful strategy that is complementary to both structured and unstructured methods.

3.4 Toolkit Support and Requirements of Applications

This final discussion session was focused on what type of toolkit support was desirable and what features were needed by applications. The first criterion seemed to be ease of use. The focus quickly turned to the topic of the effort that was needed to create something that was truly easy to use and yet flexible enough to encourage experimentation. It is also difficult to craft a toolkit that is both sophisticated and broadly applicable, and often the latter must be sacrificed in favor of the former. See, for example the FEMLAB site at <http://www.math.chalmers.se/Math/Research/Femlab/>, which only includes modules for initial value problems, two-point boundary value problems, and convection-diffusion problems in two dimensions. The lack of readily-used toolkits inhibits the adoption of adaptive mesh technology in new application areas and makes it difficult to directly compare different approaches. It also inhibits experimentation with new approaches. Everyone was encouraged to improve code portability, robustness and documentation, and to make the results of their efforts available to the community.

3.5 Abstracts

Following are abstracts of the talks delivered by the invited speakers. Since the symposium encompassed both structured and unstructured methods, the talks were arranged to interleave presentations on each approach with descriptions of applications that would benefit from the use of adaptive methods. Where possible the abstracts include speakers' addresses and home page. Not included among the abstracts are presentations given by representatives of the ASCI Alliance Partners from California Institute of Technology, the University of Chicago, and the University of Utah on some of the most demanding and challenging applications currently under investigation.

Adaptive and Parallel Computational Tools for Solving Partial Differential Equations

Joseph E. Flaherty
Department of Computer Science
Rensselaer Polytechnic Institute
Troy, NY 12180
flaherje@cs.rpi.edu
<http://www.cs.rpi.edu/~flaherje>

Abstract

We are developing a framework for adaptive and parallel computation that is capable of (i) generating three-dimensional unstructured meshes of tetrahedral elements, (ii) automatically refining and coarsening these meshes, (iii) partitioning the computation into subdomains that may be processed in parallel, and (iv) maintaining a balanced parallel computation through element migration. Parallel tools for handling computation on heterogeneous computers involve descriptions of the problem at partition, process, and computer levels. Partitioning based on an octree decomposition of the space-time domain of the partial differential system is efficient and is linked to our mesh generation procedures. Predictive strategies for anticipating load imbalances reduce migration times and provide partitions that maintain balance for a greater portion of the solution process. Applications involving transient compressible flows will be presented.

AMRITA—Animal, Vegetable, or Mineral?

James J. Quirk
Graduate Aeronautical Laboratories
California Institute of Technology
Pasadena, CA 91125
jjq@galcit.caltech.edu
<http://www.galcit.caltech.edu/~jjq>

Abstract

AMRITA was originally developed as an interactive teaching aid to allow students to explore the practical aspects of compressible, computational fluid dynamics using an adaptive mesh refinement algorithm. Hence – Adaptive Mesh Refinement Interactive Teaching Aid. Over time, however, AMRITA's mandate has become far broader and so its name is now best taken at face value[†].

In the context of this workshop, AMRITA can be viewed as a software system for automating numerical investigations. The system is driven using its own powerful scripting language, *Amrita*, which facilitates both the composition and archiving of complete numerical investigations, as distinct from isolated computations. Once archived, an AMRITA investigation can later be reproduced by any interested party, and not just the original investigator, for no cost other than the raw CPU time needed to parse the archived script. Automation is seen as the key to improving numerical reliability, repeatability and productivity to the point where software-bound algorithms could be improved through mass scrutiny.

In this lecture I will present an overview of the general needs and concerns which shaped AMRITA's development from common-or-garden CFD code to computational operating system. A more detailed answer to the question posed in the title of this talk can be found at <http://www.amrita-cfd.org>.

[†]By coincidence amrita (*am-rē-tā*) also happens to be the drink of the Hindu gods!

Parallel Unstructured Tetrahedral Mesh Algorithms for Transient Compressible and Reacting Flow Problems Using Domain Decomposition

Martin Berzins and Paul Selwood

Computational PDEs Unit
School of Computer Studies
The University of Leeds
Leeds LS2 9JT, UK
martin@scs.leeds.ac.uk

<http://agora.scs.leeds.ac.uk/staff/martin.html>

Abstract

The solution of transient compressible flow problems with chemical reactions using unstructured tetrahedral meshes will be considered. The approach described makes of adaptive meshes in both space and time, with mesh refinement guided by local error estimators.

The spatial discretization methods used are based on cell-vertex and cell-centered finite volume schemes. The issue of what constitutes an appropriate mesh will be considered. Time integration is using standard o.d.e. integration algorithms carefully modified to avoid excessive computation in the solution of the nonlinear equations. The application of the algorithm to problems in atmospheric dispersion will be described.

The need for computational efficiency and reliability has made it necessary to adopt a parallel mesh adaptation approach. Serial and Parallel algorithms for the refinement of tetrahedral meshes will be described. In the parallel case an MPI-based approach using both standard load balancing tools and a new method is employed.

The merits of the different tools will be discussed as will the scalability of the mesh adaptation and the underlying flow solver. Computational experiments on a variety of parallel machines will be used to illustrate both scalability and portability.

A key issue in the use of such methods on parallel computers is to make parallel programming of such complex algorithms easier while maintaining

the efficiency that can be achieved using message passing. A programming abstraction to help achieve this will be briefly outlined.

In order to make such codes easier to use we will briefly describe how state-of-the art problem solving environments such as SCIRUN developed at the University of Utah may be used to transform the user's view of the solution process.

Acknowledgements. The work described is joint with A. Tomlin and S. Ghorai (reacting flow), Jon Nash, P. K. Jimack and N. Touheed (parallel computing), at Leeds and C. R. Johnson, L. Zhukov and R. Coffey (SCIRUN) at Utah.

Introduction to the Immersed Interface Method and Recent Progress

Zhilin Li

Center for Research in Scientific Computation

Department of Mathematics

North Carolina State University

Raleigh, NC 27695

zhilin@pams.ncsu.edu

<http://www4.ncsu.edu/~zhilin>

Abstract

Many physical problems involve interfaces. Examples include phase transition problems where the interface separates the solid and liquid regions, bubble simulation, Hele-Shaw flow, composite materials, and many other important physical phenomena. Mathematically, interface problems usually lead to differential equations whose input data and solutions have discontinuities or non-smoothness across interfaces. As a result, many standard numerical schemes do not work or work poorly for interface problems.

In the Immersed Interface Method, we use Cartesian grids and modify finite difference methods instead of using adaptive or body fitting grids to resolve the solutions near or on interface. In this talk, I will briefly explain how the Immersed Interface Method is used to solve certain partial differential equations; the advantages and problems associated with the method; recently advances in this area, particularly, the finite element version of the method which guarantees the stability and convergence of the method; and applications if time permits.

Parallel Mesh Generation Using a Distributed Octree

Mark S. Shephard
Scientific Computation Research Center
Rensselaer Polytechnic Institute
Troy, NY 12180-3590
shephard@scorec.rpi.edu
<http://www.mech.e.rpi.edu/faculty/member/shephard.html>

Abstract

Scalable parallel computers are allowing solutions on meshes of tens of millions of elements. As mesh sizes become this large, automatic mesh generation on a serial computer becomes problematic in terms of computational effort and memory requirements. The scalable parallel mesh generator presented here is capable of working directly from solid models. The parallel mesh generator builds upon a set of underlying structures that also effectively support parallel adaptive analysis [2, 5, 6]. Key to an effective parallel mesh generator is balancing the computational effort among the processors while controlling interprocessor communications. Since the computational effort required depends on the distribution of elements, which is not well known till the mesh is generated, maintaining load balance is difficult. A distributed octree structure is used to obtain estimates of the computational work load and to distribute the effort to maintain load balance while keeping interprocessor communications to a minimum. Since knowledge of the computational effort is only determined as the meshing proceeds, the redistribution of the octree is used to maintain load balance during mesh generation.

Effective parallelism demands that the octree be distributed and able to redistribute work load as the mesh evolves and better estimates of work load are known. To meet these needs, a parallel octree data structure coupled with octant migration procedures has been designed. With this design, the storage and work load associated with a set of octants can be distributed across the processors in a MIMD environment [8, 7]. The distributed octree includes links across processors, allowing an octant to have its parent and/or children residing on other processors. An octant migration procedure allows

arbitrary redistribution of octants across the processors while maintaining the interconnections between the octants. As the mesh evolves and more accurate estimates of workload are obtained, migration procedures can be invoked to repartition the domain of interest such that the processors are load balanced.

The parallel mesh generator is responsible for automatically creating the mesh in parallel with the mesh distributed to the various processor memories as it is generated [3, 4, 1, 6, 8, 7]. The parallel mesh generator and all component tools are implemented using MPI to maximize portability. The first step in the parallel mesh generator is obtaining workload estimates for the octree partitioner. The cost associated with a given octant is the anticipated number of elements to be generated. The approximate number of elements required can be estimated from the mesh sizes and model entity sizes. Once the surface mesher [3, 1] computes the approximate number of elements required to mesh a model face, it gives the results to the octree partitioner [8, 7]. The partitioner refines the tree until the meshing load can be estimated. It then assigns octants to processors, and finally migrates the octants to their assigned processors. The octree created by the octree partitioner is only refined and partitioned to a level where the load can be reasonably balanced across the set of processors.

Once the partition tree is constructed, model faces are assigned to the processor which contains the largest portion of its surface area in the processor's partition. Each model face is then meshed by an individual processor in a hierarchical fashion. Mesh faces are then migrated to the processors which contain their centroid so that they may be inserted into the octree for localization. Efficient localization during mesh generation requires that each terminal octant be roughly the same size as its containing mesh entities. Thus, each processor will continue to refine its portions of the partition tree asynchronously. Scalability of the parallel surface meshing procedure has been guaranteed by adding a model face subdivision capability.

Given the domain's surface mesh and a partitioned octree whose terminal octants have been refined to the approximate size as their containing mesh faces and edges, the region mesh is generated by (i) adjusting the octree for proper gradation of element sizes throughout the domain, (ii) pre-processing the octree so that each terminal octant knows about all the mesh faces intersecting their volume, (iii) classifying the terminal octants as either interior, exterior, or boundary octants, (iv) applying pre-defined meshes or templates to interior terminal octants of the octree, (v) setting inter-processor links between the mesh entities created on the partition boundaries, and (vi) connecting the surface triangulation to the interior mesh coming from the

application of templates via an advancing front procedure.

Key to efficient parallel volume meshing is the repartitioning of the domain remaining to be meshed after a face removal step. The original algorithm to perform the repartitioning was a distributed parallel inertial bisection algorithm [1]. Scalability is not an issue here since, as long as partitions are bulky enough, it can be guaranteed. Efficiency was an issue. A new approach that explicitly considers the interprocessor faces, the interprocessor edges, and the interprocessor vertices has been developed [1]. This procedure is substantially more efficient.

References

- [1] H. L. DE COUGNY, *Distributed Parallel Mesh Generation*, PhD thesis, Scientific Computation Research Center, RPI, Troy, NY, 1998.
- [2] H. L. DE COUGNY, K. D. DEVINE, J. E. FLAHERTY, R. M. LOY, C. OZTURAN, AND M. S. SHEPHARD, *Load balancing for the parallel solution of partial differential equations*, Applied Numerical Mathematics, 16 (1995), pp. 157–182.
- [3] H. L. DE COUGNY AND M. S. SHEPHARD, *Surface meshing using vertex insertion*, in 5th Int. Meshing Roundtable '96, Sandia Corp., 1996, pp. 243–256. SAND96-2301.
- [4] H. L. DE COUGNY, M. S. SHEPHARD, AND C. OZTURAN, *Parallel three-dimensional mesh generation on distributed memory mimm computers*, Eng. with Comp., 12 (1996), pp. 94–106.
- [5] M. S. SHEPHARD, J. E. FLAHERTY, C. L. BOTTASSO, H. L. DE COUGNY, C. OZTURAN, AND M. L. SIMONE, *Parallel automated adaptive analysis*, Parallel Computing, 23 (1997), pp. 1327–1347.
- [6] M. S. SHEPHARD, J. E. FLAHERTY, H. L. DE COUGNY, C. L. BOTTASSO, AND C. OZTURAN, *Parallel automatic mesh generation and adaptive mesh control*, in Solving Large Scale Problems in Mechanics: Parallel and Distributed Computer Applications, M. Papadrakakis, ed., John Wiley & Sons, Ltd, Chichester, 1997, pp. 459–493.
- [7] M. L. SIMONE, H. L. DE COUGNY, AND M. S. SHEPHARD, *Tools and techniques for parallel grid generation*, in Numerical Grid Generation in Computational Field Simulations, B. K. Soni, J. F. Thompson,

J. Hauser, and P. R. Eisemann, eds., vol. II, NSF Research Center for Computational Field Simulation, Mississippi State, MS, 1996, pp. 1165–1174.

[8] M. L. SIMONE, R. M. LOY, M. S. SHEPHARD, AND J. E. FLAHERTY, *A distributed octree and algorithms for parallel mesh generation*, Tech. Rep. SCOREC Rep. No. 23-1996, Scientific Computation Research Center, RPI, 1997. To be submitted for publication.

SAMRAI: A Framework for Structured AMR

Scott Kohn

Center for Applied Scientific Computing

Lawrence Livermore National Laboratory

P.O. Box 808, L-561

Livermore, CA 94551

kohn1@llnl.gov

<http://www.llnl.gov/CASC/people/kohn>

Abstract

SAMRAI is an object-oriented framework that provides computational scientists with general and extensible software support for the prototyping and development of parallel structured adaptive mesh refinement (AMR) applications. The primary goal of the SAMRAI effort is to facilitate numerical and algorithmic exploration for applications that require complicated numerical schemes, complex data structures, and high performance computing hardware.

Achieving software flexibility and extensibility in a high performance computing environment requires careful attention to the design of the framework architecture. In particular, SAMRAI makes extensive use of object oriented techniques and various design patterns, such as Abstract Factory, Strategy, and Chain of Responsibility. In this talk, we will give an overview of the SAMRAI architecture, design principles, and parallelization strategy.

Acknowledgement. This is joint work with Xabier Garaizar and Richard Hornung.

A Posteriori Error Estimation for Finite Element Approximation

Mark Ainsworth

Department of Mathematics and Computer Science

University of Leicester

University Road

Leicester LE1 7RH, UK

ain@le.ac.uk

<http://www.mcs.le.ac.uk/~mainsworth>

Abstract

The talk will survey recent developments in a posteriori error estimation for adaptive finite element computations. Topics to be discussed include: the equilibrated residual method, selection of subspaces for solving local residual problems, robust error estimation for singularly perturbed problems and error estimation in the presence of locking.

An Adaptive Immersed Boundary Method: Implementation Issues, Applications and Work in Progress

Alexandre M. Roma
Universidade de São Paulo
Instituto de Matemática e Estatística
Caixa Postal 66281 (Agência Cidade de São Paulo)
São Paulo, SP 05315-970, Brazil
roma@ime.usp.br

Abstract

Many problems in biofluid dynamics involve the interaction between a nonstationary, incompressible viscous fluid and a visco-elastic biological tissue, which may have time-dependent configuration, time-dependent elastic properties, or both (e.g., the interaction between blood, heart muscles and heart valve leaflets). Although these problems can be handled in a robust manner by the *Immersed Boundary Method* and qualitatively good results be obtained, this method suffers from a certain “lack of resolution” which is related to limitations of computers such as speed and storage. Finer immersed boundary geometric details and flow features can be adequately resolved only if the computational mesh is dense. If a uniform mesh is used, this requirement is inevitably extended to the entire computational domain, and the resulting mesh may exceed the storage capacity of the computer. A computational setting for the Immersed Boundary Method which employs the *Adaptive Mesh Refinement Technique* will be presented. This approach is capable of removing the original uniform mesh restriction, enhancing the accuracy of the method by covering locally an immersed boundary vicinity with a sequence of nested, progressively finer rectangular grid patches which dynamically follow the immersed boundary motion.

Results obtained for a two-dimensional model problem show that, with this approach, the accuracy attained by refining only an immersed boundary vicinity is the *same* as if the whole computation had been performed on an uniform mesh with the resolution of the *finest level* used in the locally refined mesh. Implementation issues, applications in sight and the work in progress will be discussed.

Overture: A Framework for Solving Partial Differential Equations on Moving Geometries with Overlapping Grids

William Henshaw
Scientific Computing Group
Computing, Information, and Communications Division
Los Alamos National Laboratory
Los Alamos, NM 87545
henshaw@lanl.gov
<http://www.c3.lanl.gov/~henshaw/home.html>

Abstract

The Overture Framework is an object-oriented environment for solving PDEs in two and three space dimensions. It is a collection of C++ libraries that enables the use of finite difference and finite volume methods at a level that hides the details of the associated data structures. Overture can be used to solve problems in complicated, moving geometries using the method of overlapping grids. It merges geometry, grid generation, difference operators, boundary conditions, data-base access and graphics into an easy to use high level interface. This talk will describe the current status of the framework, including our work on incompressible and all-speed flow solvers.

Dynamic Load-Balancing Tools for Adaptive Computation

Karen D. Devine
MS 1111
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-1111
kddevin@sandia.gov
<http://www.cs.sandia.gov/~kddevin>

Abstract

Dynamic load balancing is an important component of many parallel, adaptive computations. Adaptive finite element methods, for example, create load imbalance dynamically by changing the number of degrees of freedom in regions of the domain. The requirements for dynamic load-balancing algorithms extend beyond those of static partitioning techniques used by traditional finite element methods. Because a dynamic load-balancing tool runs side-by-side with an application, it must run quickly in parallel and use little memory and communication. In addition, to minimize the cost of moving data, a dynamic load-balancing technique should account for data's current processor assignment in determining the data's location in the new decomposition.

While several general tools for static decomposition are available (e.g., Chaco [2] and Metis [3]), dynamic load-balancing algorithms have typically been implemented as part of a single application or closely related set of codes. This fact is due to the complexity of dynamic load-balancing tools relative to static partitioners. Static partitioners typically use file-based interfaces to applications. Dynamic load-balancing tools, however, must be designed as subroutines callable by the application. Thus, they must have simple interfaces to make them usable by a variety of applications. In addition, their data structures must be independent of those of the applications. This restriction adds the cost of building separate data structures for load-balancing, but allows clean abstraction of the load balancing from the application.

In this talk, we describe our development of a dynamic load-balancing library. The object-oriented interface and design are presented. The first

target applications are adaptive finite element methods, but the interface supports other problems, such as particle simulations and contact-detection methods. We present results for mesh refinement and mesh sequencing with dynamic load balancing in MPSalsa [1], a finite element program that solves coupled three-dimensional fluid flow and detailed reaction chemistry systems for modeling chemically reacting flow.

References

- [1] K. DEVINE, G. HENNIGAN, S. HUTCHINSON, A. SALINGER, J. SHADID, AND R. TUMINARO, *High performance MP unstructured finite element simulation of chemically reacting flows*, in Proceedings of Supercomputing 97, 1997. <http://www.supercomp.org/sc97/proceedings/BELL/DEVINE/INDEX.HTM>.
- [2] B. HENDRICKSON AND R. LELAND, *The Chaco user's guide, version 2.0*, Tech. Rep. SAND95-2344, Sandia National Laboratories, Albuquerque, NM, 1995. <http://www.cs.sandia.gov/CRF/chac.html>.
- [3] G. KARYPIS AND V. KUMAR, *Metis: A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices*, University of Minnesota, 1997. <http://www-users.cs.umn.edu/~metis/metis/main.html>.

FAMe – Fully Adaptive Multigrid (with extrapolation)

Ulrich Rüde
Institut für Mathematik
Universität Augsburg
D-86135 Augsburg, Germany
rude@math.uni-augsburg.de
<http://scicomp.math.uni-augsburg.de/rude>

Abstract

Fully adaptive multigrid is a framework combining efficient multilevel solvers with a natural error estimator and mesh adaptivity. Unlike in most other iterative methods, the order of processing and sequence of operations is not determined a-priori. The core of the method consists of a relaxation scheme with an active set strategy and can be viewed as an efficient implementation of the Gauss-Southwell relaxation. The flexibility in choosing the order of operations can also be used to derive nonstandard parallel implementations. If used in a multilevel setting, the algorithm can be shown to have asymptotically optimal convergence rate. One advantage of the method compared to plain multigrid is an improved robustness and efficiency for nontrivial problems, including those with non-smooth coefficients or structural singularities.

Computational Inverse Problems and Adaptive Methods

Chris Johnson
Department of Computer Science
University of Utah
Salt Lake City, Utah 84112
`crj@cs.utah.edu`
<http://www.cs.utah.edu/~crj/home.html>

Abstract

Most adaptive methods focus on so-called "forward" or "direct" solutions to partial differential equations. For the most part, adaptive methods for large-scale inverse problems have been largely overlooked. In this talk, I will present an overview of the issues encountered involving adaptive methods for large-scale inverse problems, focusing mainly on inverse bioelectric field imaging problems.

Object-Oriented Parallel Adaptive Mesh Refinement

Dan Quinlan
Scientific Computing Group
Computing, Information, and Communications Division
Los Alamos National Laboratory
Los Alamos, NM 87545
dquinlan@lanl.gov
<http://www.c3.lanl.gov/~dquinlan>

Abstract

Adaptive Mesh Refinement (AMR) computations are complicated by their dynamic nature. In the serial environment they require substantial infrastructures to support the regridding processes, intergrid operations, and local bookkeeping of positions of grids relative to one another. In the parallel environment the dynamic behavior is more problematic because it requires dynamic distribution support and load balancing. Parallel AMR is further complicated by the substantial task parallelism, in addition to the obvious data parallelism, this task parallelism requires additional infrastructure to support efficiently. The degree of parallelism is typically dependent upon the algorithms in use and the equations being solved. Different algorithms have significant compromises between computation and communication. Substantial research work is often required to define efficient methods and suitable infrastructure. The purpose of this talk is to introduce AMR++ as an object-oriented library which forms a part of the OVERTURE framework, a much larger object-oriented numerical framework developed and supported at Los Alamos National Laboratory and distributed on the Web for the last several years. A particular focus of this talk is upon the parallel aspects of AMR for elliptic solvers. Specifically, we will present the current status of our research on parallel elliptic solvers for use within the AMR++ library.

Hierarchical Basis Methods

Maria Elizabeth Ong
Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
P.O. Box 808, L-153
Livermore, CA 94551
ong4@llnl.gov
<http://www.llnl.gov/CASC/people/ong>

Abstract

Hierarchical basis methods for second order elliptic problems will be discussed. We begin with a discussion of the implementation and convergence of the methods using piecewise linear finite elements on a quasi-uniform and non-uniform mesh. We then discuss variants of the methods, including the stabilized version which leads to a wavelet-like discretization with better convergence properties, and the use of higher order polynomials to improve the accuracy of the solution.

Full Domain Partitions to Reduce Communication in Parallel PDE Solvers

William F. Mitchell
Mathematical and Computational Sciences Division
National Institute of Standards and Technology
Gaithersburg, MD 20899
william.mitchell@nist.gov
<http://math.nist.gov/mcsd/Staff/WMitchell/index.html>

Abstract

Adaptive multilevel methods, which combine adaptive mesh refinement with multigrid solution techniques, have been shown to be very efficient methods for the numerical solution of partial differential equations on sequential computers. The use of these methods on parallel computers is currently a research topic. In particular, effective use of these techniques in a high-latency/low-bandwidth environment, like a network of workstations, is especially challenging. In this talk we will present a parallel adaptive multilevel method for elliptic partial differential equations that obtains high parallel efficiency in this environment. Recent developments in parallelizing the method via overlapping subdomains on each refinement level will be featured. The approach produces a full domain partition, in which the usual subdomain on each processor is extended to cover the full domain. This facilitates parallel algorithms for adaptive refinement, grid partitioning and multigrid solution that require only a few communication steps each.

SAMRAI: Structured AMR Applications

Richard Hornung
Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
P.O. Box 808, L-561
Livermore, CA 94551
hornung@llnl.gov
<http://www.llnl.gov/CASC/people/hornung>

Abstract

We present several preliminary applications that demonstrate the applicability of the SAMRAI framework to different problem domains. These include shock hydrodynamics, shear band formation in granular materials, and an energy balance flow equation for porous media. By way of the hydrodynamics application, we will present the development of a traditional AMR application (in the sense of Berger and Oliger) in SAMRAI. This example will be used to demonstrate how an application developer can interact with the framework. Additionally, we will include discussion of arbitrary multi-block domains and modeling complex embedded internal boundaries. Through the shear band problem, we will explore the extension of the algorithmic components in SAMRAI and the use of the framework to implement user-defined data types. The energy balance problem illustrates preliminary work in combining SAMRAI with other software packages, in this case the KINSol nonlinear solver package and the HYPRE linear solver library (both are under development in the Center for Applied Scientific Computing at LLNL).

Acknowledgement. This is joint work with Xabier Garaizar and Scott Kohn.

Unstructured Mesh Improvement Strategies

Lori Freitag
Mathematics and Computer Science Division
Argonne National Laboratory
9700 S. Cass Avenue
Argonne, IL 60439
freitag@mcs.anl.gov
<http://www-unix.mcs.anl.gov/~freitag/index.html>

Abstract

In this talk I will present two aspects of our research on scalable algorithms for unstructured mesh improvement strategies, including adaptive refinement, edge and face swapping, and node point smoothing. The first aspect I will discuss pertains to the effect of mesh quality on solution efficiency for both finite element and finite volume applications. I will review theoretical results available in the literature that relate mesh distortion to the spectrum of the resulting linear system and, therefore, the expected efficiency of the linear solver. I will then present a series of experiments that quantify the tradeoffs between the cost of mesh improvement with the increased efficiency of a number of preconditioned conjugate gradient and GMRES solvers. For simple finite element applications, we compare our results to the theoretical predictions, and, for more complex applications, we show that performing mesh improvement techniques is beneficial, even when starting with a fairly good mesh.

I will then discuss the general framework we have developed for parallel implementation of these mesh improvement strategies. This framework is based on defining, for each algorithm, an elemental operation and a task graph. By choosing the correct task graph, we can ensure the correct parallel execution of the algorithms independent of implementation. Experimental results obtained on the Argonne IBM SP and the ASCI systems are presented to show the scaling and relative costs of these algorithms.