ECCM 2010
IV European Conference on Computational Mechanics
Palais des Congrès, Paris, France, May 16-21, 2010

Multilevel preconditioners for simulations and optimization on dynamic, adaptive meshes

E. de Sturler[1], G.H Paulino[2], and S. Wang[3]

[1] *Department of Mathematics, Virginia Tech, USA, sturler@vt.edu*
[2] *Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, USA, paulino@illinois.edu*
[3] *Energy Trading and Marketing Group, Credit Suisse, New York, wangshun98@gmail.com*

We present adaptive preconditioners for parallel, time-dependent simulations and nonlinear optimization problems with dynamic mesh adaptation. Adaptive meshing greatly reduces the computational cost of simulations and optimization. Unfortunately, it also carries a number of problems for preconditioning in iterative linear solvers, as changes in the mesh lead to structural changes in the linear systems we must solve. As a result, a new preconditioner must be computed after every change in the mesh, which might be prohibitively expensive. Here, we propose preconditioners that are cheap to update for dynamic changes to the mesh as well as for changes in the matrix due to nonlinearity of the underlying problem; more specifically, we propose preconditioners that require only *local changes to the preconditioner* for *local changes in the mesh and nonlinear terms*. Our preconditioners combine sparse approximate inverses with multilevel correction [1] and [2].

For adaptive mesh refinement (AMR), especially on parallel machines, the computational domain is usually partitioned into many small blocks, each consisting of a small number of mesh cells representing locally a uniform mesh. The refinement and derefinement of the mesh lead to structural changes in the system matrix. Moreover, to maintain a good load balance in parallel implementations, the mesh blocks are redistributed over the processors after (each) mesh refinement or derefinement, even on processors where the mesh did not change. We need preconditioners that accommodate the frequent changes in the mesh and the data redistributions. Unfortunately, these consequences of AMR make many popular preconditioners unfavorable. Preconditioners that depend explicitly on the matrix and the matrix ordering, such as incomplete factorizations like ILU and IC, are hard to update for structural changes to the matrix. Even localized changes in the system matrix generally affect the factorization of many rows and columns. In addition, the forward and backward substitution in these preconditioners lead to high synchronization costs in parallel implementations, and the redistribution of mesh blocks for load balancing tends to thwart techniques to mitigate these costs. Domain decomposition preconditioners also appear less suitable for AMR, if the frequency of mesh adaptation is relatively high and regions with high mesh resolution traverse the computational domain. The decomposition into subdomains will change frequently, so that local factorizations, coarse grid solvers, and/or Schur complement preconditioners need to be recomputed often.

A good preconditioner for AMR should have the following properties. Computing or updating the preconditioner should require only local information from the mesh and the discretization (method), local updates of the preconditioner should be sufficient to maintain quality, and such local changes should be cheap. Finally, the cost of multiplying with the preconditioner should be relatively insensitive to redistribution of the mesh blocks. Explicit Sparse Approximate Inverses (SAI) satisfy these requirements. Each column of the preconditioner depends only on the mesh in the immediate neighborhood of the mesh cell with which the column is associated. Indeed, the use of ghost cells for each mesh block can limit updates of the preconditioner to new mesh cells only (for linear PDEs). This makes updating the approximate inverse very cheap. Moreover, since SAI are explicitly available in matrix form, the redistribution of mesh blocks does not seriously affect the cost of the matrix-vector product.

Unfortunately, SAI do not approximate the inverse well for the smooth, global components of the solution that are often important in elliptic (type) problems, leading to slow convergence. We propose to remedy this problem at low cost by combining SAI with multilevel corrections using SAI at coarser meshes. We can do this efficiently by exploiting the hierarchical nature of AMR meshes. This leads to an approach that is highly efficient in computing and updating the preconditioner and highly effective in reducing the number of iterations of a Krylov subspace method. In addition, our preconditioners yield significant runtime reductions for time-dependent problems and optimal design using topology optimization. Finally, we have experimentally demonstrated level-independent convergence rates for the time-dependent problem and near level-independent convergence rates for topology optimization problems. As an example, we give convergence and timing results for a two-dimensional convection-diffusion problem with large jumps in the velocity field and the diffusion coefficient using adaptive meshing [1]. We also show the solution and mesh for two, representative, subsequent time steps.

Table 1: Convergence and timing results for a convection-diffusion problem.

time step	1	2	3	4	5
$\ell_{\max}$	5	6	7	8	8
n	4096	6208	12064	23056	32848
	convergence (niters)				
No Preconditioner	832	692	1270	3985	10051
One-level SAI	140	125	169	249	342
Two-level SAI	90	80	84	98	92
Full multi-level SAI	22	18	19	21	21
	timing (secs)				
No Preconditioner	9.47	12.00	41.88	248.78	882.01
One-level SAI	2.65	3.62	9.23	25.83	49.64
Two-level SAI	3.08	3.97	7.30	15.55	19.95
Full multi-level SAI	1.25	1.72	3.46	7.35	9.83
Update SAI on all levels	0.14	0.69	0.22	0.36	0.32

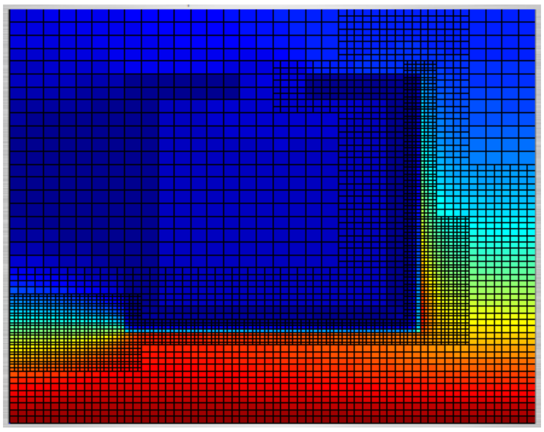

Figure 1: Distribution of pollutant before time step on adapted mesh

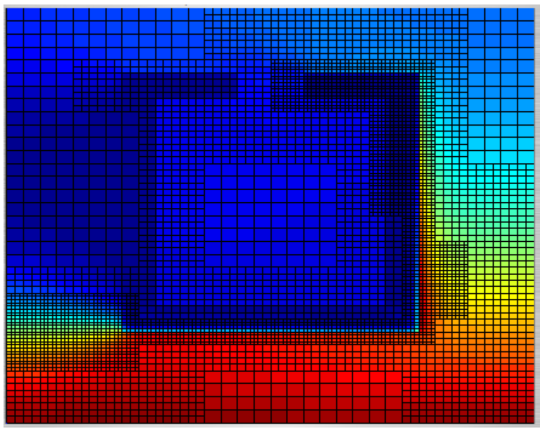

Figure 2: Distribution of pollutant after time step on adapted mesh

References

[1] Shun Wang and Eric de Sturler. Multilevel sparse approximate inverse preconditioners for adaptive mesh refinement. *Linear Algebra Appl.*, 431:409–426, 2009.

[2] Shun Wang. *Krylov subspace methods for topology optimization on adaptive meshes*. PhD thesis, University of Illinois at Urbana-Champaign, Department of Computer Science, September 2007. Advisor: Eric de Sturler, Co-Advisor: Glaucio H. Paulino.