Energy-Minimizing AMG for PDE Systems

J. Gaidamour, J. Hu, C. Siefert, and R. Tuminaro (consultant)

Many Sandia parallel simulations rely on the Trilinos framework for the repeated implicit solution of linear systems. One crucial linear solver is ML, whose primary methods are based on smoothed aggregation (SA) algebraic multigrid. A successful multigrid solver like ML accelerates the solution of the linear system of interest by employing related coarser (smaller) linear systems that must capture certain near-kernel components of the fine level system. SA has a number of known weaknesses, however. First, the cost to apply SA can grow unacceptably for systems of PDEs and for large number of near-kernel components. This problem is exacerbated by anisotropic phenomena and higher dimensionality. Second, SA also has rigid requirements for the sparsity pattern of the matrices that transfer data between coarse matrices.

In this CSRF project we have developed a family of novel energy-minimizing coarsening algorithms that address the aforementioned shortcomings in SA. These new algorithms have the flexibility to take any sparsity pattern for the transfer matrices. This allows us to choose patterns which are tailored for anisotropic problems, for instance. In addition, the cost of applying these algorithms is lower than SA because the number of degrees of freedom per grid point does not grow with the number of near-kernel components. Our algorithms still achieve accurate interpolation of the near nullspace components while minimizing the energies of the transfer matrices. A scaling study of iteration counts and complexities (cost to apply the preconditioner) for SA and energy minimization is summarized in Table 1. The test problem is 3D linear elasticity on a logically rectangular mesh. The stretching factor in the x direction is given by ϵ. For the isotropic case, energy minimization is cheaper to apply (meaning faster run times) than SA but has comparable convergence rates. For the anisotropic cases, energy minimization is cheaper and converges faster than SA. As an example, the new method is about 2.5 times faster than SA for $\epsilon = 100$ on the 40^3 mesh.

Mesh	$\epsilon = 1$				$\epsilon = 10$				$\epsilon = 100$			
	SA		Emin		SA		Emin		SA		Emin	
10^3	6	*1.32*	7	*1.10*	7	*2.05*	8	*1.24*	8	*1.98*	8	*1.24*
15^3	8	*1.19*	8	*1.05*	9	*1.69*	11	*1.16*	10	*1.68*	12	*1.16*
20^3	8	*1.24*	8	*1.06*	9	*1.82*	9	*1.18*	11	*1.76*	10	*1.18*
25^3	9	*1.26*	9	*1.08*	10	*1.98*	9	*1.21*	12	*2.02*	9	*1.21*
30^3	10	*1.22*	10	*1.06*	11	*1.77*	11	*1.17*	13	*1.75*	13	*1.17*
35^3	10	*1.24*	9	*1.06*	11	*1.99*	9	*1.18*	13	*1.91*	11	*1.18*
40^3	10	*1.26*	10	*1.07*	12	*1.83*	9	*1.19*	16	*1.77*	10	*1.19*

Table 1: Iteration count and complexity (lower complexity = faster run time) for SA and energy minimization for various mesh sizes and stretch factors for a 3D linear elasticity problem.