UCRL-TR-205838

LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

# A Note on the Relationship Between Adaptive AMG and PCG

*R. D. Falgout*

**August, 2004**

This work was done in December of 2001, while on sabbatical at the University of Colorado at Boulder

## Disclaimer

# A NOTE ON THE RELATIONSHIP BETWEEN ADAPTIVE AMG AND PCG

ROBERT D. FALGOUT*

**Abstract.** In this note, we will show that preconditioned conjugate gradients (PCG) can be viewed as a particular adaptive algebraic multigrid algorithm (adaptive AMG). The relationship between these two methods provides important insight into the construction of effective adaptive AMG algorithms.

## 1. Introduction.

The *adaptive multigrid* method employs the idea of "using the method to improve the method." Recent algorithms that utilize this basic idea are Wagner and Wittum's adaptive filtering [7, 8], Brandt's bootstrap algebraic multigrid (bootstrap AMG) [2], adaptive smoothed-aggregation [3], and the Ruge-Stüben-based adaptive AMG algorithm described in [4]. These methods exhibit the optimal convergence properties of multigrid, but are generally more robust (i.e., they apply to larger classes of problems) than their non-adaptive counterparts.

The *preconditioned conjugate gradient* (PCG) method [1, 5] is a generalization of the classic conjugate gradient (CG) method of Hestenes and Stiefel [6]. These so-called Krylov methods are well-known and widely used in practice. Unlike multigrid, they are not (in general) optimal. In particular, the convergence rate of CG (resp., PCG) depends on the condition number of the system matrix (resp., the preconditioned system matrix). However, CG and PCG are extremely robust. For example, CG is provably convergent for all symmetric positive definite systems.

At first glance, adaptive multigrid looks nothing like PCG. However, we will show in this note that PCG, preconditioned by any symmetric fixed-point residual correction method (e.g., Jacobi, symmetric Gauss-Seidel, multigrid), can be viewed as a particular adaptive AMG algorithm. The relationship between these two seemingly different methods provides important insight into the construction of effective adaptive AMG algorithms.

## 2. Preliminaries.

Consider solving the linear system

$$Au = f, \tag{2.1}$$

where $A$ is a real symmetric positive definite (SPD) matrix on a Euclidean vector space $\mathcal{V}$. Now consider a residual correction method of the form ($j \geq 1$)

$$u_j = u_{j-1} + M^{-1} r_{j-1}, \tag{2.2}$$

where $M$ is symmetric and $r_j$ is the residual at the $j^{th}$ iteration, $f - Au_j$. This is the iterative method that we will use to precondition CG and to produce the so-called *prototypes* in the adaptive method. The operator $M$ must satisfy additional criteria in order to insure convergence of the two methods, but for this paper it is sufficient to simply assume that the methods are convergent. The iteration in (2.2) can be

*Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, P.O. Box 808, L-561, Livermore, CA 94551. E-mail: rfalgout@llnl.gov. This work was performed under the auspices of the U. S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract W-7405-Eng-48. This work was done in December of 2001, while on sabbatical at the University of Colorado at Boulder.

rewritten in terms of the initial data as follows

$$u_j = H^j u_0 + \sum_{i=0}^{j-1} H^i M^{-1} f, \tag{2.3}$$

where

$$H = (I - M^{-1}A). \tag{2.4}$$

The error propagation iteration can similarly be written as

$$e_j = H^j e_0, \tag{2.5}$$

where $e_j = u - u_j$ is the error such that $r_j = Ae_j$. These equations will be useful later in the discussion.

**3. PCG.** PCG is just the CG method applied to the preconditioned system

$$\hat{A}u = \hat{f}; \quad \hat{A} = CA; \quad \hat{f} = Cf, \tag{3.1}$$

where $C$ is the preconditioning matrix given by the second term in (2.3). The three-term recurrence of CG results from an algebraic simplification of the following iteration, applied here to (3.1):

$$e_k = (I - Q_k)e_0, \tag{3.2}$$

where $Q_k$ is the $A$-orthogonal projection onto the Krylov subspace

$$\begin{aligned} \mathcal{K}_k &= \{\hat{r}_0, \hat{A}\hat{r}_0, ..., \hat{A}^{k-1}\hat{r}_0\} \\ &= \{Cr_0, (CA)Cr_0, ..., (CA)^{k-1}Cr_0\} \\ &= \{(CA)e_0, (CA)^2 e_0, ..., (CA)^k e_0\}. \end{aligned} \tag{3.3}$$

So, the above iteration describes the error propagation for the PCG method being considered here. But, from (2.3) and (2.4), we have that

$$CA = \sum_{i=0}^{j-1} H^i M^{-1} A = \sum_{i=0}^{j-1} H^i (I - H) = \sum_{i=0}^{j-1} H^i - \sum_{i=1}^{j} H^i = I - H^j.$$

Hence, we can rewrite the space in (3.3) as

$$\mathcal{K}_k = \{(I - H^j)e_0, H^j(I - H^j)e_0, ..., H^{(k-1)j}(I - H^j)e_0\}. \tag{3.4}$$

To see this, consider the second vector in (3.3). This vector is just the sum of two components,

$$(I - H^j)^2 e_0 = (I - H^j)e_0 - H^j(I - H^j)e_0,$$

the first of which is already in the space. Hence, only the second component is needed in (3.4). The rest of (3.4) is derived similarly.

**4. Adaptive AMG.** In this section, we will simultaneously describe the adaptive algorithm being considered here, and characterize it in a form that will make it possible to compare to the PCG method in the previous section. We will compare the two methods in the next section.

*Step 1:* Relax on $Ax = 0$ using the iterative method in (2.2) with initial guess $x_0$. Hence, from (2.3),

$$x_1 = H^j x_0.$$

Define interpolation by

$$P_1 = [x_1].$$

Then, the error propagation for the new method is given by

$$e_1 = (I - Q_1) H^j e_0; \quad Q_1 = P_1 (P_1^T A P_1)^{-1} P_1^T A,$$

so that $Q_1$ is just an $A$-orthogonal projection onto the space

$$\mathcal{K}_1 = \{H^j x_0\}.$$

*Step 2:* Relax on $Ax = 0$ using the new adaptive method with initial guess $x_1$. Hence, we have that

$$x_2 = (I - Q_1) H^j x_1 = (I - Q_1) H^{2j} x_0.$$

Define interpolation by

$$P_2 = [P_1, x_2].$$

Then, the error propagation for the new method is given by

$$e_2 = (I - Q_2) H^j e_0; \quad Q_2 = P_2 (P_2^T A P_2)^{-1} P_2^T A,$$

so that $Q_2$ is just an $A$-orthogonal projection onto the space

$$\mathcal{K}_2 = \{H^j x_0, H^{2j} x_0\}.$$

*Step k:* Relax on $Ax = 0$ using the new adaptive method with initial guess $x_{k-1}$. Hence, we have that

$$\begin{aligned} x_k &= (I - Q_{k-1}) H^j x_{k-1} \\ &= (I - Q_{k-1}) H^j (I - Q_{k-2}) H^{(k-1)j} x_0 \\ &= (I - Q_{k-1}) H^{kj} x_0 - (I - Q_{k-1}) H^j Q_{k-2} H^{(k-1)j} x_0 \\ &= (I - Q_{k-1}) H^{kj} x_0. \end{aligned}$$

The last step above follows since $H^j Q_{k-2} H^{(k-1)j} x_0$ is in the space $\mathcal{K}_{k-1}$ and since $Q_{k-1}$ is a projection onto that space. Define interpolation by

$$P_k = [P_{k-1}, x_k].$$

Then, the error propagation for the new method is given by

$$e_k = (I - Q_k) H^j e_0; \quad Q_k = P_k (P_k^T A P_k)^{-1} P_k^T A,$$

so that $Q_k$ is just an $A$-orthogonal projection onto the space

$$\mathcal{K}_k = \{H^j x_0, H^{2j} x_0, ..., H^{kj} x_0\}.$$

**5. The Connection.** In the previous two sections, we characterized the PCG method and a particular adaptive method both as subspace projection methods. Before we compare them, let's first summarize what we have done.

- The *PCG method* can be characterized by the following error iteration
$$e_k^p = (I - Q_k^p)e_0^p,$$
where $Q_k^p$ is the $A$-orthogonal projection onto the space
$$\mathcal{K}_k^p = \{(I - H^j)e_0^p, ..., H^{(k-1)j}(I - H^j)e_0^p\}.$$
- The *adaptive method* can be characterized by the following error iteration
$$e_k^a = (I - Q_k^a)H^j e_0^a,$$
where $Q_k^a$ is the $A$-orthogonal projection onto the space
$$\mathcal{K}_k^a = \{H^j x_0, ..., H^{kj} x_0\}.$$

From the above, it is clear that the two methods have essentially the same properties. However, by choosing appropriate initial guesses for the two methods, it is also possible to get them to produce exactly the same iterates. To do this, let $u_0^a$ (the initial guess for the adaptive method) be some arbitrary initial guess $u_0$, and let $u_0^p$ (the initial guess for the PCG method) be the result of applying $j$ steps of (2.2) to the system (2.1) with initial guess $u_0$. From this, we have that

$$e_0^a = e_0, \quad e_0^p = H^j e_0. \tag{5.1}$$

Now, let $x_0$ be the preconditioned initial residual, i.e. let

$$x_0 = Cr_0 = (I - H^j)e_0. \tag{5.2}$$

Substituting (5.1) and (5.2) into the methods above, we see that both PCG and the adaptive method have the following error iteration

$$e_k = (I - Q_k)H^j e_0,$$

where $Q_k$ is the $A$-orthogonal projection onto the space

$$\mathcal{K}_k = \{H^j(I - H^j)e_0, ..., H^{kj}(I - H^j)e_0\}.$$

**6. Comments and Conclusions.** The adaptive method considered here differs in a number of ways from most other adaptive AMG methods:

- It starts out with a single coarse grid point, and adds a new degree of freedom to the coarse grid every time a new prototype is generated.
- It is only a 2-level method.
- It does not start with a random initial guess to form each consecutive prototype. Note, however, that if $x_0$ is rich in all eigenmodes of $A$, then the approach used here is reasonable.
- It does not test the prototype being generated to see whether it really *is* a good representative. Instead, it just assumes that each $x_k$ must automatically be dealt with on a "coarse grid".

However, the adaptive idea of "using the method to improve the method" is definitely present in this algorithm. So, why should we expect adaptive multigrid methods to be any better than PCG?

The reason lies in the way the prototypes are used. A subtle, yet crucial point about adaptive AMG methods is that *the prototype is only a representative for constructing what it means to be smooth locally.* The key word here is *locally.* The adaptive algorithm in this note only deals with the $x_k$ globally, and that is why it is no better than PCG.

## REFERENCES


[1] S. F. Ashby, T. A. Manteuffel, and P. E. Saylor, *A taxonomy for conjugate gradient methods*, SIAM J. Numer. Anal., 27 (1990), pp. 1542–1568.

[2] A. Brandt, *General highly accurate algebraic coarsening*, Electronic Transactions on Numerical Analysis, 10 (2000), pp. 1–20.

[3] M. Brezina, R. Falgout, S. MacLachlan, T. Manteuffel, S. McCormick, and J. Ruge, *Adaptive smoothed aggregation ($\alpha$SA)*, SIAM J. Sci. Comput., (2003). To Appear. Also available as LLNL technical report UCRL-JC-151149.

[4] ———, *Adaptive algebraic multigrid methods*, SIAM J. Sci. Comput., (2004). Submitted. Also available as LLNL technical report UCRL-JRNL-203501.

[5] P. Concus, G. H. Golub, and D. P. O'Leary, *A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations*, in Sparse Matrix Computations, J. R. Bunch and D. J. Rose, eds., Academic Press, New York, 1976, pp. 309–332.

[6] M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 409–435.

[7] C. Wagner and G. Wittum, *Adaptive filtering*, Numerische Mathematik, 78 (1997), pp. 305–328.

[8] ———, *Filtering decompositions with respect to adaptive test vectors*, in Multigrid Methods V, W. Hackbusch and G. Wittum, eds., vol. 3 of Lecture Notes in Computational Science and Engineering, Berlin, 1998, Springer, pp. 320–334. Proc. of the Fifth European Multigrid Conference held in Stuttgart, Germany, October 1–4, 1996.