SIAM J. NUMER. ANAL.
Vol. 42, No. 1, pp. 130–152
© 2004 Society for Industrial and Applied Mathematics

ASYNCHRONOUS FAST ADAPTIVE COMPOSITE-GRID METHODS FOR ELLIPTIC PROBLEMS: THEORETICAL FOUNDATIONS*

BARRY LEE†¶, STEPHEN F. MCCORMICK‡, BOBBY PHILIP§¶, AND DANIEL J. QUINLAN†¶

Abstract. Accurate numerical modeling of complex physical, chemical, and biological systems requires numerical simulation capability over a large range of length scales, with the ability to capture rapidly varying phenomena localized in space and/or time. Adaptive mesh refinement (AMR) is a numerical process for dynamically introducing local fine resolution on computational grids during the solution process, in response to unresolved error in a computation. Fast adaptive composite-grid (FAC) methods are a class of algorithms that exploit the multilevel structure of AMR grids to solve elliptic problems efficiently. This paper develops a theoretical foundation for AFACx, an asynchronous FAC method. A new multilevel condition number estimate establishes that the convergence rate of the AFACx algorithm does not degrade as the number of refinement levels in the AMR hierarchy increases.

Key words. adaptive mesh refinement, asynchronous, fast adaptive composite-grid, elliptic solvers, FAC, AFAC, AFACx

AMS subject classifications. 65F10, 65N22, 65N50, 65N55

DOI. 10.1137/S0036142902400767

1. Introduction.

Adaptive mesh refinement (AMR) is a numerical process for dynamically introducing local fine resolution on computational grids during the solution process in response to unresolved error in a computation. Local fine resolution is achieved by dynamically adapting the existing computational grid based on additional grid points (point-based AMR) or finer local grids (block-structured AMR). AMR approaches are attractive because they often achieve orders of improvement in computational efficiency and memory usage. AMR techniques were first introduced by Brandt [19] in the early 1970s for general problems in a multilevel context and by Berger and Oliger [6] in the 1980s for hyperbolic problems. Since then, AMR research has been pursued by several groups (cf. [1, 2, 5, 29, 42, 43]).

For elliptic problems, when numerical simulations involve a large number of refinement levels and are extremely large, effective parallel methods for AMR must be considered. It is then desirable to develop elliptic solvers that asynchronously process all grids, or at least asynchronously process grids at a fixed refinement level. In addition, as the number of refinement levels increases, the convergence rate should not degrade as a function of the number of refinement levels.

*Received by the editors January 9, 2002; accepted for publication (in revised form) June 16, 2003; published electronically January 6, 2004. This work was performed by an employee of the U.S. Government. The U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes. Copyright is owned by SIAM to the extent not limited by these rights.

http://www.siam.org/journals/sinum/42-1/40076.html

†Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94550 (lee123@llnl.gov, dquinlan@llnl.gov).

‡Department of Applied Mathematics, University of Colorado at Boulder, Boulder, CO 80309 (stevem@newton.colorado.edu).

§Current Address: CCS-3, Modeling, Algorithms, and Informatics Group, Los Alamos National Laboratory, Los Alamos, NM 87545 (bphilip@lanl.gov).

¶The work of these authors was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract W-7405-Eng-48.

The fast adaptive composite-grid (FAC) method was developed in the 1980s [32, 33, 34, 35] to provide more robust discretization and solution methods for elliptic problems on AMR grids. Its strength lies in its ability to use existing single grid solvers on uniform meshes for different refinement levels, with the combined effect of solving a nonuniform composite-grid problem. Though FAC allows for asynchronous processing of disjoint grids at a given refinement level and its convergence rate is bounded independently of the number of refinement levels, the multiplicative way it treats the various refinement levels imposes sequentialness in its processing. For large-scale parallel AMR applications, this sequential nature of FAC, like that of other AMR techniques, represents a serious bottleneck to full scalability.

This difficulty led to the development of the asynchronous version of FAC, called AFAC [28, 33, 36, 37]. AFAC, like FAC, is blessed with level-independent convergence bounds and the convenience of enabling uniform grid solvers. But it has the added advantage of allowing asynchronous processing of all refinement levels. This important asynchronous feature is obtained at the cost of only a modest fixed decrease in convergence rates [33].

Further research into improving computational efficiency associated with the uniform grid solvers on local grid patches led to the development of AFACx [41]. AFACx is very inexpensive because it uses only simple relaxation methods on all but the coarsest grid. Numerical results [39, 41] show that the attendant reduction in computational and communication costs of AFACx comes with no significant degradation in convergence rates compared to AFAC based on multigrid solvers.

Convergence bounds for FAC were established in [32, 38] under certain regularity assumptions. Widlund and Dryja proposed and analyzed variants of FAC [45, 23]. Reusken and Ferket [24] compared FAC with the local defect correction (LDC) method [27] introduced by Hackbusch. AFAC was introduced by Hart and McCormick in [28]. Optimality in the multilevel case for AFAC applied to a model problem was shown in [31]. This was followed by the development of AFACx [41]. Cheng [21, 22] established optimal bounds on the condition number of the multilevel AFAC iteration operator with exact solvers. Moe [39, 7] presented performance results for FAC and AFAC on parallel machines. Quinlan, in his thesis [41], presented a two-level convergence analysis for AFACx assuming a sufficient number of smoothing steps at each level and showed that it is closely related to the convergence rate of AFAC. Shapira [44] compared the performance of AFAC and AFACx. However, a multilevel theory for AFACx remained a gap in the theory of multilevel FAC-type methods.

Closely related to AFACx are the additive preconditioners of Bramble, Pasciak, Xu [16] and Bramble, Pasciak, and Vassilevski [18]. The theoretical framework developed by Bramble, Pasciak, Xu, Wang, Oswald, Griebel, and others [16, 15, 12, 14, 13, 17, 11, 47, 48, 50, 9, 8, 49, 26] presents a powerful tool for analyzing multilevel methods. Relying heavily on this modern multilevel framework for multilevel methods and some of the assumptions therein, we present in this paper a new multilevel condition number estimate for the AFACx operator.

The new theoretical results presented in this paper are strongly backed by numerical evidence [41, 40] and performance results [41]. Recent numerical work [40, 30] shows that AFACx can be applied successfully to elliptic PDE systems arising from first-order system least squares (FOSLS) formulations on adaptively refined curvilinear AMR grids. As increasingly complex PDE systems are simulated and the need for AMR is increasingly crucial, theoretical and computational analyses of fast, parallelizable, and efficient multilevel solvers and preconditioners such as AFACx and BPX [16] become increasingly important for validating the results of complex simulations.

This paper develops new multilevel estimates establishing that the condition number of the AFACx operator, like that for AFAC, is bounded independently of the number of refinement levels. We start by introducing the model problem and necessary preliminaries in section 2. In section 3, we introduce the various FAC-type methods, and finally, in section 4, we establish the theory.

2. Model problem. Consider a linear, self-adjoint, second-order elliptic boundary value problem in $\mathbb{R}^n$, $n = 2, 3$, of the form

$$\begin{cases} Lu \equiv -\sum_{i,j=1}^{n} \frac{\partial}{\partial x_i}(a_{ij}(x)\frac{\partial u}{\partial x_j}) = & f \quad \text{in } \Omega, \\ \qquad\qquad\qquad\qquad u = & 0 \quad \text{on } \partial\Omega, \end{cases} \tag{2.1}$$

where u is the unknown, $f \in L^2(\Omega)$ is the source term, and a_{ij} are appropriate coefficients. Assume that

- domain $\Omega \subset \mathbb{R}^n$ is convex polygonal;
- coefficients $a_{ij}(x) \in C^0(\overline{\Omega}), 1 \leq i, j \leq n$;
- matrix $[a_{ij}(x)]_{1\leq i,j\leq n}$ is symmetric almost everywhere in Ω; and
- operator L is uniformly elliptic in the sense that there exists a constant $\theta > 0$ such that $\sum_{i,j=1}^{n} a_{ij}(x)\xi_i\xi_j \geq \theta|\xi|^2$ for almost all x in Ω and all ξ in $\mathbb{R}^n$, where $|\cdot|$ is the Euclidean norm.

This section is concerned with the numerical solution of the algebraic equations that arise from discretizing problem (2.1) on adaptively refined curvilinear grids. We focus on the plane $\mathbb{R}^2$ for simplicity.

2.1. Variational formulation. Under the above assumptions, the natural linear space in which to seek a weak solution of (2.1) is $V := H_0^1(\Omega)$, and the variational problem is Find $u \in V$ such that

$$a(u, v) = f(v) \quad \forall v \in V, \tag{2.2}$$

where the respective bilinear and linear forms are

$$a(u, v) = \int_{\Omega} \sum_{i,j=1}^{n} a_{ij}(x) \frac{\partial u}{\partial x_j} \frac{\partial v}{\partial x_i} \mathrm{d}\Omega, \tag{2.3}$$

$$f(v) = \int_{\Omega} f v \mathrm{d}\Omega. \tag{2.4}$$

It is known [25] that (2.2) has a unique solution, $u \in V$. Moreover, $a(\cdot, \cdot) : V \times V \to \mathbb{R}$ is symmetric and continuous [25], so uniform ellipticity of L and Poincaré's inequality (cf. [10]) imply that $a(\cdot, \cdot)$ is coercive on V: there exists a constant $\gamma > 0$ such that

$$a(u, u) \geq \gamma\|u\|_V^2 \quad \forall u \in V. \tag{2.5}$$

Coercivity, in turn, implies that $a(\cdot, \cdot)$ defines an equivalent inner product over space V. Furthermore, by the Riesz representation theorem (cf. [10]), $a(\cdot, \cdot)$ induces a bounded linear operator $\mathcal{A} : V \to V$ uniquely determined by

$$a(u, v) = (\mathcal{A}u, v) \quad \forall u, v \in V. \tag{2.6}$$

2.2. Partially refined meshes. To discretize (2.2) on partially refined meshes, we introduce the following notation. Let $\Omega_J \subseteq \Omega_{J-1} \subseteq \cdots \subseteq \Omega_1 \equiv \Omega$ be a nested sequence of nonempty bounded open polygonal Lipschitz domains. Subdomains $\Omega_k,\ k = 2, 3, \dots, J$, can be viewed as regions where the solution may vary on increasingly finer scales and, hence, regions where local refinement patches are generated during the AMR process. Let $\mathcal{T}_1^c = \{\tau_i^1\}_{i=1}^{N_1}$ be a triangulation of $\Omega_1,\ N_1 \geq 4$, meaning that they cover Ω_1 and do not overlap in the sense that the intersection of any two triangles in the triangulation is either empty, a common vertex, or a common edge. Assume that $\mathcal{T}_1^c$ is quasi-uniform. We assume also that the boundaries of Ω_2 align with the edges of elements in $\mathcal{T}_1^c$, and at least one edge of $\mathcal{T}_1^c$ is contained in Ω_2. Triangulation $\mathcal{T}_k^c = \{\tau_i^k\}_{i=1}^{N_k}$, $k = 2, 3, \dots, J$, of Ω, is obtained from $\mathcal{T}_{k-1}^c$ in the following manner. Since $\Omega_k \subseteq \Omega_{k-1}$ and its boundary aligns with elements of $\mathcal{T}_{k-1}^c$, then there exists a local "coarse" triangulation, $\mathcal{T}_k^{h_{k-1}} = \{\tau_{i_j}^{k-1}\}_{j=1}^{M_k}, M_k \leq N_{k-1}$, consisting of elements of $\mathcal{T}_{k-1}^c$ that cover Ω_k, where h_{k-1} is the length of the longest edge of triangles in $\mathcal{T}_k^{h_{k-1}}$. $\mathcal{T}_k^{h_{k-1}}$ is then a quasi-uniform triangulation of Ω_k. Now we uniformly refine elements of $\mathcal{T}_k^{h_{k-1}}$ by subdividing each triangle into four triangles by connecting the midpoints of the edges. This yields a "fine" local triangulation $\mathcal{T}_k^{h_k}$ of Ω_k, which is regular in the sense of Bank, Dupont, and Yserentant [3]. Elements of $\mathcal{T}_{k-1}^c$ that lie in the complement of Ω_k and the elements of $\mathcal{T}_k^{h_k}$ together form the elements of $\mathcal{T}_k^c = (\mathcal{T}_{k-1}^c \backslash \mathcal{T}_k^{h_{k-1}}) \cup \mathcal{T}_k^{h_k}$. This process leads to a series of nested triangulations $\{\mathcal{T}_k^c\}_{k=1}^J$ of Ω that form partially refined locally quasi-uniform meshes.

2.3. Finite element spaces. Henceforth, we assume that conforming piecewise linear finite elements are used, although our results will clearly apply to more general cases. We thus define $V_k^c \subset H_0^1(\Omega), k = 1, 2, \dots J$, to be the space spanned by standard piecewise linear nodal basis functions with local support about the nodes of triangulation $\mathcal{T}_k^c$. Because of the conformity of the finite elements, note that there are no degrees of freedom associated with fine nodes that lie on boundary $\partial\Omega_k$. Continuity implies that these "slave" nodes are evaluated simply by interpolation from adjacent coarse nodes. Now, the "fine" local finite element space defined in the interior of domain Ω_k is $V_k^{h_k} = V_k^c \cap H_0^1(\Omega_k)$. By our use of $H_0^1(\Omega_k)$ here, we mean that functions in $V_k^{h_k}$ have support only in the interior of Ω_k. Similarly, we define "coarse" local finite element spaces by $V_k^{h_{k-1}} = V_{k-1}^c \cap H_0^1(\Omega_k), V_k^{h_{k-1}} \subset V_k^{h_k},\ k = 2, \dots, J$. Note that the local spaces are nested: $V_1^c \subseteq V_2^c \subseteq \cdots \subseteq V_J^c \subset H_0^1(\Omega)$. However, the coarse local spaces are generally nonnested because they typically correspond to increasingly smaller local subdomains.

2.4. The discrete variational problem. Having chosen finite-dimensional composite-grid space V_J^c, the discrete variational problem is Find $u^c \in V_J^c$ such that

$$a(u^c, v) = f(v) \ \ \forall v \in V_J^c. \tag{2.7}$$

This problem is equivalent to solving the linear system

$$A^c u^c = f^c, \tag{2.8}$$

where A^c is a symmetric positive-definite matrix induced by the linear operator $\mathcal{A}^c$ defined over composite-grid space V_J^c. Note that V_J^c is a finite-dimensional subspace of $H_0^1(\Omega)$. For notational convenience, denote spaces V_k^c by $V_k,\ k = 1, 2, \dots, J$; operator $\mathcal{A}^c$ by A; A-inner product $a(\cdot,\cdot)$ on V_J by $A(\cdot,\cdot)$; and the induced A-norm by $|||\cdot|||$. The L^2 inner product on V_J is denoted by $(\cdot,\cdot)$ and its induced norm by $\|\cdot\|$.

2.5. Stationary linear iteration. A consistent stationary linear iterative process for linear system

$$Au = f$$

can be written in the form

$$u^{n+1} = u^n + B(f - Au^n), \tag{2.9}$$

where B is an approximate inverse of A. Here we are thinking of iteration (2.9) as one of our FAC-type algorithms defined below. If B is symmetric with respect to the A inner product, then the process is said to be symmetric. Letting $e^n = u - u^n$ denote the error in the nth iterate, then (2.9) implies that

$$e^{n+1} = (I - BA)e^n. \tag{2.10}$$

Therefore, for the iteration to converge in general, we must have $\rho(I - BA) < 1$, where $\rho(\cdot)$ denotes the spectral radius. For common multiplicative-type algorithms, it is often easy to establish this condition. However, for additive-type multilevel solvers, typically all that can be shown is that $\kappa(BA)$, the condition number of the operator BA, is independent of the number of levels. Such a result implies that the "damped" linear iteration

$$u^{n+1} = u^n + \omega B(f - Au^n) \tag{2.11}$$

converges for sufficiently small ω. It is this type of a result that we establish for the AFACx algorithm defined below.

To describe the FAC algorithms, we need to define operators that approximate (2.7) at the different refinement levels, projection operators that transfer data from fine to coarse spaces, and smoothing operators on the different spaces, all in terms of the discrete inner products $(\cdot, \cdot)$ and $A(\cdot, \cdot)$.

2.6. Approximating composite-grid operators on coarser levels.

DEFINITION 1. *For $k = 1, 2, \ldots, J$, define operator $A_k : V_k \longrightarrow V_k$ by*

$$(A_k w, \phi) = A(w, \phi) \quad \forall \phi \in V_k.$$

Note that operator A_k is symmetric and positive-definite in inner products $A(\cdot, \cdot)$ and $(\cdot, \cdot)$.

2.7. Projection operators. We introduce the following projection operators typically used in multilevel theory.

DEFINITION 2. *For $k = 1, 2, \ldots, J$, define "elliptic projection" operator $P_k : V_J \longrightarrow V_k$ by*

$$A(P_k w, \phi) = A(w, \phi) \quad \forall \phi \in V_k.$$

DEFINITION 3. *For $k = 1, 2, \ldots, J$, define "L^2 projection" operator $Q_k : V_J \longrightarrow V_k$ by*

$$(Q_k w, \phi) = (w, \phi) \quad \forall \phi \in V_k.$$

It can be shown that P_k and Q_k are orthogonal projection operators satisfying the following basic properties:

- $P_k P_l = P_l, \;\; P_l P_k = P_l, \;\; Q_k Q_l = Q_l, \;\; Q_l Q_k = Q_l$ for $l \leq k$.

Additionally, P_k and Q_k are related according to

$$A_k P_k = Q_k A. \tag{2.12}$$

2.8. Smoothing operators. We can write one step of a general stationary linear smoothing procedure applied to

$$A_k u_k = f_k \tag{2.13}$$

in the form

$$u_k^{n+1} = u_k^n + R_k(f_k - A_k u_k^n), \tag{2.14}$$

where $R_k : V_k \longrightarrow V_k$. Note that (2.14) is of the same form as (2.9), but we use R here and below (possibly with subscripts and hats) to signify smoothing. Now the error, $e_k^n = u_k - u_k^n$, obeys the following propagation equation:

$$e_k^{n+1} = (I - R_k A_k)e_k^n \tag{2.15}$$

or

$$e_k^{n+1} = (I - T_k)e_k^n, \tag{2.16}$$

where $T_k : V_J \longrightarrow V_k$ is defined by $T_k = R_k A_k P_k$. For simplicity, we assume that $R_1 = A_1^{-1}$ and that $R_k,\ k = 2, 3, \ldots, J$, are symmetric with respect to the L^2 inner product. Consider the special case $R_k = \hat{R}_k \equiv \frac{1}{\lambda_k} I$, where λ_k is the spectral radius of A_k. The smoothing process is then just Richardson's iteration defined by

$$u_k^{n+1} = u_k^n + \frac{I}{\lambda_k}(f_k - A_k u_k^n). \tag{2.17}$$

Corresponding to $\hat{R}_k$, we define $\hat{T}_k = \hat{R}_k A_k P_k$.

To further quantify the properties that a simple smoother must satisfy, we make the following assumptions commonly made in modern multilevel analyses. While we do briefly comment on the motivation for each assumption and the conditions under which they hold, we refer the reader to [11, 46, 15, 16, 12, 13, 48, 17, 47, 50] for further details. It suffices to state that the assumptions are valid (cf. [11]) for our model problem with partially refined locally quasi-uniform meshes and simple smoothers like Richardson, damped Jacobi, and symmetric Gauss–Seidel.

The first assumption concerns the Richardson operator, $\hat{R}_k$.

A.1. There exist constants $\epsilon \in (0, 1)$ and $\gamma > 0$ such that

$$A(\hat{T}_k w, w) \le (\gamma \epsilon^{k-l})^2 A(w, w) \quad \forall w \in V_l,\ l \le k, \quad k = 1, 2, \ldots, J. \tag{2.18}$$

Roughly speaking, assumption A.1 asserts that the smoother attenuates "smooth" error components slowly; i.e., energy reduction in "smooth" components (represented by components in subspaces $V_l, l < k$) is small compared to energy reduction in the "oscillatory" components. This assumption is a generalization of the strengthened Cauchy–Schwarz inequalities first introduced by Yserentant [49] for hierarchical bases and used extensively in multilevel theory [47, 14, 11, 48]. Constant γ depends on the ellipticity of the boundary value problem and the variation of the coefficients in (2.1). For our model boundary value problem (2.1) discretized with piecewise linears on simplices, (2.18) has been shown to hold (cf. [14, 48]). However, it is apparently not known whether assumption A.1 holds when the coefficients in (2.1) are not very smooth, e.g., when they are only bounded and measurable (cf. [11]).

The next two assumptions allow for more general smoothers, R_k.

A.2. There exist constants $a_0 \in (0,1)$ and $a_1 > 1$ such that

$$a_0 \frac{\|u\|^2}{\lambda_k} \leq (R_k u, u) \leq a_1 \frac{\|u\|^2}{\lambda_k} \quad \forall u \in V_k, 2 \leq k \leq J. \tag{2.19}$$

Assumption A.2 can also be written in the form

$$a_0(\hat{R}_k u, u) \leq (R_k u, u) \leq a_1(\hat{R}_k u, u) \quad \forall u \in V_k, 2 \leq k \leq J, \tag{2.20}$$

which implies that smoothing operator $R_k,\ k = 2, 3, \ldots, J$, is spectrally equivalent to the Richardson smoothing operator $\hat{R}_k$. It is easy to see that A.2 implies that

$$a_0\ A(\hat{T}_k u, u) \leq A(T_k u, u) \leq a_1\ A(\hat{T}_k u, u) \quad \forall u \in V_J,\ k = 2, 3, \ldots, J. \tag{2.21}$$

Spectral equivalence of symmetric Gauss–Seidel to the Richardson smoother is shown in [48]. The upper inequality in (2.21) holds in general for point-smoothers (cf. [11]).

A.3. There exists a constant $\theta \in (0,2)$ such that

$$A(T_k v, T_k v) \leq \theta A(T_k v, v) \quad \forall v \in V_k,\ k = 1, 2, \ldots, J. \tag{2.22}$$

Assumption A.3 is a natural consequence of assuming that operators $I - T_k$ are contractive in the energy norm, i.e.,

$$|||I - T_k||| < 1, \quad k = 1, 2, \ldots, J. \tag{2.23}$$

Note that $\theta = 1$ for Richardson, $\theta < 1$ for under-damped Richardson, and $\theta = 1$ for suitably scaled Jacobi and block Jacobi smoothers. In [13], (2.22) is shown to hold for various line and point-based Jacobi and Gauss–Seidel smoothers.

Assumption A.3 can be derived from assumption A.1 under a suitable assumption on γ and spectral equivalence of the smoothers to Richardson iteration. However, in general, the assumption on γ cannot be established without special scaling of the smoothers, so we choose to state both assumptions separately.

In addition to assumptions A.1–A.3 on smoothers R_k and $\hat{R}_k$, a "weak regularity" assumption is required. This condition replaces the standard full regularity and approximation assumption (cf. [12]) with a weaker assumption on operator A and smoothers $\hat{R}_k, k = 2, 3, \ldots, J$.

A.4. There exists a constant $\eta > 0$ such that

$$A(v, v) \leq \eta \sum_{k=1}^{J} A(\hat{T}_k v, v) \quad \forall v \in V_J. \tag{2.24}$$

Assumption A.4 is shown to hold for our model problem discretization in [11]. However, it is also noted in [11] that, in the application to second-order elliptic equations for coefficients with large jumps, A.4 is not known to hold independent of the size of the jumps.

3. Algorithms.

We now describe the FAC, AFAC, and AFACx algorithms and complete the section with a discussion of existing theory.

3.1. FAC. Let $u_c^n \in V_J$ denote the current approximation to the solution of composite-grid equation (2.8).

ALGORITHM 1. *One iteration of the basic FAC algorithm consists of the following steps.*

For $k = 1, 2, \ldots, J$, *do:*
find $w_k \in V_k^{h_k}$ *such that*
$$a(u_c^{n+(k-1)/J} + w_k, v) = f(v) \quad \forall v \in V_k^{h_k};$$
set $u_c^{n+k/J} = u_c^{n+(k-1)/J} + w_k$.

As can be seen from this pseudolanguage, FAC involves the solution of the residual equation on all refinement levels. The correction on a coarse level is computed before the correction on the next finer level, thus providing boundary conditions for the finer level equations. FAC is multiplicative, since it can be represented as a product of linear operators. Multiplicative algorithms are inherently sequential because each operation depends on its predecessor, making them less attractive in a parallel environment.

3.2. AFAC. Processing on each level in FAC attempts to resolve all components of the solution to the composite-grid residual equation that are represented on a refinement level and coarser levels. On the other hand, processing of each level by AFAC [33, 36, 37] attempts only to resolve components that can be represented on that refinement level. This objective is not dependent on resolving components of the solution to the residual equation that are represented on coarser or finer levels, so it provides for independent level processing. The principal step in AFAC is resolving solution components on each composite-grid level. Let $u_c^n \in V$ denote the current approximation to the solution of composite-grid equation (2.8).

ALGORITHM 2. *One iteration of the AFAC algorithm consists of the following steps.*

For $k = 1, 2, \ldots, J$, *do:*
find $w_k^f \in V_k^{h_k}$ *such that*
$$a(u_c^n + w_k^f, v) = f(v) \quad \forall v \in V_k^{h_k};$$
if $(k > 1)$, *then*
find $w_k^r \in V_k^{h_{k-1}}$ *such that*
$$a(u_c^n + w_k^r, z) = f(z) \quad \forall z \in V_k^{h_{k-1}};$$
set $w_1^r = 0$;
set $u_c^{n+1} = u_c^n + \sum_{k=1}^{J}(w_k^f - w_k^r)$.

AFAC appears to have optimal or near-optimal complexity in a parallel computing environment because it allows for simultaneous processing of all refinement levels. This is important because the solution process on each grid, even with the most efficient solvers, dominates computational complexity. This is especially true for systems where the solution process is significantly more computationally intensive than the evaluation of the residuals. Coupled with multigrid processing on each level and nested iteration [32] on the composite-grids, the computational cost of AFAC is proportional to the cost of a global-grid solve alone (see Hart and McCormick [28] and McCormick [33] for further details).

The following two-grid result is proved in [32].

THEOREM 3.1. *Suppose A^c is positive-definite. Then the spectral radii of the two-level exact solver forms of $AFAC^c$ and FAC^c satisfy*

$$\rho(AFAC^c) = \rho^{\frac{1}{2}}(FAC^c). \tag{3.1}$$

Here, FAC^c and $AFAC^c$ denote the respective FAC and AFAC error propagation op-

erators on the composite-grid space, and $\rho(\cdot)$ *denotes spectral radius. The convergence factor for one iteration of the two-level exact solver form of AFAC satisfies*

$$|||AFAC^c||| \le \left(\frac{\delta}{1+\delta}\right)^{\frac{1}{4}}, \tag{3.2}$$

where constant $\delta > 0$ *is independent of* h *but depends on the regularity of* (2.1) *and the approximation properties of its discretization (see* [32] *for further details).*

Now, assume for each k, $1 \le k \le J$, that there exists a bounded Lipschitz polyhedral region $\hat{\Omega}_k$ such that $\Omega_k \subset \hat{\Omega}_k$, $(\hat{\Omega}_k \setminus \Omega_k) \cap \Omega = \emptyset$, and $\partial\hat{\Omega}_k \cap \Omega_{k+1} = \emptyset$, and that the Lipschitz constants of $\hat{\Omega}_k \backslash \Omega_{k+1}$ are uniformly bounded. In addition, assume there exist constants $\gamma_1 \ge \gamma_0 > 0$ and $q \in (0,1)$ such that $\gamma_0 q^k \le h_k \le \gamma_1 q^k$, $k = 1, 2, \ldots, J$. Under these assumptions, the following theorem was proved in [21].

THEOREM 3.2. *The AFAC operator has a condition number that is bounded independent of the number of refinement levels and the number of degrees of freedom.*

It is important to note that the results hold when the exact solvers on each level in FAC and AFAC are replaced by approximate solvers (e.g., multigrid solvers), provided that they give a fixed local error reduction (see [32] and [21]). Note also that in [32], the two-level results do not depend on the refinement ratios (h_{k+1}/h_k).

3.3. AFACx. AFAC removes the sequential nature inherent in the FAC algorithm. However, it is possible to further reduce the computational effort on each level by carefully replacing the local solvers in AFAC with smoothers. AFACx is exactly such an algorithm.

To define this scheme, we introduce auxiliary bilinear forms $b_k^r(\cdot,\cdot) : V_k^{h_{k-1}} \times V_k^{h_{k-1}} \longrightarrow \mathbb{R}$ and $b_k^f(\cdot,\cdot) : V_k^{h_k} \times V_k^{h_k} \longrightarrow \mathbb{R}$. These forms correspond to symmetric positive-definite operators $B_k^r : V_k^{h_{k-1}} \longrightarrow V_k^{h_{k-1}}$ and $B_k^f : V_k^{h_k} \longrightarrow V_k^{h_k}$ that represent the action of smoothers on the "restricted" local coarse grid space $V_k^{h_{k-1}}$ and the local "fine" grid space $V_k^{h_k}$. Let $u_c^n \in V$ denote the current approximation to the solution of composite-grid equation (2.8).

ALGORITHM 3. *One iteration of the AFACx algorithm consists of the following steps.*

For $k = 1, 2, \ldots, J$*, do:*
- *if* $(k = 1)$*, then*
  - *find* $u_1^f \in V_1^{h_1}$ *such that*
    - $a(u_c^n + u_1^f, v) = f(v) \;\; \forall v \in V_1^{h_1}$;
- *else*
  - *find* $w_k^r \in V_k^{h_{k-1}}$ *such that*
    - $b_k^r(w_k^r, z) = f(z) - a(u_c^n, z) \;\; \forall z \in V_k^{h_{k-1}}$;
  - *find* $u_k^f \in V_k^{h_k}$ *such that*
    - $b_k^f(w_k^r + u_k^f, v) = f(v) - a(u_c^n, v) \;\; \forall v \in V_k^{h_k}$;

set $u_c^{n+1} = u_c^n + \sum_{k=1}^J u_k^f$.

The above pseudolanguage shows that AFACx replaces the solves on the local restricted coarse and fine levels in AFAC on all but the coarsest level by smoothing steps. Smoothing is performed on the restricted coarse level to obtain the correction w_k^r on each level k. Smoothing on the fine level with initial guess w_k^r then yields u_k^f, which approximates the component of the correction that is representable only on level k.

AFACx is generally more efficient than AFAC because the various uniform grids (the local fine and restricted coarse refinement levels) are processed only by smoothing, instead of the somewhat more expensive multigrid solvers used in AFAC. This reduction in cost apparently comes with no significant degradation in convergence rates. The following two-level result is due to Quinlan [41].

THEOREM 3.3. *Consider the two-level AFACx algorithm that involves one smoothing step on the fine grid patch and n smoothing steps on the restricted coarse grid. Then, for sufficiently large n, the spectral radius of the AFACx error propagation operator is bounded uniformly by a constant less than one, assuming only that this is true for the AFAC error operator.*

3.4. Symmetric AFACx. The operator corresponding to the AFACx algorithm described above is not symmetric with respect to the A inner product. To facilitate condition number estimates, we work instead with a symmetrized form of AFACx developed as follows. Let $u_c^n \in V$ denote the current approximation to the solution of composite-grid equation (2.8).

ALGORITHM 4. *One iteration of the symmetrized AFACx algorithm consists of the following steps.*

For $k = 1, 2, \ldots, J$, *do:*
  if $(k = 1)$, *then*
    find $u_1^f \in V_1^{h_1}$ *such that*
      $a(u_c^n + u_1^f, v) = f(v) \ \ \forall v \in V_1^{h_1}$;
  else
    find $w_{k,0}^r \in V_k^{h_{k-1}}$ *such that*
      $b_k^r(w_{k,0}^r, z) = f(z) - a(u_c^n, z) \ \ \forall z \in V_k^{h_{k-1}}$;
    find $w_{k,0}^f \in V_k^{h_k}$ *such that*
      $b_k^f(w_{k,0}^r + w_{k,0}^f, v) = f(v) - a(u_c^n, v) \ \ \forall v \in V_k^{h_k}$;
    find $w_{k,1}^f \in V_k^{h_k}$ *such that*
      $b_k^f(w_{k,1}^f, v) = f(v) - a(u_c^n, v) \ \ \forall v \in V_k^{h_k}$;
    find $w_{k,1}^r \in V_k^{h_{k-1}}$ *such that*
      $b_k^r(w_{k,1}^r, z) = f(z) - a(u_c^n + w_{k,1}^f, z) \ \ \forall z \in V_k^{h_{k-1}}$;
    set $w_{k,1}^f = w_{k,1}^f + w_{k,1}^r$;
    set $w_{k,2}^f = w_{k,1}^f - w_{k,0}^r$;
    set $u_k^f = (w_{k,0}^f + w_{k,2}^f)/2$;
set $u_c^{n+1} = u_c^n + \sum_{k=1}^J u_k^f$.

The pseudolanguage above principally involves computing two approximations $w_{k,0}^f$ and $w_{k,2}^f$ and averaging them to form u_k^f at each level k. u_k^f then approximates the component of the composite-grid correction that can be represented at level k. $w_{k,0}^f$ is obtained in the following manner: smooth on the coarse-grid residual equation and interpolate to the local fine level to obtain $w_{k,0}^r$, then smooth on the fine level with initial guess $w_{k,0}^r$ to obtain $w_{k,0}^f$. To compute $w_{k,2}^f$, we first compute $w_{k,1}^f$ by applying a two-level correction scheme (as described in [20, p. 33]) on the local fine and restricted levels. $w_{k,2}^f$ is then set to be the difference $w_{k,1}^f - w_{k,0}^r$. In practice, the unsymmetric form of AFACx is used, while the symmetric form is useful for theoretical analysis.

4. Condition number estimates for AFACx. In this section, a new condition number estimate is developed for the multilevel AFACx algorithm. We show that the condition number of the symmetrized AFACx operator (Algorithm 4) is bounded independently of the number of refinement levels.

The following lemma, which is a generalization of the standard Cauchy–Schwarz inequality, is used extensively in the proofs that follow.

LEMMA 4.1 (see [47]). *Let $T \in \mathcal{L}(W)$ be a nonnegative self-adjoint operator with respect to $\langle\cdot,\cdot\rangle$, where $(W, \langle\cdot,\cdot\rangle)$ is a finite-dimensional inner product space and $\mathcal{L}(W)$ is the space of linear operators that map W into itself. Then*

$$|\langle Tu, v\rangle| \leq \langle Tu, u\rangle^{\frac{1}{2}} \langle Tv, v\rangle^{\frac{1}{2}} \quad \forall u, v \in W. \tag{4.1}$$

It is easy to show that the following lemma holds for operator $T_k = R_k A_k P_k$.

LEMMA 4.2 (see [11]). *Operator $T_k : V_J \longrightarrow V_J$, $k = 1, 2, \dots, J$, is nonnegative and self-adjoint with respect to the A inner product on V_J.*

4.1. Full refinement. Consider first the case of full refinement: $\Omega_1 = \Omega_2 = \cdots = \Omega_J$. Note that the "restricted coarse" grid is the entire global coarse grid, so that $V_k^{h_{k-1}} = V_{k-1}$, $k = 2, 3, \dots, J$, and the "local fine" grid is the entire global fine grid, so that $V_k^{h_k} = V_k$, $k = 1, 2, \dots, J$. Define $R_0 = 0$, $P_0 = 0$, and $Q_0 = 0$. Then the operator corresponding to one iteration of AFACx (Algorithm 3) with a single smoothing step each on the fine grid and the restricted coarse grid can be expressed as

$$B^a = \sum_{k=1}^{J} (R_k Q_k - R_k A_k R_{k-1} Q_{k-1}) A. \tag{4.2}$$

To avoid theoretical complications in satisfying assumption A.3 for $\theta \in (1, 2)$ for general smoothers, we work instead with the operator

$$B^a = \sum_{k=1}^{J} \left(R_k Q_k - \frac{1}{2} R_k A_k R_{k-1} Q_{k-1} \right) A, \tag{4.3}$$

which corresponds to damping the restricted coarse grid smoothing by an additional factor of $\frac{1}{2}$. Using relation (2.12), B^a may be rewritten as

$$B^a = \sum_{k=1}^{J} T_k \left(I - \frac{T_{k-1}}{2} \right), \tag{4.4}$$

where T_0 is identically zero. Expressing P_k as the telescoping series $\sum_{l=1}^{k} (P_l - P_{l-1})$, we can then write

$$B^a = \sum_{k=1}^{J} T_k (P_k - P_{k-1}) + \sum_{k=1}^{J} \sum_{l=1}^{k-1} T_k \left(I - \frac{T_{k-1}}{2} \right) (P_l - P_{l-1}). \tag{4.5}$$

Interchanging the order of summation in the second term in (4.5) allows us to rewrite B^a as

$$B^a = \sum_{l=1}^{J} T_l (P_l - P_{l-1}) + \sum_{l=1}^{J-1} \sum_{k=l+1}^{J} T_k \left(I - \frac{T_{k-1}}{2} \right) (P_l - P_{l-1}). \tag{4.6}$$

4.2. Richardson smoothing. First, consider the case when Richardson iteration is used as the smoother. We then have $R_k = \hat{R}_k$ and $T_k = \hat{T}_k$. Let $P_0 \equiv 0$ and define $w_l = (P_l - P_{l-1})v$, $l = 1, 2, \ldots, J$, for a given $v \in V_J$.

Our next lemma establishes a simple but important approximation property on each level.

LEMMA 4.3. *Let* T_k, $k = 2, 3, \ldots, J$, *satisfy bound* (2.22). *Then*

$$\begin{aligned} &A\left(T_k\left(I - \frac{T_{k-1}}{2}\right)w_l, \left(I - \frac{T_{k-1}}{2}\right)w_l\right) \\ &\qquad \leq \left(1 + \frac{\gamma\sqrt{\theta}}{2}\right)^2 (\gamma\epsilon^{k-l})^2 A(w_l, w_l), \quad k = 2, 3, \ldots, J,\ l < k. \end{aligned} \tag{4.7}$$

Proof. First, note that

$$\begin{aligned} &A\left(T_k\left(I - \frac{T_{k-1}}{2}\right)w_l, \left(I - \frac{T_{k-1}}{2}\right)w_l\right) \\ &\qquad \leq A(T_k w_l, w_l) + |A(T_k w_l, T_{k-1} w_l)| + \frac{1}{4} A(T_k T_{k-1} w_l, T_{k-1} w_l). \end{aligned} \tag{4.8}$$

Setting $T = T_k$, $u = w_l$, and $v = T_{k-1} w_l$ in Cauchy–Schwarz inequality (4.1), we have

$$|A(T_k w_l, T_{k-1} w_l)| \leq A(T_k w_l, w_l)^{\frac{1}{2}} A(T_k T_{k-1} w_l, T_{k-1} w_l)^{\frac{1}{2}}. \tag{4.9}$$

From (4.8) and (4.9), we thus have

$$\begin{aligned} &A\left(T_k\left(I - \frac{T_{k-1}}{2}\right)w_l, \left(I - \frac{T_{k-1}}{2}\right)w_l\right) \\ &\qquad \leq \left(A(T_k w_l, w_l)^{\frac{1}{2}} + \frac{1}{2} A(T_k T_{k-1} w_l, T_{k-1} w_l)^{\frac{1}{2}}\right)^2. \end{aligned} \tag{4.10}$$

Using assumption A.1 with $w = T_{k-1} w_l$ and applying assumption A.3, we see that the last term in (4.10) is bounded according to

$$A(T_k T_{k-1} w_l, T_{k-1} w_l) \leq (\gamma\epsilon)^2 \theta A(T_{k-1} w_l, w_l). \tag{4.11}$$

Applying assumption A.1 again, we have

$$A(T_{k-1} w_l, w_l) \leq (\gamma\epsilon^{k-l-1})^2 A(w_l, w_l). \tag{4.12}$$

Combining (4.11) and (4.12) yields

$$A(T_k T_{k-1} w_l, T_{k-1} w_l) \leq \gamma^2 \theta (\gamma\epsilon^{k-l})^2 A(w_l, w_l). \tag{4.13}$$

Now, using assumption A.1 to bound the first term on the right-hand side of (4.10) and (4.13) to bound the second term, we have

(4.14)

$$A\left(T_k\left(I - \frac{T_{k-1}}{2}\right)w_l, \left(I - \frac{T_{k-1}}{2}\right)w_l\right) \leq \left(1 + \frac{\gamma\sqrt{\theta}}{2}\right)^2 (\gamma\epsilon^{k-l})^2 A(w_l, w_l). \qquad \square$$

Before we prove the main results of this section, we state the following useful identity (cf. [11, 47]).

LEMMA 4.4. *Let* $w_l = (P_l - P_{l-1})v,\ v \in V_J,\ l = 1, 2, \ldots, J,$ *with* $P_0 = 0$. *Then*

$$\sum_{l=1}^{J} A(w_l, w_l) = A(v, v). \tag{4.15}$$

The next few lemmas are used to show that a symmetrized version of B^a has a uniformly bounded condition number. We first show that B^a is bounded uniformly in the A-norm.

LEMMA 4.5. *There exists constant* $C_1 > 0$*, independent of the number of levels* J*, such that*

$$A(B^a v, v) \le C_1 A(v, v) \quad \forall v \in V_J.$$

Proof. From (4.6), we have

$$\begin{aligned} A(B^a v, v) &= \sum_{l=1}^{J-1} \sum_{k=l+1}^{J} A\left(T_k\left(I - \frac{T_{k-1}}{2}\right) w_l, v\right) + \sum_{l=1}^{J} A(T_l w_l, v) \\ &= \sum_{l=1}^{J-1} \sum_{k=l+1}^{J} A\left(T_k\left(I - \frac{T_{k-1}}{2}\right) w_l, P_k v\right) + \sum_{l=1}^{J} A(T_l w_l, P_l v). \end{aligned}$$

Expressing P_k as a telescoping series, $P_k = \sum_{j=1}^{k}(P_j - P_{j-1})$, we have

$$\begin{aligned} A(B^a v, v) &= \sum_{l=1}^{J-1} \sum_{k=l+1}^{J} A\left(T_k\left(I - \frac{T_{k-1}}{2}\right) w_l, \sum_{j=1}^{k}(P_j - P_{j-1})v\right) + \sum_{l=1}^{J} A(T_l w_l, P_l v) \\ &= \sum_{l=1}^{J-1} \sum_{k=l+1}^{J} \sum_{j=1}^{k} A\left(T_k\left(I - \frac{T_{k-1}}{2}\right) w_l, w_j\right) + \sum_{l=1}^{J} A(T_l w_l, P_l v). \end{aligned} \tag{4.16}$$

Now, applying Cauchy–Schwarz inequality (4.1) with $T = T_k,\ u = (I - T_{k-1})w_l$, and $v = w_j$ yields

$$\begin{aligned} &A\left(T_k\left(I - \frac{T_{k-1}}{2}\right) w_l, w_j\right) \\ &\qquad \le A\left(T_k\left(I - \frac{T_{k-1}}{2}\right) w_l, \left(I - \frac{T_{k-1}}{2}\right) w_l\right)^{\frac{1}{2}} A(T_k w_j, w_j)^{\frac{1}{2}}. \end{aligned} \tag{4.17}$$

Bounding the first factor on the right-hand side of (4.17) using Lemma 4.3 and the second factor using assumption A.1 yields

$$A\left(T_k\left(I - \frac{T_{k-1}}{2}\right) w_l, w_j\right) \le \left(1 + \frac{\gamma\sqrt{\theta}}{2}\right)(\gamma\epsilon^{k-l})A(w_l, w_l)^{\frac{1}{2}}(\gamma\epsilon^{k-j})A(w_j, w_j)^{\frac{1}{2}}. \tag{4.18}$$

Finally, applying the arithmetic-geometric mean inequality in (4.18) yields

$$A\left(T_k\left(I - \frac{T_{k-1}}{2}\right) w_l, w_j\right) \le \frac{(1 + \frac{\gamma\sqrt{\theta}}{2})\gamma^2\epsilon^{2k-l-j}}{2}(A(w_l, w_l) + A(w_j, w_j)). \tag{4.19}$$

Using bound (4.19) for the individual terms in (4.16), we have

$$A(B^a v, v) \le \left(\frac{(1+\frac{\gamma\sqrt{\theta}}{2})\gamma^2}{2}\right)\left(\sum_{l=1}^{J-1}\sum_{k=l+1}^{J}\sum_{j=1}^{k}\epsilon^{2k-l-j}(A(w_l,w_l)+A(w_j,w_j))\right)$$
$$+\sum_{l=1}^{J} A(T_l w_l, P_l v). \tag{4.20}$$

We now bound each of the three terms on the right-hand side of (4.20) individually.

Term I. First write

$$S_1 \equiv \sum_{l=1}^{J-1}\sum_{k=l+1}^{J}\sum_{j=1}^{k}\epsilon^{2k-l-j}A(w_l,w_l) = \sum_{l=1}^{J-1}\sum_{k=l+1}^{J}\epsilon^{k-l}A(w_l,w_l)\left(\sum_{j=1}^{k}\epsilon^{k-j}\right).$$

Hence,

$$S_1 \le \left(\frac{1}{1-\epsilon}\right)\sum_{l=1}^{J-1}A(w_l,w_l)\left(\sum_{k=l+1}^{J}\epsilon^{k-l}\right).$$

Again, bounding terms involving powers of ϵ and applying Lemma 4.4, we have

$$S_1 \le \frac{\epsilon}{(1-\epsilon)^2}\sum_{l=1}^{J}A(w_l,w_l) = \frac{\epsilon}{(1-\epsilon)^2}A(v,v). \tag{4.21}$$

Term II. Let

$$S_2 \equiv \sum_{l=1}^{J-1}\sum_{k=l+1}^{J}\sum_{j=1}^{k}\epsilon^{2k-l-j}A(w_j,w_j)$$
$$= \sum_{l=1}^{J-1}\sum_{k=l+1}^{J}\epsilon^{k-l}\left(\sum_{j=1}^{k}\epsilon^{k-j}A(w_j,w_j)\right). \tag{4.22}$$

Now, let $\underline{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_J)^t$, where $\alpha_j = A(w_j, w_j)$, $j = 1, 2, \ldots, J$. Also, let $\mathcal{E} = (\mathcal{E}_{ij})_{1\le i,j\le J}$ denote the $J \times J$ lower triangular matrix with entries given by

$$\mathcal{E}_{ij} = \begin{cases} \epsilon^{i-j} & : \quad i \ge j, \\ 0 & : \quad i < j. \end{cases}$$

Finally, let $\underline{\beta}_l = (\beta_{1l}, \beta_{2l}, \ldots, \beta_{Jl})^t$, $l = 1, 2, \ldots, J$, denote column vectors with entries given by

$$\beta_{il} = \begin{cases} 0 & : \quad i \le l, \\ \epsilon^{i-l} & : \quad i > l, \end{cases}$$

and define $\tilde{\beta}^t = (1, 1, \ldots, 1)$. Then, (4.22) can be written as

$$S_2 = \sum_{l=1}^{J-1}\sum_{k=1}^{J}\beta_{kl}\left(\sum_{j=1}^{J}\mathcal{E}_{kj}\alpha_j\right) = \sum_{l=1}^{J-1}\underline{\beta}^t_l\mathcal{E}\,\underline{\alpha} = \left(\sum_{l=1}^{J-1}\underline{\beta}^t_l\right)\mathcal{E}\,\underline{\alpha}. \tag{4.23}$$

Each entry of the column vector $\sum_{l=1}^{J-1} \underline{\beta}^t_{\ l}$ can be bounded by $\frac{\epsilon}{1-\epsilon}$. Since all quantities are nonnegative,

$$S_2 \leq \left(\frac{\epsilon}{1-\epsilon}\right) \underline{\tilde{\beta}}^t \mathcal{E}\ \underline{\alpha} = \left(\frac{\epsilon}{1-\epsilon}\right) \sum_{k=1}^{J} \sum_{j=1}^{k} \mathcal{E}_{kj}\alpha_j. \tag{4.24}$$

Interchanging the order of summation and noting that $\mathcal{E}$ is lower triangular, we thus have

$$\begin{aligned} S_2 &\leq \left(\frac{\epsilon}{1-\epsilon}\right) \sum_{j=1}^{J} \sum_{k=j}^{J} \mathcal{E}_{kj}\alpha_j \\ &= \left(\frac{\epsilon}{1-\epsilon}\right) \sum_{j=1}^{J} \sum_{k=j}^{J} \epsilon^{k-j} A(w_j, w_j) \\ &= \left(\frac{\epsilon}{1-\epsilon}\right) \sum_{j=1}^{J} A(w_j, w_j) \left(\sum_{k=j}^{J} \epsilon^{k-j}\right) \\ &\leq \frac{\epsilon}{(1-\epsilon)^2} \sum_{j=1}^{J} A(w_j, w_j). \end{aligned} \tag{4.25}$$

Applying Lemma 4.4 in (4.25), we therefore have

$$S_2 \leq \frac{\epsilon}{(1-\epsilon)^2} A(v, v). \tag{4.26}$$

Term III. Again using the telescoping series $P_k = \sum_{l=1}^{k}(P_l - P_{l-1})$, we have

$$\begin{aligned} S_3 \equiv \sum_{k=1}^{J} A(T_k w_k, P_k v) &= A(P_1 v, P_1 v) + \sum_{k=2}^{J} A(T_k w_k, P_k v) \\ &= A(P_1 v, P_1 v) + \sum_{k=2}^{J} A\left(T_k w_k, \sum_{l=1}^{k}(P_l - P_{l-1})v\right) \\ &= A(P_1 v, P_1 v) + \sum_{k=2}^{J} \sum_{l=1}^{k} A(T_k w_k, w_l). \end{aligned}$$

Let $\hat{S}_3 = \sum_{k=2}^{J} \sum_{l=1}^{k} A(T_k w_k, w_l)$. Then applying Cauchy–Schwarz inequality (4.1) followed by assumption A.1 yields

$$\begin{aligned} \hat{S}_3 = \sum_{k=2}^{J} \sum_{l=1}^{k} A(T_k w_k, w_l) &\leq \sum_{k=2}^{J} \sum_{l=1}^{k} A(T_k w_k, w_k)^{\frac{1}{2}} A(T_k w_l, w_l)^{\frac{1}{2}} \\ &\leq \sum_{k=2}^{J} \sum_{l=1}^{k} \gamma A(w_k, w_k)^{\frac{1}{2}} (\gamma \epsilon^{k-l}) A(w_l, w_l)^{\frac{1}{2}} \\ &\leq \gamma^2 \sum_{k=1}^{J} \sum_{l=1}^{J} A(w_k, w_k)^{\frac{1}{2}} (\epsilon^{|k-l|}) A(w_l, w_l)^{\frac{1}{2}}. \end{aligned} \tag{4.27}$$

Set $\alpha_k = A(w_k, w_k)^{\frac{1}{2}}$. Then

$$\hat{S}_3 \le \gamma^2 \sum_{k=1}^{J} \sum_{l=1}^{J} (\epsilon^{|k-l|}) \alpha_k \alpha_l = \gamma^2 \langle\langle \hat{\mathcal{E}} \vec{\alpha}, \vec{\alpha} \rangle\rangle,$$

where $\hat{\mathcal{E}}$ is the $J \times J$ symmetric positive-definite matrix with entries $\epsilon^{|k-l|}$, $\vec{\alpha}$ is the $J \times 1$ column vector with entries α_k, $k = 1, 2, \ldots, J$, and $\langle\langle \cdot, \cdot \rangle\rangle$ is the Euclidean inner product. The largest eigenvalue of $\hat{\mathcal{E}}$ is bounded by its maximal row sum, which in turn is bounded by $\frac{2}{1-\epsilon}$. Therefore,

$$\begin{aligned} \hat{S}_3 &\le \gamma^2 \left(\frac{2}{1-\epsilon} \right) \langle\langle \vec{\alpha}, \vec{\alpha} \rangle\rangle \\ &= \left(\frac{2\gamma^2}{1-\epsilon} \right) \left(\sum_{k=1}^{J} \alpha_k^2 \right) \\ &= \left(\frac{2\gamma^2}{1-\epsilon} \right) \left(\sum_{k=1}^{J} A(w_k, w_k) \right). \end{aligned} \tag{4.28}$$

From Lemma 4.4 and (4.28), we thus have

$$\begin{aligned} S_3 &= A(P_1 v, P_1 v) + \hat{S}_3 \\ &\le A(v, v) + \left(\frac{2\gamma^2}{1-\epsilon} \right) A(v, v) \\ &= \left(1 + \frac{2\gamma^2}{1-\epsilon} \right) A(v, v). \end{aligned} \tag{4.29}$$

Substituting (4.21), (4.26), and (4.29) into (4.20), we therefore conclude that

$$A(B^a v, v) \le \left(\frac{(1 + \frac{\gamma \theta^{\frac{1}{2}}}{2}) \gamma^2 \epsilon}{(1-\epsilon)^2} + 1 + \frac{2\gamma^2}{1-\epsilon} \right) A(v, v), \tag{4.30}$$

which proves the lemma with

$$C_1 = \left(\frac{\left(1 + \frac{\gamma \theta^{\frac{1}{2}}}{2} \right) \gamma^2 \epsilon}{(1-\epsilon)^2} + 1 + \frac{2\gamma^2}{1-\epsilon} \right). \qquad \square$$

The next lemma shows that B^a is coercive in the A inner product.

LEMMA 4.6. *Under assumptions* A.3 *and* A.4, *there exists constant* $C_0 > 0$, *independent of the number of levels* J, *such that*

$$A(B^a v, v) \ge C_0 A(v, v) \quad \forall v \in V_J.$$

Proof. From (4.4), we have that

$$
\begin{aligned}
A(B^a v, v) &= \sum_{l=1}^{J} A\left(T_l\left(I - \frac{T_{l-1}}{2}\right)v, v\right) \\
&= \sum_{l=1}^{J} A(T_l v, v) - \sum_{l=1}^{J} \frac{1}{2} A(T_l T_{l-1} v, v) \qquad (4.31)\\
&= \sum_{l=1}^{J} A(T_l v, v) - \sum_{l=1}^{J} \frac{1}{2} A(T_{l-1} v, T_l v). \qquad (4.32)
\end{aligned}
$$

Applying the standard Cauchy–Schwarz inequality yields

$$
|A(T_{l-1}v, T_l v)| \le A(T_{l-1}v, T_{l-1}v)^{\frac{1}{2}} A(T_l v, T_l v)^{\frac{1}{2}}. \tag{4.33}
$$

Applying assumption A.3 to (4.33), we get

$$
|A(T_{l-1}v, T_l v)| \le \theta A(T_{l-1}v, v)^{\frac{1}{2}} A(T_l v, v)^{\frac{1}{2}}. \tag{4.34}
$$

Hence, using the standard Cauchy–Schwarz inequality and nonnegativeness of operator T_J with respect to the A inner product we have

$$
\begin{aligned}
\sum_{l=1}^{J} A(T_l T_{l-1} v, v) &\le \theta \sum_{l=1}^{J} A(T_{l-1}v, v)^{\frac{1}{2}} A(T_l v, v)^{\frac{1}{2}} \\
&\le \theta \left(\sum_{l=1}^{J} A(T_{l-1}v, v)\right)^{\frac{1}{2}} \left(\sum_{l=1}^{J} A(T_l v, v)\right)^{\frac{1}{2}} \\
&\le \theta \left(\sum_{l=1}^{J} A(T_l v, v)\right). \qquad (4.35)
\end{aligned}
$$

Relations (4.32) and (4.35) and assumption A.4 combine to show that

$$
\begin{aligned}
A(B^a v, v) &\ge \left(1 - \frac{\theta}{2}\right)\left(\sum_{l=1}^{J} A(T_l v, v)\right) \\
&\ge \frac{1}{2\eta}(2-\theta)A(v,v) \quad \forall v \in V_J. \qquad (4.36)
\end{aligned}
$$

∎

The following theorem is a direct consequence of Lemmas 4.5 and 4.6.

THEOREM 4.7. *There exists constant $C > 0$, independent of the number of levels J, such that*

$$
\kappa(B^s) \le C,
$$

where $B^s = \frac{1}{2}(B^a + (B^a)^*)$ is the operator corresponding to symmetrized AFACx (Algorithm 4), with $(B^a)^*$ denoting the adjoint of B^a with respect to the A inner product.

4.3. More general smoothers. The estimates established above apply when the smoother used on each level is a Richardson iteration. In practice, simple but more robust smoothers such as damped Jacobi or Gauss–Seidel are usually employed. Assumption A.2 now becomes important in establishing condition number estimates for AFACx with general symmetric smoothers on each level. Lemma 4.3 is restated as follows for general $R_k \neq \hat{R}_k$ that is symmetric in the L^2 inner product.

LEMMA 4.8. *Let T_k satisfy* (2.21). *Then*

$$\begin{aligned} &A\left(T_l\left(I-\frac{T_{l-1}}{2}\right)w_k, \left(I-\frac{T_{l-1}}{2}\right)w_k\right) \\ &\leq a_1\left(1+\frac{\gamma\sqrt{a_1\theta}}{2}\right)^2\left(\gamma\epsilon^{k-l}\right)^2 A(w_k,w_k), \quad l=2,3,\ldots,J,\ k<l. \end{aligned} \tag{4.37}$$

The proof is along the same lines as that for Lemma 4.3.

Also, Lemma 4.5 now reads as follows.

LEMMA 4.9. *We have*

$$A(B^a v, v) \leq C_1 A(v,v) \quad \forall v \in V_J,$$

where

$$C_1 = \left(a_1\left(1+\frac{\gamma\sqrt{a_1\theta}}{2}\right)\gamma^2\frac{\epsilon}{(1-\epsilon)^2}+1+\frac{2a_1\gamma^2}{(1-\epsilon)}\right).$$

Lemma 4.6 becomes the following.

LEMMA 4.10. *We have*

$$A(B^a v, v) \geq C_0 A(v,v) \quad \forall v \in V_J,$$

where $C_0 = \frac{a_0}{2\eta}(2-\theta)$.

For symmetric smoothers that are spectrally equivalent to Richardson iteration, the condition number of the symmetrized AFACx operator is therefore again bounded independently of the number of levels.

4.4. Partial refinement. For the case of partial refinement, local "restricted coarse" $V_k^{h_{k-1}}$ is a subspace of $V_k^{h_k} \cap V_{k-1}$, and local "fine" $V_k^{h_k}$ is a subspace of V_k, $k=2,3,\ldots,J$. However, spaces $V_k^{h_k}$, $k=2,3,\ldots,J$, need not be nested. To treat this more general setting, we need to define operators at the different levels, projection operators between levels, and smoothing operators. Note that, in what follows, superscripts of "f" and "r" denote linear operators mapping to local "fine" $V_k^{h_k}$ and "restricted coarse" $V_k^{h_{k-1}}$, respectively, for given level k.

DEFINITION 4. *For $k=2,\ldots,J$, define operator $A_k^f : V_k^{h_k} \longrightarrow V_k^{h_k}$ by*

$$(A_k^f w, \phi) = A(w,\phi) \quad \forall \phi \in V_k^{h_k},\ w \in V_k^{h_k}.$$

DEFINITION 5. *For $k=2,\ldots,J$, define operator $A_k^r : V_k^{h_{k-1}} \longrightarrow V_k^{h_{k-1}}$ by*

$$(A_k^r w, \phi) = A(w,\phi) \quad \forall \phi \in V_k^{h_{k-1}},\ w \in V_k^{h_{k-1}}.$$

Orthogonal "elliptic" projection operators P_k^f, $k=1,2,\ldots,J$, and P_k^r, $k=2,3,\ldots,J$, are defined as follows.

DEFINITION 6. $P_k^f : V_J \longrightarrow V_k^{h_k}$ *is defined by*

$$A(P_k^f w, \phi) = A(w, \phi) \quad \forall \phi \in V_k^{h_k},\ w \in V_J.$$

DEFINITION 7. $P_k^r : V_J \longrightarrow V_k^{h_{k-1}}$ *is defined by*

$$A(P_k^r w, \phi) = A(w, \phi) \quad \forall \phi \in V_k^{h_{k-1}},\ w \in V_J.$$

Orthogonal "L^2" projection operators Q_k^f, $k = 1, 2, \ldots, J$, and Q_k^r, $k = 2, 3, \ldots, J$, are defined as follows.

DEFINITION 8. $Q_k^f : V_J \longrightarrow V_k^{h_k}$ *is defined by*

$$(Q_k^f w, \phi) = (w, \phi) \quad \forall \phi \in V_k^{h_k},\ w \in V_J.$$

DEFINITION 9. $Q_k^r : V_J \longrightarrow V_k^{h_{k-1}}$ *is defined by*

$$(Q_k^r w, \phi) = (w, \phi) \quad \forall \phi \in V_k^{h_{k-1}},\ w \in V_J.$$

Symmetric positive-definite smoothing operators $R_k^f : V_k \longrightarrow V_k^{h_k}$ and $R_k^r : V_{k-1} \longrightarrow V_k^{h_{k-1}}$ are also assumed to be defined.

The following relationships hold between the various operators: $Q_k^f A = A_k^f P_k^f$, $Q_k^r A = A_k^r P_k^r$, $R_k^f = R_k^f Q_k^f$, and $R_k^r = R_k^r Q_k^r$, $k = 2, 3, \ldots, J$.

For the case of partial refinement, we present the proof for only the AFACx operator with Richardson iteration as the smoother. It is obvious by now that the case of more general symmetric smoothers that are spectrally equivalent to Richardson iteration is easily handled through conditions like assumption A.2. Henceforth, let $R_1^f = (A_1^f)^{-1}$, $R_k^f = \frac{1}{\lambda_k} I$, $k = 2, 3, \ldots, J$, and $R_k^r = \frac{1}{\lambda_{k-1}} I$, $k = 2, 3, \ldots, J$. Define $T_k^f = R_k^f A_k^f P_k^f$ and $T_k^r = R_k^r A_k^r P_k^r$.

The following lemma is needed for the case of partial refinement.

LEMMA 4.11. *We have*

$$A(T_k^r v, v) \le A(T_{k-1}^f v, v) \quad \forall v \in V_J,\ k = 2, 3, \ldots, J. \tag{4.38}$$

Proof. From the basic properties of the L^2 projection operators listed in section 2.7, we have

$$\begin{aligned}
\|(Q_{k-1}^f - Q_k^r)u\|^2 &= ((Q_{k-1}^f - Q_k^r)u,\ (Q_{k-1}^f - Q_k^r)u) \\
&= \|Q_{k-1}^f u\|^2 + \|Q_k^r u\|^2 - 2(Q_{k-1}^f u, Q_k^r u) \\
&= \|Q_{k-1}^f u\|^2 + \|Q_k^r u\|^2 - 2(Q_{k-1}^f u, (Q_k^r)^2 u) \\
&= \|Q_{k-1}^f u\|^2 + \|Q_k^r u\|^2 - 2(Q_k^r Q_{k-1}^f u, Q_k^r u) \\
&= \|Q_{k-1}^f u\|^2 + \|Q_k^r u\|^2 - 2(Q_k^r u, Q_k^r u) \\
&= \|Q_{k-1}^f u\|^2 - \|Q_k^r u\|^2 \qquad \forall u \in V_J,\ k = 2, 3, \ldots, J. \qquad (4.39)
\end{aligned}$$

Since $\|(Q_{k-1}^f - Q_k^r)u\|^2 \ge 0$, (4.39) implies that

$$\|Q_k^r u\|^2 \le \|Q_{k-1}^f u\|^2 \quad \forall u \in V_J,\ k = 2, 3, \ldots, J. \tag{4.40}$$

Let $u = Av$. Then

$$\begin{aligned}
\|Q_k^r Av\|^2 &\le \|Q_{k-1}^f Av\|^2 \\
\Rightarrow \|A_k^r P_k^r v\|^2 &\le \|A_{k-1}^f P_{k-1}^f v\|^2 \\
\Rightarrow \frac{1}{\lambda_{k-1}}(A_k^r P_k^r v, A_k^r P_k^r v) &\le \frac{1}{\lambda_{k-1}}(A_{k-1}^f P_{k-1}^f v, A_{k-1}^f P_{k-1}^f v) \\
\Rightarrow (R_k^r A_k^r P_k^r v, A_k^r P_k^r v) &\le (R_{k-1}^f A_{k-1}^f P_{k-1}^f v, A_{k-1}^f P_{k-1}^f v) \\
\Rightarrow A(T_k^r v, v) &\le A(T_{k-1}^f v, v). \qquad \square
\end{aligned}$$

Assumptions similar to A.1 and A.3 in the previous section are made for the case of partial refinement. We refer to [14] for proof that the assumptions made below are valid in the case of partial refinement.

A.5. There exist constants $\epsilon \in (0, 1)$ and $\gamma > 0$ such that

$$A(T_k^f w, w) \le (\gamma \epsilon^{k-l})^2 A(w, w) \quad \forall w \in V_l^{h_l},\ l \le k, \quad k = 1, 2, \ldots, J. \tag{4.41}$$

Then, from Lemma 4.11 and assumption A.5, we have

$$A(T_k^r w, w) \le (\gamma \epsilon^{k-l-1})^2 A(w, w) \quad \forall w \in V_l^{h_l},\ l \le k-1, \quad k = 2, \ldots, J. \tag{4.42}$$

A.6. There exists constant $\theta \in (0, 2)$ such that

$$A(T_k^f v, T_k^f v) \le \theta A(T_k^f v, v) \quad \forall v \in V_J \tag{4.43}$$

and

$$A(T_k^r v, T_k^r v) \le \theta A(T_k^r v, v) \quad \forall v \in V_J, \quad k = 1, 2, \ldots, J. \tag{4.44}$$

In addition to the assumptions on the smoothers, a weak regularity assumption analogous to A.4 is also needed.

A.7. There exists a constant $\eta > 0$ such that

$$A(v, v) \le \eta \sum_{k=1}^{J} A(T_k^f v, v) \quad \forall v \in V_J. \tag{4.45}$$

Finally, we make the following assumption.

A.8. Range$(P_k - P_{k-1}) \subseteq V_k^{h_k}$, $\ k = 1, 2, \ldots, J$.

AFACx operator B^a for the case of partial refinement can be written as

$$\begin{aligned}
B^a &= \sum_{k=1}^{J} R_k^f A_k^f P_k^f \left(I - \frac{1}{2} R_k^r A_k^r P_k^r \right) \\
&= \sum_{k=1}^{J} T_k^f \left(I - \frac{T_k^r}{2} \right).
\end{aligned} \tag{4.46}$$

The proofs of the following lemmas are virtually the same as the proofs for Lemmas 4.5 and 4.6, respectively. Assumptions A.5–A.8 and Lemma 4.11 take the place of assumptions A.1, A.3, and A.4.

Lemma 4.12. *Under assumptions* A.5, A.6, *and* A.8, *there exists a constant* $C_1 > 0$*, independent of the number of levels* J*, such that*

$$A(B^a v, v) \le C_1 A(v, v) \ \ \forall v \in V_J. \tag{4.47}$$

LEMMA 4.13. *Under assumptions* A.5–A.8, *there exists a constant* $C_0 > 0$, *independent of the number of levels* J, *such that*

$$A(B^a v, v) \geq C_0 A(v, v) \ \ \forall v \in V_J. \tag{4.48}$$

The following theorem follows immediately from Lemmas 4.12 and 4.13.

THEOREM 4.14. *There exists constant* $C > 0$, *independent of the number of levels* J, *such that*

$$\kappa(B^s) \leq C,$$

where $B^s = \frac{1}{2}(B^a + (B^a)^*)$ *is the operator corresponding to symmetrized AFACx (Algorithm* 4*).*

5. Conclusions. In this paper, we have presented a new multilevel condition number estimate for the AFACx algorithm. This estimate shows that the condition number of the AFACx operator does not degrade as the number of refinement levels in the AMR hierarchy increases. Numerical results supporting these theoretical estimates are presented in a forthcoming paper.

REFERENCES

[1] APPLIED NUMERICAL ALGORITHMS GROUP, *Chombo Reference Manual*, available online at http://seesar.lbl.gov/anag/software/chombo.html.

[2] S. BADEN, N. CHRISOCHOIDES, D. GANNON, AND M. NORMAN, EDS., *Structured Adaptive Mesh Refinement (SAMR) Grid Methods*, IMA Vol. Math. Appl. 117, Springer, New York, 2000.

[3] R. E. BANK, T. F. DUPONT, AND H. YSERENTANT, *The hierarchical basis multigrid method*, Numer. Math., 52 (1988), pp. 427–458.

[4] M. BERGER, *Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations*, Ph.D. thesis, Stanford University, Stanford, CA, 1982.

[5] M. BERGER AND R. LEVEQUE, *AMRCLAW software*, available online at http://www.amath.washington.edu/r̃jl/amrclaw.

[6] M. J. BERGER AND J. OLIGER, *Adaptive mesh refinement for hyperbolic partial differential equations*, J. Comput. Phys., 53 (1984), pp. 484–512.

[7] P. E. BJORSTAD, R. MOE, AND M. D. SKOGEN, *Parallel domain decomposition and iterative refinement algorithms*, in Parallel Algorithms for Partial Differential Equations (Proceedings of the Sixth GAMM-Seminar, Kiel, 1990), W. Hackbusch, ed., Notes Numer. Fluid Mech. 31, Vieweg Verlag, Wiesbaden, 1990, pp. 216–227.

[8] F. BORNEMANN AND H. YSERENTANT, *A basic norm equivalence for the theory of multilevel methods*, Numer. Math., 64 (1993), pp. 455–476.

[9] F. A. BORNEMANN, *Interpolation spaces and optimal multilevel preconditioners*, in Domain Decomposition Methods in Scientific and Engineering Computing, D. E. Keyes and J. Xu, eds., Contemp. Math. 180, AMS, Providence, RI, 1994, pp. 3–8.

[10] D. BRAESS, *Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics*, Cambridge University Press, Cambridge, UK, 1997.

[11] J. BRAMBLE, *Multigrid Methods*, Pitman Research Notes in Mathematics Series 24, Longman Scientific and Technical, Harlow, UK, 1993.

[12] J. BRAMBLE AND J. PASCIAK, *New convergence estimates for multigrid algorithms*, Math. Comp., 49 (1987), pp. 311–329.

[13] J. BRAMBLE AND J. PASCIAK, *The analysis of smoothers for multigrid algorithms*, Math. Comp., 58 (1992), pp. 467–488.

[14] J. BRAMBLE AND J. PASCIAK, *New estimates for multilevel algorithms including the V-cycle*, Math. Comp., 60 (1993), pp. 447–471.

[15] J. BRAMBLE, J. PASCIAK, J. WANG, AND J. XU, *Convergence estimates for multigrid algorithms without regularity assumptions*, Math. Comp., 57 (1991), pp. 23–45.

[16] J. BRAMBLE, J. PASCIAK, AND J. XU, *Parallel multilevel preconditioners*, Math. Comp., 55 (1990), pp. 1–22.

[17] J. BRAMBLE AND J. XU, *Some estimates for a weighted L^2 projection*, Math. Comp., 56 (1991), pp. 463–476.

[18] J. H. Bramble, J. E. Pasciak, and P. S. Vassilevski, *Computational scales of Sobolev norms with application to preconditioning*, Math. Comp., 69 (1999), pp. 463–480.
[19] A. Brandt, *Multi-level adaptive solutions to boundary-value problems*, Math. Comp., 31 (1977), pp. 333–390.
[20] W. L. Briggs, V. E. Henson, and S. F. McCormick, *A Multigrid Tutorial*, 2nd ed., SIAM, Philadelphia, 2000.
[21] H. Cheng, *Iterative Solution of Elliptic Finite Element Problems on Partially Refined Meshes and the Effect of Using Inexact Solvers*, Ph.D. thesis, Courant Institute of Mathematical Sciences, New York University, New York, 1993.
[22] H. Cheng, *On the optimality of some FAC and AFAC methods for elliptic finite element problems*, Taiwanese J. Math., 2 (1998), pp. 405–426.
[23] M. Dryja and O. B. Widlund, *On the optimality of an additive iterative refinement method*, in Proceedings of the Fourth Copper Mountain Conference on Multigrid Methods, J. Mandel, S. F. McCormick, J. E. Dendy, C. Farhat, G. Lansdale, S. V. Porter, J. W. Ruge, and K. Stüben, eds., SIAM, Philadelphia, 1989, pp. 161–170.
[24] P. Ferket and A. Reusken, *Further analysis of the local defect correction method*, Computing, 56 (1996), pp. 117–139.
[25] D. Gilbarg and N. Trudinger, *Elliptic Partial Differential Equations of Second Order*, Springer-Verlag, New York, 1977.
[26] M. Griebel and P. Oswald, *On the abstract theory of additive and multiplicative Schwarz algorithms*, Numer. Math., 70 (1995), pp. 163–180.
[27] W. Hackbusch, *Local defect correction method and domain decomposition techniques*, in Defect Correction Methods, K Böhner and H. J. Stetter, eds., Comput. Suppl. 5, Springer, Vienna, 1984, pp. 89–113.
[28] L. Hart and S. McCormick, *Asynchronous multilevel adaptive methods for solving partial differential equations: Basic ideas*, Parallel Comput., 12 (1989), pp. 131–144.
[29] S. Kohn, X. Garaizar, R. Hornung, and S. Smith, *SAMRAI*, available online at http://www.llnl.gov/CASC/SAMRAI/.
[30] B. Lee, S. F. McCormick, B. Philip, and D. J. Quinlan, *Asynchronous fast adaptive composite-grid methods: Numerical results*, SIAM J. Sci. Comput., 25 (2003), pp. 682–699.
[31] J. Mandel and S. McCormick, *Iterative solution of elliptic equations with refinement: The model multi-level case*, in Domain Decomposition Methods (Los Angeles), T. F. Chan, R. Glowinski, J. Periaux, and O. B. Widlund, eds., SIAM, Philadelphia, 1989, pp. 81–92.
[32] S. McCormick, *Fast adaptive composite grid (FAC) methods: Theory for the variational case*, in Defect Correction Methods: Theory and Applications, K. Böhmer and H. J. Stetter, eds., Comput. Suppl. 5, Springer, Vienna, 1984, pp. 115–122.
[33] S. F. McCormick, *Multilevel Adaptive Methods for Partial Differential Equations*, Frontiers Appl. Math. 6, SIAM, Philadelphia, 1989.
[34] S. F. McCormick, *Multilevel Projection Methods for Partial Differential Equations*, CBMS-NSF Reg. Conf. Ser. Appl. Math. 62, SIAM, Philadelphia, 1992.
[35] S. McCormick, S. McKay, and J. Thomas, *Computational complexity of the fast adaptive composite grid (FAC) method*, Appl. Numer. Math., 6 (1989), pp. 315–327.
[36] S. McCormick and D. Quinlan, *Asynchronous multilevel adaptive methods for solving partial differential equations on multiprocessors: Performance results*, Parallel Comput., 12 (1989), pp. 145–156.
[37] S. McCormick and D. Quinlan, *Idealized analysis of asynchronous multilevel methods*, in Adaptive Multilevel and Hierarchical Computational Strategies, Appl. Mech. Div. Ser. 157, American Society of Mechanical Engineers, New York, 1992, pp. 1–8.
[38] S. McCormick and J. Thomas, *The fast adaptive composite-grid method for elliptic equations*, Math. Comp., 46 (1986), pp. 439–456.
[39] R. Moe, *Iterative Local Uniform Mesh Refinement Methods and Parallel Processing*, Ph.D. thesis, University of Bergen, Bergen, Norway, 1992.
[40] B. Philip, *Asynchronous Fast Adaptive Composite Grid Methods for Elliptic Problems on Adaptively Refined Curvilinear Grids*, Ph.D. thesis, University of Colorado at Boulder, Boulder, CO, 2001.
[41] D. Quinlan, *Adaptive Mesh Refinement for Distributed Parallel Architectures*, Ph.D. thesis, University of Colorado at Denver, Denver, CO, 1993.
[42] D. Quinlan, *AMR++ Manual*, Technical report LA-UR-97-4325, Los Alamos National Laboratory, Los Alamos, NM, 1997.
[43] D. Quinlan, *AMR++ Tutorial*, Technical report, Lawrence Livermore National Laboratory, Livermore, CA, 2000.

[44] Y. Shapira, *Multigrid for locally refined meshes*, SIAM J. Sci. Comput., 21 (1999), pp. 1168–1190.

[45] O. B. Widlund, *Optimal iterative refinement methods*, in Domain Decomposition Methods (Los Angeles), T. F. Chan, R. Glowinski, J. Periaux, and O. B. Widlund, eds., SIAM, Philadelphia, 1989, pp. 114–125.

[46] J. Xu, *Theory of Multilevel Methods*, Ph.D. thesis, Cornell University, Ithaca, NY, 1989.

[47] J. Xu, *Iterative methods by space decomposition and subspace correction*, SIAM Rev., 34 (1992), pp. 581–613.

[48] J. Xu, *An introduction to multigrid convergence theory*, in Iterative Methods in Scientific Computing, T. Chan, R. Chan, and G. Golub, eds., Springer-Verlag, Singapore, 1997, pp. 169–241.

[49] H. Yserentant, *Old and new convergence proofs for multigrid methods*, Acta Numer., (1993), pp. 285–326.

[50] X. Zhang, *Multilevel Schwarz methods*, Numer. Math., 63 (1992), pp. 521–539.