AIAA JOURNAL
Vol. 48, No. 4, April 2010

Critical Study of Agglomerated Multigrid Methods for Diffusion

Hiroaki Nishikawa* and Boris Diskin†
National Institute of Aerospace, Hampton, Virginia 23666
and
James L. Thomas‡
NASA Langley Research Center, Hampton, Virginia 23681

DOI: 10.2514/1.J050055

Agglomerated multigrid techniques used in unstructured-grid methods are studied critically for a model problem representative of laminar diffusion in the incompressible limit. The studied target-grid discretizations and discretizations used on agglomerated grids are typical of current node-centered formulations. Agglomerated multigrid convergence rates are presented using a range of two- and three-dimensional randomly perturbed unstructured grids for simple geometries with isotropic and stretched grids. Two agglomeration techniques are used within an overall topology-preserving agglomeration framework. The results show that a multigrid with an inconsistent coarse-grid scheme using only the edge derivatives (also referred to in the literature as a thin-layer formulation) provides considerable speedup over single-grid methods, but its convergence can deteriorate on highly skewed grids. A multigrid with a Galerkin coarse-grid discretization using piecewise-constant prolongation and a heuristic correction factor is slower and also can be grid dependent. In contrast, nearly grid-independent convergence rates are demonstrated for a multigrid with consistent coarse-grid discretizations. Convergence rates of multigrid cycles are verified with quantitative analysis methods in which parts of the two-grid cycle are replaced by their idealized counterparts.

I. Introduction

MULTIGRID techniques [1] are used to accelerate convergence of current Reynolds averaged Navier–Stokes solvers for steady and unsteady flow solutions, especially for structured-grid applications. Mavriplis [2–4] and Mavriplis and Pirzadeh [5] pioneered agglomerated multigrid methods for large-scale unstructured-grid applications. Impressive improvements in efficiency over single-grid computations have been demonstrated. During a recent development of multigrid methods for unstructured grids [6], it was realized that some of the current approaches for coarse-grid discretization of viscous fluxes used in state-of-the-art codes have serious limitations on highly refined grids. The purpose of this paper is to critically study the current techniques for a simple Poisson equation (representing laminar diffusion in the incompressible limit), assess their performance in grid refinement, and develop improved approaches.

The paper is organized as follows. The model diffusion equation and control-volume partitions are presented from a general finite volume discretization (FVD) standpoint in Sec. II. Elements of multigrid algorithms are described, including a tabulation of target and coarse-grid discretizations in Sec. III. Quantitative analysis methods, in which parts of the actual multigrid cycle are replaced by their idealized counterparts, are described in Sec. IV. The target grids and typical agglomerated grids developed within a topology-preserving framework are shown in Sec. V, followed by two- and three-dimensional results in Secs. VI and VII, respectively. Results from applying analysis methods to 3-D computations are also reported in Sec. VII. Section VIII contains conclusions.

Received 10 July 2009; revision received 18 September 2009; accepted for publication 1 October 2009. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0001-1452/10 and $10.00 in correspondence with the CCC.

*Research Scientist, 100 Exploration Way; hiro@nianet.org.
†100 Exploration Way; bdiskin@nianet.org; currently Visiting Associate Professor, Mechanical and Aerospace Engineering Department, University of Virginia, Charlottesville. Associate Fellow AIAA.
‡Senior Research Scientist, Computational Aerosciences Branch, Mail Stop 128; James.L.Thomas@nasa.gov. Fellow AIAA.

II. Model Diffusion Equation and Boundary Conditions

The FVD schemes considered are derived from the integral form of the diffusion equation,

$$\oint_{\Gamma} (\nabla U \cdot \hat{\mathbf{n}})\, \mathrm{d}\Gamma = \iint_{\Omega} f\, \mathrm{d}\Omega \tag{1}$$

where f is a forcing function independent of the solution U, Ω is a control volume with boundary Γ, $\hat{\mathbf{n}}$ is the outward unit normal vector, and ∇U is the solution gradient vector. The boundary conditions are taken as Dirichlet, that is, specified from a known exact solution over the computational boundary. Tests are performed for simple manufactured solutions, namely, collections of polynomial or sine functions. The corresponding forcing functions are found by substituting these solutions into the differential form of the diffusion equation,

$$\Delta U = f \tag{2}$$

and boundary conditions. The *discretization error*, $E_d = U - U^h$, is defined as the difference between the exact continuous solution, U, to the differential Eq. (2) and the exact discrete solution, U^h, of the discretized Eq. (1). The *algebraic error* is the difference between the approximate and exact discrete solutions. A scheme is considered as design-order accurate if its discretization errors computed on a sequence of consistently refined grids [7,8] converge with the design order in the norm of interest.

The general FVD approach requires partitioning the domain into a set of nonoverlapping control volumes and numerically implementing Eq. (1) over each control volume. Node-centered schemes define solution values at the mesh nodes. In two dimensions, the primal meshes are composed of triangular and quadrilateral cells; in three dimensions, the primal cells are tetrahedral, prismatic, pyramidal, or hexahedral. The *median-dual* partition [9,10] used to generate control volumes is illustrated in Fig. 1 for two dimensions. These nonoverlapping control volumes cover the entire computational domain and compose a mesh that is dual to the primal mesh.

The control volumes of each agglomerated grid are found by summing control volumes of a finer grid. Any agglomerated grid can be defined in terms of a conservative agglomeration operator, R_0, as

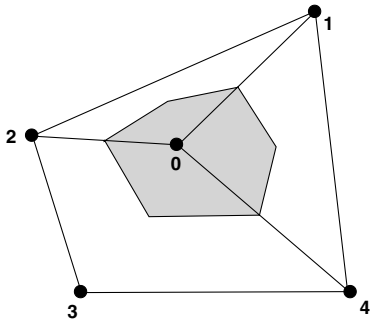


Fig. 1 Illustration of a node-centered median-dual control volume (shaded). Dual faces connect edge midpoints with primal cell centroids. Numbers 0–4 denote grid nodes.

$$\Omega^c = R_0 \Omega^f \tag{3}$$

where the superscripts c and f denote entities on coarser and finer grids, respectively. On the agglomerated grids, the control volumes become geometrically more complex than their primal counterparts and the details of the control-volume boundaries are not retained. The directed area of a coarse-grid face separating two agglomerated control volumes, if required, is found by lumping the directed areas of the corresponding finer-grid faces and is assigned to the virtual edge connecting the centers of the agglomerated control volumes.

III. Multigrid

Elements of the multigrid algorithm are presented in this section. A V cycle [1], denoted as $V(\nu_1, \nu_2)$, uses ν_1 relaxations performed at each grid before proceeding to the coarser grid and ν_2 relaxations after coarse-grid correction; the coarsest grid is solved exactly (with many relaxations). Residuals, r, corresponding to the integral equation (1) are restricted to the coarse grid using R_0, as

$$r^c = R_0 r^f \tag{4}$$

The prolongations P_0 and P_1 are exact for piecewise-constant and linear functions, respectively. The prolongation P_0 is the transpose of R_0. The operator P_1 is constructed locally using linear interpolation from a triangle (two dimensions) or tetrahedra (three dimensions) defined on the coarse grid. The geometrical shape is anchored at the coarser-grid location of the agglomerate that contains the given finer control volume. Other nearby points are found using the adjacency graph. An enclosing simplex is sought that avoids prolongation with nonconvex weights and, in situations in which multiple geometrical shapes are found, the first one encountered is used. Where no enclosing simplex is found, the simplex with minimal nonconvex weights is used. The coarse-grid solution approximation is restricted as

$$U^c = \frac{R_0(U^f \Omega^f)}{\Omega^c} \tag{5}$$

The correction δU to the finer grid is prolonged typically through P_1 as

$$(\delta U)^f = P_1 (\delta U)^c \tag{6}$$

The available consistent target-grid discretizations are the Green–Gauss and the average least squares (Avg-LSQ). These schemes are representative of viscous discretizations used in Reynolds averaged Navier–Stokes unstructured-grid codes. The main target discretization of interest is the Green–Gauss scheme [6], which is the most widely used viscous discretization for node-centered schemes and is equivalent to a Galerkin finite element discretization for triangular/tetrahedral grids. For mixed elements, edge derivatives are used to increase the h ellipticity [1] of the operator and thus avoid checkerboard instabilities [6,10]. Typically, the flux at a face is formed by the edge derivative computed as the divided difference of the solutions at the edge nodes and the Green–Gauss gradient projected onto the directions normal to the edge. The Avg-LSQ scheme defines the flux by the edge derivative and the average of the dual-volume least-squares (LSQ) gradients projected onto the directions normal to the edge [10,11]. The stencils for the dual-volume LSQ gradients include all edge-connected neighbors. The LSQ minimization enforces the given solution at the central node. In both formulations, Dirichlet boundary conditions are implemented strongly.

The exact linear operator is used in the iterative phase of the Green–Gauss scheme, enabling a robust multicolor Gauss–Seidel relaxation. The Avg-LSQ scheme has a comparatively larger stencil, and its exact linearization is not used in iterations; instead, relaxation of the Avg-LSQ scheme relies on an approximate edge-terms-only linearization, which approximates face gradients as edge derivatives. So far, we observe good smoothing rates with this approach, but previous analysis has shown that the smoothing rate can deteriorate on highly skewed grids [6]. The estimates for the smoothing rates obtained with quantitative analysis methods [12] are shown in Sec. VI. The Green–Gauss scheme relies on an element-based data structure and is not considered for agglomerated grids. Note that the Green–Gauss scheme can be written as an edge-based formulation for simplicial grids.

The available coarse-grid discretizations are two possible direct discretizations (Avg-LSQ and edge terms only) and two possible Galerkin discretizations ($R_0 A^f P_0^*$ and $R_0 A^f P_1$) in which the coarse-grid operators are derived from the fine-grid operator. Dirichlet boundary conditions are enforced strongly. The coarse-grid operator is overwritten with the boundary condition linearization at boundary nodes.

The edge-terms-only discretization is often cited as a thin-layer discretization in the literature [2,3,5]; it is a positive scheme but on nonorthogonal grids it is not consistent (i.e., its discrete solution does not converge to the exact continuous solution with consistent grid refinement) [7,8,13]. An orthogonal grid would have each edge node across a face be colinear with the corresponding directed area vector. Another possible coarse-grid discretization strategy, not considered here, is to construct simplicial grids from the coarse-grid vertices.

The Galerkin coarse-grid operator [1] is denoted by RAP. Because the governing equation is a second-order equation, the Galerkin construction, $R_0 A^f P_0$, is formally inconsistent [2,3]; the heuristic correction factor adopted by Mavriplis [2] is used:

$$A^c = R_0 A^f P_0^* = \frac{1}{2} R_0 A^f P_0 \tag{7}$$

The correction factor, applied per agglomerated cell, is derived by enforcing consistency on uniformly agglomerated hexahedral meshes. The Galerkin construction, $R_0 A^f P_1$, is consistent, but was found to be unstable in a multigrid.

IV. Quantitative Analysis of Unstructured Multigrid Solvers

The quantitative analysis methods for unstructured multigrid solvers considered in this section are idealized relaxation (IR) and idealized coarse-grid (ICG) iterations, introduced in [12]. The methods analyze the main complementary parts of a multigrid cycle: relaxation and coarse-grid correction. In a multigrid, relaxation and coarse-grid correction are assigned certain tasks: relaxation is required to smooth the algebraic error, and coarse-grid correction is required to reduce smooth algebraic errors.

To apply the analysis, we first choose a desired sample fine-grid solution (zero is a natural choice for linear problems) and substitute it into the equations to generate the corresponding source and boundary data. Then we form an initial guess (for example, a random perturbation of the solution); thus, the fine-grid algebraic error is known. In the analysis, idealized iterations probe the actual two-grid cycle to identify parts limiting the overall efficiency. In these iterations, one part of the cycle is actual, and its complementary part is replaced with an idealized part. The idealized parts do not depend on the operators to be solved. They are numerical procedures acting

directly on the known algebraic error to efficiently fulfill the task assigned to the corresponding part of the two-grid cycle. The results of the analysis are not single-number estimates; they are rather convergence patterns of the iterations that may either confirm or refute our expectations as to what part of the actual cycle is not efficient in carrying out the assigned task. These IR and ICG analysis methods can be regarded as a numerical extension of the Fourier analysis to problems in which the classical Fourier analysis is inapplicable, in particular, to unstructured-grid solvers.

IR and ICG iterations are analysis methods that test computational efficiency of a two-grid cycle. The two-grid cycle amplification matrix, M, transforms the initial fine-grid algebraic error, $\mathbf{e}^{\text{old}}$, into the after-cycle error, $\mathbf{e}^{\text{new}}$:

$$\mathbf{e}^{\text{new}} = M\mathbf{e}^{\text{old}} \tag{8}$$

The amplification matrix can be defined as

$$M = S^{\nu_2} C S^{\nu_1} \tag{9}$$

Here, ν_1 and ν_2 are small nonnegative integers representing the number of pre- and postrelaxation sweeps, S is the fine-grid relaxation amplification matrix, and C is the amplification matrix of the coarse-grid correction:

$$C = E - P_0 (A^c)^{-1} R_0 A^f \tag{10}$$

where A^c and A^f are the coarse and fine-grid operator matrices, P_0 and R_0 are the prolongation and agglomeration matrices, and E is the fine-grid identity matrix.

For IR iterations, the coarse-grid correction part is actual and the relaxation is idealized. The idealized relaxation may be defined as an explicit error-averaging procedure. In this paper, we employ the IR procedure that replaces the algebraic error at each dual cell with an average of algebraic errors at edge-adjacent cells. At each relaxation step, the known exact solution, if not zero, is subtracted from the current approximation to obtain the algebraic error function. The explicit averaging procedure is applied directly to the error function. The number of sweeps throughout the grid is taken as ν_1 or ν_2, and we denote the corresponding cycles as $\text{IR}(\nu_1, \nu_2)$. The exact solution is then added back. Slow convergence of IR iterations indicates insufficient coarse-grid correction.

In ICG iterations, the relaxation scheme is actual and the coarse-grid correction is idealized. Assuming that the agglomeration and prolongation operators are suitable for efficient multigrid solution, the idealized coarse-grid correction involves idealized fine and coarse operators, A^f_{ideal} and A^c_{ideal}, such that $D^c_\Omega (A^c_{\text{ideal}})^{-1}$ is an accurate approximation to $D^f_\Omega (A^f_{\text{ideal}})^{-1}$ for smooth error components. Here, D^c_Ω and D^f_Ω are diagonal matrices with corresponding coarse- and fine-grid volumes on the diagonals. The simplest idealized operators are corresponding fine- and coarse-grid identity matrices. With this choice, the idealized coarse-grid correction becomes

$$C_{\text{ideal}} = E - P_0 (D^c_\Omega)^{-1} R_0 D^f_\Omega \tag{11}$$

Note that the operator $(D^c_\Omega)^{-1} R_0 D^f_\Omega$ represents volume-weighted averaging. In ICG analysis, the idealized C_{ideal} is applied directly to the known algebraic errors obtained after prerelaxation sweep(s) of the actual relaxation. In implementation, the algebraic error is averaged to the coarse grid, changed in sign, and then prolonged to the fine grid. The slow convergence observed in the ICG iterations is a sign of poor smoothing in relaxation. We denote the ICG cycle as $\text{ICG}(\nu_1, \nu_2)$.

V. Target Grids and Agglomerations

The grids considered are generated by splitting isotropic mapped Cartesian grids into triangular (two-dimensional) or tetrahedral (three-dimensional) elements and then randomly perturbing the grid points by up to one-quarter in two dimensions and one-sixth in three dimensions of the local mesh size. A typical target grid is shown in Fig. 2 for two dimensions with 33 points in each direction. An orthographic view of the boundary grids of a typical target 3-D grid is shown in Fig. 3, again for 33 points in each direction.

The grids are agglomerated within a topology-preserving framework, in which hierarchies are assigned based on connections to the computational boundaries. Corners are identified as grid points with three or more boundary-condition-type closures (or three or more boundary slope discontinuities). Ridges are identified as grid points with two boundary-condition-type closures (or two boundary slope discontinuities). Valleys are identified as grid points with a single boundary-condition-type closure, and interiors are identified as grid points with no boundary closure. The agglomerations proceed hierarchically from seeds within the topologies, first corners, then ridges, then valleys, and finally interiors. Rules are enforced to maintain the boundary condition types of the finer grid within the agglomerated grid. Candidate volumes to be agglomerated are vetted against the hierarchy of the currently agglomerated volumes using the rules summarized in Table 1. The allowed entries denote that interior volumes can be agglomerated to any existing agglomerate. The single disallowed entry enforces that two corners cannot be agglomerated. The conditional entries denote that further inspection of the connectivity of the topology must be considered before agglomeration is allowed. For example, a ridge can be agglomerated into a corner if the ridge is part of the boundary condition

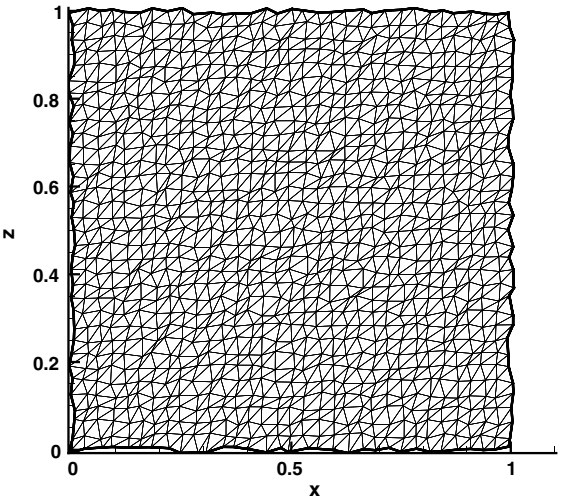


Fig. 2 Typical 2-D target grid.

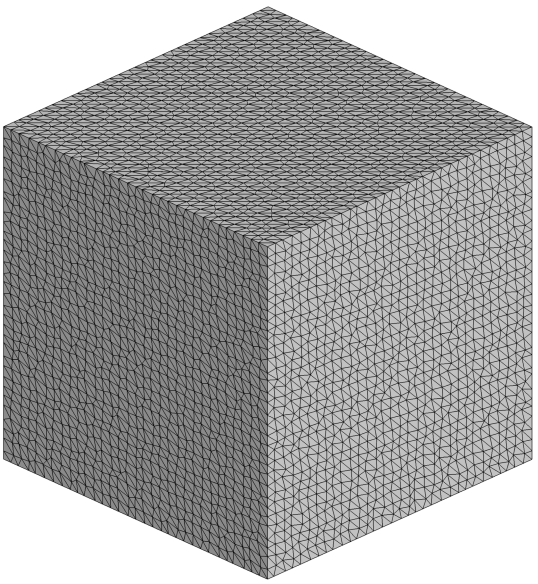


Fig. 3 Orthographic view of a typical 3-D target grid.

Table 1 Admissible agglomerations

Hierarchy of agglomeration	Hierarchy of added volume	Agglomeration decision
Corner	Interior	Allowed (corner to interior)
Corner	Valley	Conditional
Corner	Ridge	Conditional
Corner	Corner	Disallowed (two corners)
Ridge	Interior	Allowed (ridge to interior)
Ridge	Valley	Conditional
Ridge	Ridge	Conditional
Valley	Interior	Allowed (valley to interior)
Valley	Valley	Conditional
Interior	Interior	Allowed (interior to interior)

specification associated with the corner. As another example, a ridge can be agglomerated into an existing ridge agglomeration if the two boundary conditions associated with each ridge are the same. Also, the prolongation operator P_1 is modified to prolong only from hierarchies equal to or above the hierarchy of the prolonged point. Hierarchies on each agglomerated grid are inherited from the finer grid.

There are two agglomeration schemes, referred to as schemes I and II, that have evolved historically within this development. The agglomeration scheme I orders the possible points within a hierarchy using the distance from the corners of the grid and the closest points are taken first. Given a seed, a triad is constructed using a surrounding cloud of points, defined from the adjacency list. The first leg of the triad is defined by the seed and the nearest point. The next leg of the triad is defined by including another point from the entries in the cloud such that the leg is most orthogonal to the first leg. The third leg is found as the one most parallel to the cross product of the first two legs. Points within the volume defined by the triads (extended to infinite length) are taken, first for the edge adjacencies in the cloud and subsequently for the entire adjacency, to satisfy a global coarsening goal (four volumes agglomerated for two dimensions and eight for three dimensions). The agglomeration scheme II also starts from the corners. After all corners have been agglomerated, a front list is defined by collecting nodes adjacent to the agglomerated corners. It then proceeds to agglomerate nodes in the list (while updating the list as the agglomeration proceeds) in the following order: ridges, valleys, interiors. A node is selected among those in the same hierarchy that has the least number of nonagglomerated neighbors to reduce the occurrences of agglomerations with small numbers of volumes. For a given seed, it collects all neighbors and agglomerates them up to a specified maximum number, for example, eight in three dimensions. The agglomeration continues until the front list becomes empty. For either agglomeration scheme, agglomerations containing only a few volumes are combined with other agglomerations, as is typical of the methods used in the literature.

Figure 4 shows three agglomerated grids generated from the primal grid in Fig. 2 using agglomeration schemes I and II. Figure 5 shows three agglomerated grids generated from the primal grid in Fig. 3 using agglomeration scheme II. The agglomerations are representative of those in the literature.

For meshes stretched toward a surface, implicit lines are used. They are defined in the direction normal to the surface by the shortest distance between nodes, constructed on the primal grid, and terminated in the isotopic region [1–3]. The agglomerations are first constructed along the boundary of the grid (corners, ridges, and valleys) and then the cells are agglomerated from the boundary within the implicit lines associated with the stretched grid. The

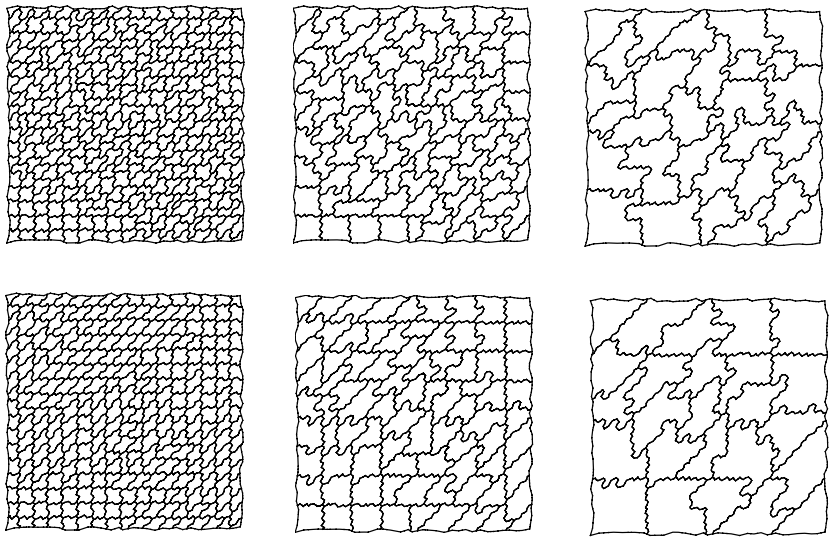


Fig. 4 Control-volume boundaries (nonlumped) for 2-D agglomerations using scheme I (top row) and scheme II (bottom row).

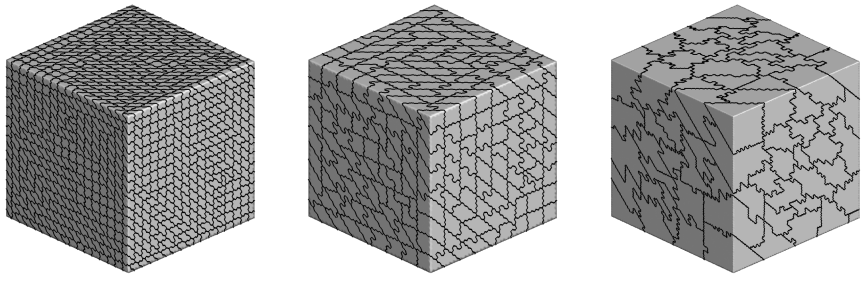


Fig. 5 Control-volume boundaries (nonlumped) for 3-D agglomerations using scheme II.

Table 2 Summary of multilevel asymptotic convergence rates per $V(2,1)$ multigrid cycle with agglomeration scheme I for the Green–Gauss scheme on the fine grid with various coarse-grid operators; cycles to convergence are in parentheses

Fine grid	Direct discretization		Galerkin discretization	
	Avg-LSQ	Edge terms only	$R_0A^fP_0^*$	$R_0A^fP_1$
33×33	0.15(12)	0.20(13)	0.51(23)	Divergent
65×65	0.18(12)	0.29(15)	0.58(25)	Divergent
129×129	0.21(12)	0.33(16)	0.60(24)	Divergent
257×257	0.19(12)	0.44(18)	0.62(24)	Divergent

Table 3 Summary of multilevel asymptotic convergence rates per $V(2,1)$ multigrid cycle with agglomeration scheme II for the Green–Gauss scheme on the fine grid with various coarse-grid operators; cycles to convergence are in parentheses

Fine grid	Direct discretization		Galerkin discretization	
	Avg-LSQ	Edge terms only	$R_0A^fP_0^*$	$R_0A^fP_1$
33×33	0.16(11)	0.29(15)	0.47(23)	Divergent
65×65	0.16(11)	0.42(19)	0.58(27)	Divergent
129×129	0.18(12)	0.54(26)	0.68(31)	Divergent
257×257	0.18(12)	0.82(60)	0.71(34)	Divergent

Table 4 Summary of asymptotic convergence rates per relaxation and the number of relaxations to converge in single-grid calculations (the Green–Gauss scheme is used)

	Convergence per relaxation	Number of relaxations
33×33	0.99710	1278
65×65	0.99926	5440
129×129	0.99945	16320

boundary agglomerate is merged with the volumes corresponding to the next node in the line. The agglomeration continues to the end of the shortest line in the boundary agglomerate, merging two cells in the normal direction at a time. After agglomeration of lines, the algorithm uses the point agglomeration method for the rest of the domain. Illustrations of stretched grids and corresponding agglomerations are shown in Section VI.

VI. Two-Dimensional Results

A summary of $V(2,1)$ multigrid cycle convergence rates is compiled in Tables 2 and 3 for the two agglomeration schemes, respectively. The computations are performed for the Green–Gauss scheme on the fine grid with various coarse-grid operators. The asymptotic convergence per cycle and the number of cycles to reach machine-precision residuals from a random initial perturbation are tabulated. Multigrid cycles employ as many levels as possible; for example, there are six levels used for the 129×129 target grid and four levels for the 33×33 target grid. Table 4 shows convergence rates per relaxation and the number of relaxations to converge for single-grid calculations. Somewhat surprisingly, with the Galerkin coarse-grid operator constructed via $R_0A^fP_1$, the multigrid algorithm is divergent. The reason, confirmed by analysis, is that the coarse-grid operator, although accurate, loses h ellipticity [1]. This loss of h ellipticity for the Galerkin operator with simplex-based P_1 prolongation has been observed even with quadrilateral grids, for which bilinear prolongation is known to result in h elliptic coarse-grid operators.

With the Galerkin coarse-grid operator $R_0A^fP_0^*$, the multigrid algorithm is stable. However, the convergence rates degrade on finer grids with either agglomeration scheme. With the coarse-grid operator using only the edge terms, the convergence per cycle is generally better, but again shows a deterioration on finer grids. The deterioration is noticeably worse with the agglomeration scheme II, although it is hard to judge the reason from visual inspection of the agglomerated grids. With the Avg-LSQ scheme, the convergence per cycle is 0.21 or better and grid independent. In any case, the multigrid algorithm, whether grid dependent or grid independent, gives considerable speedup over a single-grid method; compare Tables 2 and 3 with Table 4.

The dependence on the number of levels in the multigrid cycle is shown in Table 5 using the two agglomeration schemes. In all cases, the coarsest-grid residual was reduced 2 orders of magnitude from the initial coarsest-grid residual; the results were insensitive to reducing the coarsest-level residual further. Typically, convergence in a two-level cycle is a lower bound of the convergence in a multilevel cycle; such behavior is observed with the coarse grids discretized using the Avg-LSQ scheme. The observed multilevel cycle convergence is very similar to the two-level cycle convergence. With the coarse grids discretized using the edge-terms-only scheme, the results are unexpected; the six-level cycle convergence is significantly better than the two-level cycle convergence. This is true for both agglomeration schemes, although the effect is considerably more pronounced with agglomeration scheme II. A possible explanation is that the coarser agglomeration grids have a less consistently high skewing, thus mitigating inconsistency of the edge-terms-only discretization. Although we did not tabulate the results, the dependence on the number of levels in the multigrid cycle for the heuristic Galerkin construction is more or less as would be expected; the two-level cycle converges best, and performance falls off with increasing number of levels.

The grid-dependent convergence of multigrid cycles with the edge-terms-only scheme (Tables 2 and 3) is attributed to the poor coarse-grid correction, which is confirmed by quantitative analysis. Both ICG and IR were applied to a family of element-based grids (33×33, 65×65, 129×129, and 257×257) with coarser grids constructed in turn using each of the two agglomeration schemes. Convergence of the ICG(3,3) scheme was less than 0.1 per cycle in all cases, indicating that the multicolor relaxation is not a source of the grid-dependent convergence. The results of applying IR(3,3) are shown in Table 6 with the coarse-grid correction using the Avg-LSQ and the edge-terms-only schemes for each of the two agglomeration schemes. With the coarse-grid correction using the Avg-LSQ scheme, the convergence rates per cycle are grid independent and

Table 5 Asymptotic convergence per $V(2,1)$ cycle for the Green–Gauss scheme on the target 129×129 grid with various coarse-grid operators; cycles to convergence are in parentheses

	Agglomeration scheme I		Agglomeration scheme ii	
	Coarse-grid discretization		Coarse-grid discretization	
Multigrid levels	Avg-LSQ	Edge terms only	Avg-LSQ	Edge terms only
6	0.21(12)	0.33(16)	0.18(12)	0.54(26)
5	0.21(12)	0.33(16)	0.18(12)	0.54(26)
4	0.20(12)	0.35(16)	0.18(12)	0.60(30)
3	0.19(12)	0.43(19)	0.18(12)	0.69(39)
2	0.18(12)	0.41(18)	0.17(12)	0.81(55)

Table 6 Asymptotic convergence per cycle using IR(3,3) analysis; cycles to convergence are in parentheses

	Agglomeration scheme I		Agglomeration scheme ii	
	Coarse-grid discretization		Coarse-grid discretization	
Element-based grid	Avg-LSQ	Edge terms only	Avg-LSQ	Edge terms only
33×33	0.11 (9)	0.32 (15)	0.14 (10)	0.55 (26)
65×65	0.13 (10)	0.49 (21)	0.15 (10)	0.72 (44)
129×129	0.20 (11)	0.54 (26)	0.21 (12)	>0.99 (>200)
257×257	0.17(10)	0.61 (28)	0.20 (11)	>0.99 (>500)

Table 7 Asymptotic convergence per cycle using ICG (3,3) analysis for family of agglomerated grids; cycles to convergence are in parentheses (the Avg-LSQ scheme is used on all grids)

	Agglomeration scheme	
Agglomeration level	I	II
4	0.06 (8)	0.06 (8)
3	0.06 (8)	0.05 (7)
2	0.07 (8)	0.07 (8)
1	0.06 (8)	0.08 (8)
0	0.07 (8)	0.08 (8)

Table 8 Asymptotic convergence for two-level cycle for sheared primal grids; cycles to convergence are in parentheses (the Green–Gauss scheme is used on the primal grids)

	Agglomeration scheme I		Agglomeration scheme II	
	Coarse-grid discretization		Coarse-grid discretization	
Primal grid	Avg-LSQ	Edge terms only	Avg-LSQ	Edge terms only
17×17	0.17(12)	0.48(27)	0.15(12)	0.63(39)
33×33	0.18(12)	0.67(40)	0.20(12)	unstable
65×65	0.21(12)	0.88(102)	0.22(13)	unstable

better than 0.21; the number of cycles to convergence is 12 at most. With the coarse-grid correction using the edge-terms-only scheme, the convergence rates and number of cycles to converge are grid dependent.

With a consistent coarse-grid discretization, such as the Avg-LSQ scheme, we expect good two-level convergence rates. With the Avg-LSQ scheme, relaxation is implemented within a defect-correction setting in which the approximate linearization based on the edge-terms-only scheme is used as a driver. The viability of this approach is checked using ICG(3,3) for the family of grids agglomerated from the parent 257×257 grid. The convergence per cycle is shown in Table 7 for different agglomeration levels, where the target element-based grid is denoted as level 0. In all cases, the edge-terms-only scheme provides adequate relaxation, yielding an order of magnitude convergence per ICG(3,3) cycle.

The spatial convergence of discretization error for agglomerate families with the Avg-LSQ target-grid discretization is shown in Fig. 6. Results with the edge-terms-only discretization are also shown for reference. The manufactured solution is $U = \sin(\pi x + 0.8\pi y) + 0.1x + 0.2y$ and the coarser grids were generated using agglomeration scheme II. Each agglomerate family is composed of a target element-based grid and agglomerated grids generated recursively; a particular agglomerate family is denoted by the density of the primal mesh in parentheses. The L_1 norm of the discretization error is shown versus an equivalent mesh size, taken as the L_1 norm of a local characteristic distance, that is, $h_V = \|\Omega^{1/d}\|$, where d is the number of spatial dimensions. The edge-terms-only discretization shows no order property, as expected, but the Avg-LSQ scheme shows a second-order convergence of discretization errors. Thus, the Avg-LSQ scheme is second-order accurate and provides a viable way of discretizing diffusion terms on agglomerated coarse grids.

For the finer agglomerate family, multigrid convergence is shown in Fig. 7 using the Avg-LSQ discretization on all grids. Multilevel $V(2, 2)$ cycles are used with two levels on the coarsest agglomerate and six levels on the primal mesh. The initial conditions are taken as the exact solution with a randomly perturbed error on each grid. Grid-independent convergence is shown with approximately an order of magnitude reduction in residual per cycle.

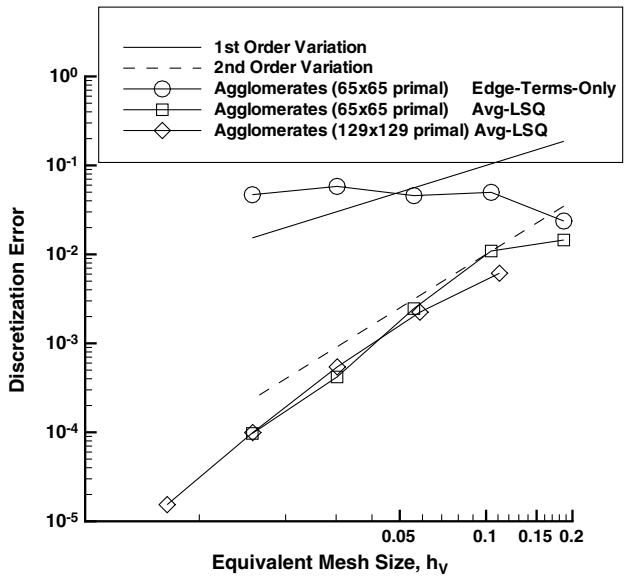


Fig. 6 Spatial convergence of discretization error for agglomerate families with the Avg-LSQ and edge-terms-only schemes.

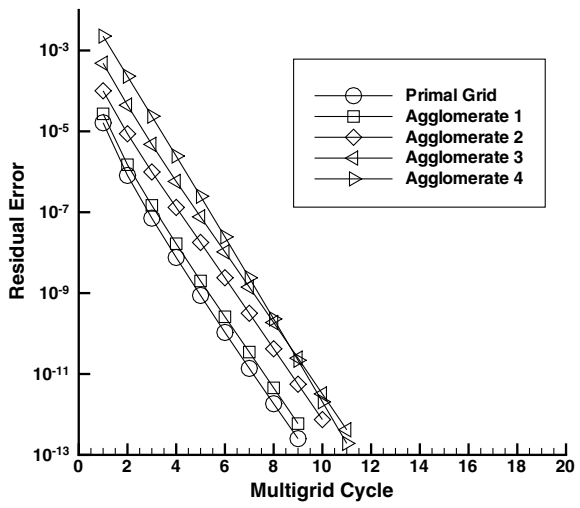
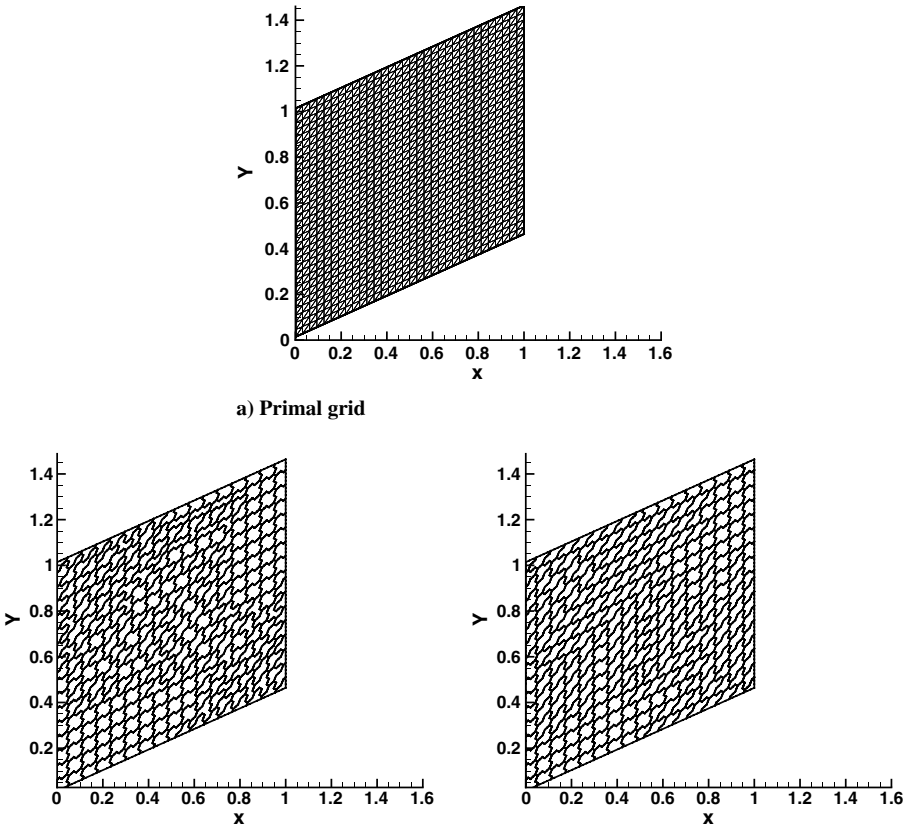


Fig. 7 Multigrid convergence for agglomerate family composed of the 129 × 129 primal mesh and its coarsened agglomerates, using the Avg-LSQ scheme in all levels.



a) Primal grid

b) Agglomerate with scheme I

c) Agglomerate with scheme II

Fig. 8 Primal (33 × 33) grid and agglomerated grids.

Finally, to demonstrate that multigrid convergence with the coarse-grid edge-terms-only discretization is grid dependent, a series of sheared primal grids is considered with skew angles consistently greater than 45 deg. A typical primal grid and the agglomerated grids using the two agglomeration schemes are shown in Fig. 8. The convergence of two-level multigrid cycles is shown in Table 8 using the two agglomeration schemes with different coarse-grid discretizations. Convergence with the coarse-grid Avg-LSQ discretization is very similar using either agglomeration scheme and nominally grid independent. With the coarse-grid discretized using the edge-terms-only scheme, the convergence is grid-dependent for agglomeration scheme I; the multigrid cycle is unstable beyond the coarsest grid with agglomeration scheme II. Note the variability in convergence with the edge-terms-only coarse-grid discretization between agglomeration schemes I and II even though the agglomerations from the two schemes are quite regular and similar (Fig. 8).

VII. Three-Dimensional Results

Multigrid asymptotic convergence rates are shown in Table 9 with various coarse-grid operators for a range of isotropic 3-D grids ($9 \times 9 \times 9$ to $129 \times 129 \times 129$). Results are obtained with multiple-level $V(3,3)$ multigrid cycles. Two-grid results are not shown but are very similar to the multiple-level results. Agglomerated grids are generated with scheme II.

Table 9 Summary of multilevel asymptotic convergence rates per $V(3,3)$ multigrid cycle with agglomeration scheme II for the Green–Gauss scheme on the fine grid with various coarse-grid operators

Fine grid	Direct discretization		Galerkin discretization	
	Avg-LSQ	Edge terms only	$R_0A^fP_0^*$	$R_0A^fP_1$
$9 \times 9 \times 9$	0.05	0.05	0.15	divergent
$17 \times 17 \times 17$	0.11	0.16	0.35	divergent
$33 \times 33 \times 33$	0.14	0.26	0.54	divergent
$65 \times 65 \times 65$	0.16	0.30	0.67	divergent
$97 \times 97 \times 97$	0.24	0.33	0.73	divergent
$129 \times 129 \times 129$	0.22	0.34	0.76	divergent

The 3-D results are consistent with the 2-D results. With the Galerkin coarse-grid operator constructed via $R_0A^fP_0^*$, the multigrid algorithm is stable, but the convergence degrades on finer grids. The Galerkin coarse-grid operator constructed via $R_0A^fP_1$ was again found to be divergent. With agglomerated grids using the edge-terms-only scheme, the convergence per cycle is better but again shows a deterioration on finer grids. Note that the deterioration observed in three dimensions is weaker than that in two dimensions. With agglomerated grids using the Avg-LSQ scheme, the convergence per cycle is practically grid-independent; the asymptotic convergence per cycle is similar to that in two dimensions. In any case, the multigrid method gives considerable speedup over a single-grid method, as clearly seen in Fig. 9, which shows the residual convergence versus work units for the $65 \times 65 \times 65$ grid case. Here, the work unit is defined as the work required for one residual evaluation and relaxation on the target grid; a multigrid $V(3,3)$ cycle requires about 7 work units; restriction and prolongation work is small and has been neglected. The multigrid method converged in 108 work units using the Avg-LSQ scheme, 144 using the edge-terms-only scheme, and 425 with the Galerkin coarse-grid operator constructed via $R_0A^fP_0^*$, whereas the single-grid method converged in 10,335 work units. Some dependence on the number of levels in the multigrid cycle similar to that for 2-D cases as shown in Table 5 was observed also in three dimensions, but the variation was smaller.

The multigrid $V(3,3)$ cycle is tested with a line agglomeration/relaxation for stretched grids typical in high-Reynolds-number flow simulations. The grids are regular tetrahedral $9 \times 9 \times 17$, $13 \times 13 \times 25$, $17 \times 17 \times 33$, $24 \times 24 \times 47$, $33 \times 33 \times 65$, $49 \times 49 \times 97$ grids with exponential stretching applied in the z direction. The stretching is applied only in the lower half region; the upper half remains isotropic. A representative grid is shown in Fig. 10. A line

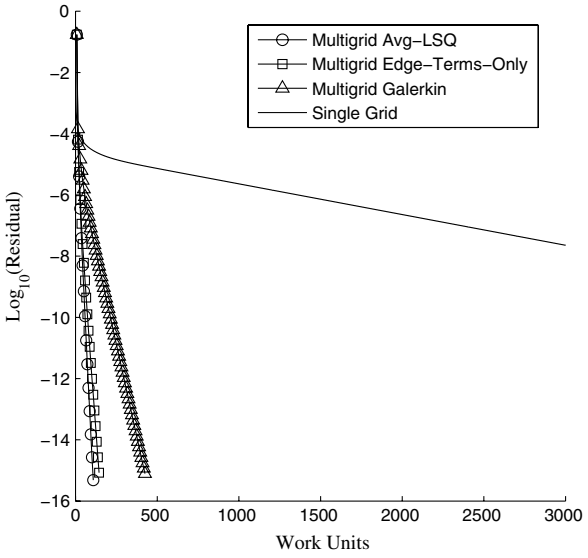


Fig. 9 Residual versus work units for the $V(3,3)$ multilevel multigrid methods with agglomeration scheme II and a single-grid method on the 65 × 65 × 65 grid. The Green–Gauss scheme is used on the fine grid.

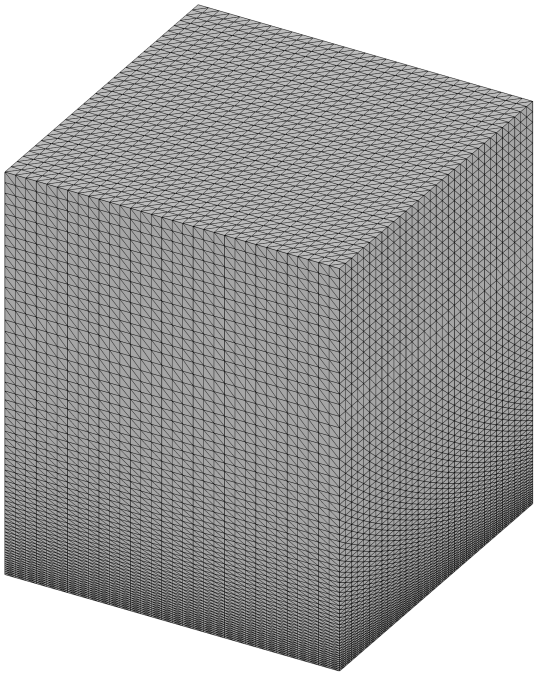


Fig. 10 33 × 33 × 65 stretched grid with the maximum aspect ratio of 6.25.

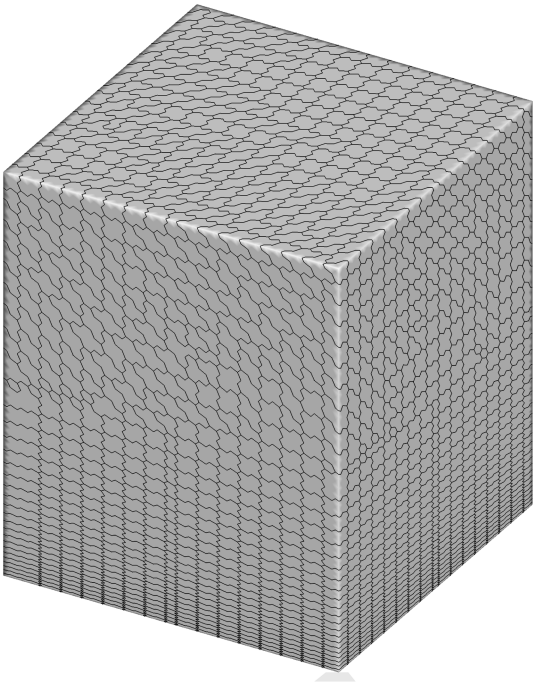


Fig. 11 Coarse grid for the 33 × 33 × 65 stretched grid with the maximum aspect ratio of 6.25.

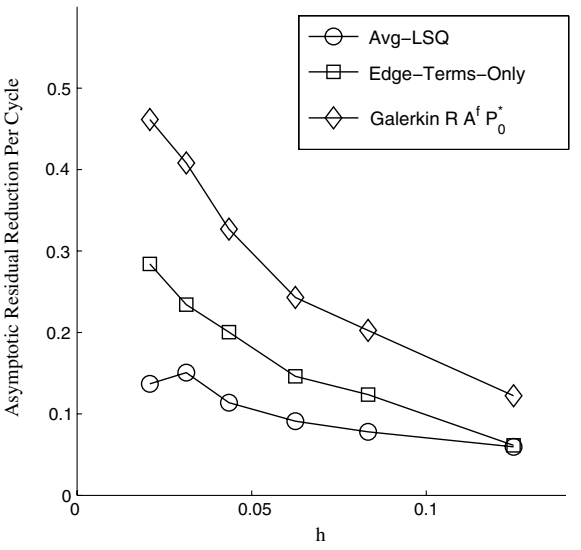


Fig. 12 Asymptotic convergence rate per $V(3,3)$ multigrid two-level cycle with agglomeration scheme II and a line agglomeration/relaxation in the stretched region. The Green–Gauss scheme is used on the fine grid.

agglomeration and a line relaxation are applied in the stretched region. A representative coarse grid is shown in Fig. 11. The results are shown in Fig. 12. The mesh size h corresponds to $1/(N^{1/3}-1)$, where N is the total number of nodes. Again, multigrid with either the edge-terms-only or the Galerkin coarse-grid operator shows a deterioration on finer grids, whereas a multigrid with the Avg-LSQ scheme gives nearly grid-independent results. One would have to consider even higher mesh densities to clearly indicate the behavior of the convergence rate with mesh refinement.

The IR and ICG analysis methods have been applied within a two-grid multigrid cycle on perturbed isotropic tetrahedral grids to evaluate relaxation smoothing and efficiency of coarse-grid correction. The point relaxation scheme has been tested on a $33 \times 33 \times 33$ grid for three formulations: Green–Gauss, Avg-LSQ, and edge terms only. Convergence rates observed in ICG iterations and collected in Table 10 show that the tested relaxation is an efficient error smoother for all three schemes; the high-frequency error reduction is better than 0.55, which is an excellent smoothing factor.

IR iterations have been performed to analyze the quality of coarse-grid correction with two different coarse-grid schemes: Avg-LSQ and edge-terms-only approximation. The results are shown in Table 11. To provide robust grid-independent convergence rates in a

Table 10 Summary of smoothing rates of three relaxation schemes obtained from ICG(1,0) on a 33 × 33 × 33 perturbed isotropic tetrahedral grid (the Green–Gauss scheme is used on the fine grid)

Green–Gauss	Avg-LSQ	Edge-terms-only
0.545	0.470	0.358

Table 11 Summary of convergence rates for two coarse-grid correction schemes obtained from IR(3,3) on a 33 × 33 × 33 perturbed isotropic tetrahedral grid

Coarse grid	Avg-LSQ	Edge terms only
P_0 prolongation	0.124	0.303
P_1 prolongation	0.125	0.303

multigrid cycle, the coarse-grid correction is expected to reduce smooth errors by an order of magnitude. Convergence rates observed in IR iterations with six explicit error-averaging sweeps show that the coarse-grid correction is adequate for the Avg-LSQ scheme. The rates observed for the edge-terms-only scheme are slow and further deteriorate on grids with consistent high skewing. Both schemes appear insensitive to the prolongation order, demonstrating almost identical convergence rates for either P_0 or P_1 prolongation operator.

VIII. Conclusions

Agglomerated multigrid techniques used in unstructured-grid methods have been critically studied for a model problem representative of laminar diffusion in the incompressible limit. The studied target-grid discretizations and discretizations used on agglomerated grids are typical node-centered formulations. Agglomerated multigrid convergence rates are compiled using a range of two- and three-dimensional randomly perturbed unstructured grids for simple geometries, including isotropic and stretched grids. Two agglomeration techniques are used within an overall topology-preserving agglomeration framework. The results show that a multigrid with an inconsistent coarse-grid scheme using only the edge terms (also referred to in the literature as a thin-layer formulation) provides considerable speedup over single-grid methods, but its convergence can deteriorate on consistently skewed grids. A multigrid with a formally inconsistent Galerkin coarse-grid discretization using piecewise-constant prolongation and a heuristic correction is slower and also can be grid dependent. A consistent Galerkin coarse-grid construction using simplex prolongation was found to be unstable because the discretization lacked h ellipticity. Nearly grid-independent convergence rates are demonstrated for a multigrid with consistent coarse-grid discretizations. Additional study with higher mesh densities is required to determine grid-independence for 3-D high-aspect-ratio grids. The results from the actual cycle are verified using discrete analysis methods in which parts of the cycle are replaced by their idealized counterparts.

Acknowledgments

The three-dimensional results presented were computed within the FUN3D suite of codes at NASA Langley Research Center.[§] The contributions of E. J. Nielsen, J. A. White, and R.T. Biedron of NASA to the implementation within FUN3D are gratefully acknowledged. Nishikawa was supported by the National Institute of Aerospace under the NASA Fundamental Aeronautics Program through NASA Research Announcement Contract NNL07AA23C. Diskin was supported by the National Institute of Aerospace under NASA Fundamental Aeronautics Program through NASA Research Announcement Contract NNL07AA31C.

References

[1] Trottenberg, U., Oosterlee, C. W., and Schüler, A., *Multigrid*, Academic Press, London, 2000.

[2] Mavriplis, D. J., "Multigrid Techniques for Unstructured Meshes," VKI Lecture Series, Von Karman Institute for Fluid Dynamics VKI-LS 1995-02, Rhode-Saint-Genese, Belgium 1995.

[3] Mavriplis, D. J., "Unstructured Grid Techniques," *Annual Review of Fluid Mechanics*, Vol. 29, 1997, pp. 473–514. doi:10.1146/annurev.fluid.29.1.473

[4] Mavriplis, D. J., "An Assessment of Linear Versus Non-Linear Multigrid Methods for Unstructured Mesh Solvers," *Journal of Computational Physics*, Vol. 175, 2002, pp. 302–325. doi:10.1006/jcph.2001.6948

[5] Mavriplis, D. J., and Pirzadeh, S., "Large-Scale Parallel Unstructured Mesh Computations for 3D High-Lift Analysis," *Journal of Aircraft*, Vol. 36, No. 6, 1999, pp. 987–998. doi:10.2514/2.2540

[6] Diskin, B., Thomas, J. L., Nishikawa, H., Nielsen, E. N., and White, J. A., "Comparison of Node-Centered and Cell-Centered Unstructured Finite-Volume Discretizations Part I Viscous Fluxes," AIAA Paper 2009-597, Jan. 2009.

[7] Diskin, B., and Thomas, J. L., "Accuracy Analysis for Mixed-Element Finite-Volume Discretization Schemes," National Institute of Aerospace Technical Report 2007-8, Aug. 2007.

[8] Thomas, J. L., Diskin, B., and Rumsey, C. L., "Towards Verification of Unstructured Grid Methods," *AIAA Journal*, Vol. 46, No. 12, December 2008, pp. 3070–3079. doi:10.2514/1.36655; also AIAA Paper 2008-0666.

[9] Barth, T. J., "Numerical Aspects of Computing High-Reynolds Number Flow on Unstructured Meshes," AIAA Paper 91-0721, Jan. 1991.

[10] Haselbacher, A. C., "A Grid-Transparent Numerical Method for Compressible Viscous Flow on Mixed Unstructured Meshes," Ph.D. Thesis, Loughborough Univ., Loughborough, England, UK, 1999.

[11] Weiss, J. M., Maruszeski, J. P., and Smith, W. A., "Implicit Solution of Preconditioned Navier–Stokes Equations Using Algebraic Multigrid," *AIAA Journal*, Vol. 37, No. 1, Jan. 1999, pp. 29–36. doi:10.2514/2.689

[12] Diskin, B., Thomas, J. L., and Mineck, R., "On Quantitative Analysis Methods for Multigrid Solutions," *SIAM Journal on Scientific Computing*, Vol. 27, No. 1, 2005, pp. 108–129. doi:10.1137/030601521

[13] Svärd, M., Gong, J., and Nordström, J., "An Accuracy Evaluation of Unstructured Node-Centered Finite-Volume Methods," *Applied Numerical Mathematics*, Vol. 58, No. 8, 2008, pp. 1142–1158. doi:10.1016/j.apnum.2007.05.002; also available as National Institute of Aerospace Rept. 2005-04, NASA CR-2006-214293, April 2006.

W. Anderson
Associate Editor

[§]Data available online at http://fun3d.larc.nasa.gov/ [retrieved 4 December 2009].