

# A Critical Study of Agglomerated Multigrid Methods for Diffusion

Hiroaki Nishikawa, National Institute of Aerospace, Hampton, VA 23666\*

Boris Diskin, National Institute of Aerospace, Hampton, VA 23666†

James L. Thomas, NASA Langley Research Center, Hampton VA 23681‡

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 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 deteriorates on finer grids. Multigrid with a Galerkin coarse-grid discretization using piecewise-constant prolongation and a heuristic correction factor is slower and also grid-dependent. In contrast, grid-independent convergence rates are demonstrated for 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.

## Introduction

Multigrid techniques<sup>1</sup> are used to accelerate convergence of current Reynolds-Averaged Navier-Stokes solvers for steady and unsteady flow solutions, especially for structured-grid applications. Mavriplis et al.<sup>2-5</sup> 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,<sup>6</sup> 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 is presented from a general finite-volume discretization (FVD) standpoint. Elements of multigrid algorithms are described, including a tabulation of target and coarse-grid discretizations. Quantitative analysis methods, in which parts of the actual multigrid cycle are replaced by their idealized counterparts, are described in the next section. The target grids and typical agglomerated grids developed within a topology-preserving framework are next shown, followed by two dimensional (2D) and three dimensional (3D) results. Results from applying analysis methods to 3D computations are also reported. The final section contains conclusions.

## I. Model Diffusion Equation and Boundary Conditions

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

\*Research Scientist, 100 Exploration Way, [hiro@nianet.org](mailto:hiro@nianet.org). This author was supported by the National Institute of Aerospace under NASA fundamental Aeronautics Program through NRA Contract NNL07AA23C

†Associate Fellow, 100 Exploration Way, [bdiskin@nianet.org](mailto:bdiskin@nianet.org); Visiting Associate Professor, Mechanical and Aerospace Engineering Department, University of Virginia, Charlottesville. This author was supported by the National Institute of Aerospace under NASA Fundamental Aeronautics Program through NRA Contract NNL07AA31C

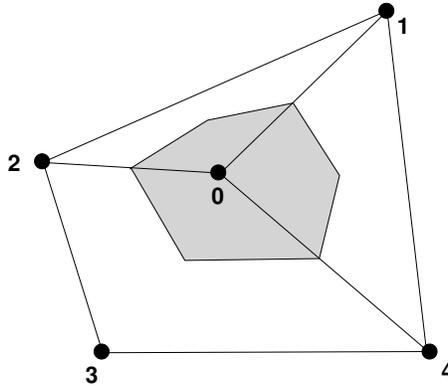
‡Senior Research Scientist, Computational Aerosciences Branch, Mail Stop 128, Fellow AIAA, [James.L.Thomas@nasa.gov](mailto:James.L.Thomas@nasa.gov).

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

where  $f$  is a forcing function independent of the solution  $U$ ,  $\Omega$  is a control volume with boundary  $\Gamma$ ,  $\hat{\mathbf{n}}$  is the outward unit normal vector, and  $\nabla U$  is the solution gradient vector. The boundary conditions are taken as Dirichlet, i.e., 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,

$$\Delta U = f, \quad (2)$$

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 equation (2) and the exact discrete solution,  $U^h$ , of the discretized equation (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<sup>7,8</sup> converge with the design order in the norm of interest.



**Figure 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.**

The general FVD approach requires partitioning the domain into a set of non-overlapping control volumes and numerically implementing equation (1) over each control volume. Node-centered schemes define solution values at the mesh nodes. In 2D, the primal meshes are composed of triangular and quadrilateral cells; in 3D, the primal cells are tetrahedral, prismatic, pyramidal, or hexahedral. The *median-dual* partition<sup>9,10</sup> used to generate control volumes is illustrated in Figure 1 for 2D. These non-overlapping 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

$$\Omega^c = R_0 \Omega^f, \quad (3)$$

where 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.

## II. Multigrid

Elements of the multigrid algorithm are presented in this section. A V-cycle,<sup>1</sup> denoted as  $V(\nu_1, \nu_2)$ , uses  $\nu_1$  relaxations performed at each grid before proceeding to the coarser grid and  $\nu_2$  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

$$r^c = R_0 r^f. \quad (4)$$

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 (2D) or tetrahedra (3D) 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 non-convex weights and, in situations where multiple geometrical shapes are found, the first one encountered is used. Where no enclosing simplex is found, the simplex with minimal non-convex weights is used. The coarse-grid solution approximation is restricted as

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

The correction  $\delta U$  to the finer grid is prolonged typically through  $P_1$ , as

$$(\delta U)^f = P_1(\delta U)^c. \quad (6)$$

Target-Grid Discretizations
Green-Gauss
Avg-LSQ

**Table 1. Summary of target-grid discretizations.**

The available consistent target-grid discretizations are listed in Table 1 where Avg-LSQ stands for average-least-squares. 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, 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-based contributions are used to increase the  $h$ -ellipticity of the operator. The avg-LSQ scheme uses the average of the dual-volume LSQ gradients to determine a gradient at the face, which is augmented with the edge-based directional contribution to determine the gradient used in the flux.

A full linearization has been implemented for the Green-Gauss scheme, enabling a robust multicolor Gauss-Seidel relaxation. The avg-LSQ scheme has a comparatively larger stencil and its full linearization has not been implemented; instead relaxation of the avg-LSQ scheme relies on an approximate linearization, which consists of edge terms only. 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.<sup>6</sup> The estimates for the smoothing rates obtained with quantitative analysis methods<sup>12</sup> are shown in Section VI. Note that the Green-Gauss scheme relies on an element-based data structure and hence is not considered for agglomerated grids.

Coarse-Grid Discretizations	
Direct Discretization	Avg-LSQ
	Edge-Terms-Only
Galerkin Discretization	$R_0 A^f P_0^*$
	$R_0 A^f P_1$

**Table 2. Summary of coarse-grid discretizations.**

The available coarse-grid discretizations are summarized in Table 2, listing two possible direct discretizations and two possible Galerkin discretizations, in which the coarse grid operators are derived from the fine grid operator. The edge-terms-only discretization is often cited as a thin-layer discretization in the literature;<sup>2-4</sup> it is a positive scheme but is not consistent (i.e., its discrete solution does not converge to the exact continuous solution with consistent grid refinement) unless the grid is orthogonal.<sup>7,8,11</sup> An orthogonal grid would have each edge connecting nodes across a face being co-linear with the face-normal direction  $\hat{n}$ .

The Galerkin coarse-grid operator<sup>1</sup> is denoted by  $RAP$ . Because the governing equation is a second-order equation, the Galerkin construction,  $R_0A^fP_0$ , is formally inconsistent; the heuristic correction factor adopted by Mavriplis<sup>2</sup> is used,

$$A^c = R_0A^fP_0^* = \frac{1}{2}R_0A^fP_0. \quad (7)$$

The correction factor is derived by enforcing consistency on uniformly-agglomerated hexahedral meshes. The Galerkin construction,  $R_0A^fP_1$ , is consistent, but was found to be unstable. Dirichlet boundary conditions are enforced strongly. The coarse-grid operator is overwritten with the boundary condition linearization at boundary nodes.

### III. 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 Ref. [12]. The methods analyze the main complementary parts of a multigrid cycle: relaxation and coarse-grid correction. In 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 where 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,  $e^{\text{old}}$ , into the after-cycle error,  $e^{\text{new}}$ ,

$$e^{\text{new}} = Me^{\text{old}}. \quad (8)$$

The amplification matrix can be defined as

$$M = S^{\nu_2}CS^{\nu_1}. \quad (9)$$

Here,  $\nu_1$  and  $\nu_2$  are small nonnegative integers representing the number of pre- and post-relaxation sweeps,  $S$  is the fine-grid relaxation amplification matrix, and  $C$  is the amplification matrix of the coarse-grid correction.

$$C = E - P_0(A^c)^{-1}R_0A^f, \quad (10)$$

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  $\nu_1$  or  $\nu_2$  and we denote results 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  $A_{\text{ideal}}^f$  and  $A_{\text{ideal}}^c$ , such that  $D_{\Omega}^c(A_{\text{ideal}}^c)^{-1}$  is an accurate approximation to  $D_{\Omega}^f(A_{\text{ideal}}^f)^{-1}$  for smooth error components. Here,  $D_{\Omega}^c$  and  $D_{\Omega}^f$  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

$$C_{\text{ideal}} = E - P_0(D_{\Omega}^c)^{-1}R_0D_{\Omega}^f. \quad (11)$$

Note that the operator  $(D_{\Omega}^c)^{-1}R_0D_{\Omega}^f$  represents volume-weighted averaging. In ICG analysis, the idealized  $C_{\text{ideal}}$  is applied directly to the known algebraic errors obtained after pre-relaxation 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. Slow convergence observed in the ICG iterations is a sign of poor smoothing in relaxation. We denote ICG results as  $\text{ICG}(\nu_1, \nu_2)$ .

#### IV. Target Grids and Agglomerations

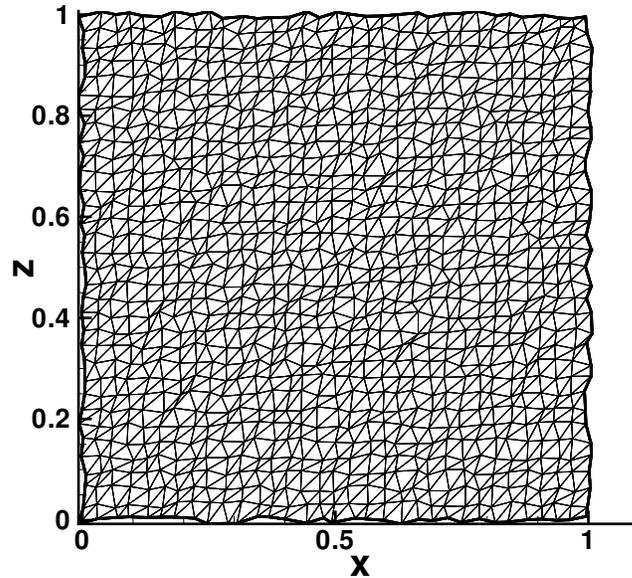
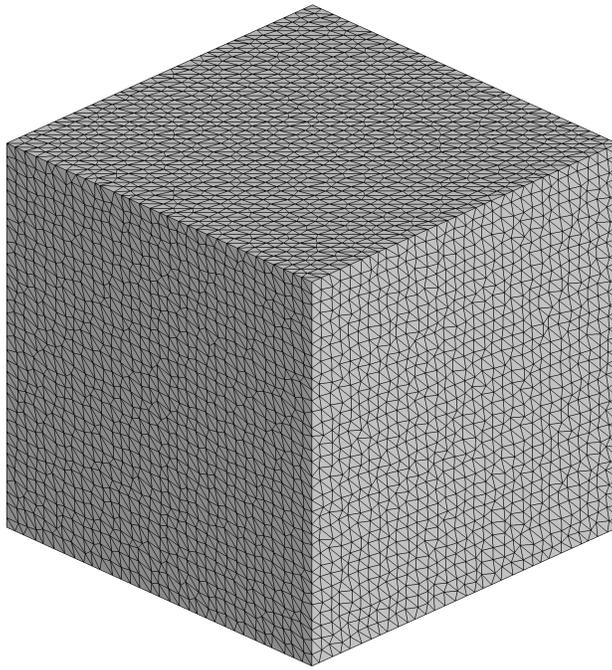


Figure 2. A typical 2D target grid.

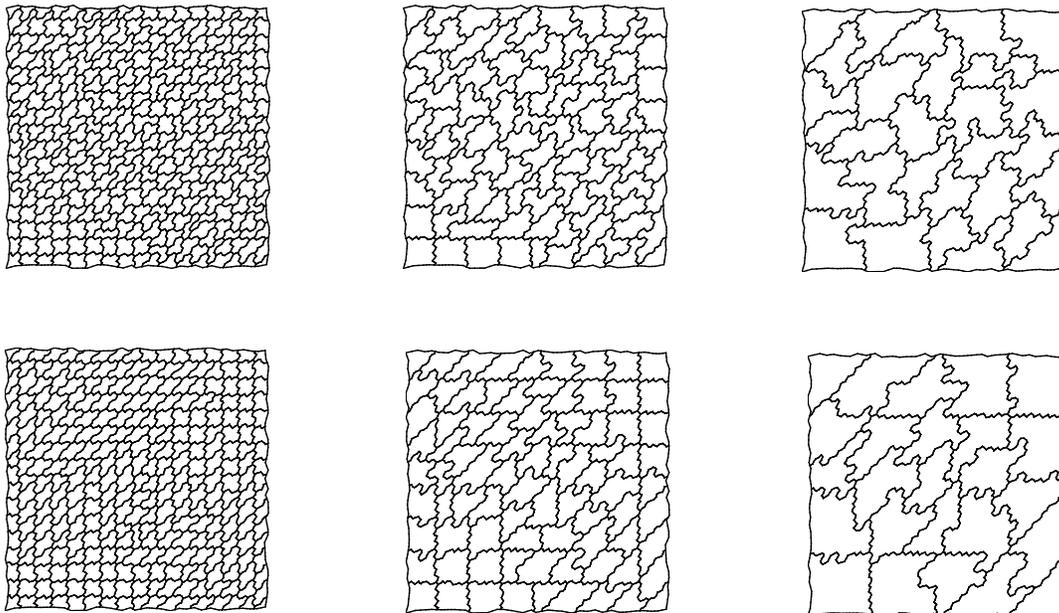
The grids considered are generated by splitting isotropic mapped Cartesian grids into triangular (2D) or tetrahedral elements (3D) and then randomly perturbing the grid points by up to one quarter of the local mesh size. A typical target grid is shown in Figure 2 for 2D with 33 points in each direction. An orthographic view of the boundary grids of a typical target 3D grid is shown in Figure 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 3. 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 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 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 scheme 1 and 2, that have evolved historically within this development. The agglomeration scheme 1 orders the possible points within a hierarchy using the distance from the



**Figure 3. Orthographic view of a typical 3D target grid.**



**Figure 4. Control volume boundaries (non-lumped) for 2D agglomerations using scheme 1 (top row) and scheme 2 (bottom row).**

corners of the grid and 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

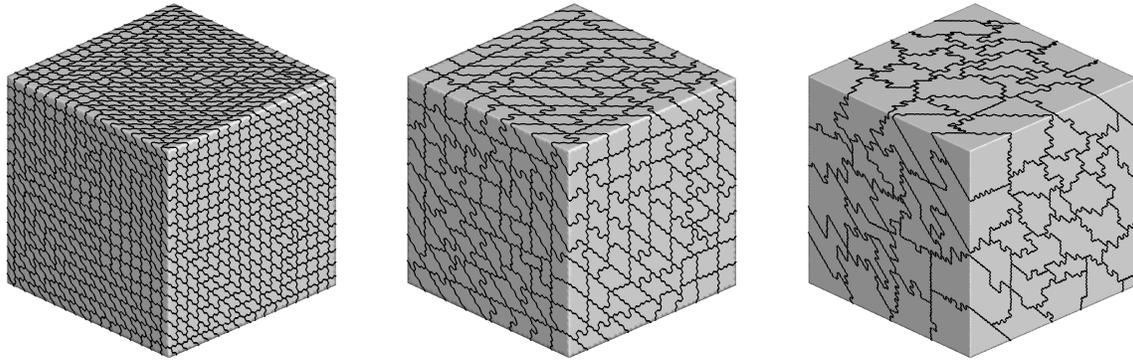


Figure 5. Control volume boundaries (non-lumped) for 3D agglomerations using scheme 2.

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)

Table 3. Admissible agglomerations.

subsequently for the entire adjacency, to satisfy a global coarsening goal (4 volumes agglomerated for 2D and 8 for 3D). The agglomeration scheme 2 also starts from 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 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 non-agglomerated 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, e.g., 8 in 3D. 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.

For highly stretched meshes, the agglomerations are first constructed along the boundary of the grid (corners, ridges, and valleys) and then the cells are agglomerated from the boundary to the ends of the implicit lines associated with the stretched grid. The boundary agglomerate is merged with the volumes corresponding to the next point in the line. The agglomeration continues to the end of the shortest line in the boundary agglomerate, forming an agglomeration from the volumes associated with the next two entries in the line. After agglomeration of lines, the algorithm uses the point agglomeration method for the rest of the domain.

Figure 4 shows three agglomerated grids generated from the primal grid in Figure 2 using schemes 1 and 2. Figure 5 shows three agglomerated grids generated from the primal grid in Figure 3 using scheme 2. The agglomerations are representative of those in the literature.

Fine Grid	Direct Discretization		Galerkin Discretization	
	Avg-LSQ	Edge-Terms-Only	$R_0 A^f P_0^*$	$R_0 A^f P_1$
33x33	0.15	0.20	0.51	divergent
65x65	0.22	0.26	0.53	divergent
129x129	0.21	0.32	0.60	divergent
257x257	0.17	0.45	—	—

Table 4. Summary of multi-level asymptotic convergence rates per V(2,1) multigrid cycle with agglomeration scheme 1 for various coarse-grid operators.

Fine Grid	Direct Discretization		Galerkin Discretization	
	Avg-LSQ	Edge-Terms-Only	$R_0 A^f P_0^*$	$R_0 A^f P_1$
33x33	0.17	0.29	0.49	divergent
65x65	0.19	0.44	0.58	divergent
129x129	0.18	0.54	0.68	divergent

Table 5. Summary of multi-level asymptotic convergence rates per V(2,1) multigrid cycle with agglomeration scheme 2 for various coarse-grid operators.

## V. Two Dimensional Results

A summary of multigrid convergence rates is compiled in Tables 4 and 5 for the two agglomeration schemes, respectively, for a V(2,1) multigrid cycle with various coarse-grid operators. Multigrid cycles employ as many levels as possible; e.g., there are 6 levels used for the 129x129 target grid and 4 levels for the 33x33 target grid. Somewhat surprisingly, with the Galerkin coarse-grid operator constructed via  $R_0 A^f P_1$ , the multigrid algorithm is divergent. The reason, confirmed by analysis, is that the coarse-grid operator, although accurate, loses  $h$ -ellipticity.<sup>1</sup> 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 constructed via  $R_0 A^f P_0^*$ , 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 better, but again shows a deterioration on finer grids. The deterioration is noticeably worse with the agglomeration scheme 2, 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 0.22 or better and grid-independent.

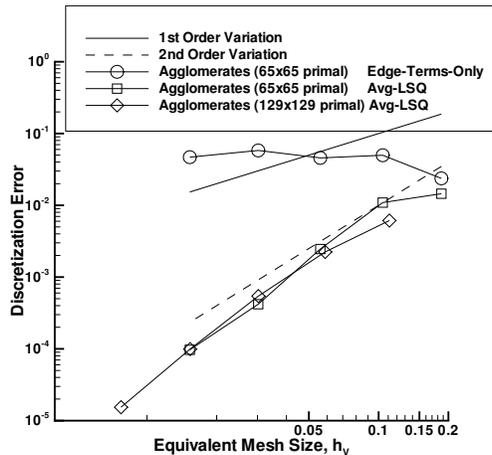


Figure 6. Spatial convergence of discretization error for agglomerate families.

The spatial convergence of discretization error for agglomerate families with the avg-LSQ target-grid discretization is shown in Figure 6. Results with the edge-terms-only (thin-layer) 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 2. Each agglomerate family is composed of a single primal 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, i.e.,  $h_V = \|\Omega^{1/d}\|$ , 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 error. This result is believed to be the first to show order properties for diffusion on a family of agglomerated meshes.

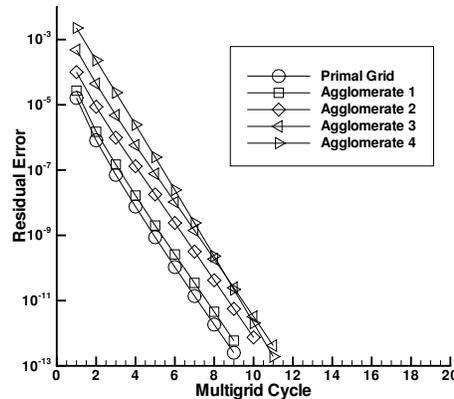
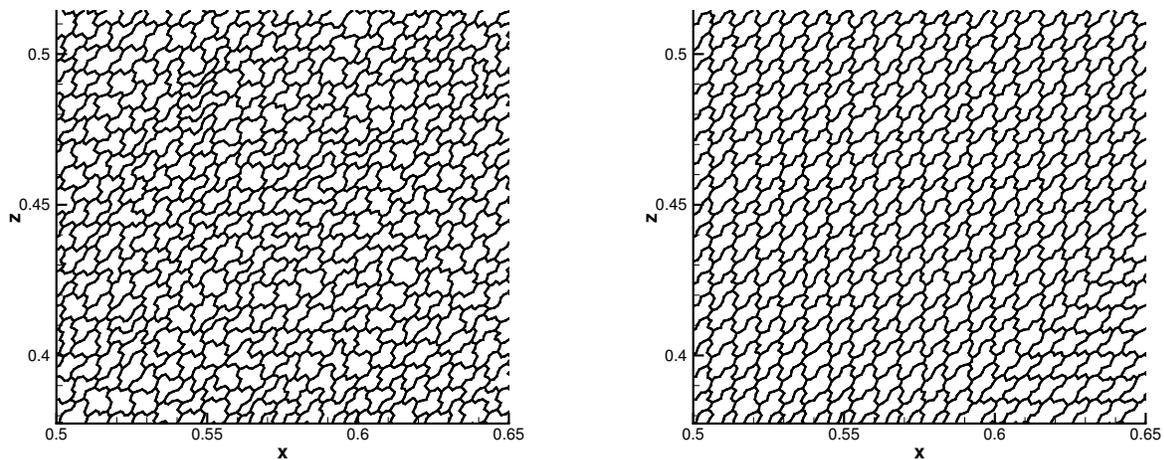


Figure 7. Multigrid convergence for agglomerate family composed of the 129x129 primal mesh and its coarsened agglomerates.

For the finer agglomerate family, multigrid convergence is shown in Figure 7 using the avg-LSQ discretization on all grids. Multilevel V(2,2) cycles are used with 2 levels on the coarsest agglomerate and 6 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.



(a) Agglomeration scheme 1.

(b) Agglomeration scheme 2.

Figure 8. Interior view of agglomerated grids generated from 257x257 element-based grid.

The grid-dependent convergence of multigrid cycles with the edge-terms-only scheme (Tables 4 and 5) 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 (33x33, 65x65, 129x129, and 257x257) 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 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. The asymptotic convergence per cycle and the number of cycles to reach machine precision residuals from the random perturbation applied initially are tabulated. With the coarse-grid correction using the avg-LSQ scheme, the convergence rates per cycle are grid-independent and better than 0.21; the number of cycles to converge 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.

As in the actual cycle results shown previously, IR results with the agglomeration scheme 2 show more dependence on the grid. Comparing the two for the 257x257 parent grid, agglomeration schemes 1 and 2 provide agglomeration ratios of 3.3 and 3.9, respectively; thus, scheme 2 is closer to the requested ratio of 4. An interior section of the two agglomerations is shown in Fig. 8. Scheme 2 results in a piecewise near-regular grid with a distinct skew angle near 45 deg; the grid from scheme 1 is noticeably less regular. The maximum skew angle is higher with scheme 1 (70 deg as compared to 60 deg); however, with scheme 2, 47 percent of the nodes have control volumes with skew angles greater than 40 deg as compared to only 23 percent of the nodes with scheme 1. It is hypothesized the consistent skewing of scheme 2 causes the increased degradation with the edge-terms-only discretization. In any event, the differences in convergence between the two schemes are far smaller with a consistent discretization on the coarse grid.

Element-Based Grid	Agglomeration Scheme 1 Coarse-Grid Discretization		Agglomeration Scheme 2 Coarse-Grid Discretization	
	Avg-LSQ	Edge-Terms-Only	Avg-LSQ	Edge-Terms-Only
33x33	0.11 (9)	0.32 (15)	0.14 (10)	0.55 (26)
65x65	0.13 (10)	0.49 (21)	0.15 (10)	0.72 (44)
129x129	0.20 (11)	0.54 (26)	0.21 (12)	>0.99 (>200)
257x257	0.17(10)	0.61 (28)	0.20 (11)	>0.99 (>500)

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

With a consistent coarse-grid discretization, such as the avg-LSQ scheme, we expect good 2-level convergence rates. With the avg-LSQ scheme, relaxation is implemented within a defect-correction setting in which only the linerization of the edge-terms-only scheme is retained. The viability of this approach is checked using ICG(3,3) for the family of grids agglomerated from the parent 257x257 grid. The convergence per cycle is shown in Table 7 for different agglomeration levels, where the 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.

Agglomeration Level	Agglomeration Scheme	
	1	2
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 7. Asymptotic convergence per cycle using ICG(3,3) analysis for family of agglomerated grids; cycles are in parentheses.**

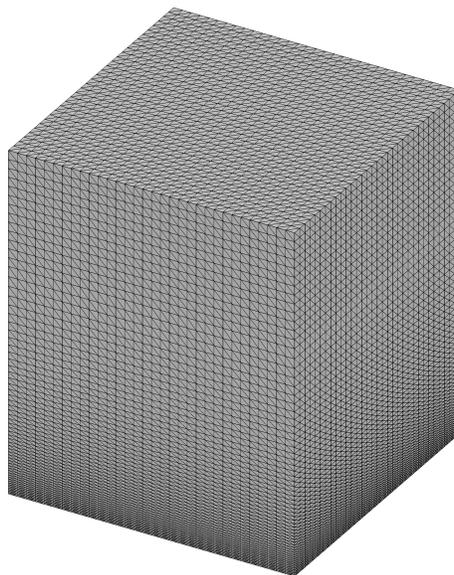
## VI. Three Dimensional Results

Multigrid asymptotic convergence rates are shown in Table 8 with various coarse-grid operators for a range of grids (9x9x9 to 129x129x129). 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 2.

Fine Grid	Direct Discretization		Galerkin Discretization	
	Avg-LSQ	Edge-Terms-Only	$R_0 A^f P_0^*$	$R_0 A^f P_1$
9x9x9	0.05	0.05	0.15	divergent
17x17x17	0.11	0.16	0.35	divergent
33x33x33	0.14	0.26	0.54	divergent
65x65x65	0.16	0.30	0.67	divergent
97x97x97	0.24	0.33	0.73	divergent
129x129x129	0.22	0.34	0.76	divergent

**Table 8.** Summary of multi-level asymptotic convergence rates per V(3,3) multigrid cycle with agglomeration scheme 2 for various coarse-grid operators.

The 3D results are consistent with the 2D results. With the Galerkin coarse-grid operator constructed via  $R_0 A^f P_0^*$ , the multigrid algorithm is stable, but the convergence degrades on finer grids. The Galerkin coarse-grid operator constructed via  $R_0 A^f P_1$  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. 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 2D.



**Figure 9.** 33x33x65 stretched grid with the maximum aspect ratio of 6.25.

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 9x9x17, 13x13x25, 17x17x33, 24x24x47, 33x33x65, 49x49x97 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 Figure 9. A line-agglomeration and a line-relaxation are applied in the stretched region. The results are shown in Figure 10. 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 while multigrid with the avg-LSQ scheme gives practically grid-independent results.

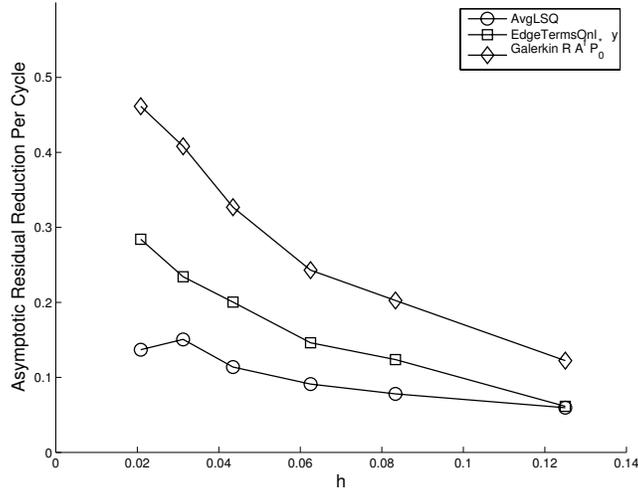


Figure 10. Asymptotic convergence rate per V(3,3) multigrid 2-level cycle with agglomeration scheme 2 and a line agglomeration/relaxation in the stretched region.

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 fine-grid point relaxation scheme has been tested on a 33x33x33 grid for three formulations: Green-Gauss, avg-LSQ, and thin-layer. Convergence rates observed in ICG iterations and collected in Table VI 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.

Green-Gauss	Avg-LSQ	Edge-Terms-Only
0.545	0.470	0.358

Table 9. Summary of smoothing rates of three relaxation schemes obtained from ICG(1,0) on a 33x33x33 perturbed isotropic tetrahedral grid.

IR iterations have been performed to analyze the quality of coarse-grid correction with two different coarse-grid schemes: avg-LSQ and thin-layer approximation. The results are shown in Table VI. To provide a robust grid-independent convergence rates in a 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 high skewing. Both schemes appear insensitive to the prolongation order, demonstrating almost identical convergence rates for either  $P_0$  or  $P_1$  prolongation operator.

Coarse Grid	Avg-LSQ	Edge-Terms-Only
$P_0$ prolongation	0.124	0.303
$P_1$ prolongation	0.125	0.303

Table 10. Summary of convergence rates for two coarse grid correction schemes obtained from IR(3,3) on a 33x33x33 perturbed isotropic tetrahedral grid.

## VII. Concluding Remarks

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 conver-

gence rates are compiled using a range of two- and three-dimensional randomly perturbed unstructured grids for simple geometries, including isotropic and highly-stretched grids. Two agglomeration techniques are used within an overall topology-preserving agglomeration framework. The results show that 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 deteriorates on finer grids. Multigrid with a formally inconsistent Galerkin coarse-grid discretization using piecewise-constant prolongation and a heuristic correction is slower and also grid-dependent. A consistent Galerkin coarse-grid construction using simplex prolongation was found to be unstable because the discretization lacked h-ellipticity. Grid-independent convergence rates are demonstrated for multigrid with consistent coarse-grid discretizations. 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 (<http://fun3d.larc.nasa.gov/>). The contributions of E. J. Nielsen, J. A. White, and R.T. Biedron of NASA in the implementation within FUN3D are gratefully acknowledged.

## References

- <sup>1</sup>Trottenberg U., Oosterlee, C. W., and Schüller, A., *Multigrid*, Academic Press, London, 2000.
- <sup>2</sup>Mavriplis, D. J., "Multigrid Techniques for Unstructured Meshes," VKI Lecture Series VKI-LS 1995-02, Von Karman Institute for Fluid Dynamics, Rhode-Saint-Genese, Belgium 1995.
- <sup>3</sup>Mavriplis, D. J., "Unstructured Grid Techniques," *Annual Review of Fluid Mechanics*, Vol. 29, 1997, pp. 473-514.
- <sup>4</sup>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.
- <sup>5</sup>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.
- <sup>6</sup>Diskin, B., Thomas, J. L., H. Nishikawa, 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, January 2009.
- <sup>7</sup>Diskin, B. and Thomas, J. L., "Accuracy Analysis for Mixed-Element Finite-Volume Discretization Schemes," NIA Technical Report 2007-8, August 2007.
- <sup>8</sup>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 (also AIAA Paper 2008-0666).
- <sup>9</sup>Barth, T. J., "Numerical Aspects of Computing High-Reynolds Number Flow on Unstructured Meshes," AIAA Paper 91-0721, January 1991.
- <sup>10</sup>Haselbacher, A. C., *A Grid-Transparent Numerical Method for Compressible Viscous Flow on Mixed Unstructured Meshes*, PhD thesis, Loughborough University, 1999.
- <sup>11</sup>Svärd, M. and Nordström, J., "An Accuracy Evaluation of Unstructured Node-centered Finite-volume Methods," *Applied Numerical Mathematics*, Vol. 58, No. 8, 2008, pp. 1142-1158; also available as NIA Report 2005-04, NASA CR-2006-214293, April 2006.
- <sup>12</sup>Diskin, B., Thomas, J. L., and Mineck, R., "On Quantitative Analysis Methods for Multigrid Solutions," *SIAM J. Sci. Comput.*, Vol. 27, No. 1, 2005, pp. 108-129.