ICAS 2021 SHANGHAI
International Council of the Aeronautical Sciences
32nd Congress of the International Council of the Aeronautical Sciences
September 6-10, 2021
Pudong Shangri-La, Shanghai, China

METHODS OF DIRECTIONAL COARSENING FOR UNSTRUCTURED GRIDS AND APPLICATIONS

Zhongyun Xiao, Lu Zhang*, Xingda Cui, Bin Mou

Computational Aerodynamics Institute of Chia Aerodynamics Research and Development Center

Abstract

Coarse grids are generated for geometric multigrid by agglomeration of cells. In the simulation of high Reynolds number flows, highly stretched grids in boundary layer tend to cause low quality coarse grid cells in the aspects of skewness and convexity, which decrease the acceleration effect of multigrid. For this issue, directional coarsening methods are proposed to construct coarse grid levels in boundary layer. Firstly original grids are divided into isotropic and anisotropic parts by line searching of marching cells along the direction normal to wall surface. Then for the anisotropic grids, directional coarsening is applied to cells normal to the wall, and surface agglomeration is applied to grids on the wall, both results are integrated to generate volume cells on coarse levels. Various parameters are studied to improve the coarse grid quality. By agglomeration, coarse grid cells are polyhedrons consisting of many faces, to reduce the complexity of flux computation, multiple faces between two neighbor cells are merged, and new grid connections are built. Highly stretched unstructured grids are generated on a cube, M6 wing and CHNT-1 model in this paper, which are successively coarsened for geometric multigrid computation, results show that directional coarsening improves the qualities of coarse grids in orthogonalities and aspect ratios, and by RANS based simulation, the speeding up effects and accuracies of multigrid are demonstrated.

Keywords: multigrid, agglomeration, unstructured grids, boundary layer

1. General Introduction

Multigrid are viewed as one of the ten most advanced numerical methods in 20th century[1][2], which are highly efficient on solving systems of partial differential equations(PDEs). Nowadays unstructured grids have become quite popular because automatic grid generation is much easier than the generation of block-structured grids for complicated 3D domains. Multigrid methods of unstructured grids usually generate coarse grids via cell agglomeration[3], in which the coarse grids are nested and geometric conservations are automatically satisfied. Grid coarsening must meet the requirements of agglomeration ratios and cell qualities, in which the former relate to the computation complexity of coarser levels, and the latter may brings in numerical errors which are transferred level by level in multigrid computation. As for irregular unstructured grids with highly stretched cells, grid coarsening may cause amouts of skewed or concave grid cells, and evenmore produce illegal cells whose centers lies beyond the volume, which cause convergence deceleration or even lead to divergence. Therefore new grid coarsening strategies are needed to be developed to fix these problems.

Extensive researches have been done about unstructured grid agglomeration. In Pandya's method[4] boundary cells adjacent to the wall and outer domain were firstly agglomerated to pursue a better quality, then neighboring inner cells were successively picked out and agglomerated. Mahmutyazicioglu[5] studied octree based unstructured grid coarsening method for 3d applications, which agglomerated cells based on their distribution on an octree data structure. Moulitsas[6] developed unstructured grid coarsening method based on graph partitioning, which uses multilevel algorithms to optimize the overall quality of the fused elements by treating aggloeration ratio as a constraint, and aspect ratios as object functions. Dargaville[7] compared seven coarsening methods, including greedy algorithm, node cancellation, graph partitioning, etc. all methods create legal coarse grids, among which the graph partitioning based method achieved comparatively better quality in aspect ratios. Recently, unstructured multigrid are applied with high order discontinuous

Galerkin algorithms. For example, Pan[8] used greedy algorithm for grid coarsening, Zenoni[9] optimized coarse elements for the best computation accuracy by application of adjoint error estimation method. In Nishikawa[10][11] work, three base rules are summarized for grid coarsening. The first is automation, which means that elements on coarser levels are agglomerated automatically from finer elements. The second is solvability, which means that in flow equations and turbulence model equations, the computation residuals can reach machine zero on fine and different coarse grid levels. The third is that the speeding up effects of unstructured multigrid is comparable with the structured one.

Currently grid coarsening method based on graph partitioning are widely used. Studies have shown that for highly stretched grids this method may produce concave elements, and in more worse condition, illegal elements whose center locates outside the domain. To address this issue, a new coarsening method is designed in this paper, which uses directional coarsening for extruding cells in boundary layer, and graph partitioning based agglomeration for surface grids on the wall. By applications of current methods in three cases, including a cube, a standalone wing and a transport aircraft model, parameters effects on grid coarsening and efficiency of multigrid computation are discussed in detail.

2. Multigrid Methods

2.1 Grid Coarsening in Boundary Layer

Cell centered scheme is used for discretization on base and coarse levels of multigrid, and the aspect ratios are chosen as the common metrics to evaluate cell quality, which is defined as $AR = S^{3/2} / V$ in 3D and $AR = l^2 / S$ in 2D. Aspect ratio indicate the convexity of grid cells, which tend to be regular polygons or regular polyhedrons in optimum cases. taking 3D for example, the aspect ration of a sphere is used as reference for normalization, and the non-dimensional aspect ratio is defined as $AR_{nom} = \left(AR_{act} - AR_{opt} \right) / AR_{act}$, which varies from 0 to 1, where 0 means regular polyhedron like elements, and 1 means degenerated elements whose volume tend to be zero. For cell agglomeration in 2D and 3D, an graph partitioning based agglomeration tool named MGridGen[12][13] is used in this paper.

Semi-structured grids in boundary layer are utilized to guide grid coarsening. Firstly from any surface grid on the wall, the normal cells can be found and listed in sequence by line searching algorithms. Then coarse elements are generated along the line by a specified ratio. There are various ways of searching normal cells. To start with, generally cell sizes in flow and spanwise directions are much larger than normal direction, and the face area of the advancing frontiers is much larger than others. Thus the cell adjacent the maximum face is the normal cell we're looking for. Secondly, on condition that the normal surface areas aren't superior than other faces, the normal vectors can be used as a guide, and the included angle between the advancing face and wall surface is always much smaller than other faces. Thirdly, if the boundary layer grids only consist of prims, line searching would be very simple, which is changed to find adjacent cells of the triangular faces of prisms.

To improve the coarsening ratio, surface grids on wall are also agglomerated, and the surface coarsening and normal coarsening are integrated together to generate coarse grids. In aerospace applications, wall surface grids are generally anisotropically distributed, where regions such as the front and rear edges of wing are covered by fine grids. Taking Figure 2(a) as an example, highly stretched triangular cells are generated near the edges of a cube to simulate the fast geometric and flow changes. Surface grid coarsening can be done by graph partitioning based algorithms, tool libraries like Mgridgen provide such functions. And the connection weights are added to control the agglomeration direction in this paper, which are valued by the length of the public sides. The sum and maximum of coarse element aspect ratios are the optimization goals of grid coarsening, which means that among all the neighboring cells the longest side adjacent cell is the precedent one to be joined. In Figure 2(a), it's possible that cells on different sides of the edges are joined together, which make vertices of the coarse elements not coplanar seriously. To avoid such situations, the included angle of neighboring surfaces are calculated. If the included angle is larger than an threshold value, the adjacent cells are avoided to be joined by deleting the connection relation.

In implementation of multigrid in parallel computation, there are choices of grid partitioning first or grid coarsening first. Strategies of grid partitioning ahead of grid coarsening are employed in this paper by considering the computation complexity of coarsening is much larger than partitioning for equal number of cells. In current strategy, grid cells are divided into small groups to do coarsening, which decrease the time consuming by reducing the involving cells. But in this method the surface normal cells can be divided into different zones, which interrupt the grid coarsening in normal direction. To reduce such situations, a weight function is added in the grid partitioning, which make the partition boundary more likely to be perpendicular to the wall surface.

Figure 1 shows the flow chart of grid coarsening in this paper. In current method, Metis libraries are used to partition grids, after which the grid coarsening are conducted zone by zone, which can also be done in parallel. Before the grid coarsening, the anisotropic boundary layer grids are marked from isotropic ones by line searching. In anisotropic part, surface coarsening and directional line coarsening are integrated for coarse element generation. In isotropic part, cell agglomeration is carried out with graph partitioning based method MGridGen. Grid coarsening for each level of multigrid is completely nested, and before each agglomeration cell connections need to be built. In the end, the anisotropic and isotropic part are merged together to generate grids on all coarse levels.

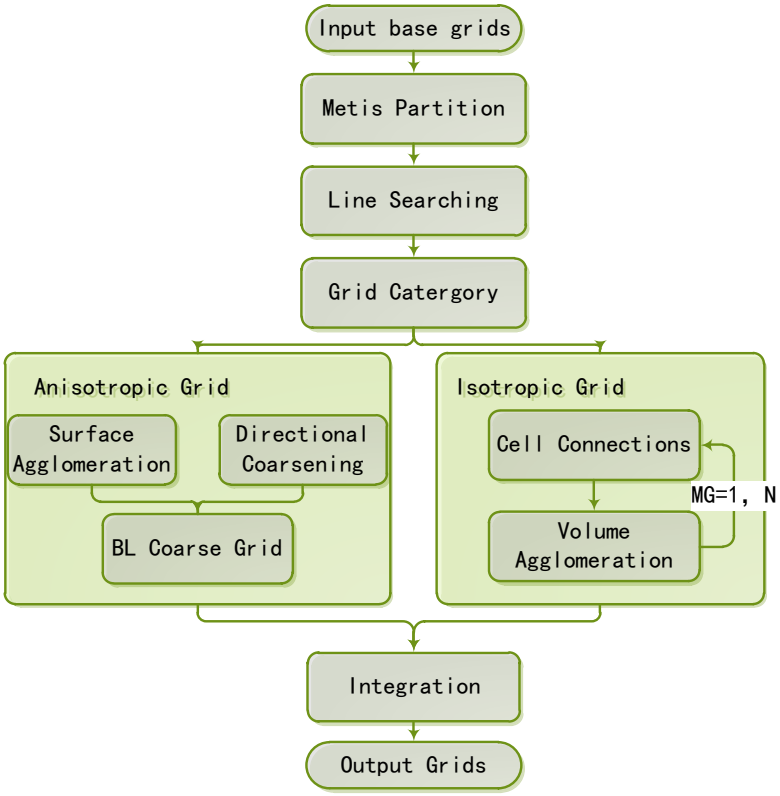


Figure 1 Flow chart of grid coarsening.

2.2 Multigrid Discretization

Nested iterations are central of multigrid, relaxations on coarse grid level are efficient to cancel low frequency errors on base grid level[14][15]. On each coarse level, errors are restricted from finer grids, and the flow corrections are prolonged back after relaxation. The discretization form of N-S equations on base grid is as the following.

$$L_h U_h = f_h \tag{1}$$

Where L represents difference operator, U is the flow variable, f is a known function, and subscript h represents grid size. After relaxation, the approximate flow values and errors satisfy

$$L_h \bar{U}_h = f_h + R_h \tag{2}$$

Substract (2) from (1), then

$$L_h U_h - L_h \bar{U}_h = -R_h \tag{3}$$

Restrict the error equations to coarser grids, then

$$L_{2h}U_{2h} - L_{2h}\left(I_h^{2h}\right)\bar{U}_h = -I_h^{2h}R_h \tag{4}$$

This is the equation to be relaxed on coarse grids. In case of nonlinear equations, the first and second term on the left side can't be merged to get simplified error equations, so full quantity equations need to be relaxed on coase levels, which is called full approximation scheme(FAS) of multigrid. By defining force function as

$$f_{2h} = L_{2h}\left(I_h^{2h}\right)\bar{U}_h - I_h^{2h}R_h \tag{5}$$

Then the approximate values on coarse grids satisfy,

$$L_{2h}\bar{U}_{2h} = f_{2h} + R_{2h} \tag{6}$$

It shows that the iteration formulation on coarse grids is equivalent to that of the fine grids. Restriction and prolong operators are needed to transfer flow information in multigrid hierarchies, which are straight injection and linear interpolation respectively in this paper.

Polyhedrons are generated by grid coarsening, the averaged face number of coarse elements increase by the agglomeration times. Taking hexahedron as an example, if the agglomeration ratio in all directions is 2:1, the face number of coarse elements is 24 after an agglomeration, which is 96 after the second one. Too many faces increase the complexities of flux computation. To avoid this problem, public faces of two neighboring coarse elements are merged before computation, and new faces and connections are constructed, which significantly reduce the time consuming on flux integration.

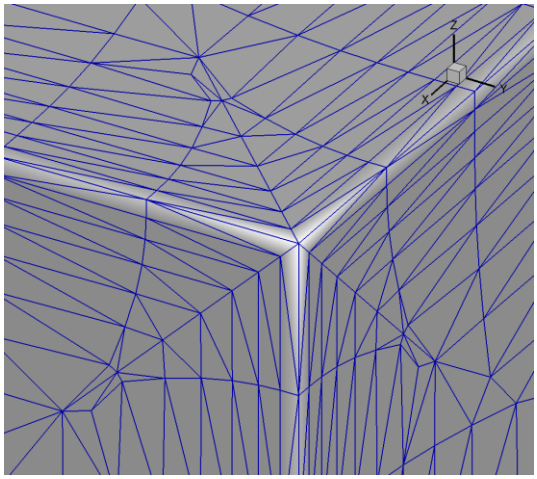
Cell centered scheme is used to discretize the flow equations on fine and coarse elements. On coarse levels of multigrid hierarchies, convective fluxs are calculated by flux difference splitting based Roe upwind scheme, and the 1st order reconstructions of flow variables are used to assure computation robustness, which set the face variables by cell center value. On coarse levels diffusive fluxs are calculated by thin layer approximation, and the turbulence models are solved only on the finest grid, turbulence quantities on coarse grids are obtained by interpolation.

3. Results and Analysis

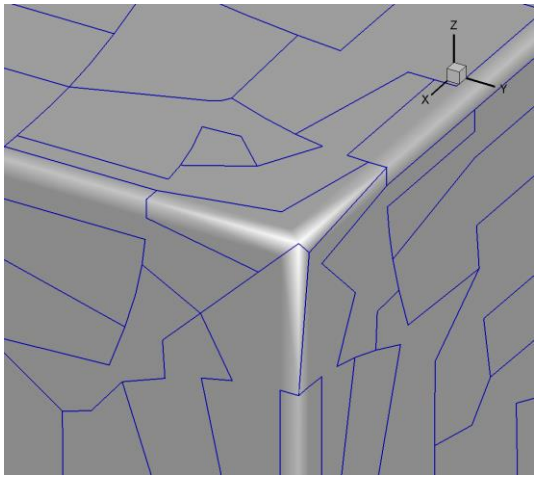
3.1 Cube

Grid coarsening is firstly validated on a cube for its typical geometrical features. The primal surface grids are shown in Figure 2(a), whose grid points on each edge are evenly distributed, and the number of grid points is 13. Surface grids are generated by extrusion from the edges, and the cell heights near the edges are 1% of edge length. Volume grids are generated by advancing layer method. The wall distance of first layer grid is 1% of edge length, and the growth ratio is 1.1. Volume cells consist of prisms in boundary layer and tetrahedrons in outer domain.

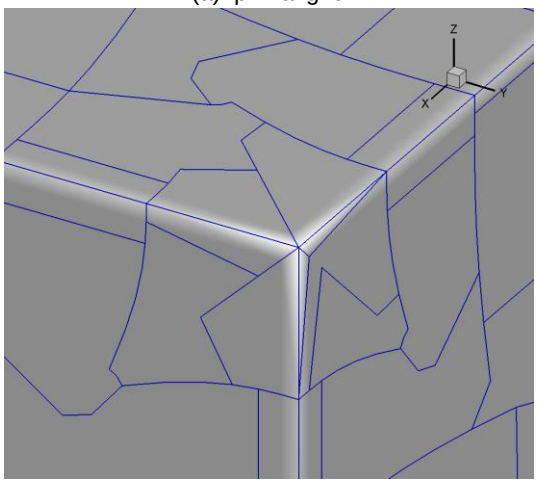
Three agglomeration methods are compared in Figure 2 through coarse surface grids, which are generated from primal grids in Figure 2(a). Figure 2(b) shows the result by volume agglomeration, which treats all volume cells equally and the coarsening ratio is 1:8. Figure 2(b) indicates current algorithm doesn't identify the edges, and as a result cells on different faces are fused together. It shows the result of directional coarsening in Figure 2(c) and Figure 2(d). In the process of surface agglomeration, included angles of surface are calculated to identify whether they are on the same faces, otherwise they are not allowed to be fused. In Figure 2(c), surface identifications are done after the fusion, and the fusions are canceled if they are located on different faces. In Figure 2(d), surface identifications are executed on primal surface grids, if neighboring cells are not on the same faces, their connections are canceled. By comparing Figure 2(c) with Figure 2(d), it shows that strategy of Figure 2(d) is more advantageous, where the shape and aspect ratio of coarse cells near the edge are more regular.



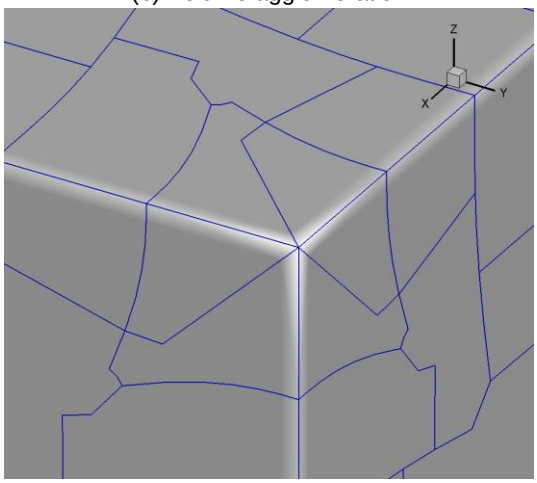
(a) primal grid



(b) Volume agglomeration



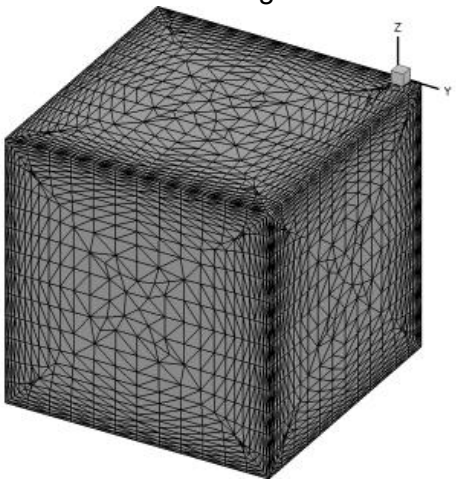
(c) Directional coarsening(way 1)



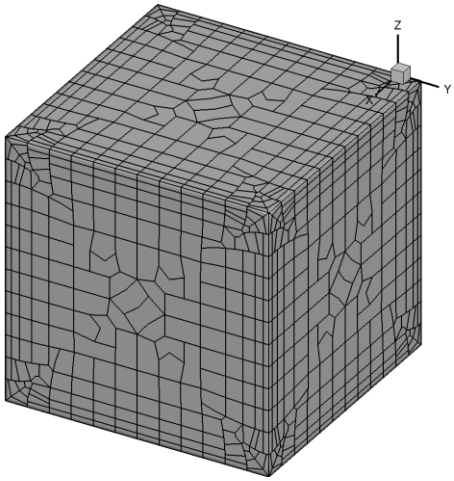
(d) Directional coarsening(way 2)

Figure 2 Coarse grids near edges by various methods.

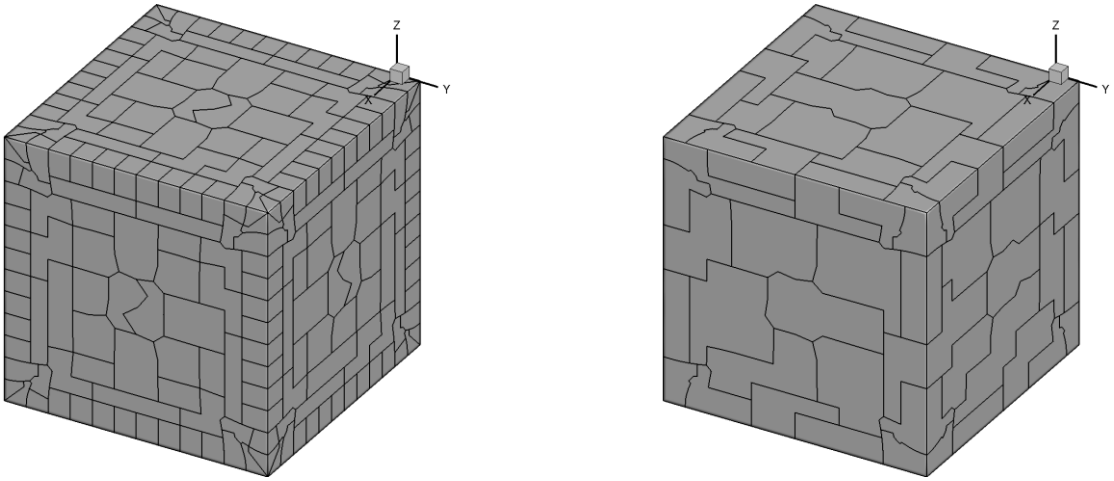
By directional coarsening method as in Figure 2(d), we have got 4-level multigrid hierarchies whose surface grids are presented in Figure 3. It shows that by setting coarsening ratio as 1:4, coarse elements on the second level are mainly rectangles of low aspect ratios. Elements on the third and fourth levels are mainly quadrilaterals and polygons. Generally these elements are compact in shape, which conform to the goal of minimum aspect ratio. Besides, it shows in Figure 3 that all edges of the cube are saved on coarse levels, which indicates the coarse surface elements don't cross the edge and are beneficial to finite volume discretization.



(a) primal grid



(b) level 2 coarse grid



(c) level 3 coarse grid (d) level 4 coarse grid

Figure 3 Coarse surface grids on multigrid levels.

3.2 M6 Wing

For validation of current method, M6 wing is studied on unstructured grid coarsening and multigrid computation. The primal grid is shown in Figure 4, which consists of .532 million cells including prisms in boundary layer and tetrahedrons in outer domain. In the boundary layer, the first grid distance is 3×10^{-5} m, and the number of advancing layer is 32 with growth ratio 1.2.

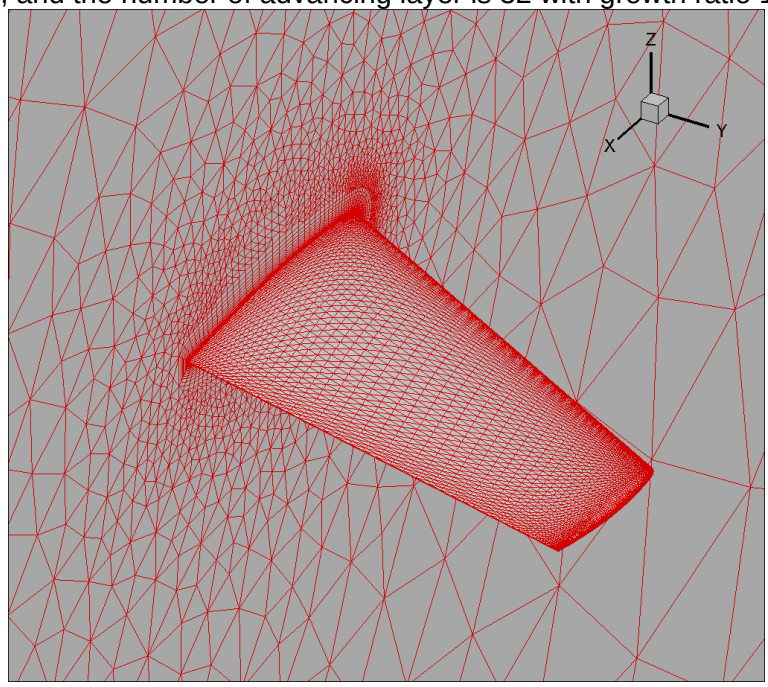
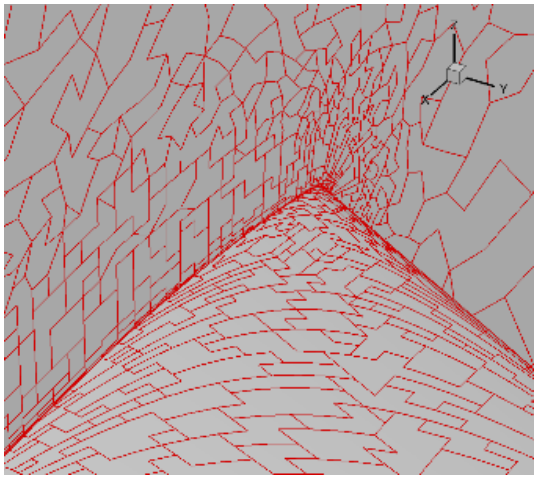
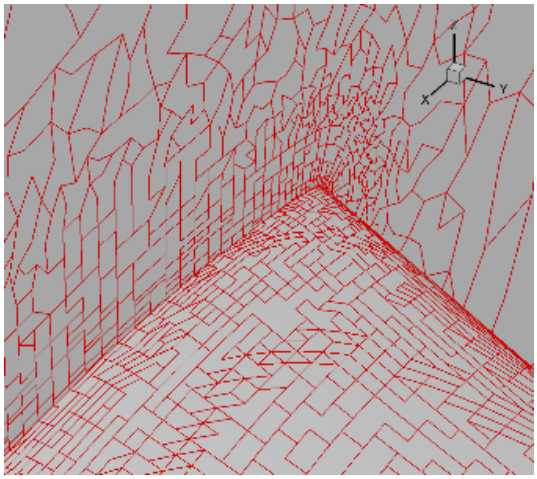


Figure 4 Coarse surface grids on multigrid levels.

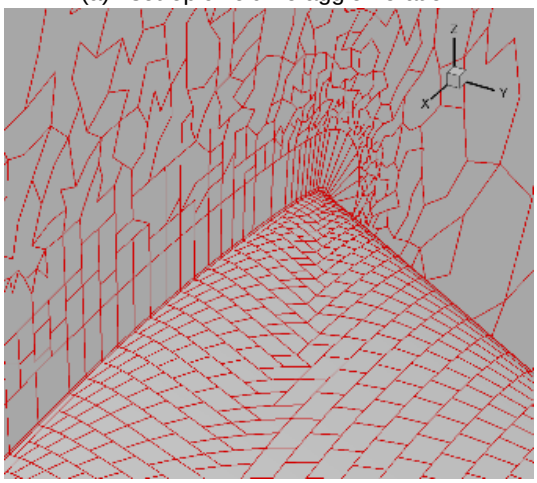
Four agglomeration methods are compared for grid coarsening. The first one is isotropic volume agglomeration, where volume cells are treated equally with the same connection weights. The second is anisotropic volume agglomeration, which treats the face area as the weight of cell connection. The third and fourth both use directional coarsening method, and the difference lies in coarsening direction. In the third method, a coarse element consists of 2 surface faces and 4 normal cells, and in the fourth method a coarse element consists of 4 surface faces and 2 normal cells. Surface grids obtained by four agglomeration methods are compared in fig.5, where the surface grids of the wing and symmetry plane on the third multigrid level are displayed. It shows that the directional coarsening method improves the coarse surface grids significantly, and the ratio of surface coarsening to normal coarsening can be adjusted freely.



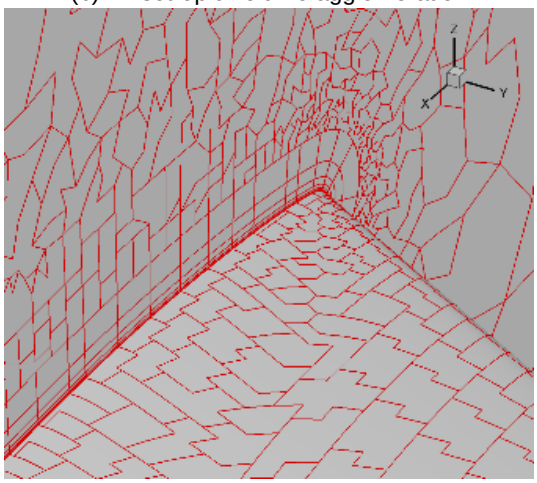
(a) Isotropic volume agglomeration



(b) Anisotropic volume agglomeration



(c) Directional coarsening(surface 1:2, normal 1:4)



(d) Directional coarsening(surface 1:4, normal 1:2)

Figure 5 Coarse surface grids by various methods.

Effects of agglomeration methods on speeding up convergence are displayed in fig.6. in the simulation the free stream mach number is 0.78, and the attack angle is 4.5 degree. Three level w-cycle multigrid is applied in the computation, and the relaxation steps on each level is 10. convergence histories of different multigrid levels are compared in fig.7, where coarse grids are generated by directional coarsening, with surface coarsening ratio of 1:2 and normal coarsening ratio of 1:4. It shows in fig.7 that speeding up effects of multigrid are notable, where the L2 norm of density residuals takes 4800 steps to reach 10^{-11} by single grid, and it take 800 steps and 500 steps respectively by 3-level and 4-level multigrid, whose speeding up ratios are 6 and 10.

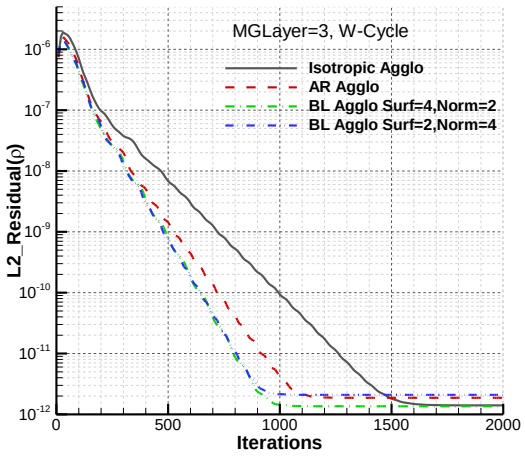


Figure 6 Density residuals by various agglomeration methods.

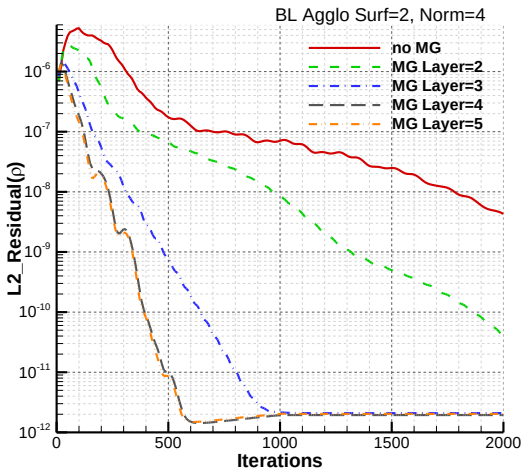
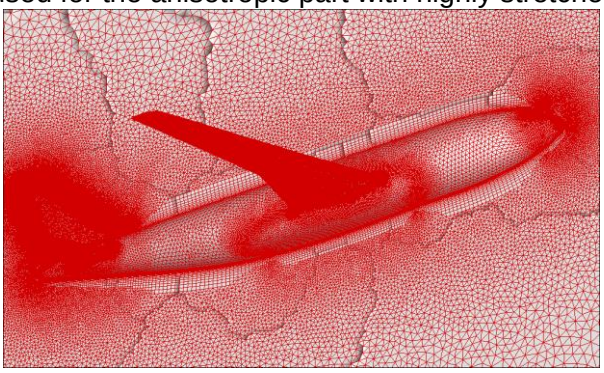


Figure 7 Effects of multigrid levels on convergence.

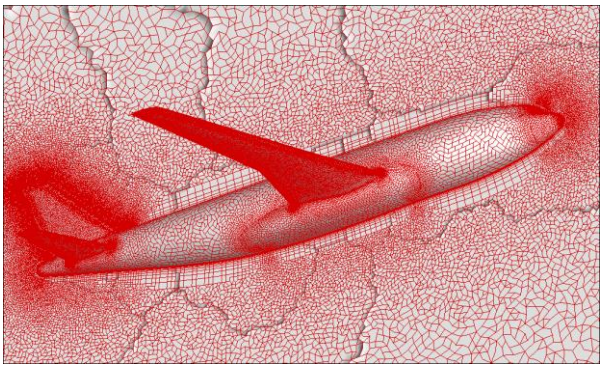
3.3 Transport Aircraft Model CHNT-1

For further validate current methods in engineering applications, geometric multigrid computations are carried out on an transport aircraft model(CHNT-1). The model conists of airframe, wing, horizontal and vertical tails, and the wind tunnel tests are carried out in 2.4m × 2.4m transonic wind tunnel of CARDC, and 2.4m × 2.0m high Reynolds number transonic wind tunnel of ETW. The model and experimental data are extensively discussed in the national seminar of CFD credibility in 2018[16][17][18]. test cases in CARDC are simulated in this paper, the model scale is 0.052; free stream mach number is 0.78, and the chord based Reynolds number is 3.3×10^{6}. Hybrid unstructured grids with medium densities are used in the simulation, as shown in fig.8(a), extrusion grid are generated in the boundary layer with the first layer grid height being 1.0×10^{-6}m, and growth ratio being 1.2. the total number of grid cells are 17.38M, which consists of 8.44M tetrahedrons, 8.88M prisms and 0.05M pyramids.

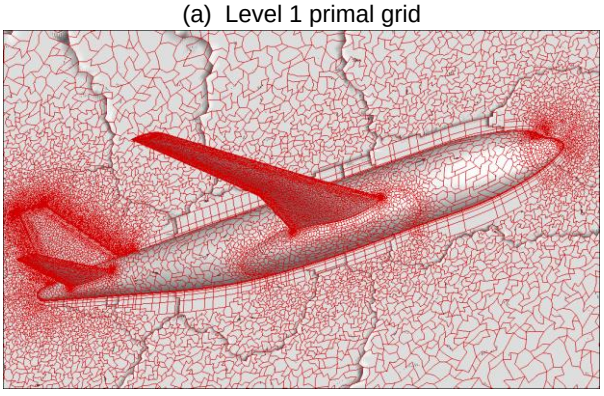
fig.8(a)~(e) shows the primal grid on CHNT-1 model and sequence of coarse grids generated by grid coarsening. In primal grid, there are 214 thousand surface cells in total, and the maximum number of advancing layer is 45 in boundary layer, the minimum number is 32. There are 5 levels in the multigrid hierarchy, and from the finest to the coarsest level, volume elements are totally nested. In these figures, the shadow lines indicate the boundaries of parallel partitions. For each zone, grid cells are divide into anisotropic part and isotropic part, and the directional coarsening method is used for the anisotropic part with highly stretched cells.



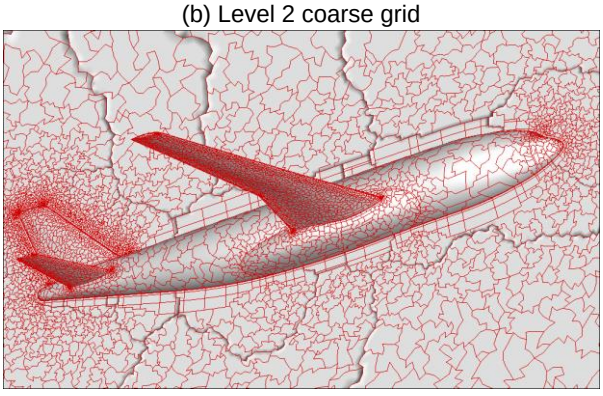
(a) Level 1 primal grid



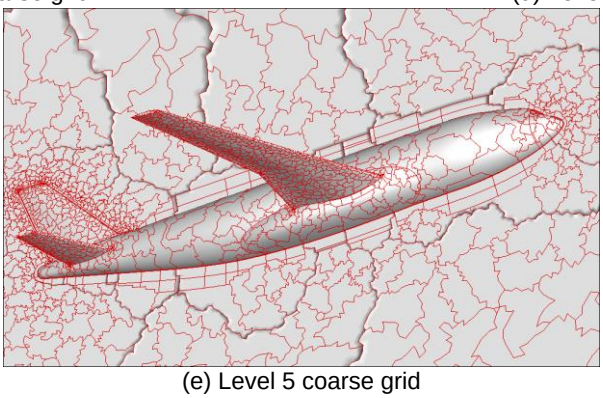
(b) Level 2 coarse grid



(c) Level 3 coarse grid



(d) Level 4 coarse grid



(e) Level 5 coarse grid

Figure 8 Primal and coarse levels of CHNT-1 grid.

Owing to surface agglomeration in directional coarsening method, the coarse cell shapes are given the chance to be optimized with low aspect ratio, which are shown in fig.8. For example, stretched triangular cells are generated near leading and rear edges of the wing in primal grid, and on the second level more regular quadrilaterals are generated by agglomeration. As the coarse level increases, it's possible that there are less and less available neighboring cells to be fused, which cause irregular concave elements that can be found in fig.8(e). surface grid on the symmetry plane indicate the normal distribution of coarse elements, it shows in fig.8(b)~fig.8(e) that in current method normal coarse elements are regular and ordered, which are beneficial to the simulation of boundary layer flows.

W-cycle multigrid are applied in this computation, where the smoother is implicit lower-upper symmetric Gauss Seidal(LU-SGS) scheme, and the steps of relaxation on each level are 10. Effects of multigrid level on speeding up convergence are compared in fig.9 by L2 norm of density residual. It shows that the speeding up ratios are significant for all multigrid levels, where the 3-level multigrid computation accelerate the convergence by 10 times by comparison to single grid computation, and the 4-level multigrid is slightly faster than 3-level. It's notable that the 5-level multigrid show almost the same speeding up effects with 4-level, and the minimum error can't reach as low as 4-level. From this we can infer that qualities of coarse grids have equal importance on the multigrid efficiency with multigrid levels. On one side, higher level improves the speeding up ratio, but on the other side grid qualities are hard to keep for higher levels, so it's vital to find a balance point in the computation.

Comparisons of lift coefficients and drad coefficients are shown in fig.10 and fig.11, where square symbol represents experimental data, shadow region represent statistic data from national seminar of CFD credibility in 2018. In the CFD seminar data, the center dash line is the averaged value of 22 proposed CFD results, and the upper and lower boundaries are the standard deviation. It shows from these figures that current computational results agree quite well with experimental data, and are basically consistent with other CFD data, which in some extent verify current method.

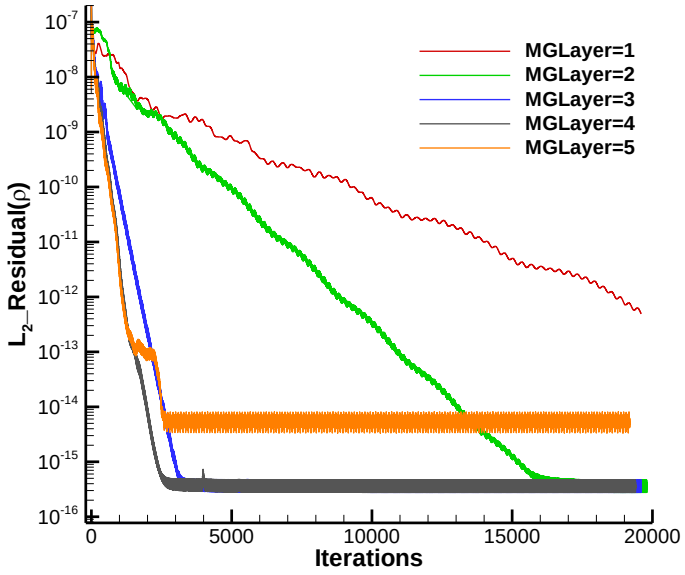


Figure 9 Speeding up effects by various multigrid levels .

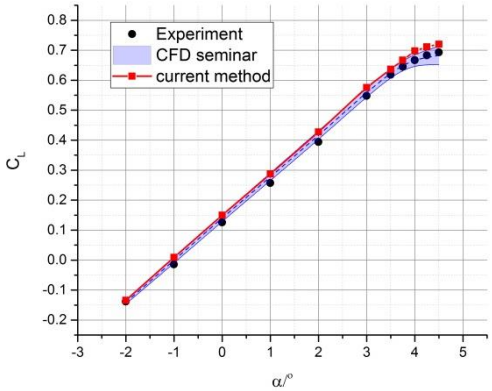


Figure 10 Comparisons of lift coefficients.

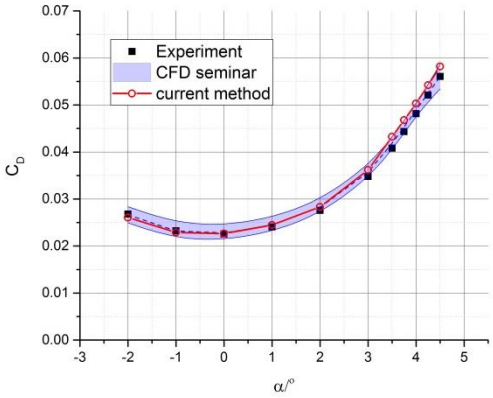


Figure 11 Comparisons of drag coefficients..

4. Concluding Remarks

Coarsening strategies of unstructured multigrid for viscous flows are discussed in this paper. A new method is proposed to address the highly stretched grids in boundary layer, which use line coarsening in extrusion direction and surface agglomeration on wall. Grid qualities and speeding up effects of multigrid are investigated, and main concluding remarks are as follows.

(1) In the process of surface agglomeration, normal vectors of neighboring surface cells are calculated to decide whether they are located on the same face, otherwise the clusterings are avoided by deleting the cell connection, which helps to keep the geometric features on coarse grid levels.

(2) Directional coarsening method based on line searching helps to control the direction and size ratio of grid clustering in boundary layer, which improve the element's quality on coarse levels of multigrid.

(3) Significant speeding up effects are achieved by directional coarsening method in cases of M6 wing and CHN-T1 transport aircraft model. By comparison with isotropic agglomeration, directional coarsening speeds up the convergence by 1.58 times for 3 level W-cycle multigrid, and the optimum multigrid computation is about 10 times faster than single grid.

References

[1] Xu X W. Parallel Algebraic Multigrid Algorithms: Current Status and Challenges of Large-scale Computing Applications[J]. Numerical calculation and computer application, 2019, 40(4): 243-260. (in Chinese)

[2] Strauss D,Azevedo J L F. On the development of an agglomeration multigrid solver for turbulent flows [J]. Journal of the Brazil society of mechanical science and engineering, 2003, 15(4): 315-324

[3] Li Z Z. Research on parallel multigrid algorithm for unstructured grid[D]. Doctoral dissertation, National university of defense technology, 2012. (in Chinese)

[4] Pandya M,Frink N T. Agglomeration multigrid for an unstructured grid flow solver [R]. AIAA 2004-759, 2004

[5] Mahmutyazicioglu E, Tuncer I H,Aksel H. Octree based 3-D unstructured grid coarsening and multigrid viscous flow solutions [R]. AIAA 2011-757, 2011

[6] Moulitsas I,Karypis G. Multilevel algorithms for generating coarse grids for multigrid methods [A]. In.Supercomputing 2001 conference proceedings [C]. 2001.

[7] Dargaville S, Buchan A G,Smedley-Stevenson R P. A comparison of element agglomeration algorithms for unstructured geometric multigrid [J]. arXiv:2005.09104v1, 2020:

[8] Pan Y,Persson P O. Agglomeration-based geometric multigrid solvers for compact discontinuous galerkin discretizations on unstructured meshes [J]. aRxIV:2012.08024V1, 2020:

[9] Zenoni G, Leicht T, Colombo A. An agglomeration-based adaptive discontinuous galerkin method for compressible flows [J]. International Journal for Numerical Methods in Engineering, 2017, 13(2): 1-20

[10] Nishikawa H,Diskin B. Recent advances in agglomerated multigrid [R]. AIAA 2013-0863, 2013

[11] Nishikawa H,Diskin B. Development and application of parallel agglomerated multigrid methods for complex geometries [R]. AIAA 2011-3232, 2011

[12] Karypis G,Kumar V. Multilevel algorithms for generating coarse grids for multigrid methods [J]. Journal of parallel and distributed computing, 1998, 48(1): 96-129

[13] Moulitsas I,karypis G. Mgridgen/parmgridgen serial/parallel library for generating coarse grids for multigrid methods version 1.0 [M]. Minneapolis, MN 55455: University of Minnesota, 2001.

[14] Gerlinger P, Stoll P,Bruggemann D. an implicit multigrid method for the simulation of chemically reacting

flows [J]. Jounal of computational physics, 1998, 146: 322-345
[15] Trottenberg U, Oosterlee C W,Schuller A. Multigrid [M]. Elervier(Singapore) Pte Ltd., 2015.
[16] Wang Y T, Liu G, Chen Z B. Summary of the first aviation CFD credibility seminar[J]. Acta Aerodynamica Sinica, 2019, 37(2): 247-261. (in Chinese)
[17] Yu Y G, Zhou Z, Huang J T. Design of CHN-T1 aerodynamic standard model for single pass airliner[J]. Acta Aerodynamica Sinica, 2018, 36(3): 505-513. (in Chinese)
[18] Li W, Wang Y T, Hong J W. Simulation of the aerodynamic characteristics of CHN-T1 model by using TRIP3.0[J]. Acta Aerodynamica Sinica, 2019, 37(2): 272-279. (in Chinese)