

Virtual Control Volumes for Two-Dimensional Unstructured Elliptic Smoothing

Steve L. Karman Jr.

SimCenter: National Center for Computational Engineering
University of Tennessee at Chattanooga
Chattanooga TN, USA
Steve-Karman@utc.edu

Summary

A two-dimensional unstructured elliptic smoothing method is described where the Winslow equations are discretized using a finite volume approach. Virtual control volumes for each node are constructed with element shapes that are nearly ideal. Green-Gauss theorem is used to formulate gradients over an element or a collection of elements for a node, which ultimately leads to a coupled non-linear system of equations. Modifications enable the scheme to reproduce results similar to structured mesh schemes. Results are included that demonstrate basic mesh smoothing and boundary motion. In addition, layers of quadrilateral elements can be added to selected boundaries and the interior point positions are determined via elliptic smoothing.

Keywords: Unstructured elliptic smoothing, Winslow smoothing, Finite-volume control volume, Mesh quality improvement

1. Introduction

Elliptic smoothing has been routinely used with structured meshes for many years [1]. It is a very mature technology that allows mesh generation practitioners to create smooth structured meshes. Forcing functions have been developed that can influence the grid spacing and grid line angularity to meet the specific needs of the analysis tool [2, 3, 4]. This technology has been mostly unavailable to the unstructured mesh community, due to the lack of a global mapping from physical space to computational space. Knupp described an approach for smoothing two-dimensional unstructured meshes using a finite difference method where the local logical space was created using trigonometric functions [5]. Finite difference formulae were then derived using Taylor series expansions about the central node. Results in his article show quadrilateral meshes, although the derivation includes an arbitrary number of neighboring nodes, which could be part of triangles and quadrilaterals. Knupp also refers to a proprietary publication by Tipton that is an extension of the method to three dimensions using a finite element approach [6]. Karman et. al described a finite-volume method in two and three dimensions for unstructured meshes comprised of mixed elements [7]. In the finite-volume approach elliptic smoothing of unstructured meshes was possible if an existing valid mesh is available to serve as the computational domain. Virtual control volumes were then introduced where a local computational space for each node is constructed that is independent from a neighboring node's computational space [8]. These virtual control volumes are the key component of the method and the current paper describes in detail the construction and use of these virtual control volumes for meshes comprised of quadrilaterals and

triangles. Aspects of the discretization strategy are explored to allow for the same type of grid control afforded to the structured community. Results for smoothing of static and dynamic problems are included. The cases include hybrid meshes where the quadrilateral elements are added to existing meshes with the points positioned using elliptic smoothing.

2. Winslow Smoothing on Structured Meshes

Most elliptic smoothing methods utilize the Winslow equations as the governing equations for mesh distribution [1]. These equations are derived from a Laplacian operator or Poisson operator applied to the computational coordinates. The Laplacian operator form is shown in equations (1). When satisfied, these relationships describe a smooth distribution of computational coordinates (ξ, η) in physical space (X, Y).

$$\begin{aligned}\nabla^2 \xi &= \xi_{xx} + \xi_{yy} = 0 \\ \nabla^2 \eta &= \eta_{xx} + \eta_{yy} = 0\end{aligned}\quad (1)$$

Theoretically the Laplace equations satisfy the max-min property, which states that the parameter on the interior of the domain will not exceed the values on the boundary. For mesh generation this means the grid lines will not cross. The known variables are the computational coordinates ξ and η . The unknown variables are x and y. So a transformation is necessary to cast the equations in computational space, resulting in the Winslow equations given in equations (2).

$$\begin{aligned}\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} &= 0 \\ \alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} &= 0 \\ \alpha &= x_\eta^2 + y_\eta^2 \\ \beta &= x_\xi x_\eta + y_\xi y_\eta \\ \gamma &= x_\xi^2 + y_\xi^2\end{aligned}\quad (2)$$

Winslow smoothing is not the same as Laplacian smoothing. Many researchers will perform Laplacian smoothing, which is based on the Laplacian operator in equation (1) cast in computational space, shown in equation (3).

$$\begin{aligned}\nabla^2 x &= x_{\xi\xi} + x_{\eta\eta} = 0 \\ \nabla^2 y &= y_{\xi\xi} + y_{\eta\eta} = 0\end{aligned}\quad (3)$$

Laplacian smoothing for structured meshes is essentially an averaging of surrounding nodes physical coordinates. The same is true when implemented for unstructured meshes. It does not contain the cross derivative term of the Winslow equations and can result in grid crossing, especially for non-convex domains.

Finite Difference

For structured meshes the Winslow equations are typically solved on an equally spaced Cartesian grid in computational space. The stencil typically used to discretize the first and second derivative terms is shown in Figure 1. The numerical formulae for the first and second derivatives for x in ξ and η directions is shown in equations (4). These are centered difference formulae for the central node. Notice that the derivatives in the ξ direction use only nodes on a constant j line

and the derivatives in the η direction using only nodes on a constant i line, while the cross derivative formula uses only the corner nodes. Figure 2 shows the resulting mesh for a spike case where the center of the bottom boundary protrudes into the middle of the domain. The boundary nodes are fixed in position. Winslow smoothing produces a mesh without crossing at the spike. This illustrates that the max-min property of the underlying Laplacian operator is being enforced.

$$\begin{aligned}
 x_{\xi} &= \frac{x_{i+1,j} - x_{i-1,j}}{2\Delta\xi} \\
 x_{\eta} &= \frac{x_{i,j+1} - x_{i,j-1}}{2\Delta\eta} \\
 x_{\xi\xi} &= \frac{x_{i+1,j} - 2x_{i,j} + x_{i-1,j}}{\Delta\xi^2} \\
 x_{\eta\eta} &= \frac{x_{i,j+1} - 2x_{i,j} + x_{i,j-1}}{\Delta\eta^2} \\
 x_{\xi\eta} &= \frac{x_{i+1,j+1} - x_{i+1,j-1} - x_{i-1,j+1} + x_{i-1,j-1}}{\Delta\xi\Delta\eta}
 \end{aligned} \tag{4}$$

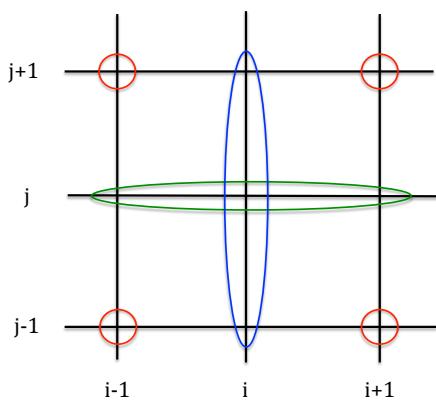


Figure 1. Finite difference stencil.

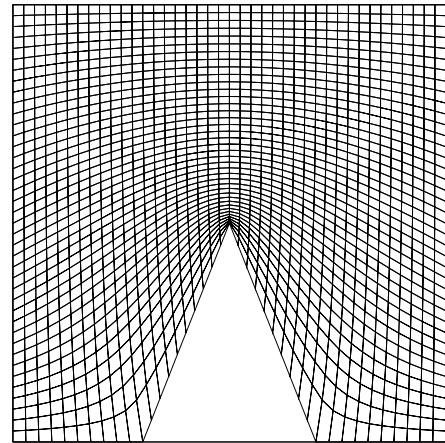


Figure 2. Structured mesh Winslow solution for spike case.

3. Virtual Control Volumes for Unstructured Meshes

In order to solve the Winslow equations on unstructured meshes a computational domain must be created. A global computational domain is possible if one uses an existing valid mesh. However, this will not provide any real benefit, since a valid mesh already exists. Instead, it is possible to construct a computational domain for each node. These “virtual control volumes” were first introduced in References [8] and [9]. Under the premise that a local computational domain comprised of nearest neighbors is necessary to assemble the global system of equations, it is possible to construct nearly ideal computational stencils for each node. This does assume that the valid global mesh connectivity does exist. These stencils are comprised of the elements connected to the central node. However, instead of using the physical coordinates for each node

in the stencil, computational coordinates are constructed. These computational elements can be created to have nearly ideal shape quality or can be manipulated to reflect a desired stretching or clustering, corresponding to a viscous layer distribution or an adaptation spacing field.

The creation of these computational stencils in two dimensions is relatively straightforward. The stencil for each node will have an arbitrary number of surrounding elements and neighboring nodes; information that is contained in the global mesh connectivity. The process involves distributing directly connected nodes on the unit circle and then positioning any secondarily connected nodes to produce ideal element shapes. Directly connected nodes are those nodes that share an edge with the central node. Secondarily connected nodes do not share an edge with the central nodes, such as the opposite corner of a quadrilateral element. Figure 3 shows a region of a hybrid mesh containing triangular and quadrilateral elements. Figure 4 shows a sample virtual control volume for a node at the edge of the quadrilateral layers (any of the 4 nodes in the center region of the plot with the same element connectivity).

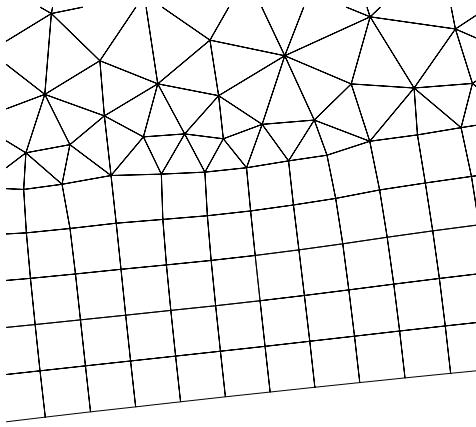


Figure 3. Physical mesh containing triangles and quadrilaterals.

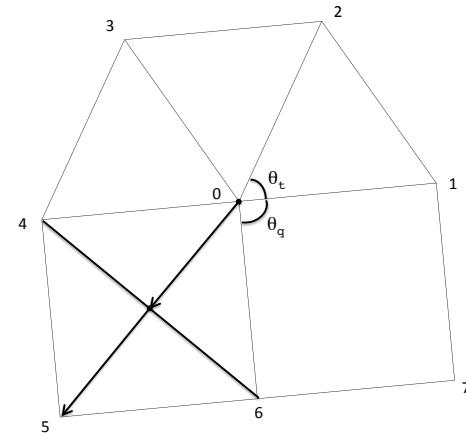


Figure 4. Sample virtual control volume for node at edge of quadrilateral layers in Figure 3.

The central node is labeled ‘0’. The surrounding nodes are numbered 1 through 7, in order. The directly connected nodes are 1, 2, 3, 4, and 6. The secondarily connected nodes are 5 and 7. The first directly connected node is positioned on the unit circle in a direction that corresponds to the direction of the physical node, although this is not necessary. It could be positioned at computational coordinate (1, 0). The remaining directly connected nodes are then distributed on the unit circle according to the angle distributions, θ_t , and θ_q . These angles are determined based on the number of triangles (nt) and the number of quadrilaterals (nq) connected to the central node, given in equations (5). The goal is to produce angles and computational coordinates that reflect the desired element shapes in physical space.

$$\begin{aligned}
 & \text{if } nt = 0 & \theta_q = \frac{2\pi}{nq} \\
 & \text{else if } nq = 0 & \theta_t = \frac{2\pi}{nt} \\
 & \text{else if } nq = 1 \text{ and } nt > 1 & \theta_t = \frac{3\pi}{2nt}, \quad \theta_q = \frac{\pi}{2} \\
 & \text{else if } nt = 1 \text{ and } nq > 1 & \theta_t = \frac{\pi}{2}, \quad \theta_q = \frac{3\pi}{2nq} \\
 & \text{else if } nt > 1 \text{ and } nq > 1 & \theta_t = \frac{\pi}{nt}, \quad \theta_q = \frac{\pi}{nq}
 \end{aligned} \tag{5}$$

The secondarily connected nodes are then placed opposite the central node using a vector emanating from the central node in the direction of the midpoint of the diagonal edge of the connected nodes, shown in Figure 4 as edge 4 – 6. That vector is copied to the midpoint of the diagonal edge and the secondarily connected node, node 5 in the figure, is positioned at the end of the vector. The length of that vector is the minimum of the original vector length and $\sqrt{3}/2$ times the length of the diagonal edge. This will result in perfect squares for the quadrilateral elements when the number of quadrilaterals is equal to 1, 2 or 4. Otherwise, the outer portion of the quadrilateral will be an equilateral triangle. Examples of virtual control volumes for all-triangular stencils and all-quadrilateral stencils are shown in Figure 5.

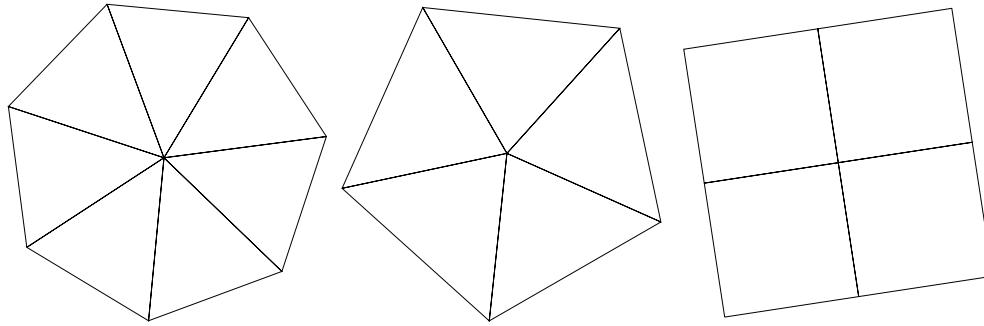


Figure 5. Examples of virtual control volumes for different numbers of triangles and an all-quadrilateral stencil.

Triangles in these virtual control volume will be equilateral only when the triangle angle, θ_t , is equal to $\pi/3$, which occurs with 6 triangles in an all-triangular stencil or 3 triangles in a hybrid stencil, such as the one shown in Figure 4. Changing the lengths of the edges emanating from the central node does not alter the angle relationships. These are the “best” control volumes possible and the physical mesh will attempt to mimic the shapes of these virtual elements.

These two-dimensional control volumes are essentially equivalent to the local logical space described by Knupp [5]. The adjacent nodes are ordered in a counter-clockwise fashion and the computational coordinates are distributed on a unit circle. Knupp’s finite difference formulae include cosine and sine functions, which can be computationally prohibitive. The current approach computes and stores computational coordinates. The method could be extended to include general polygons instead of just triangles and quadrilaterals.

4. Finite Volume Formulation of Winslow Equations

One approach to solving the Winslow equations is to consider the coefficients, α, β, γ as constant. Then integrate the first equation from (2) over $\xi\text{-}\eta$ space.

$$\alpha \iint x_{\xi\xi} d\Omega - 2\beta \iint x_{\xi\eta} d\Omega + \gamma \iint x_{\eta\eta} d\Omega = 0 \quad (6)$$

Ω represents the area in two dimensions. Examine the first integral and define \vec{v} such that

$$\begin{aligned} \vec{v} &= x_{\xi} \hat{i} + 0 \hat{j} \\ \nabla \bullet \vec{v} &= x_{\xi\xi} \end{aligned} \quad (7)$$

The divergence theorem can be used to convert the double integral to a line integral.

$$\iint \nabla \bullet F d\Omega = \oint F \bullet \hat{n} d\Gamma \quad (8)$$

Γ represents the boundary in two dimensions. Therefore, the first integral becomes

$$\iint x_{\xi\xi} d\Omega = \oint x_{\xi} \hat{n}_{\xi} d\Gamma \quad (9)$$

A similar operation is used to convert the other double integrals terms. The resulting set of integral equations become

$$\begin{aligned} \alpha \oint x_{\xi} \hat{n}_{\xi} d\Gamma - 2\beta \oint x_{\eta} \hat{n}_{\xi} d\Gamma + \gamma \oint x_{\eta} \hat{n}_{\eta} d\Gamma &= 0 \\ \alpha \oint y_{\xi} \hat{n}_{\xi} d\Gamma - 2\beta \oint y_{\eta} \hat{n}_{\xi} d\Gamma + \gamma \oint y_{\eta} \hat{n}_{\eta} d\Gamma &= 0 \end{aligned} \quad (10)$$

The numerical representation of these surface integrals is given in equations (11). The summation is over the surrounding elements. This is illustrated in Figure 6 for a hybrid node surrounded by 3 triangles and 2 quadrilaterals for a total of 5 elements.

$$\begin{aligned} \alpha \sum_{i=1}^{ne} [x_{\xi}]_i (t_{\xi})_i - 2\beta \sum_{i=1}^{ne} [x_{\eta}]_i (t_{\xi})_i + \gamma \sum_{i=1}^{ne} [x_{\eta}]_i (t_{\eta})_i &= 0 \\ \alpha \sum_{i=1}^{ne} [y_{\xi}]_i (t_{\xi})_i - 2\beta \sum_{i=1}^{ne} [y_{\eta}]_i (t_{\xi})_i + \gamma \sum_{i=1}^{ne} [y_{\eta}]_i (t_{\eta})_i &= 0 \end{aligned} \quad (11)$$

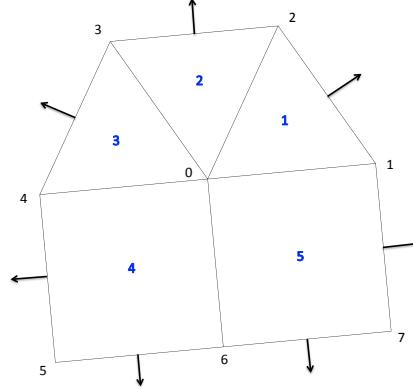


Figure 6. Control volume and external area vectors for a hybrid node.

The normal vectors, \vec{t} , correspond to the normal vectors of the external edge of the surrounding elements. The first derivative terms in (11) are replaced with a formula for the corresponding component of a gradient calculation of each element using a Green-Gauss formulation. For instance, the derivative of x with respect to ξ is given by (12).

$$x_{\xi} = \frac{\sum_{j=1}^{ns} \frac{(x_l + x_r)}{2} (n_{\xi})_j}{A} \quad (12)$$

where A represents the area of the element. The subscripts l and r correspond to the left and right node numbers for edge j . The ξ component of the normal vector for side j is $(n_{\xi})_j$. The summation is carried out over the number of sides of the element. A similar expression can be developed for the η derivative.

Substituting for all the first derivative terms will result in a system of equations for node 0 using nearest neighbor node information. Combining the equations from all nodes in the mesh will result in a sparse matrix linear system of equations that can be solved for new physical

coordinates (x,y) using techniques such as point-implicit scheme with under-relaxation. Since the coefficients α , β , and γ were assumed frozen, but are actually functions of the physical locations, an outer iteration loop is required to update these coefficient values. When a point-implicit method is used an inner iteration loop is used to converge the linear system. Full convergence of the inner iterations is not necessary. Convergence of the outer iterations is required.

Quadrilateral Case

A comparison of the finite volume approach with the structured, finite-difference approach is instructive. The structured mesh from Figure 2 was converted to an unstructured collection of quadrilaterals and solved using the finite volume methods described above. The control volume for an interior node is shown in Figure 7. In this case, the first directly connected node is positioned at computational coordinate (1, 0). The resulting unstructured Winslow solution in Figure 8 is different from the structured mesh solution. There is grid crossing at the top of the spike and some kinks in the mesh near the corners on the bottom boundary.

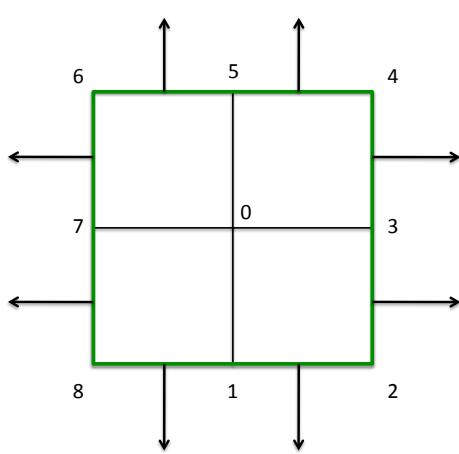


Figure 7. Control volume for central node in Finite Volume method.

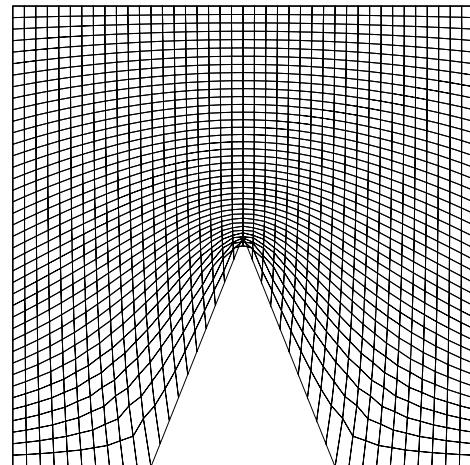


Figure 8. Result of Winslow smoothing using Finite Volume method.

An examination of first derivatives using the finite volume approach reveals differences with the finite difference formulae. Assuming unit edge lengths in the control volume, equation (13) shows influence from the corner nodes 2, 4, 6 and 8 in computing the derivative in the ξ direction, whereas the structured finite difference formula uses only information corresponding to nodes 3 and 7.

$$x_{\xi} = \frac{(x_4 + 2x_3 + x_2) - (x_6 + 2x_7 + x_8)}{8} \quad (13)$$

The finite volume numerical formula for the surface integral of term 1 in equation (10) is shown below, equation (14). Clearly there is influence from all nodes on the top and bottom edges of the control volume. This is in direct contrast to the finite difference formula for $x_{\xi\xi}$ in equations (4).

$$\oint x_{\xi} \hat{n}_{\xi} d\Gamma = \frac{x_4 - 2x_5 + x_6}{2} + x_3 - 2x_0 + x_7 + \frac{x_2 - 2x_1 + x_8}{2} \quad (14)$$

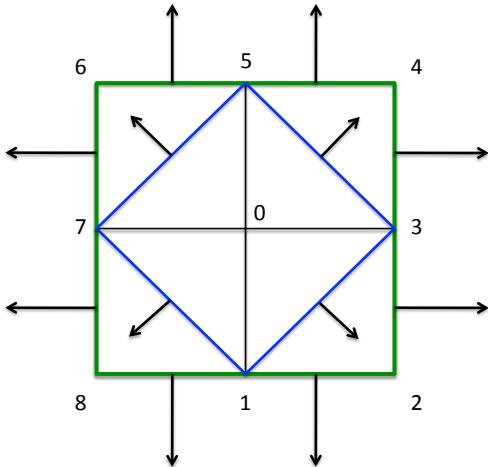


Figure 9. Control volumes for modified Finite-Volume method.

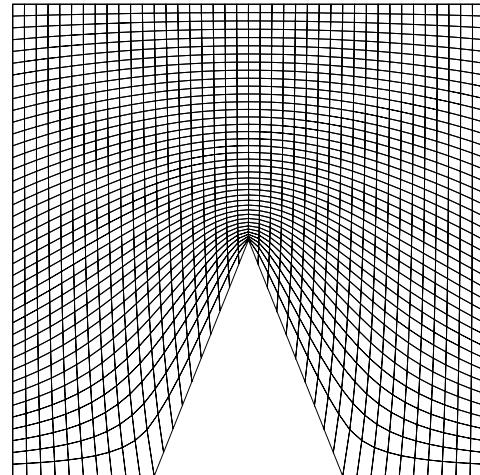


Figure 10. Winslow solution using modified control volumes.

This is a common problem associated with finite-volume solutions to diffusion type equations, such as the Laplace equation. If the structured mesh result is considered “correct” and serves as a guide then the finite volume scheme must be modified to exclude those corner nodes where appropriate. In order to replicate the structured mesh result the control volumes used with the terms involving α and γ (the first and third integral) must be replaced with the collection of triangles shown in blue in Figure 9. This is sometimes referred to as “cutting the corner” and it ignores the influence of the opposite node in the quadrilateral. The first derivatives for the central node, used to compute α , β , and γ , are also computed using this modified stencil. The finite volume result for the second integral correctly uses only the corner nodes, so the full stencil comprised of the 4 quadrilaterals is still used for the β term. The resulting smoothed mesh using this hybrid stencil is shown in Figure 10, which is identical to the structured mesh result.

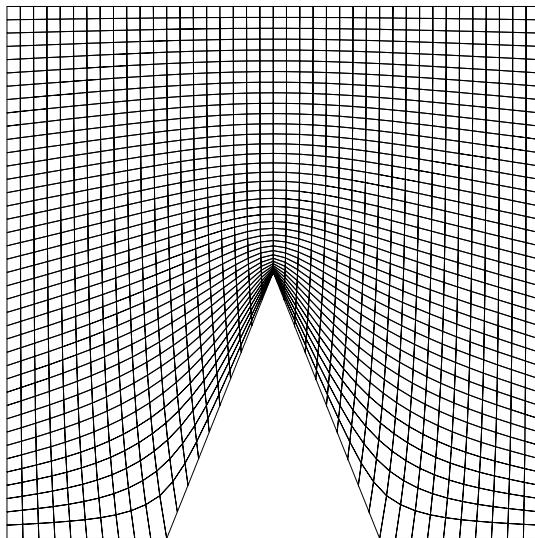


Figure 11. Unstructured Winslow solution using all triangle stencils.

If the “cut the corner” stencil (blue triangles) is used for all three integrals the result is still valid, shown in Figure 11, but different from the structured mesh result. The main difference occurs at the spike. Clearly the discretization strategy used for the β term is influencing the result. In fact, for the stencils comprised of the four blue triangles shown in Figure 9 the β term computes to exactly zero. This was verified by numerically ignoring this term and the result was identical to the mesh shown in Figure 11.

It is interesting to note that using the hybrid stencil (cut-the-corner for the α and γ terms and full stencil for the β term), but rotating the entire stencil by 45 degrees will also produce the result shown in Figure 11. Apparently the hybrid stencil is not invariant to rotation. This is not a desirable property. For nodes with

only four quadrilaterals it is easy to align the stencils with the computational coordinates. For nodes with triangles and quadrilaterals or nodes with more than 4 quadrilaterals it may be difficult to devise a strategy to align the stencil in the computational domain.

Triangular Case

Diagonalizing, in a symmetric manner, the quadrilaterals from the initial transfinite interpolation mesh for Figure 2 produces the mesh in Figure 12. Using the original finite volume scheme described above produces the result in Figure 13.

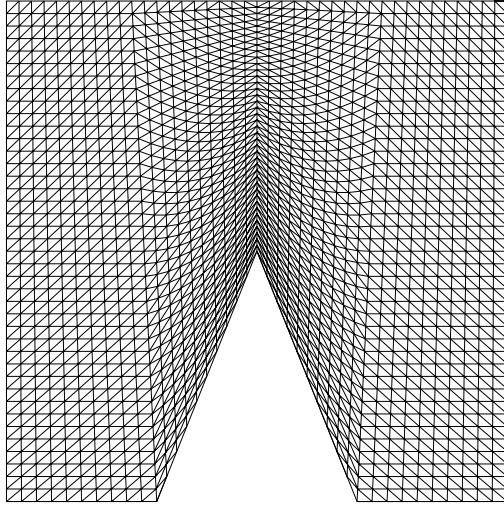


Figure 12. Quadrilateral mesh diagonalized to produce all triangles.

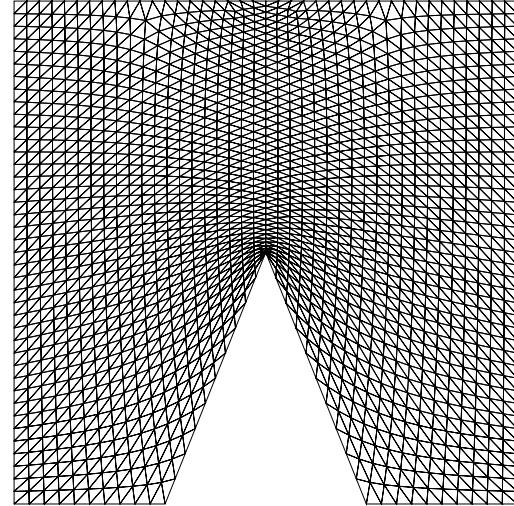


Figure 13. Winslow smoothed triangle mesh for spike.

Everything appears valid until one examines the region near the top of the spike, shown in Figure 14. The first interior node above the spike has been moved below the boundary, resulting in grid crossing. The majority of the interior nodes contain six triangles. The virtual control volume for those nodes is shown in Figure 15.

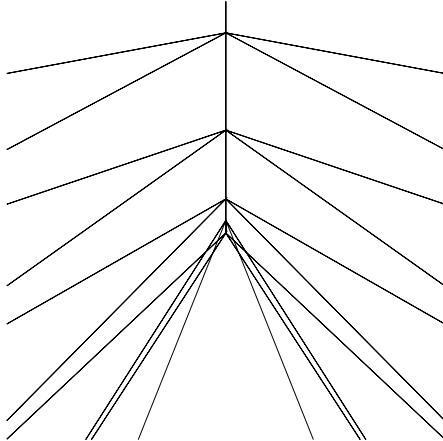


Figure 14. Magnified view of mesh near spike.

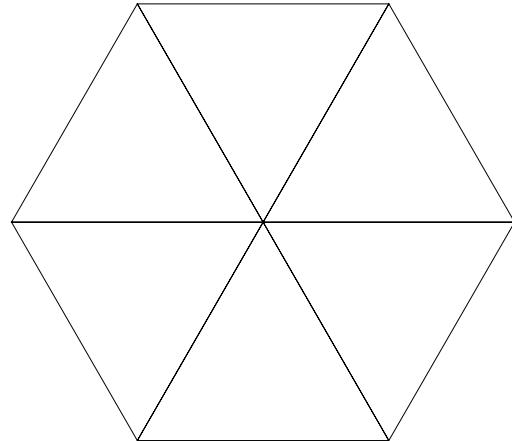


Figure 15. Virtual control volume for most of the interior nodes of the triangular spike mesh.

Rotating the stencil by 45 degrees does not change the outcome, so the original finite volume scheme appears to be invariant to rotation. All derivatives are computed using a consistent Green-Gauss derived formulation. As the stencil is rotated the gradient components are altered, but the magnitude of the derivative stays constant.

5. Augmented Stencil for Triangles

Since including the opposite corner node of quadrilaterals in the β term was helpful, it is possible to expand the stencil for triangles to bring in more information. Figure 16 shows an expanded stencil that includes the first neighboring triangle of each directly connected triangle. Also shown in bold red is a quadrilateral comprised of a directly connected triangle and the augmented neighbor. This will serve as the “quadrilateral” in the hybrid stencil described earlier. This is similar to Knupp’s use of the opposite node of the quadrilateral when the number of surrounding neighbor nodes was insufficient to determine the required finite-difference derivatives. The α and γ terms will use only the directly connected triangles and the β term will use the augmented quadrilaterals. For nodes containing a mix of triangles and quadrilaterals in the original stencil, only the triangles will be augmented with neighboring triangles.

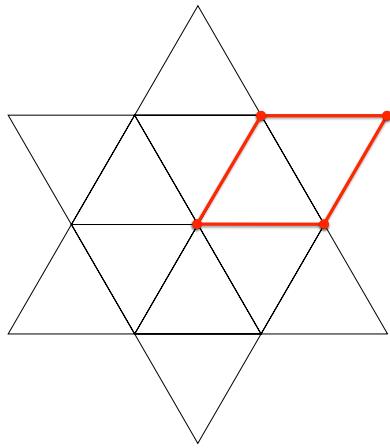


Figure 16. Augmented stencil for nodes with triangular elements.

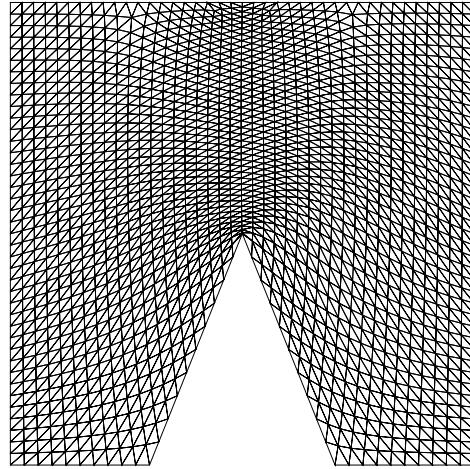


Figure 17. Triangle mesh with augmented stencils.

The smoothed mesh using the augmented triangle stencils is shown in Figure 17. This is a valid mesh with no grid crossing. However, it is not symmetric from left to right, as expected. The cause is believed to be the augmented stencils near the boundary. One of the directly connected triangles of these nodes cannot be augmented with a neighbor because a neighbor does not exist. Instead the stencil will be unbalanced and the “first” node of the stencil may differ from node to node, resulting in arbitrary alignment of the unbalanced-augmented stencil with respect to the computational coordinate directions. Rotating all the stencils by 45 degrees results in a valid mesh, however the asymmetry still exists but is flipped from right to left. If aligning node “1” with the physical edge direction is used it requires constant adjustments to the stencil and still does not produce a completely symmetric mesh.

Augmenting the stencils with adjacent triangles has provided valid meshes without grid crossing for the spike case. The sensitivity to the rotation of the control volume is bothersome. This is not a desirable trait and further research is required to fully understand the causes and devise an appropriate modification to the discretization strategy. The goal should be to devise a strategy that accurately models all three terms of the equations and is invariant to rotation in computational space. An important point to be made, though, is the augmented stencil approach does provide meshes without grid crossing. And this method can be turned on or off as needed. Even without the augmented stencil the method seems to work extremely well for most cases. The exceptions are cases with extreme sharp edges.

If the computational mesh for the structured case were rotated through any angle the original finite-difference formulae would no longer be valid. Taylor series expansions could be re-derived

for the new orientation or a least squares method could be used to compute the derivatives. But modified difference formulae have already been used for a structured mesh. The modified scheme described in the previous section was carefully constructed for the all-quadrilateral mesh in order to reproduce the structured mesh result. All stencils were comprised of exactly four quadrilaterals that were carefully aligned with the ξ and η directions. The finite-volume formulation is equivalent to performing a Taylor series expansion about the central node if the higher order terms are ignored. When those stencils were rotated 45 degrees the results were different, but still valid. This demonstrates that the discretization strategy for structured meshes is also not invariant to rotation.

6. Results

All cases shown below are computed without any manipulation of the control volumes to account for viscous-type spacing or for solution based adaptation. Some cases are computed using the original finite volume approach outlined in Section 4. For those cases, no augmented stencils were used for the triangles and the “cut-the-corner” approach was used for quadrilaterals. Thus the scheme is invariant to scaling, translation and rotation of the virtual control volumes. Other cases used the augmented stencils to obtain valid meshes where sharp corners exist.

Rotating Circle

The first case is a simple geometry consisting of an outer circular boundary and an inner circular boundary. The original mesh was created using Pointwise [10]. The mesh contains 2,415 nodes and 4,710 triangles. The inner and outer boundaries are comprised of two sections with 31 equally spaced points each for a total of 61 point for each circle. Views of the original mesh can be seen in Figure 18 and Figure 20. The original finite volume scheme without augmented stencils was used for this case. The Winslow smoothed mesh is shown in Figure 19 and Figure 21. The differences between the meshes are slightly more evident in the magnified views. The original mesh is certainly valid and contains high quality triangles. Pointwise uses a Delaunay method to triangulate the points and the distribution is very smooth. The Winslow smoothed mesh does not attempt to enforce the Delaunay property. Winslow smoothing attempts to make the physical mesh match the “ideal” element shapes in the virtual control volumes. No attempt is made to declare one mesh better or worse than the other. This merely shows that Winslow smoothing can be used with existing tessellations where the boundaries are held fixed. In cases where the initial distribution of points was less smooth the Winslow method will show more variation from the Delaunay-based result.

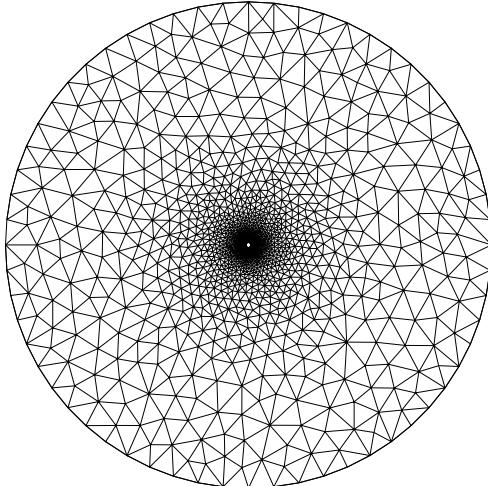


Figure 18. Original mesh for circle case.

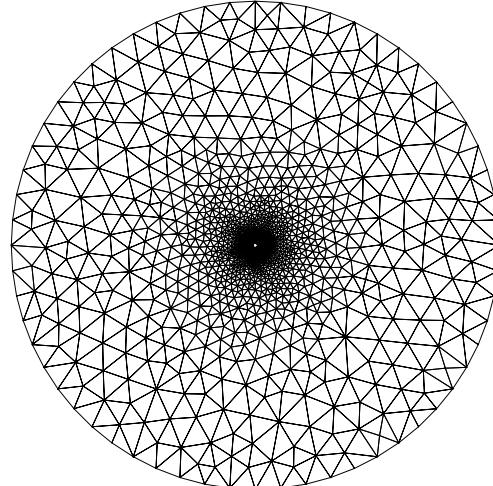


Figure 19. Winslow smoothed mesh.

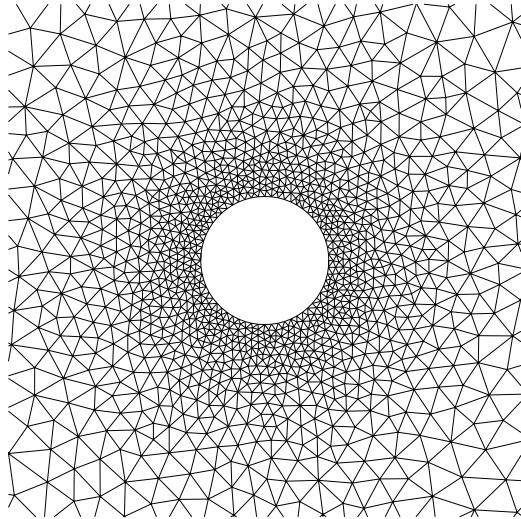


Figure 20. Magnified view of original mesh near inner boundary.

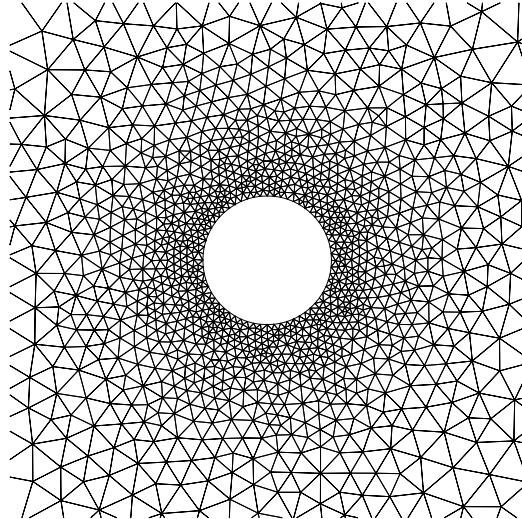


Figure 21. Winslow smoothed mesh near inner boundary.

One of the major benefits Winslow smoothing provides is the ability to perform mesh movement. So for the next demonstration the inner circle is rotated 90 degrees counterclockwise. The inner boundary points are held fixed at the new location and the outer boundary points remain in their original position. The global mesh is shown in Figure 22. The twisting of the interior mesh is evident in the global view, but is less evident in the magnified view, Figure 23. The shearing of the mesh is minimal for this rotation angle. Additional rotation of the inner boundary is certainly possible and would increase the shearing, unless the points on the boundaries were allowed to float.

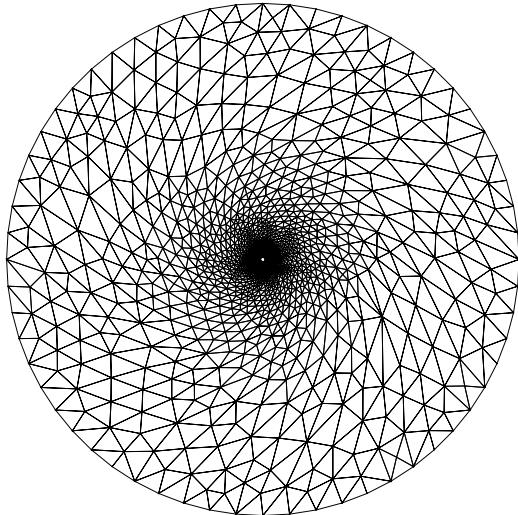


Figure 22. Global view of rotated circle mesh.

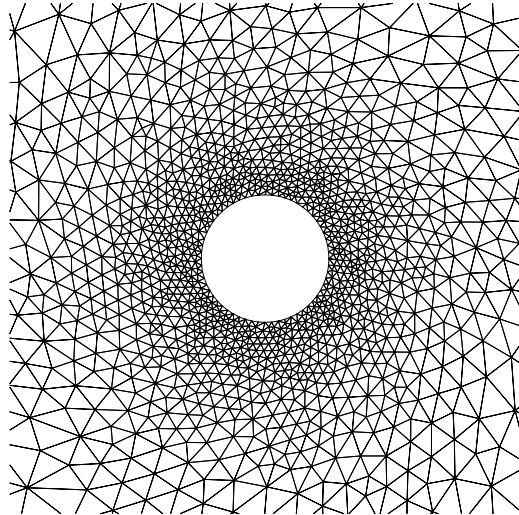


Figure 23. Magnified view of inner boundary for rotated circle mesh.

The next modification to this rotated mesh is shown in Figure 24. Ten layers of quadrilateral elements were added to the mesh connectivity at the inner boundary. The smoothed quadrilateral elements show evidence of the shearing exhibited by the triangular elements in Figure 22. The basic method was employed without any attempt at controlling mesh spacing or grid line angularity. Winslow is attempting to make the physical mesh match the virtual control volume shapes, which are nearly ideal.

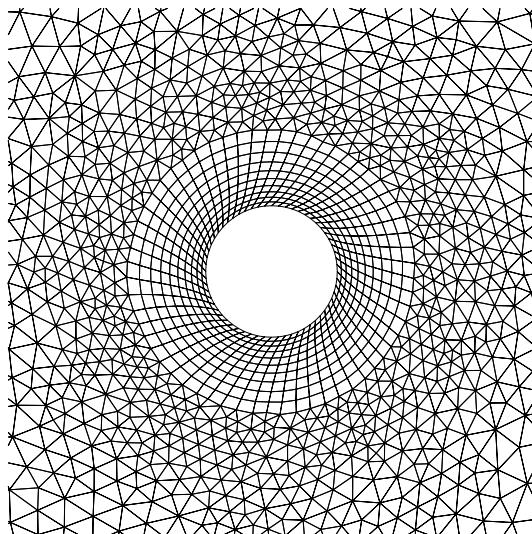


Figure 24. Quadrilateral layers added to rotated mesh.

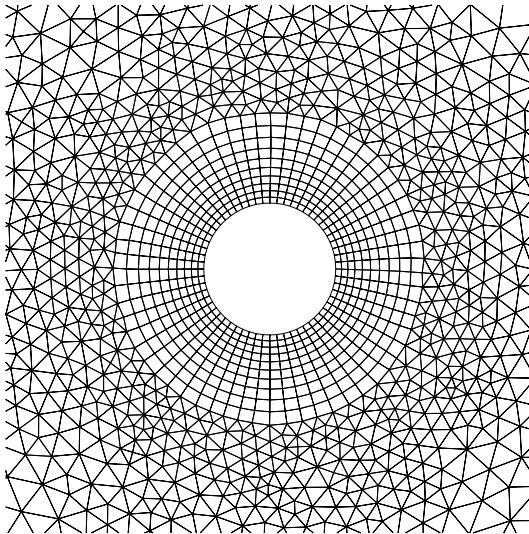


Figure 25. Hybrid mesh rotated back to original position.

When the inner boundary is rotated back to the original position the shearing of the quadrilateral elements disappears, shown in Figure 25. The sequence of smoothing, rotating and adding layers of quadrilateral elements could have been performed in any order. The same mesh would be produced had the rotation never taken place. And the creation of the quadrilateral elements does not require one to distribute the points in the normal direction away from the boundary. The physical points were actually initialized to be coincident with the original boundary points. Winslow smoothing produced the final point distribution, attempting to make the quadrilateral elements perfect squares.

NACA0012 Airfoil

The second case is the famous NACA0012 airfoil. The original triangular mesh, created in Pointwise, contains 4,392 points and 8,424 triangles. The outer boundary is in two sections with 61 points per section, equally spaced. The airfoil boundary is also defined in two sections, upper and lower, with 121 points per section. The points are clustered toward the leading and trailing edges in a symmetric manner from top to bottom. The augmented stencils were used to smooth this triangular mesh. The global mesh after Winslow smoothing is shown in Figure 26. A magnified view of the region near the airfoil is shown in Figure 27.

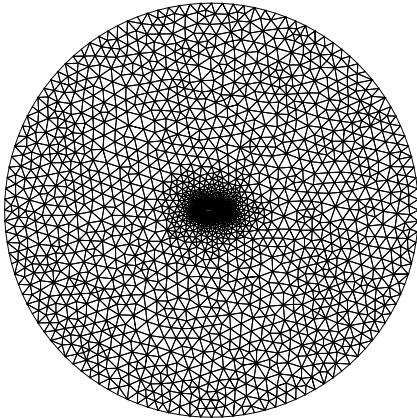


Figure 26. Global smoothed mesh for NACA0012 airfoil.

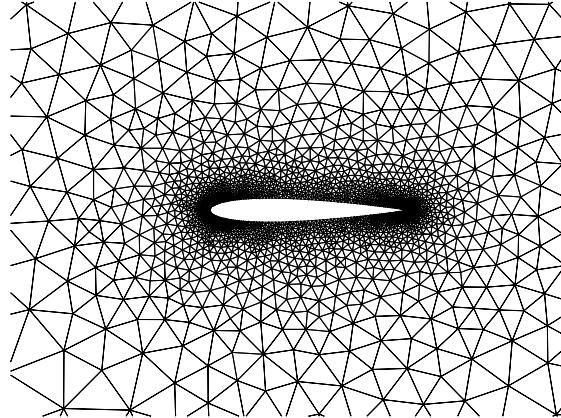


Figure 27. Magnified view of triangular mesh in region near airfoil.

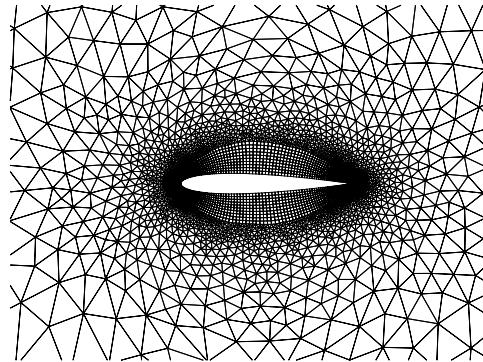


Figure 28. Smoothed mesh with 10 quadrilateral layers added near airfoil.

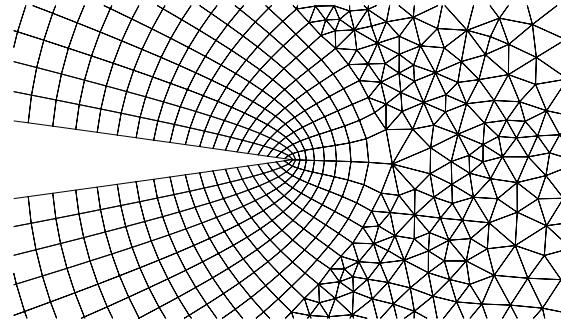


Figure 29. Magnified view of hybrid mesh near trailing edge of airfoil.

Ten quadrilateral element layers were added to the connectivity near the airfoil boundary. The β term is discretized with the full quadrilateral and augmented triangle formulations. The smoothed mesh is shown in Figure 28. At first it might seem like this is not the expected outcome. But in fact it is because no grid spacing control is being applied. The virtual control volumes for the quadrilateral elements are perfect squares and the physical mesh is attempting to mimic that shape for the quadrilaterals. So where there is clustering at the leading and trailing edge the local quadrilaterals (squares) are smaller. In the middle of the airfoil, shown in Figure 28, the point spacing along the surface is larger and the neighboring quadrilaterals are nearly perfect squares there, as well. A view of the trailing edge region is shown in Figure 29.

Finally, this hybrid mesh is translated one unit in the positive X direction and one unit in the positive Y and rotated 90 degrees. The smoothed mesh is shown in Figure 30. There was some difficulty encountered converging the solution due to the skewing of the triangles in the far field. The augmented triangles were turned off and the scheme was able to converge. The full

quadrilateral formula was used for the β term in order to accurately handle the sharp trailing edge. Figure 31 shows the magnified view of the trailing edge region. There are no negative areas or skewed elements in the mesh.

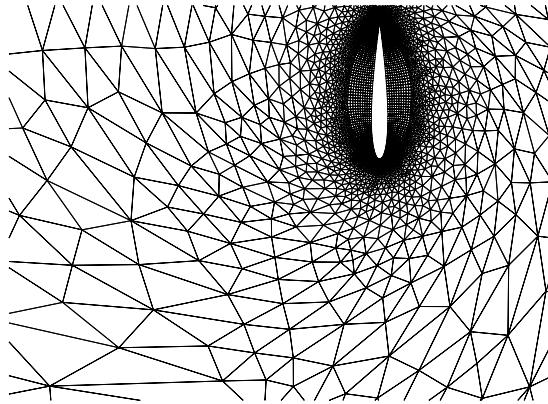


Figure 30. Airfoil translated in X and Y and rotated 90 degrees.

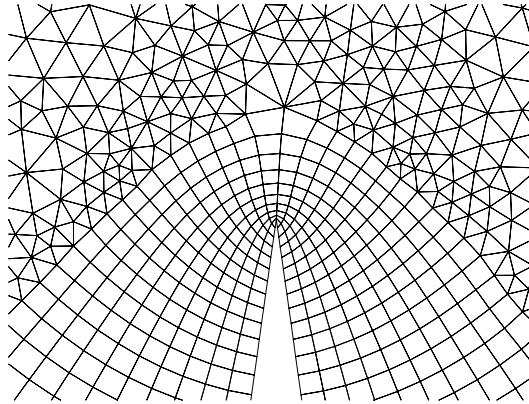


Figure 31. Magnified view of mesh near the trailing edge.

30P30N Multi-Element Airfoil

The final case is a 3-element airfoil section known as 30P30N. The triangular mesh was created using Pointwise and it consisted of 22,165 points and 42,743 triangles. The outer boundary was defined in four sections with 21 points per section, equally spaced. The slat section was divided into two sections with the sharp points of the section providing the demarcation. One section contained 200 points while the other contained 150 points. The main element was defined in three sections and had three sharp points that defined the sharp trailing edge and the cove region. The largest section contained 600 points and the other two contained 100 points and 50 points. The flap was divided into 3 sections; an upper section with 200 points, a lower section with 200 points and blunt trailing edge section with 7 points. The distributions on each section were clustered toward the sharp points. This initial mesh was smoothed using the augmented triangle stencils. Figure 32 shows the smoothed mesh with the three elements. The slat region is shown in more detail in Figure 33 and the flap is detailed in Figure 34. The gap region between the main airfoil element and the flap is magnified in Figure 35.

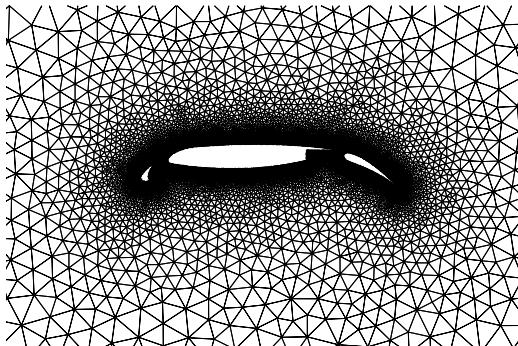


Figure 32. Three-element airfoil mesh.

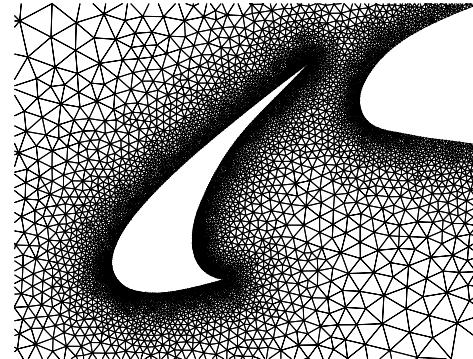


Figure 33. Magnified view of leading edge slat.

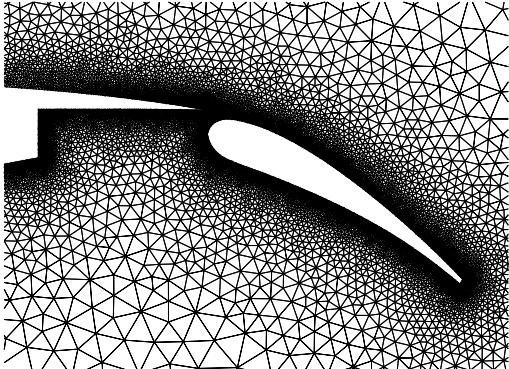


Figure 34. Magnified view of trailing edge flap.

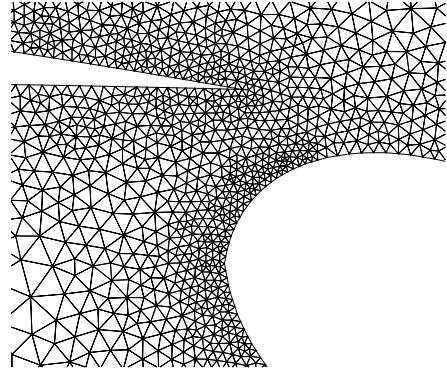


Figure 35. View of gap between main element and trailing edge flap.

The flap was rotated an additional 20 degrees downward and the slat an additional 10 degrees downward. The smoothed meshes are shown in Figure 36 and Figure 37, compared to the meshes in Figure 33 and Figure 34.

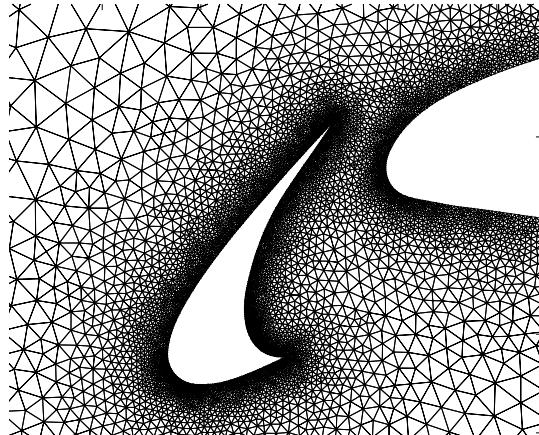


Figure 36. Magnified view of slat rotated 10 degrees downward.

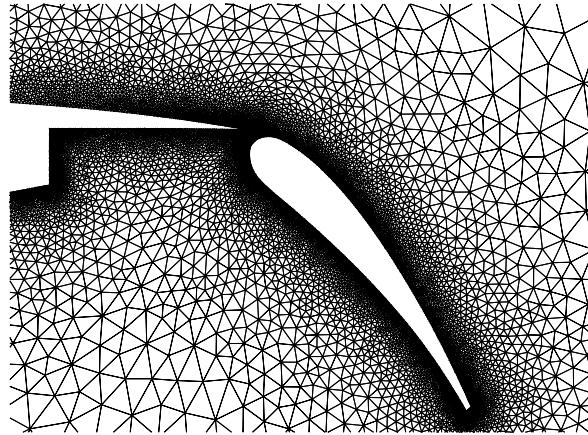


Figure 37. Magnified view of flap rotated 20 degrees downward.

Some excessive stretching was encountered at the sharp trailing edge of the main airfoil section, Figure 38. The augmented stencils were turned off to produce the mesh in Figure 39, which shows slightly less skewing.

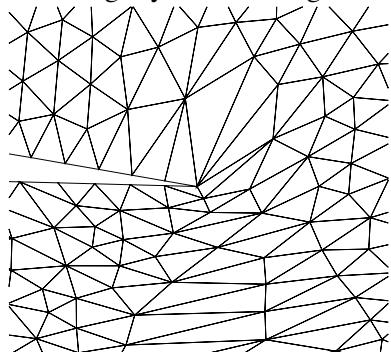


Figure 38. Main element trailing edge using augmented triangle stencils.

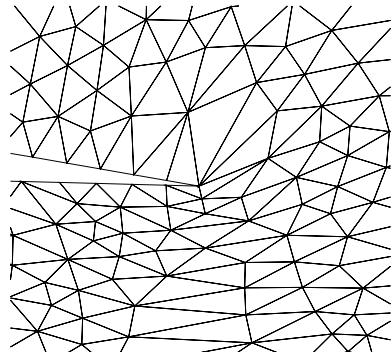


Figure 39. Main element trailing edge without using augmented triangle stencils.

With the flap rotating the elements in the gap are stretched. The elliptic smoothing is attempting to restore the elements to the ideal shapes provided in the virtual control volumes. The

augmented stencil and the β term carried the effect of the skewing around the trailing edge and on to the topside of the airfoil section. Turning off the augmented stencil lessened the effect.

Five quadrilateral element layers were added to the original, un-rotated triangular mesh from Figure 32. The smoothed mesh, computed without augmented triangle stencils and the “cut-the-corner” approach for the quadrilaterals, is shown in Figure 40. A magnified view of the gap between the slat and main element is shown in Figure 41. The mesh near the sharp tip of the slat is smooth and contains no grid crossing.

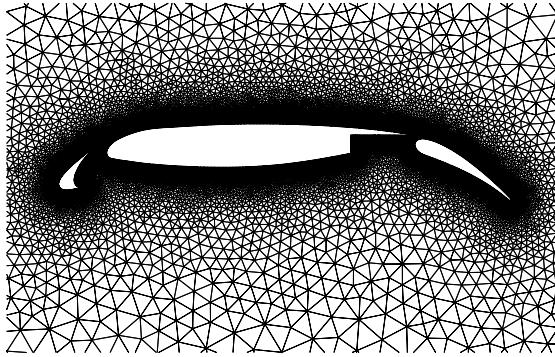


Figure 40. Three-element airfoil with five quadrilateral layers added.

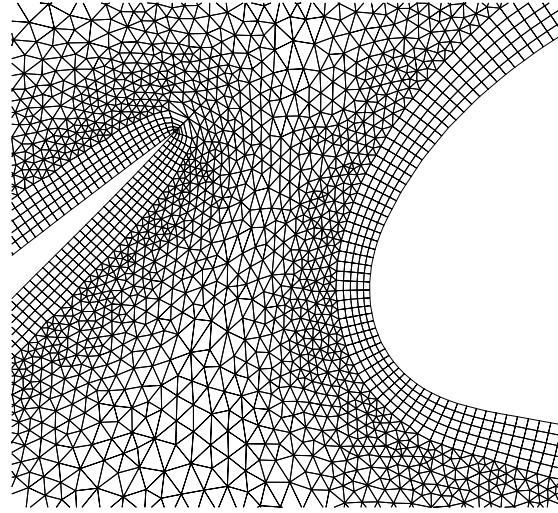


Figure 41. Magnified view of hybrid mesh in region between slat and main element.

The mesh for the backward-facing step of the cove region is shown in Figure 42. The pulling of the mesh out of the corner is typical for Winslow smoothing with structured meshes. Again, no control of grid spacing was attempted, so this is an expected result. The mesh in the gap between the main element and the flap, shown in Figure 43, shows some skewing at the sharp trailing edge. The quadrilaterals at the sharp corner are skewed, with a slightly inverted corner, but the areas are still positive. Grid spacing control is expected to lessen the skewing in this region since there will be more room between the airfoil sections for the isotropic (triangle) portion of the mesh.

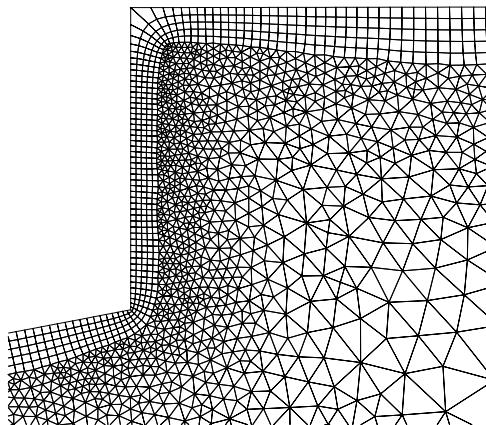


Figure 42. Hybrid mesh in cove region of main airfoil section.

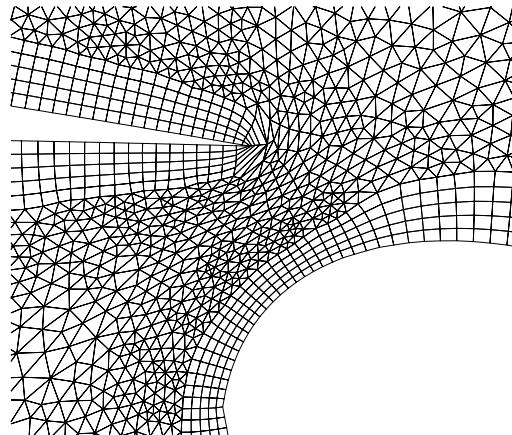


Figure 43. Hybrid mesh in gap between main element and flap.

Conclusions

The development of a two-dimensional unstructured elliptic smoothing method has been described. The Winslow equations are discretized using a finite volume approach. Individual control volumes for each node are constructed in a virtual space where the element shapes are nearly ideal. The baseline discretization uses Green-Gauss theorem to formulate gradients over an element or a collection of elements for a node. As coded this formulation would not reproduce the structured mesh elliptic smoothing behavior for a spike problem. A modification to the scheme for quadrilateral elements was successful at producing structured mesh results for the spike problem. The triangulated version of the spike mesh was smoothed resulting in one node near the spike outside the domain. A second modification to the finite-volume method incorporated additional triangles in the virtual control volume stencil that could serve as a basis for incorporating the same discretization used for quadrilateral elements, resulting in a valid mesh for the spike problem.

Results were shown for three additional cases. For each case basic smoothing was demonstrated as well as boundary motion. The resulting meshes were smooth and contained no invalid elements. The final demonstration for each case was to add layers of quadrilateral elements to selected boundaries. The points were distributed via the Winslow smoothing. No mesh spacing control was attempted and the resulting meshes were smooth, without grid crossing or invalid elements. The three-dimensional extension has been completed and will be published in the near future.

References

- 1 Winslow, Alan M., Numerical Solution of the Quasilinear Poisson Equations in a Nonuniform Triangle Mesh, *Journal of Computational Physics* 2, 149-172, 1967
- 2 Thompson, J. F., Thames, F. C., and Mastin, C. W., "Boundary-Fitted Curvilinear Coordinate Systems for Solution of Partial Differential Equations on Fields Containing Any Number of Arbitrary Two-Dimensional bodies," NASA CR-2729, July 1977.
- 3 Thomas, P. D., and Middlecoff, J. F., "Direct Control of the Grid Point Distribution in Meshes Generated by Elliptic Equations," *AIAA Journal*, Vol. 18, 1979, pp. 652-656.
- 4 Sorenson, R. L., "A Computer Program to Generate Two-Dimensional Grids About Airfoils and Other Shapes by Use of Poisson's Equations," NASA TM-81198, 1980.
- 5 Knupp, Patrick M., "Winslow Smoothing on Two-Dimensional Unstructured Meshes," *Engr. With Computers*, 15:263-268, 1999.
- 6 Tipton, R. "Grid Optimization by Equipotential Relaxation," manuscript, 26pp, Lawrence Livermore National Laboratory, November 28, 1990.
- 7 Karman, S. L. Jr., Anderson, W. K., and Sahasrabudhe, M., "Mesh Generation Using Unstructured Computational Meshes and Elliptic Partial Differential Equation Smoothing," *AIAA Journal*, Vol. 44, Number 6, 2006, pp. 1277-1286.
- 8 Sahasrabudhe, M., "Unstructured Mesh Generation and Manipulation Based on Elliptic Smoothing and Optimization," Ph.D. Dissertation, University of Tennessee at Chattanooga, August 2008.
- 9 Karman, S. L. Jr. and Sahasrabudhe, M. S., "Unstructured Elliptic Smoothing Revisited," AIAA-2009-1362, Orlando, FL, January 2009.
- 10 <http://www.pointwise.com/>