

# Adaptive Boundary Layer Meshing for Viscous Flow Simulations

Onkar Sahni<sup>1</sup>, Kenneth E. Jansen<sup>1</sup>, Mark S. Shephard<sup>1</sup>, Charles A. Taylor<sup>2</sup>, Mark W. Beall<sup>3</sup>

<sup>1</sup> Scientific Computation Research Center, Rensselaer Polytechnic Institute, Troy, NY, 12180, USA  
e-mail: [osahni@scorec.rpi.edu](mailto:osahni@scorec.rpi.edu), [kjansen@scorec.rpi.edu](mailto:kjansen@scorec.rpi.edu), [shephard@scorec.rpi.edu](mailto:shephard@scorec.rpi.edu)

<sup>2</sup> Stanford University, E350 Clark Center, 318 Campus Drive, Stanford, CA, 94305, USA  
e-mail: [taylorca@stanford.edu](mailto:taylorca@stanford.edu)

<sup>3</sup> Simmetrix Inc., Clifton Park, NY, 12065, USA  
e-mail: [mbeall@simmetrix.com](mailto:mbeall@simmetrix.com)

Received: date / Revised version: date

**Abstract** A procedure for anisotropic mesh adaptation accounting for mixed element types and boundary layer meshes is presented. The method allows to automatically construct meshes on domains of interest to accurately and efficiently compute key flow quantities, especially near wall quantities like wall shear stress. The new adaptive approach uses local mesh modification procedures in a manner that maintains layered and graded elements near the walls, which are popularly known as boundary layer or semi-structured meshes, with highly anisotropic elements of mixed topologies. The technique developed is well suited for viscous flow applications where exact knowledge of the mesh resolution over the computational domain required to accurately resolve flow features of interest is unknown *a priori*. We apply the method to two type of problem cases; the first type, which lies in the field of hemodynamics, involves pulsatile flow in blood vessels including a porcine aorta case with a stenosis bypassed by a graft whereas the other involves high-speed flow through a double throat nozzle encountered in the field of aerodynamics.

**Key words** Boundary layer meshes, Computational fluid dynamics, Mesh adaptation, Mixed/Hybrid meshes, Viscous flow simulations

---

## 1 Introduction

The reliability and accuracy of a fluid dynamics simulation is a strong function of the mesh resolution and quality. For problems of practical interest increasing the mesh resolution in a uniform global fashion to the level required for acceptable accuracy would introduce excessive demands on the computational resources. In order to expedite the convergence of the numerical solution mesh

resolution needs to be determined in a local fashion that takes into account the spatial distribution of the solution and error associated with its numerical approximation. Moreover many physical problems of interest, especially in the field of fluid mechanics, involve directional solution features, for example, boundary layers which form near the walls in viscous flows or shock waves in high speed flows. Such solution features are most effectively resolved using mesh elements which are oriented in a certain manner with significant degree of anisotropy, i.e., different resolution in different local directions.

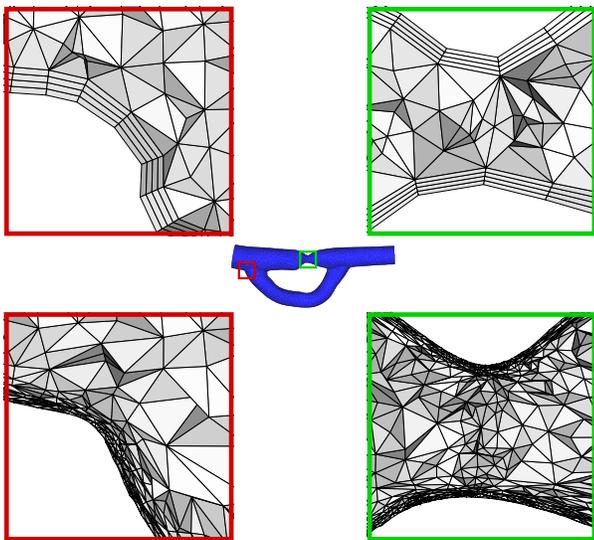
In case of viscous flow simulations, better results for key quantities of interest in regions of boundary layers, like wall shear, can be obtained with the help of semi-structured boundary layer mesh [1]. Pre-defined meshes with layered elements near the walls have been used in the field of fluid mechanics for some time, especially for cases with prior knowledge of boundary layers (see, e.g., [2–8] and literature cited therein). A common method to construct these meshes, referred as *advancing layers method*, inflates the unstructured surface mesh on no-slip walls, where boundary layers form, into the volume along the local surface normals as layers upto specified distance and fills rest of the domain with unstructured tetrahedral elements. Such a mesh possesses structure in the direction normal to the walls due to creation of extruded volume elements over unstructured surface mesh as layers. Care is taken to produce elements of acceptable shape at sharp corners with high folding angle and to prevent element overlap in regions of tight corners; and also to ensure a smooth transition between the layered and unstructured tetrahedral part of the mesh [2–8].

Favorable attributes of such meshes are high-aspect ratio, orthogonal and graded elements at the walls. The need to specify surface mesh resolution and the distribution of points along the normals to the surface [4] (e.g., spacing of first point off the wall, total thickness of the layers, growth/stretch rate between layer thicknesses, number of layers) make their use limited as it involves a

great deal of user intervention. The deficiency involved is two folds for problem cases of interest; the process can be tedious and requires expertise in terms of knowledge about the solution that is often not available. For realistic problems the mesh resolution necessary to effectively compute the numerical solution is unknown *a priori*. An effective approach to overcome this difficulty is to apply an adaptive procedure where the distribution of the spatial discretization errors introduced are estimated and controlled by modifying the mesh resolution. Hence, the automatic and adaptive construction of properly configured anisotropic meshes is central to the ability to competently perform numerical simulations.

Mesh metric field based anisotropic adaptivity have been developed for real geometries [9–17]. The anisotropic adaptive procedures based on mesh metric field to date have made no effort to create and/or preserve an appropriate semi-structured boundary layer mesh near the walls. Although this issue has only a minor influence on the overall flow field, it has a much larger impact upon local quantities of interest like wall shear stress [1].

Fig. 1 presents the two meshing techniques; the top images show magnified views of mesh faces of a pre-defined boundary layer mesh, with four layers of equal thickness, cut by a plane over a blood vessel whereas the bottom ones show mesh faces for an fully unstructured, anisotropically adapted mesh. In this article, we present a new approach for mesh adaptation that incorporates the capacity to produce anisotropic meshes suitable for viscous flows with adequate mesh resolution in an adaptive procedure to attain better results for relevant flow quantities near the walls. Key to this approach is adapting the mesh such that semi-structured nature of the boundary layer mesh is maintained.



**Fig. 1** Magnified views of clipped mesh faces through a pre-defined boundary layer mesh (top) and fully unstructured, anisotropically adapted mesh (bottom) over a by-passed porcine aorta with a stenosis.

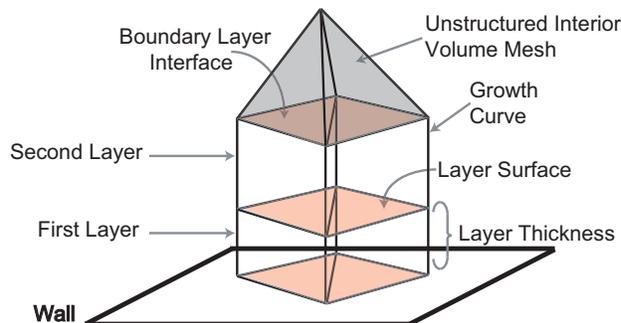
The organization of the paper is as follows. Section 2 provides the overall boundary layer mesh adaptation procedure. Section 3 and 4 detail the two important step of layer surface adaptation and thickness adjustment. Section 5 demonstrates the capability of the boundary layer adaptive procedure by applying it to problem cases involving cardiovascular and aerodynamic flows. Although some figures illustrate 2D examples the procedures presented in this work are fully developed for 3D cases.

## 2 Boundary layer mesh adaptation

### 2.1 Overview

The boundary layer mesh adaptation process is initiated on a mesh that already carries a pre-defined mesh with layered elements on no-slip walls. Subsequent mesh adaptation steps preserve the layer structure normal to the walls while at the same time attain desired element sizes in different directions as indicated by the *a posteriori* mesh size field information.

The inherent structure in the boundary layer mesh allows us to decompose them as a product of a layer surface (2D) and thickness (1D) mesh. The mesh composed of triangles located at the top of each layer will be referred to as *layer surface*, see Fig. 2, while the lines orthogonal to the wall composed of edges are called

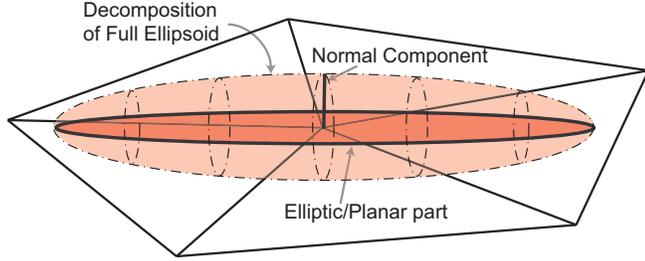


**Fig. 2** Conceptual decomposition of a boundary layer mesh.

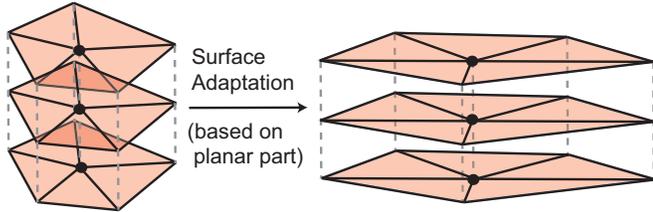
*growth curves*. The edges that belong to layer surfaces are referred to as *layer edges* and ones that reside on growth curves are called *growth edges*. The number of total nodes (or edges) on growth curves determine its *level*. The nodes on walls from which growth curves originate are referred to as *originating nodes*. Note that the top most layer of the stack of boundary layer elements shares an interface with the unstructured volume mesh, where the interior tetrahedral elements are referred to as *interface elements*.

To preserve the structure along the normals, mesh adaptation for layered part of the mesh is divided into two steps: surface adaptation and thickness adjustment. The two step mesh modification procedure is governed

by the mesh metric tensors defined by directional error indicators. In this regard, the mesh metric tensors, represented as ellipsoids, at any node of the boundary layer mesh can also be decomposed into a planar part, which dictates surface adaptation, and a normal component that controls thickness adjustment, see Fig. 3. Fig. 4 depicts surface adaptation, where mesh resolution

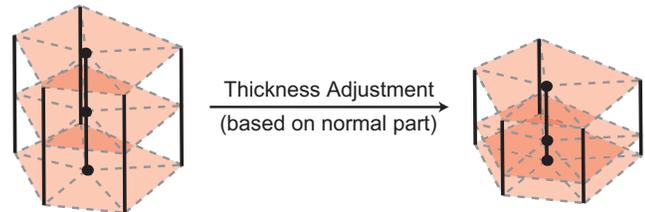


**Fig. 3** Decomposition of a mesh metric tensor at a boundary layer node.



**Fig. 4** Schematic for surface adaptation for boundary layer mesh (edges with dashed pattern are inactive in this step).

on layer surfaces is decreased (coarsened) in stream-wise direction and increased (refinement) in span-wise direction, on the other hand Fig. 5 illustrates thickness adjustment, where layer thicknesses are adjusted without modifying layer surfaces.



**Fig. 5** Schematic for thickness adjustment for boundary layer mesh (edges with dashed pattern are inactive in this step).

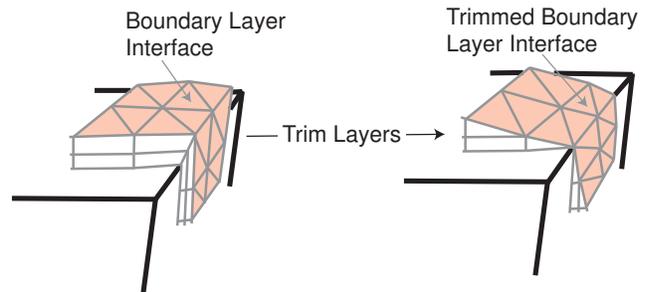
The local mesh modification operations of edge split, collapse and swap are utilized to perform the surface mesh adaptation while node movement is applied to adjust the layer thicknesses. In case of surface mesh modification any operation is carried out such that it is propagated through the stack of boundary layer elements and

affects all the layer surfaces, including the interface elements, in the same way. In case of thickness adjustment the desired normal mesh resolution for each layer, and hence total thickness, is achieved by node movement (even for the top most node of the stack) while maintaining the number and topology of the layers.

## 2.2 Geometric complexities

Care is required for geometries of interest that contain sharp and tight corners to produce elements of acceptable shape and to prevent element overlap, respectively. In the current approach, such cases are handled by terminating the stack of boundary layer elements through smoothly trimming them at the corners. This prevents the interaction of layered elements over adjacent walls that may produce poorly shaped or overlapping elements at geometric edges/corners with steep folding angle.

*Transition elements* are introduced on top of the staircase formed by trimming boundary layer elements to avoid the exposure of vertical boundary layer mesh faces to the unstructured tetrahedral interior volume mesh, see Fig. 6 for a sharp corner case with high folding angle. The topology of transition elements depends on the

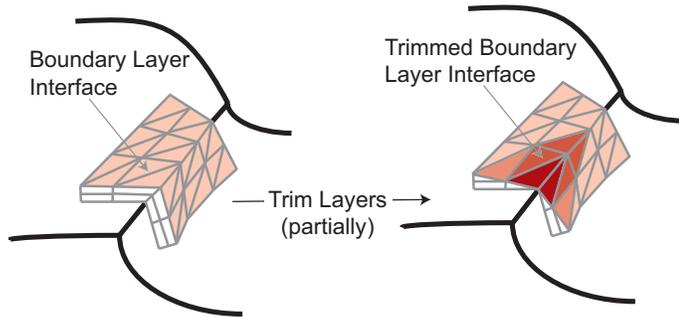


**Fig. 6** Trimming of boundary layer mesh at a sharp corner with high folding angle.

level of growth curves emanating from nodes of base mesh face on the wall; and is either pyramid or tetrahedron for triangular surface mesh (see, Fig. 9 in [4]). The edges connecting nodes, of adjacent growth curves, on different level are referred to as *transition edges*.

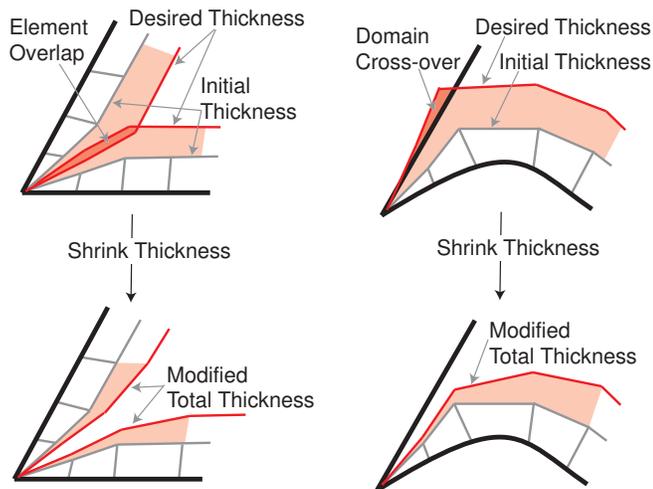
The trimmed boundary layer mesh interface allows for smooth transition to unstructured tetrahedral interior mesh at the cost of losing structure in the normal direction locally near the corners, which is acceptable as the variation in solution is typically isotropic around the corners, atleast in the perpendicular plane containing it, like in case of bifurcations [1]. In cases with variable folding angle along geometric edges surrounding non-planar surfaces, boundary layer mesh is trimmed only over portion(s) where the folding angle is high, see Fig. 7.

In geometries involving tight corners it is required to ensure that elements do not cross-over each other and/or bounding geometry during thickness adjustment. These



**Fig. 7** Partial trimming of boundary layer mesh at a corner with variable folding angle.

situations arise when the desired total thickness is high relative to the local curvature and/or due to acuteness of the angle between adjacent surfaces as depicted in Fig. 8 for 2D cases. In the current approach, the growth of the layers is limited by shrinking the total thickness to avoid overlapping elements and/or domain cross-overs.



**Fig. 8** Shrinking of desired total thickness to avoid element overlap (left) and domain cross-over (right) near tight corners for 2D cases.

### 2.3 Overall algorithm

Before going into the details for each of the two steps, which are discussed in sections 3 and 4, the overall algorithm for the current approach is presented. Provided with a computed solution over a pre-defined mesh with boundary layers the current adaptive procedure first computes the directional error indicators, see [1] and literature cited therein. Error indicators are then converted into the mesh metric field with appropriate adjustment through metric decomposition over layered part of the mesh [1]. The mesh metric field then dictates local mesh modification operations which maintain layered elements in the mesh near the walls and handles unstructured

tetrahedral interior volume mesh as discussed in previous anisotropic adaptive method [13].

For overall mesh modification, the idea is to adopt the general steps of the algorithm presented in [13] for fully unstructured anisotropic meshes (see, Fig. 6 therein). In the current algorithm, the mesh modification procedure developed for layered part of the mesh is combined with one for unstructured tetrahedral part of the mesh in the interior of the domain. The steps are carried out such that the integration is appropriately done to obtain a smooth transition at the interface. Algorithm 1 presents the pseudo code for overall adaptive procedure.

---

#### Algorithm 1 Pseudo code for overall adaptive procedure

---

- 1: compute directional error indicators
  - 2:
  - 3: construct mesh metric field
  - 4: apply metric decomposition over layered part of mesh
  - 5:
  - 6: coarsen short layer edges in metric space by collapse
  - 7: coarsen short interior edges in metric space by collapse
  - 8:
  - 9: adjust thickness for each growth curve by node movement
  - 10: assign:  $L_{ref} = \alpha L_{max}$  {where,  $\alpha$  is a given constant and  $L_{max}$  is current max. mesh edge length in metric space}
  - 11: **while** tag edges with lengths in metric space above  $L_{ref}$  **do**
  - 12:   split layer edges tagged in this iteration
  - 13:   project new nodes at curved walls onto solid surface
  - 14:   process short layer edges created in this step
  - 15:   try eliminating poorly shaped layer faces by edge swaps
  - 16:
  - 17:   split interior edges tagged in this iteration
  - 18:   project new nodes at curved bdy. onto surface [14]
  - 19:   process short interior edges created in this step
  - 20:   try to eliminate sliver interior regions by shape control
  - 21:   update  $L_{max}$  and  $L_{ref}$
  - 22:   **if**  $L_{max}$  below upper bound OR max. iterations done **then**
  - 23:     break the loop
  - 24:   **end if**
  - 25: **end while**
- 

### 2.4 Desirable features

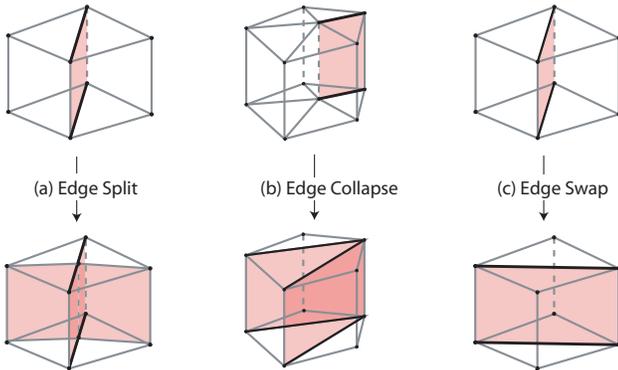
The idea to work with stack of boundary layer elements has been previously utilized, see [18,19]. The current approach generalizes these processes and includes:

1. Modification of the total thickness of the layers, i.e., including node movement for the top most node of the stack.
2. Gradation of each layer thickness based on computed size field information instead of using a constant growth rate or stretching factor between layers.

3. Dealing with mesh metric field appropriately through metric decomposition in layered part of the mesh.
4. Creation of anisotropic elements for layer surface and interior volume meshes along with smooth transition at the interface between layered and highly anisotropic unstructured tetrahedral elements.
5. Coarsening of meshes past initial one, i.e., coarsening process is not limited to undoing current refinements.
6. Working with significantly coarse initial meshes, especially in the normal direction.
7. Application of layer edge swap operations in order to improve element shape and remove the signature of initial meshes.
8. Extension to geometries of interest that contain sharp and tight corners.

### 3 Layer surface mesh adaptation

Layer surface mesh (2D) adaptation is performed by using three basic local mesh modification operations of edge split, collapse and swap, typically used for 2D meshes, see Fig. 9. Layer edges that belong to layer surfaces take part during surface mesh adaptation. For trimmed boundary layer meshes at corners transition edges also need to be handled. Hence, each of these mesh modification operations are developed to handle stack of boundary layer elements with mixed topologies, i.e., prisms, pyramids and tetrahedra. Lets look at each of the local mesh modification operation into more detail for boundary layer meshes.



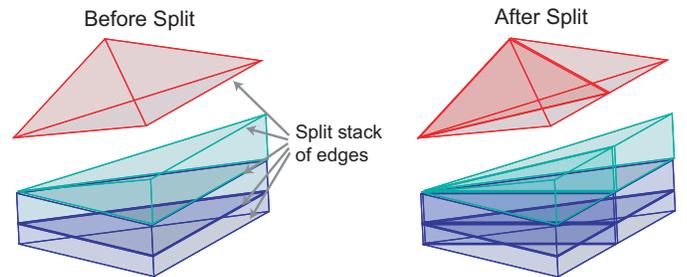
**Fig. 9** Mesh topology before (top) and after (bottom) modification : (a) edge split, (b) edge collapse and (c) edge swap.

#### 3.1 Layer edge split

The edge split operator is utilized for mesh refinement, i.e., to increase the mesh resolution. Ideally it is carried out for edges that have lengths greater than unity in the metric space. However, the fact that unit equilateral/regular elements cannot be packed to satisfy a

constant unit mesh metric field forces us to relax the criterion of achieving edges of unit length in metric space. In this regard, edge lengths in metric space are accepted when they fall under an interval close to one, referred to as  $[L_{low}, L_{up}]$ . A restriction,  $L_{low} \leq 0.5L_{up}$ , is applied in selecting the interval to prevent oscillation between refinement and coarsening of edges, for example,  $[1/\sqrt{2}, \sqrt{2}]$ . The edges whose lengths exceed the upper bound,  $L_{up}$ , in the metric space, referred to as *long edges*, are incrementally refined, see [13] for more details, where a set of long edges are marked and subdivided along with mesh entities surrounding them as done within the loop at step 11 of algorithm 1.

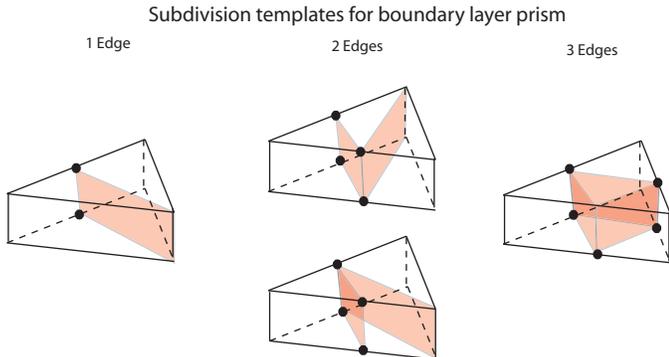
To apply this strategy for layered elements, marked edges are split in a similar fashion, see step 12 in algorithm 1. In the current approach, if edge split operation is carried out on any layer surface then it is extended to all the layers, including the interface elements, as shown in Fig. 10. The figure depicts stacks before and after the edge split operation where three boundary layer ele-



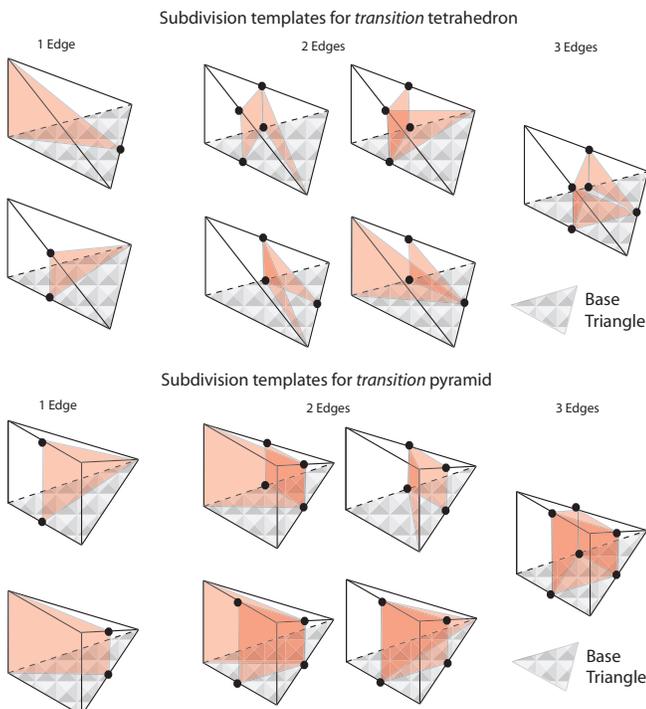
**Fig. 10** Stack of boundary layer and interface elements before (left) and after (right) split operation (interface elements are offset from boundary layer elements).

ments, i.e., two regular prisms and one transition pyramid, and one interface tetrahedron are subdivided. The current approach is conservative in nature as even if one edge in the stack is long then all the edges are split which will result in more nodes/elements but at the same time will lead to mesh resolution required for acceptable level of accuracy.

In order to apply this refinement strategy all the edges of stacks containing long edge(s) are marked for split. Subsequently, boundary layer and interface elements with marked edges are subdivided through the application of refinement templates. Fig. 11 provides the templates for regular prismatic elements and Fig. 12 provides it for transition pyramidal and tetrahedral elements. For interface tetrahedral elements, with marked edge(s), general subdivision templates defined for tetrahedron are used. Notice that subdivision of transition elements form elements of different topologies, i.e., prisms, pyramids and tetrahedra.



**Fig. 11** Subdivision templates for regular boundary layer elements based on number of marked base edges.



**Fig. 12** Subdivision templates for transition boundary layer elements based on number of marked base edges.

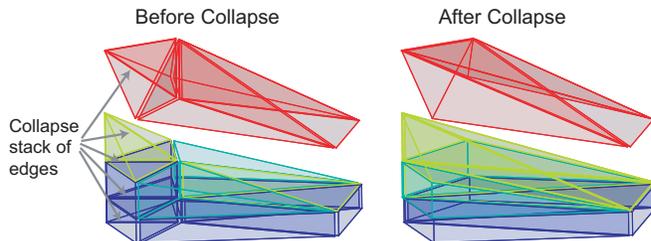
### 3.2 Layer edge collapse

The edge collapse operation is utilized for mesh coarsening, i.e., to decrease the mesh resolution. Ideally it must be carried out for edges that have lengths smaller than unity in the metric space but due to relaxed criterion on edge lengths in metric space it is performed for edges with lengths less than the lower bound,  $L_{low}$ , referred to as *short edges*. In edge collapse operation one or both ends of a short edge are pulled along the edge to collapse it. The edge collapse operation is not always attainable since it may produce flat/inverted elements or violate the topological association between the mesh and its geometric model (see appendix A in [20] for details).

The mesh modification operations in the coarsening procedure are repeatedly applied until no short edge can

be further processed. Moreover, the order in which the nodes of short edges are processed is important to maintain a good vertex distribution, see algorithm in section 3.2.2 of [13] for details, which is based on “topologically every other vertex” rule. To apply this strategy for layered part of the mesh, i.e., steps 6 and 14 in algorithm 1, base nodes of stacks containing short edges are dynamically processed in an order that is topologically every other one. Step 14 is important to carry out in order to eliminate short edges created during element subdivision in step 12, see Fig. 10 in [13] for further insight.

In the current approach, the edge collapse operation is performed on stacks that contain all short edges. This restriction is exercised to prevent oscillation between collapse and split operations as layer edge split operation is carried out even on stacks with a single long edge. Fig. 13 shows stacks of boundary layer (including transition) and interface elements before and after the



**Fig. 13** Stack of boundary layer and interface elements before (left) and after (right) collapse operation (interface elements are offset from boundary layer elements).

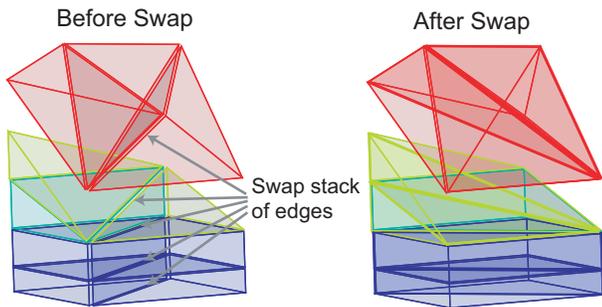
edge collapse operation where five base mesh triangles on the wall collapse into three. Notice that the topology of boundary layer elements, including prisms, can change during the edge collapse operation depending on the level of growth curves emanating from the nodes of new base triangulation on the wall.

### 3.3 Layer edge swap

The edge swap operation is utilized for mesh shape improvement, i.e., to improve the connectivity in the mesh. It is carried out for edges connected to low-quality elements under some user-defined measure in the metric space (like mean ratio etc.), which are referred to as *poorly shaped elements*, to replace them with better ones. Poorly shaped elements might be present in the initial mesh and/or created during edge collapse operation. For simplex 3D meshes the edge swap operation reconfigures tetrahedra surrounding the edge by removal of the edge and re-triangulation of the *equatorial plane* with pair of tetrahedra for each triangle [21]. The shape of the new tetrahedra in the metric space is computed for each possible configuration and the best configuration is selected. The edge swap operation is also not always attainable

since it may produce inverted elements or violate the topological association between the mesh and its geometric model (refer appendix A in [20] for details).

The mesh modification operations in the mesh shape improvement step are applied until no poorly shaped element can be further processed, see Fig. 21 in [13]. To apply this strategy for layered part of the mesh, i.e., step 15 in algorithm 1, edges connected to poorly shaped faces on layer surfaces are considered and reconfigured to replace them with better ones. Note that there is only one other possible configuration in case of layer edge swap for base layer faces under surface adaptation whereas interior tetrahedral elements at the interface are reconfigured based on the equatorial plane. Fig. 14 shows stacks of boundary layer (including transition) and interface el-



**Fig. 14** Stack of boundary layer and interface elements before (left) and after (right) swap operation (interface elements are offset from boundary layer elements).

ements before and after the edge swap operation. Again notice that the topology of boundary layer elements, including prisms, can change based on the level of growth curves emanating from the nodes of new layer faces on the wall.

#### 4 Layer thickness adjustment

In practical problems of interest, layer thicknesses for an initial boundary layer mesh are typically not ideal, i.e., either lower or higher than the desired local values for a given level of accuracy. Thus, the layer thicknesses need to be adjusted locally for each growth curve to achieve sufficient normal mesh resolution as done in step 9 of algorithm 1. In the current approach, thickness adjustment is achieved through node movement such that layer configuration is preserved.

The procedure first evaluates the target locations for nodes on all the growth curves and then tries to move the nodes to their computed target locations. Node movement through repositioning is not always possible, especially for the top most nodes, as it may introduce inverted elements, in which case local mesh modification operations are applied to the interior volume mesh to make the way for repositioning to be successful. The

idea is adopted from [14] where similar concept is applied in the context of moving newly created nodes on straight-sided edges at curved boundaries onto the solid model surfaces to improve the geometric approximation. Thus, the node movement technique developed in this work for thickness adjustment uses local mesh modifications; and do not employ mesh movement/deformation methods based on techniques like spring analogy. Note that although the figures in this section illustrate specific concepts for 2D examples the developments have been implemented and exercised for 3D boundary layer meshes as given in the results section.

It is important to note that thickness adjustment may lead to many alterations on the interior volume mesh near the interface, depending upon the magnitude of changes desired in the total thickness, due to repositioning of nodes and/or local mesh modifications. It will involve duplication of work to regain the desired mesh resolution in case if these alterations are done after the interior volume mesh has been adapted to satisfy the mesh metric field. Hence, it is appropriate to perform thickness adjustment before the interior volume mesh has been adapted. In other words, the order in which various steps of mesh adaptation are carried out is critical to maintain the efficiency. In the current approach, thickness adjustment is performed before refinement at step 9 in algorithm 1. Moreover, applying thickness adjustment before mesh refinement allows the new growth curves introduced by edge splits to inherently accomplish the desired normal resolution. For new growth curves on curved boundaries, the originating nodes are projected onto the solid model surfaces to improve geometric approximation towards actual definition, see step 13 in algorithm 1, as discussed later in the section.

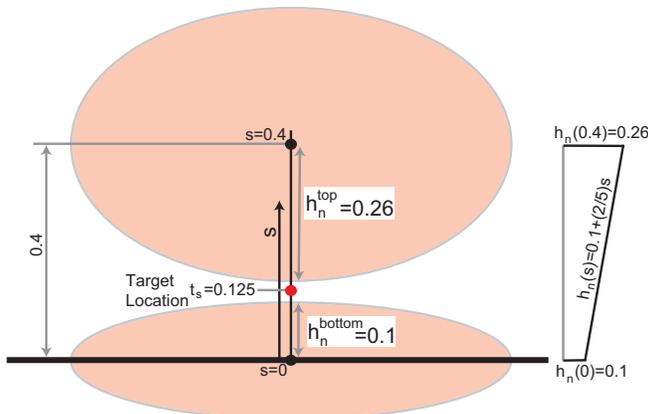
##### 4.1 Target location computation

The nodal spacing on each growth curve determine the local layer thicknesses. The desired nodal spacing or local thickness for each layer, and hence local total thickness, is determined by the normal component of the mesh metric tensors over a growth curve. In order to do this, starting with first node off the wall the length of first growth edge is computed in the metric space, the computation is 1D in nature based on normal mesh sizes,  $h_n$ . A long growth edge, in the metric space, indicates that the local thickness needs to be decreased whereas for a short growth edge it needs to be increased. Hence, the direction of node movement is dependent on whether the growth edge is long or short in the metric space.

The next step is to determine the desired amount of node movement which needs to be such that the growth edge satisfies the normal mesh sizes. As the values of  $h_n$  are dependent on the spatial location the evaluation of desired amount of node movement must account for the variation in  $h_n$ . In the current approach, the variation

in  $h_n$  is considered by using piece-wise linear interpolation of end values over the current or upper growth edge based on the direction in which the node movement is desired. The target locations are then computed based on the desired direction and amount of node movement.

To understand the evaluation process of target nodal locations an example is used. Say, values of  $h_n$  at bottom and top ends of a growth edge, with physical length of 0.4, are  $h_n^{bottom} = 0.1$  and  $h_n^{top} = 0.26$ , respectively, as shown in Fig. 15. As the growth edge is long, i.e.,  $0.4 > 0.26 > 0.1$ , the desired direction of node movement in this case is towards the bottom end, i.e., downwards on the growth curve. Now, to evaluate the amount of node movement required, the variation in  $h_n$  is accounted by using linear interpolation of  $h_n$  values at the ends of the current growth edge, i.e.,  $h_n(s) = 0.1 + ([0.26 - 0.1]/[0.4 - 0])s = 0.1 + (2/5)s$ , where  $s$  is parametric location along the edge with  $s = 0$  and  $s = 0.4$  at bottom and top ends, respectively. Thus, for the growth edge to satisfy the metric space the target location for the top end of the edge,  $t_s$ , is computed by solving this linear equation:  $t_s - 0 = (h_n(0) + h_n(t_s))/2 \implies t_s = (0.1 + 0.1 + (2/5)t_s)/2 \implies t_s = 0.125$ .



**Fig. 15** A 2D example for target location computation based on normal component,  $h_n$ , of mesh metric tensors.

Similarly, for short growth edges the desired direction of movement is upwards on the growth curve. The amount of node movement required in this case is computed by using linear interpolation,  $h_n(s)$ , over the upper level growth edge as the movement is desired in that direction. For the top most growth edge as there is no upper level growth edge the amount of movement required in case of short edge is computed by using linear extrapolation along the direction of the edge.

The target locations for all the nodes, except the originating node on the wall, of a growth curve are computed in a sequential manner starting with first node off the wall. The target locations computed for each node is stored in a list of locations that is assigned in the beginning to initial locations. In each step of the sequential

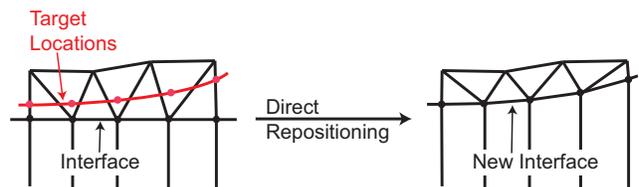
process, the amount of movement required for the current node is applied to all the locations in the list associated to upper level nodes. This is done because upper layers will cascade due to movement in any layer below. At each step as upper level node locations in the list are updated the normal mesh sizes,  $h_n$ , are also updated based on the new nodal locations. Since the movement in each step is uni-directional along the growth curve piece-wise linear interpolation over line segments defined by growth edges is used to update scalar values of  $h_n$ .

Finally, the computed target locations are modified, if required, to ensure elements of acceptable shape near tight corners, see Fig. 8. A similar approach can be applied to cases where the gap between two opposite stack of layers is limited. Moreover, the computed local layer thicknesses can be limited in terms of rate of mesh size changes along different levels of growth curve and/or among neighboring growth curves to obtain graded boundary layer elements. Once the final target locations for nodes of every growth curve is available at hand the next step is to apply node movement to reach respective target locations, which may involve local mesh modification operations when direct repositioning is not possible due to introduction of inverted elements.

#### 4.2 Application of node movement

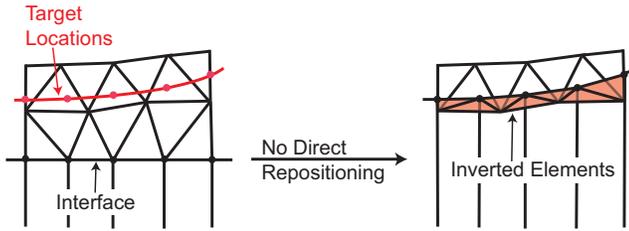
In the current approach, as the configuration of the layers during thickness adjustment is kept fixed a critical step is the realization of node movement for the top most nodes at the interface due to their connection with the interior tetrahedral volume mesh. All the other growth curve nodes can be placed after the top most nodes have been moved to their computed target locations. Hence, the focus is node movement for the top most nodes at the interface between the layered and interior volume part of the mesh.

The nodes for which movement to their target locations is feasible through repositioning, i.e., leads to no inverted elements at the interface, are directly moved as shown in Fig. 16. The nodes for which movement is



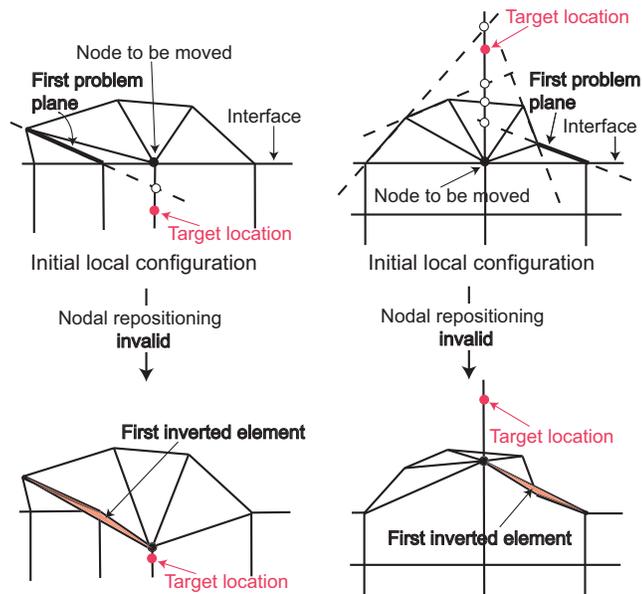
**Fig. 16** A 2D example where direct nodal repositioning to increase the total thickness leads to no inverted elements at the interface.

not addressed by direct repositioning operation due to creation of inverted elements, as shown in Fig. 17, are handled iteratively by applying local mesh modifications



**Fig. 17** A 2D example where direct nodal repositioning to increase the total thickness leads to inverted elements at the interface.

to the interior volume mesh. In each step the node under consideration is moved at least as far as to the location onto the plane that causes the first element to become flat. It is crucial to reach at least onto the first plane where the problem occurs as depicted in Fig. 18, referred to as *first problem plane*, in order to avoid situations where the process takes ever decreasing movement at every step leading to many increments with deteriorating local mesh quality, see [14] for further understanding.

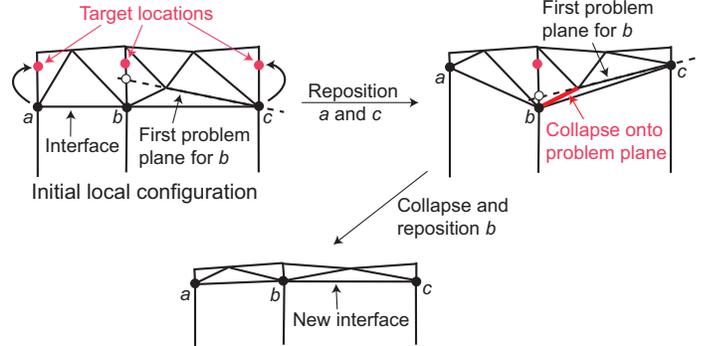


**Fig. 18** 2D examples to depict the first problem plane for two cases involving decrease in total thickness (left) and increase in total thickness (right).

To perform movement at least to the first problem plane a set of local mesh modification operations are utilized. In the current approach, the set of operations includes: *single step operations* of edge collapse, edge swap and face swap; and *compound operations* of edge split with collapse of new edge and double edge split with collapse of new edge, i.e., consisting of collapse (see [13, 14] for more details). An appropriate operation from this set is selected depending on the configuration of the local interior mesh around the node. The basis for selecting an operation is to either reach the location where the

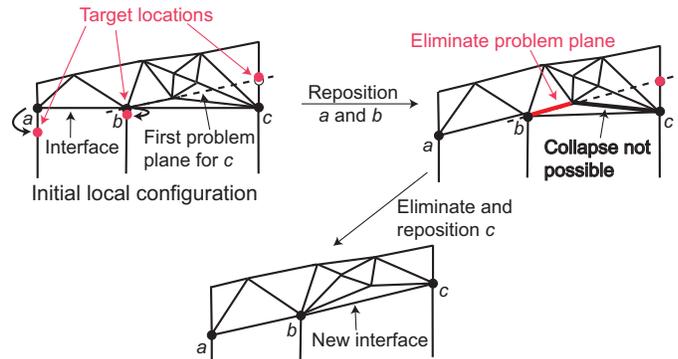
first problem occurs or eliminate the first problem plane to make the way for movement.

Applying a collapse operation on appropriate mesh entities is considered foremost to reach the location of the first problem as depicted in Fig. 19 (see Fig. 9 in [14] for further insight). For cases where collapse is not pos-



**Fig. 19** A 2D example for collapse operation onto first problem plane to move the node.

sible other modifications to the local mesh are examined with the goal of eliminating the problem plane through a single step operation or with the help of a compound operation comprised of collapse. Elimination of the problem plane is tried by removal of either the mesh face defining the problem plane through a single step operation or the interior tetrahedra that becomes flat, i.e., bounded by the node and problem mesh face, with the help of a compound operation. Fig. 20 illustrates the case



**Fig. 20** A 2D example for elimination of first problem plane to allow node movement.

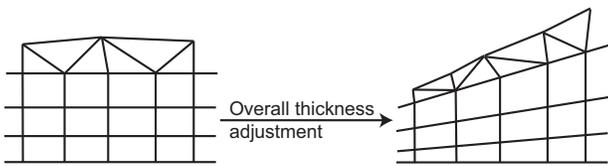
where first problem plane is eliminated to make the way for node movement to the location where first problem occurs. Further details about the mesh modification operations and key mesh entities to be considered to allow node movement at least as far as the first problem plane are provided in [14].

In the current approach, nodes at the interface are considered iteratively with the restriction that at each step any mesh modification to the interior volume mesh

is performed only if movement at least to the first problem plane is realized. Thus, in each step of the process an interface node is selected to be considered for node movement either to its target location or at least to the first problem plane. For nodes involving problem planes only a single modification operation is carried out at a time for the selected node. No further modifications are performed for the selected node to reach its target in the same step before considering movements for neighboring nodes. With such an approach a significant number of problem planes will not arise as node movements are interlinked with the neighboring ones and hence, it reduces the number of local mesh modifications applied.

There is one caution that needs to be observed in the context of applying local mesh modifications on the interior volume mesh to accomplish thickness adjustment for layered part of the mesh. As the movement of nodes at the interface is strongly coupled to movement of other neighboring ones, it is desirable to apply a strategy which accounts for the movement of the whole interface in an progressive fashion, where local mesh modifications are instantiated only when progressive movement of the interface is hindered and not when movement of only a single node is investigated. The current scheme only partly accounts for these issues. Additional improvements under development are expected to increase the effectiveness of this step of the mesh modification procedure.

The nodes that accomplish movement to their target location are no longer considered in further iterations and the process finishes when all the nodes at the interface are successively addressed. Other lower level nodes on the growth curves are placed to their respective target locations once the top most nodes have reached their target locations as depicted in Fig. 21. As mentioned before the number and topology of layers is not changed during



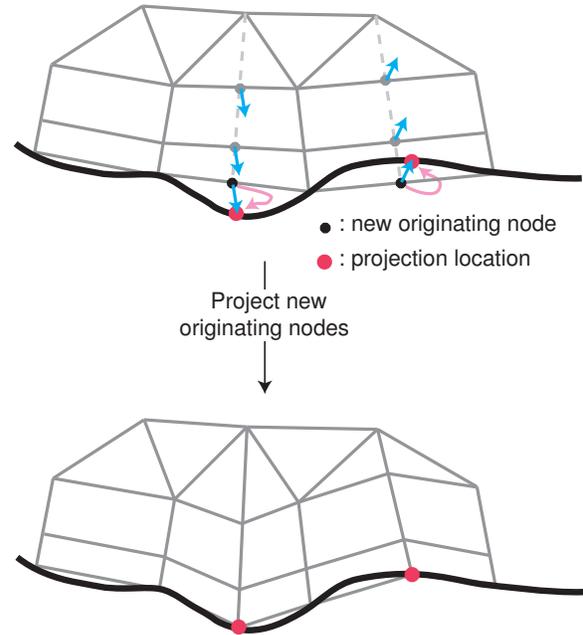
**Fig. 21** A 2D example where overall boundary layer thickness adjustment is accomplished by application of node movement.

thickness adjustment in the current adaptive procedure. Note that the interior volume mesh may have encountered many alterations in this process depending upon the amount of change in total thickness and hence, it is modified to respect the mesh metric field after this step.

#### 4.3 Projection of new nodes onto curved surfaces

In the current approach, the newly created originating nodes by refining straight-sided layer edges at curved

boundaries are projected onto to the solid model surfaces to improve the geometric approximation as the mesh resolution is increased. To do this the movement vector required for the originating node is determined and is applied to all the nodes of the growth curve as the local layer thicknesses already satisfies the desired normal mesh sizes, i.e., step 13 in algorithm 1. Fig. 22 illustrates the process for a 2D example. Similar to thickness adjustment step the movement is applied through direct repositioning or by using local mesh modifications to the interior volume mesh.



**Fig. 22** A 2D example where newly created originating nodes at curved boundaries are project onto solid model surface.

## 5 Application to viscous flows

We now demonstrate the capabilities of the boundary layer mesh adaptation procedure developed in this work by applying it on two type of problem cases. The first type involves pulsatile incompressible blood flow in blood vessels including a case of porcine aorta with a stenosis bypassed by a graft [22]. These problems fall under the category of cardiovascular flows where computations are performed using patient-specific anatomic and physiologic information to help in surgical planning [23,24] and design of medical devices [25]. One of the key quantities of interest in these problems is the wall shear stress. The other case involves a steady, laminar, compressible flow through double throat nozzle, arising in the field of aerodynamics. This problem was one of the test case of GAMM workshop [26] and consists of supersonic viscous flow. One of the interesting features of this flow is the shock-wave/boundary-layer interaction.

### 5.1 Blood Flow

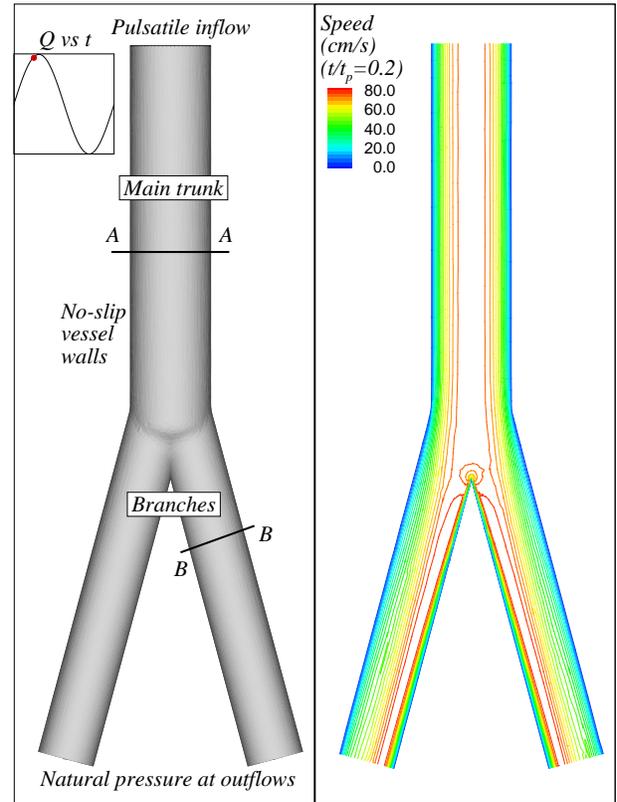
In the case of cardiovascular flows, we consider two examples. The first example consists of a cylindrical blood vessel with a symmetric bifurcation whereas the other involves a patient-specific case of porcine aorta with a stenosis bypassed by a graft. Flow computations for these cases are done using unsteady, incompressible, Navier-Stokes equations assuming Newtonian constitutive behavior with rigid and impermeable vessel walls. See [27] for recent developments considering deformable arteries.

We employ the streamline upwind/Petrov-Galerkin (SUPG) stabilization method introduced in [28] to discretize the governing equations. The stabilized finite element formulation currently utilized has been shown to be robust, accurate and stable on a variety of flow problems (see for example [23,29]). Linear finite elements, both for the pressure and the velocity field, are used in these computations. The system of non-linear ordinary differential equations obtained is discretized in time via a second-order generalized- $\alpha$  time integrator [30] resulting in a non-linear system of algebraic equations. This system is in turn linearized with Newton's method to obtain a linear algebraic system of equations that is solved using the linear algebra solver of [31].

In these cases we demonstrate the effectiveness of adapted boundary layer meshes (with layered elements near the walls) to compute the wall shear stress, see [1] for further details on its computation. To this end, we perform simulations to compare wall shear stress values computed on boundary layer meshes to those computed on fully unstructured, anisotropically adapted meshes with roughly same number of degrees of freedom.

**5.1.1 Bifurcating cylindrical vessel** A schematic of the cylindrical blood vessel with symmetric branches is shown in left column of Fig. 23. The main trunk is 5cm with a diameter of 1cm and the branches, at an angle of  $30^\circ$ , are around 4cm long with a diameter of  $1/\sqrt{2}$ cm. The inflow velocity is assumed to be a Womersley profile [32], with Womersley number  $\alpha = 5.6$  and time period  $t_p = 1.25s$ . The volume flow rate at the inlet, is depicted in the inset within Fig. 23, which is a sinusoidal wave form with an amplitude and mean of  $20cc/s$ . No-slip boundary conditions are assumed on the vessel walls and pressure and zero traction are prescribed at the outlet. The viscosity and density are assumed to be  $\mu = 0.04dynes\ s/cm^2$  and  $\rho = 1g/cm^3$ , respectively.

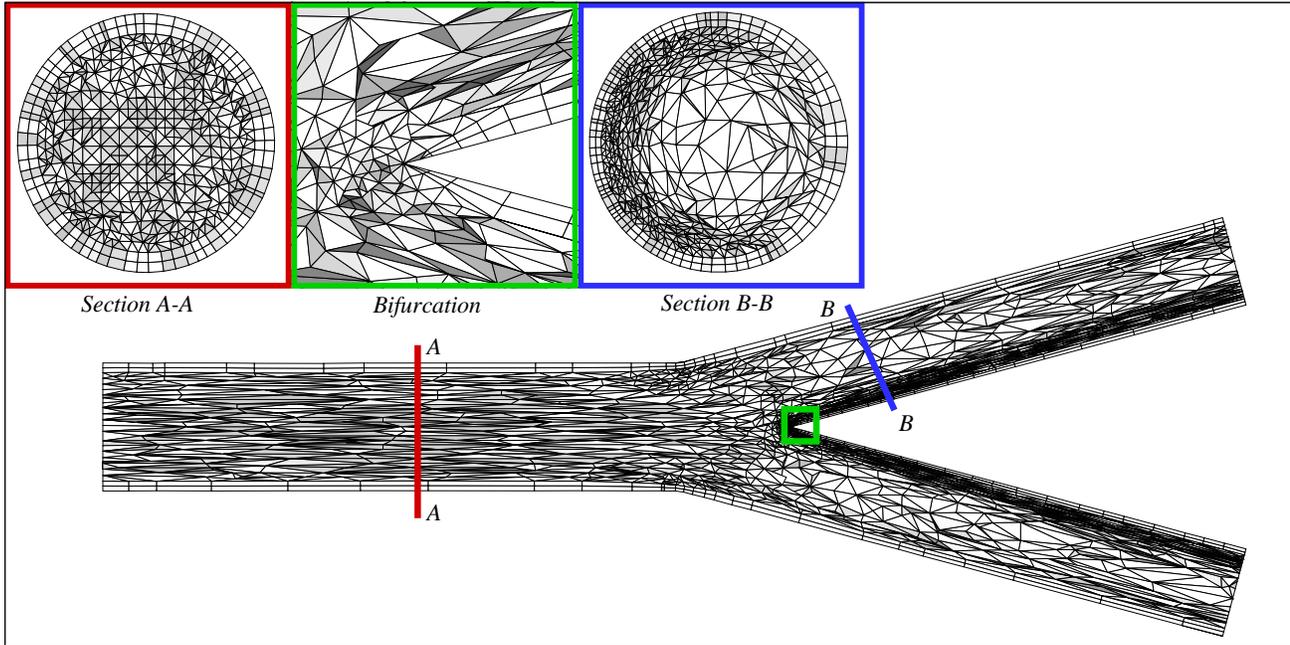
The simulations were performed for two cardiac cycles to obtain a periodic flow and thus, the results for the second cycle are presented here. Each cycle was divided into 500 time steps with a constant time step size of 0.0025s. The computations with 1000 time steps per cycle show no significant difference as compared to the ones obtained with 500 time steps per cycle, which ensures that the temporal errors are smaller than those due to the spatial discretization.



**Fig. 23** Schematic of bifurcating cylindrical blood vessel along with boundary conditions (left) and isolines of the computed flow speed on horizontal center-plane at  $t/t_p = 0.2$  (right). The inset shows the flow rate at the inlet along with the instant (see dot) corresponding to the flow profile shown on the right.

The computations are started using a pre-defined boundary layer mesh with two layers of constant thickness and an isotropic interior volume mesh consisting of approximately 25K nodes and 99K regions. Two iterations of the flow computation and mesh adaptation are performed to obtain an adapted boundary layer mesh with around 32K nodes and 140K regions. A single mesh adaptation process is applied for the whole cardiac cycle. The average speed profile obtained over the second cardiac cycle is used to compute the error indicators (see [1] for more details). A similar process is performed using fully unstructured, anisotropic adaptivity to allow comparison of the resulting solutions. Note that the number of degrees of freedom is roughly the same for both type of meshes used for comparison.

The right column of Fig. 23 shows isolines of the computed speed field on the adapted boundary layer mesh at one-fifth of the cycle, i.e.,  $t/t_p = 0.2$ . It can be observed that the flow field is fully developed in most of the main trunk of the vessel while there is axial variation in the branches. Isotropic flow gradients can be observed near the bifurcation whereas further downstream the gradients become directional in nature with steeper radial component towards the inner side of the branches.



**Fig. 24** Collection of mesh faces cut by different planes through the adapted boundary layer mesh of a cylindrical vessel with symmetric bifurcation (the windows correspond to magnified views).

Fig. 24 presents clip-plane view, i.e., a collection of mesh faces cut by the plane, of the adapted mesh. The effect of flow features can be clearly seen on the adapted mesh. Anisotropic elements well aligned with the flow are created which is common of anisotropic adaptive procedures. The adapted mesh is fairly uniform along the section A-A in the trunk. In section B-B, on the other hand, there is significant variation in mesh resolution, in both the azimuthal and radial direction, especially near the inner side of the branches where the gradients are high. Near the bifurcation element sizes are small and isotropic, see the center zoom window in Fig. 24, reflecting the fact that the solution behavior is nearly singular around this region. Similar features are observed in the fully unstructured, anisotropically adapted mesh [1].

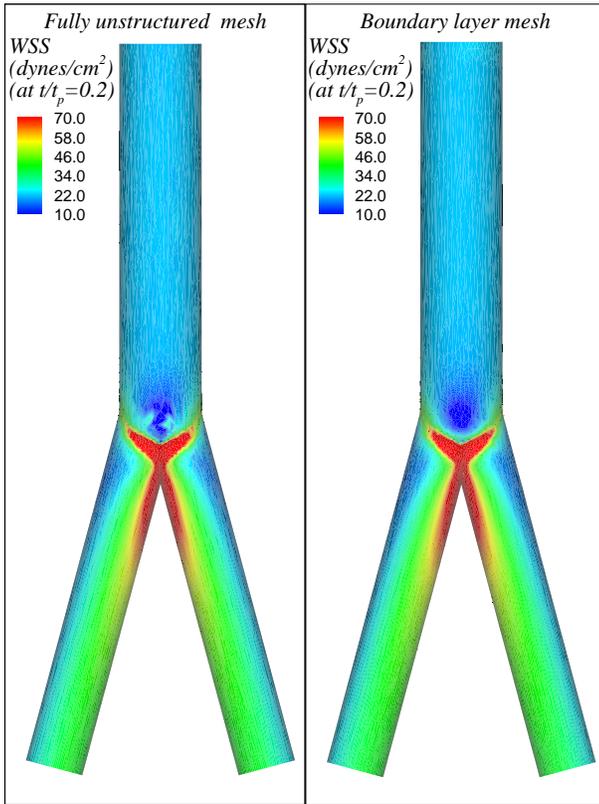
The adapted boundary layer mesh clearly demonstrates the ability to maintain layered and graded elements near the walls that incorporate mesh resolution as requested depending upon the flow features. Some other key features of the adapted boundary layer mesh that can be observed in this case are the transition of layers near the bifurcation to obtain smooth isotropic elements and the azimuthal variation in layer thicknesses along the section B-B which is accomplished by changing the local total thickness. Note that the folding angle at the bifurcation is varying and hence, we only partially trim the stack of layers in this case.

To demonstrate the effectiveness of boundary layer mesh adaptation, wall shear stress values computed at  $t/t_p = 0.2$  on anisotropically adapted, fully unstructured and boundary layer meshes are presented in Fig. 25. The results obtained on the adapted boundary layer mesh

appear to be of better quality with the same level of mesh resolution. Fig. 26 further supports this observation where wall shear stress values along the circumference of section A-A are provided for both the meshes. The exact value of wall shear stress is  $21.49 \text{ dynes/cm}^2$  for a straight cylindrical vessel with similar flow conditions. The values obtained on the adapted boundary layer mesh have smaller oscillations than the ones obtained on the fully unstructured mesh. Note that the range of ordinate in Fig. 26 is magnified to clearly show the fluctuations.

**5.1.2 Porcine aorta** The layout of the porcine aorta with a stenosis bypassed by a graft is illustrated in the top of Fig. 27. The vessel in this case is approximately  $10 \text{ cm}$  long with a diameter of  $1.6 \text{ cm}$  at the inlet. Similar boundary conditions as the bifurcating cylindrical vessel case are applied herein. For more physiological outflow boundary conditions based on resistance and impedance conditions see [33]. The wave form of the inflow volume rate is shown in the inset within Fig. 27, which is obtained from the imaging data [22], with a time period of  $t_p = 0.6 \text{ s}$ ; and a peak and trough of around  $100 \text{ cc/s}$  and  $5 \text{ cc/s}$ , respectively. For this case, the viscosity and density are assumed to be  $\mu = 0.04 \text{ dynes s/cm}^2$  and  $\rho = 1.06 \text{ g/cm}^3$ , respectively.

In this case four cardiac cycles are simulated to obtain a periodic flow and thus, results for the last cycle are presented. Computations are performed using 100 time steps with equal time step size of  $0.006 \text{ s}$  in a cardiac cycle. A similar process to the one described for the cylindrical vessel case is followed to obtain an adapted boundary layer mesh and a fully unstructured, adapted

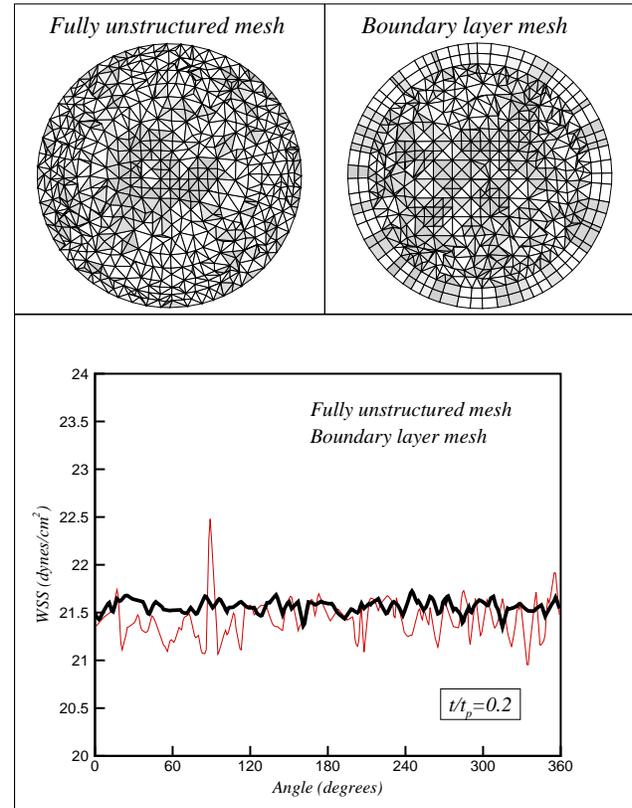


**Fig. 25** Wall shear stress values computed at  $t/t_p = 0.2$  on anisotropically adapted, fully unstructured and boundary layer meshes of a cylindrical vessel with symmetric bifurcation.

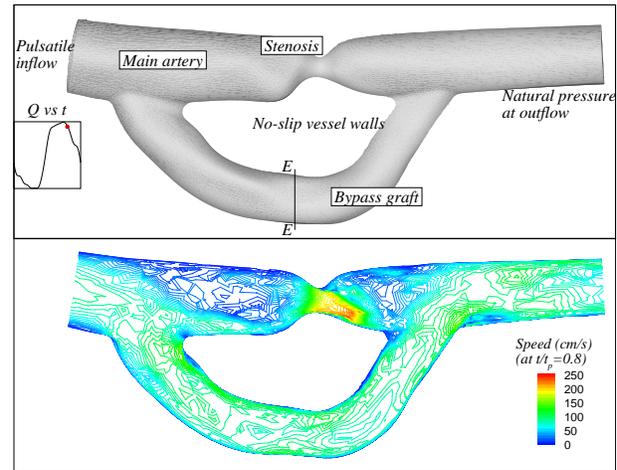
mesh. The initial mesh used in this case is fairly coarse and consists of around 10K nodes and 37K regions. After adaptation, the boundary layer mesh increases to around 136K nodes and 502K regions. As before, the number of degrees of freedom is roughly the same for both type of adapted meshes that are used for comparison.

Isolines of the computed speed field at an instant of  $t/t_p = 0.8$  is provided in the bottom of Fig. 27. The flow consists of strong boundary layers that form near the walls. Near the stenosis the flow is quite complex with significant spatial and temporal variations, including flow reversal in part of the cardiac cycle. Free shear layers form in the region where the flow re-enters the main artery from the bypass.

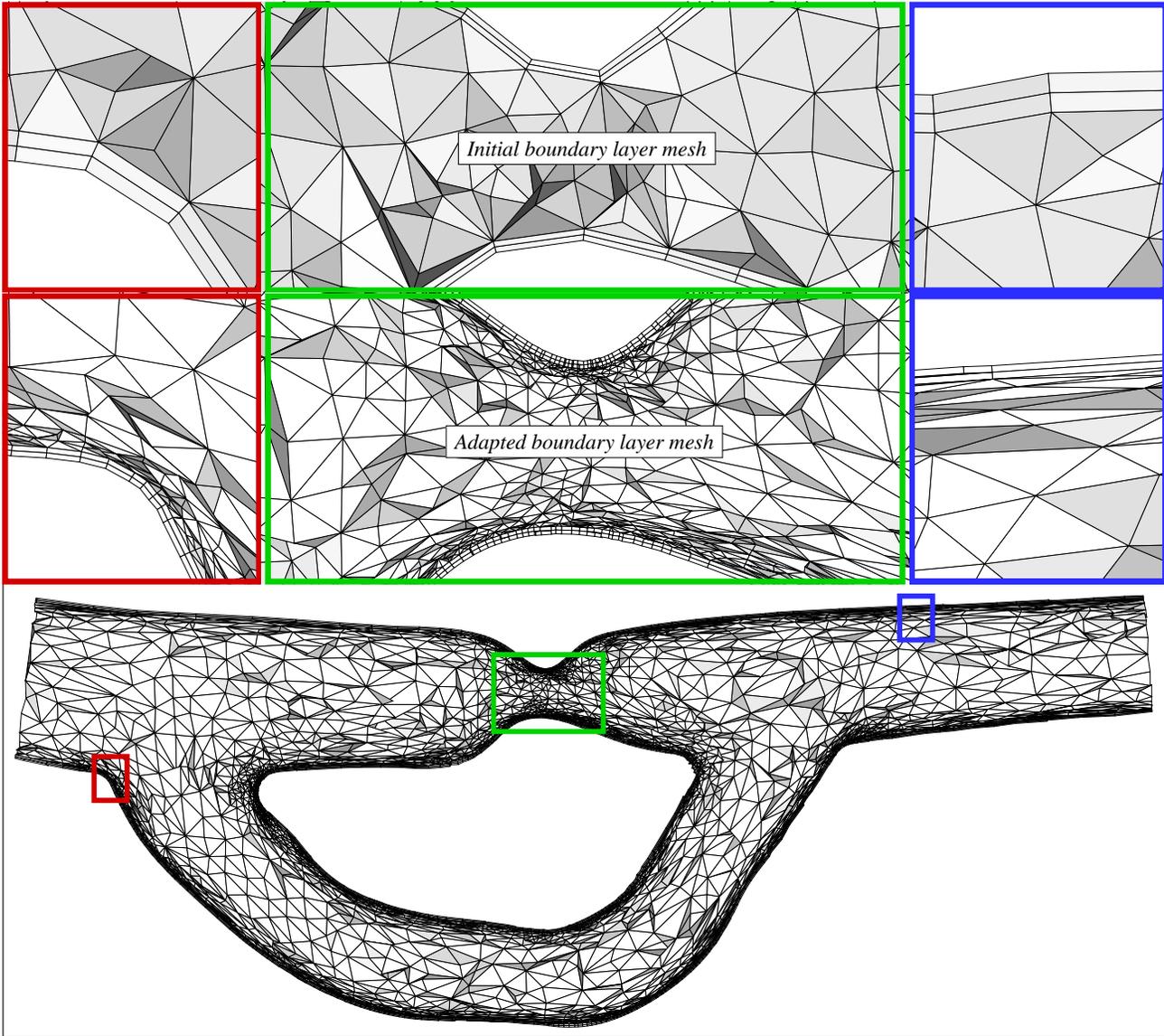
Fig. 28 presents clip-plane view of the adapted mesh. Similar to the previous case the effect of flow features can be clearly seen on the adapted mesh. Elongated elements aligned in the flow direction are created where gradients in flow speed are high such as near the vessel walls due to boundary layers. In this case the clip-plane view of the initial mesh are also provided to illustrate the effect of mesh modification procedure. Isotropic elements are created where flow variation in the stream-wise direction is significant, especially near the stenosis region. Notice that the newly created nodes in the adapted meshes are



**Fig. 26** Wall shear stress values along the circumference of section A-A at  $t/t_p = 0.2$  for a cylindrical vessel with symmetric bifurcation using anisotropically adapted, fully unstructured and boundary layer meshes (notice that the range of the ordinate is magnified). The meshes on top correspond to mesh faces cut by section A-A.



**Fig. 27** Schematic of porcine aorta along with boundary conditions (left) and isolines of computed flow speed on the vertical center-plane at  $t/t_p = 0.8$  (right). The inset shows the flow rate at the inflow along with the instant (see dot) the flow profile corresponds to.



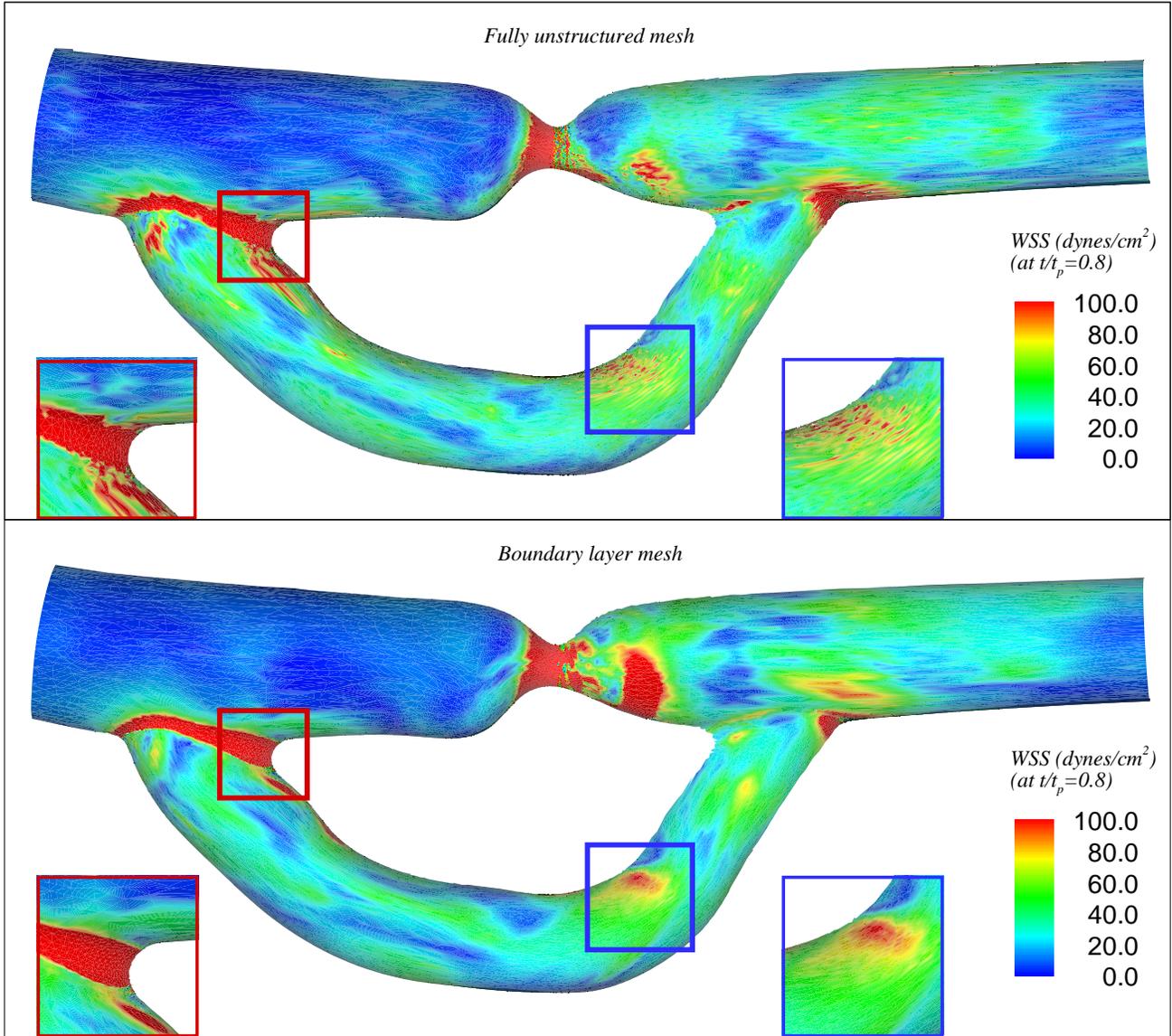
**Fig. 28** Collection of mesh faces cut by the vertical center-plane through the adapted boundary layer mesh of a porcine aorta. The windows correspond to magnified views which also show the initial boundary layer mesh.

projected onto to the solid model surface to improve the geometric approximation which is very distinct near the stenosis area. Similar features are observed in the fully unstructured, anisotropically adapted mesh, see [15].

Again, the adapted boundary layer mesh demonstrates the capability to maintain layered and graded elements near the walls. Key features of the mesh that can be observed in this case are the smooth transition of elements at the interface between layered and unstructured tetrahedral part of the mesh, even for portions with highly anisotropic elements, and the ability to improve the geometric approximation as the mesh resolution is increased.

The effectiveness of boundary layer mesh can be observed in Fig. 29 which provides wall shear stress values computed at  $t/t_p = 0.8$  on anisotropically adapted, fully unstructured and boundary layer meshes. The re-

sults obtained on the adapted boundary layer mesh are clearly superior to the ones on fully unstructured mesh which possess significant fluctuations with the same level of mesh resolution. Fig. 30 provides additional evidence to support this observation where shear force vectors at a different instant of  $t/t_p = 0.48$  along the circumference of section E-E are provided for both the meshes. Shear force vectors have significant variation due to swirling flows which are typical in cardiovascular system. The shear force vectors obtained on the adapted boundary layer mesh are considerably smoother as compared to the ones computed on the fully unstructured anisotropic mesh.



**Fig. 29** Wall shear stress values computed at  $t/t_p = 0.8$  on anisotropically adapted, fully unstructured and boundary layer meshes of a porcine aorta (the windows corresponds to magnified views).

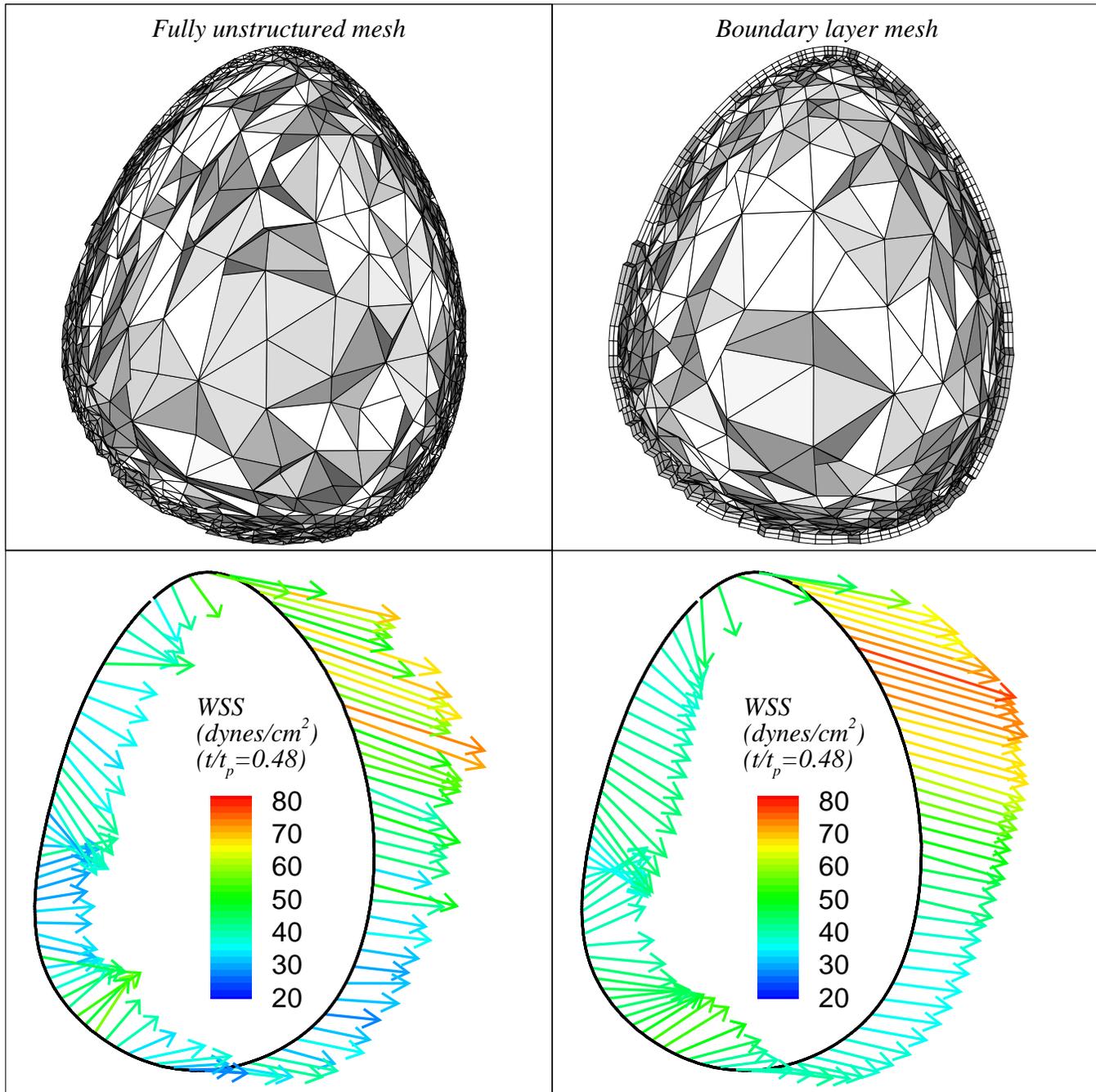
### 5.2 Double throat nozzle

Schematic of the double throat nozzle is provided in Fig. 31 along with the boundary conditions. The geometric specifications of the nozzle are available in [26]. Although the flow is 2D in nature the computations are performed over a 3D domain. The inflow conditions are based on the reservoir values. Only half of the nozzle is considered with symmetry conditions at the centerline. No-slip condition is imposed on the upper wall of the nozzle that is assumed to be isothermal with temperature set to reservoir value. A slip condition is applied to the side walls along with no heat flux. The outflow is supersonic in most of the portion and hence, no boundary conditions are prescribed at the outlet.

The flow is laminar in nature with a Reynolds number,  $Re = \rho_0 a_0 L / \mu$ , of 1600 based on the inlet stagnation

speed of sound ( $a_0$ ), density ( $\rho_0$ ) and viscosity ( $\mu$ ); and reference length ( $L$ ) as height of the first throat from the centerline. The reservoir values are taken at standard sea level conditions for pressure ( $P_0 = 101325 Pa$ ), temperature ( $T_0 = 288 K$ ) and density ( $\rho_0 = 1.225 Kg/m^3$ ). The coefficients for viscosity and thermal conductivity are assumed to be constant. The Prandtl number and ratio of specific heats are taken to be 0.72 and 1.4, respectively. The computations are carried out in the non-dimensional form with  $P_{ref} = \rho_{ref} u_{ref}^2 = \rho_0 a_0^2$  and  $T_{ref} = P_{ref} / \rho_{ref} = a_0^2$ .

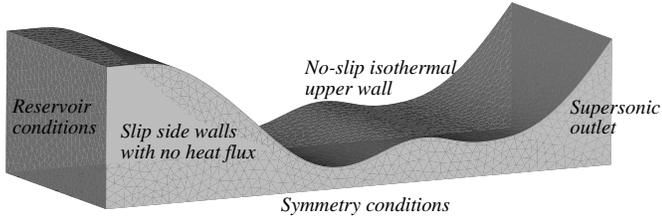
The flow solver used for this case employs streamline upwind/Petrov-Galerkin (SUPG) stabilized method, to discretize compressible Navier-Stokes equations, with linearly interpolated pressure-primitive variables, see [34, 35] for further details. A supplemental term for discon-



**Fig. 30** Tangential component of the surface traction (shear force) along the circumference of section E-E at  $t/t_p = 0.48$  in bypass graft of porcine aorta for anisotropically adapted, fully unstructured and boundary layer meshes. The meshes on top correspond to mesh faces cut by section E-E.

tinuity capturing is applied to maintain optimal rate of convergence, see [36]. As the flow is steady in nature an implicit, first-order accurate method of backward Euler is used for time integration. The non-linear system of algebraic equations obtained is in turn linearized with Newton's method which yields a linear algebraic system of equations. The GMRES method [37] with sparse storage and pre-conditioners is used to solve the linear system of equations.

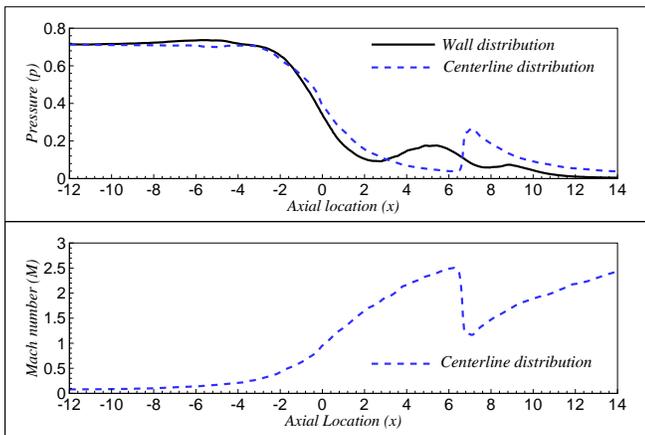
The flow computations are started using a pre-defined boundary layer mesh with two layers of constant thickness and an isotropic interior volume mesh. Three iterations of the flow computations and boundary layer mesh adaptation are performed based on the adaptive procedure developed in this work. In each adaptive cycle the mesh is adapted as per the computed mesh size field information based on the steady-state solution, see [1] for details. The initial mesh has around 15K nodes and 60K regions whereas the adapted mesh has around 281K



**Fig. 31** Schematic of double throat nozzle along with boundary conditions.

nodes and 1481K regions. In this case we only utilize the boundary layer mesh adaptivity to present results.

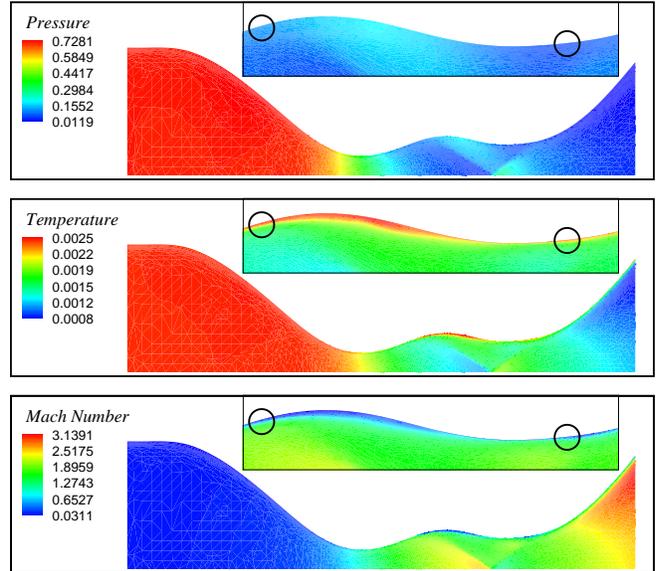
Fig. 32 shows the centerline and wall distribution of non-dimensional pressure along with the centerline



**Fig. 32** Distribution of the computed solution in double throat nozzle: (top) non-dimensional pressure at upper wall and centerline and (bottom) Mach number at centerline.

distribution of Mach number computed on the adapted mesh. The flow becomes sonic at the first throat and shock waves form further along the axis. It can be observed from the centerline distribution that the phenomenon of shock wave is sharply resolved. Fig. 33 presents the computed solution over the domain. Flow features like shock waves, shock wave reflection, boundary layers, boundary layer separation and shock-wave/boundary-layer interactions are well captured. The magnified views provided in Fig. 33 show the important viscous flow phenomenon of flow separation at two locations, one after the first throat and other around the second throat.

To demonstrate the features of the adapted mesh a clip-plane view at the center of the nozzle is presented in Fig. 34 along with the Mach number. It can be seen that the imprint of flow phenomena is strong on the adapted mesh, especially near flow features like shock wave reflection and flow separation at the location of shock-wave/boundary-layer interaction. It is interesting to note that the mesh resolution near such features directly correspond to the nature of the solution like well



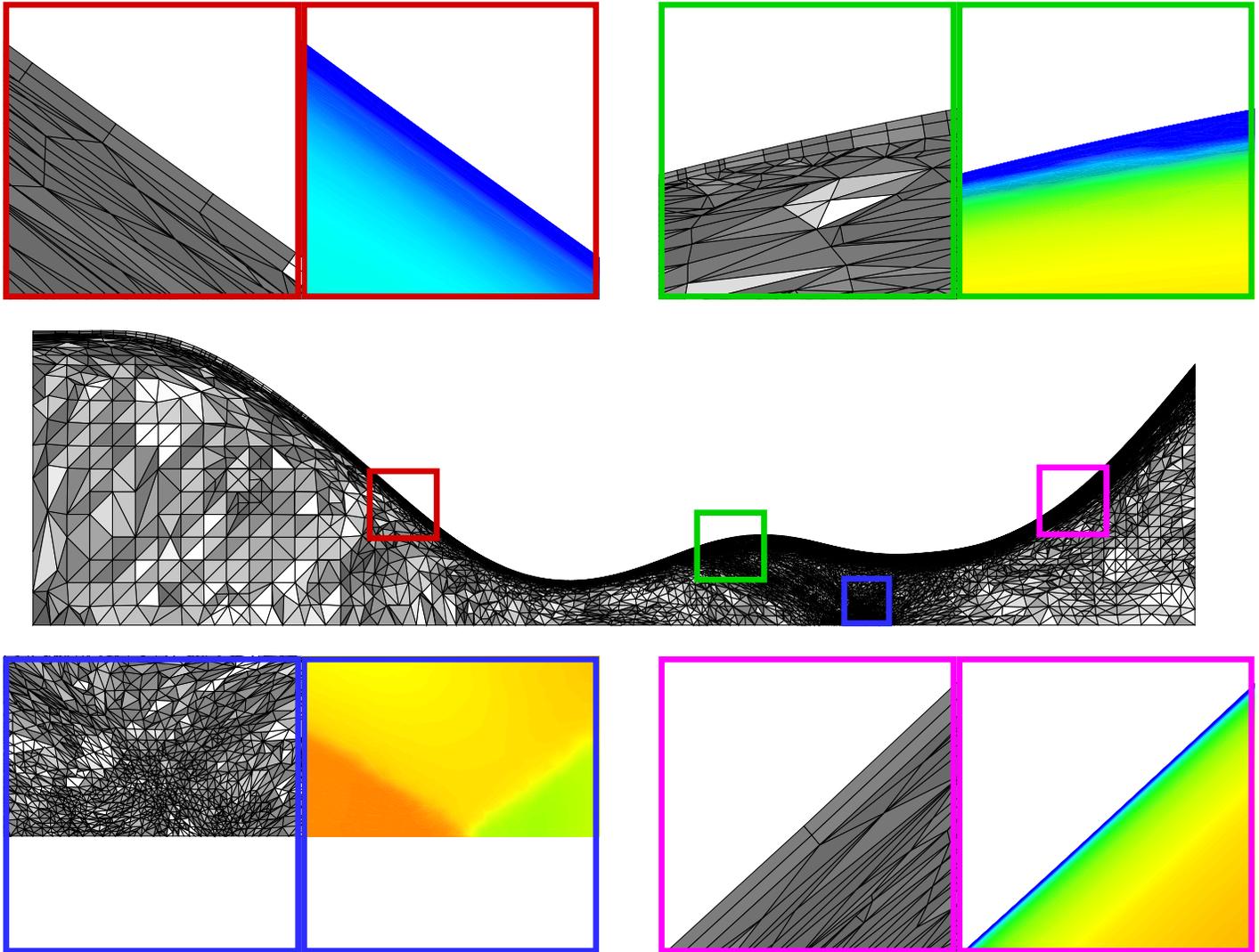
**Fig. 33** Computed solution fields for double throat nozzle on adapted boundary layer mesh (windows correspond to magnified view near the upper wall within the throats that show flow separation locations).

aligned anisotropic elements at shock waves, isotropic elements near the shock wave reflection point, high-aspect ratio layered elements where thin boundary layers form and varying layer thicknesses near the separation zone.

## 6 Closing remarks

In this article, we have presented an adaptive boundary layer meshing procedure for viscous flow simulations. The key to this approach is to maintain layered and graded elements near the wall as the mesh is modified based on the computed size field information. The approach adapts both the in-plane and normal resolution of the boundary layer elements near the walls. In this process it creates elements with smooth transition at the interface between layered and unstructured tetrahedral part of the mesh. The procedure presented also possesses the capacity to handle curved geometries of interest involving steep folding angles like bifurcations.

We demonstrated the effectiveness of the boundary layer meshes for viscous flows by considering two type of problem cases: one involved blood flow simulations in the cardiovascular system and in other we simulated high-speed flow arising in the field of aerodynamics. We illustrated that key wall quantities of interest like wall shear stress obtained on boundary layer meshes are more accurate and characterized by less fluctuations than the ones computed on fully unstructured, anisotropic meshes. We also showed that viscous flow features of boundary layer separation and shock-wave/boundary-layer interactions are well captured by using boundary layer mesh adaptivity.



**Fig. 34** A collection of mesh faces cut by the center-plane on adapted boundary layer mesh of a double throat nozzle (windows correspond to magnified views of the adapted mesh along with the computed Mach number).

## 7 Acknowledgments

We gratefully acknowledge the support of this work by NSF grant ACI-0205741. We would also like to acknowledge that the solutions presented herein for blood flow simulations made use of the linear algebra library provided by AcuSim Software.

## References

1. O. Sahni, J. Müller, K. E. Jansen, M. S. Shephard, C. A. Taylor, Efficient anisotropic adaptive discretization of the cardiovascular system, *Comp. Meth. Appl. Mech. Engng.* 195 (2006) 5634–5655.
2. S. Pirzadeh, Unstructured viscous grid generation by the advancing-layers method, *AIAA Journal* 32 (1994) 1735–1737.
3. B. M. Connell SD, Semi-structured mesh generation for three-dimensional Navier-Stokes calculations, *AIAA Journal* 33 (1995) 1017–1024.
4. R. V. Garimella, M. S. Shephard, Boundary layer mesh generation for viscous flow simulations, *Int. J. Numer. Meth. Engng.* 49 (2000) 193–218.
5. Y. Ito, K. Nakahashi, Unstructured mesh generation for viscous flow computations, in: *Proc. 11th International Meshing Roundtable*, Ithaca, NY, USA, 2002.
6. K. E. Jansen, M. S. Shephard, M. W. Beall, On anisotropic mesh generation and quality control in complex flow problems, in: *Proc. 10th International Meshing Roundtable*, Newport Beach, CA, USA, 2001.
7. A. Khawaja, Y. Kallinderis, Hybrid grid generation for turbomachinery and aerospace applications, *Int. J. Numer. Meth. Engng.* 49 (2000) 145–166.
8. R. Löhner, J. Cebral, Generation of non-isotropic unstructured grids via directional enrichment, *Int. J. Numer. Meth. Engng.* 49 (2000) 219–232.
9. J. Peraire, M. Vahdati, K. Morgan, O. C. Zienkiewicz, Adaptive remeshing for compressible flow computations, *Journal of Computational Physics* 72 (1987) 449–466.
10. G. C. Buscaglia, E. A. Dari, Anisotropic mesh optimization and its application in adaptivity, *Int. J. Numer.*

- Meth. Engng. 40 (1997) 4119–4136.
11. M. J. Castro-Díaz, F. Hecht, B. Mohammadi, O. Pironneau, Anisotropic unstructured mesh adaption for flow simulations, *Int. J. Numer. Meth. Fluids* 25 (1997) 475–491.
  12. P. J. Frey, F. Alauzet, Anisotropic mesh adaptation for CFD computations, *Comp. Meth. Appl. Mech. Engng.* 194 (2005) 5068–5082.
  13. X. Li, M. S. Shephard, M. W. Beall, 3D anisotropic mesh adaptation by mesh modifications, *Comp. Meth. Appl. Mech. Engng.* 194 (2005) 4915–4950.
  14. X. Li, M. S. Shephard, M. W. Beall, Accounting for curved domains in mesh adaptation, *Int. J. Numer. Meth. Engng.* 58 (2003) 247–276.
  15. J. Müller, O. Sahni, X. Li, K. E. Jansen, M. S. Shephard, C. A. Taylor, Anisotropic adaptive finite element method for modelling blood flow, *Comp. Meth. Biomech. Biomed. Eng.* 8 (2005) 295–305.
  16. C. C. Pain, A. P. Umpleby, C. R. E. de Oliveira, A. J. H. Goddard, Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations, *Comp. Meth. Appl. Mech. Engng.* 190 (2001) 3771–3796.
  17. J. F. Remacle, X. Li, M. S. Shephard, J. E. Flaherty, Anisotropic adaptive simulation of transient flows using Discontinuous Galerkin methods, *Int. J. Numer. Meth. Engng.* 62 (2005) 899–923.
  18. A. Khawaja, T. Minyard, Y. Kallinderis, Adaptive hybrid grid methods, *Comp. Meth. Appl. Mech. Engng.* 189 (2000) 1231–1245.
  19. Y. Kallinderis, C. Kavouklis, A dynamic adaptation scheme for general 3-D hybrid meshes, *Comp. Meth. Appl. Mech. Engng.* 194 (2005) 5019–5050.
  20. R. V. Garimella, Anisotropic tetrahedral mesh generation, Ph.D. thesis, Rensselaer Polytechnic Institute (May 1999).
  21. L. A. Freitag, C. Ollivier-Gooch, Tetrahedral mesh improvement using swapping and smoothing, *Int. J. Numer. Meth. Engng.* 40 (1997) 3979–4002.
  22. J. P. Ku, M. T. Draney, F. R. Arko, W. A. Lee, F. P. Chan, N. J. Pelc, C. K. Zarins, C. A. Taylor, In vivo validation of numerical prediction of blood flow in arterial bypass grafts, *Ann. Biomed. Eng.* 30 (2002) 743–752.
  23. C. A. Taylor, T. J. R. Hughes, C. K. Zarins, Finite element modeling of blood flow in arteries, *Comp. Meth. Appl. Mech. Engng.* 158 (1998) 155–196.
  24. C. A. Taylor, M. Draney, J. Ku, D. Parker, B. Steel, K. Wang, C. Zarins, Predictive medicine: computational techniques in therapeutic decision-making, *Computer Aided Surgery* 4 (5) (1999) 231–247.
  25. G. R. Stuhne, D. A. Steinman, Finite-element modeling of the hemodynamics of stented aneurysms, *Trans. ASME J. Biomech. Engrg.* 126 (3) (2004) 382–387.
  26. M. O. Bristeau, R. Glowinski, J. Periaux, H. Viviand, Presentation of problems and discussion of results, in: M. O. Bristeau, R. Glowinski, J. Periaux, H. Viviand (Eds.), *Numerical Simulation of Compressible Navier-Stokes Flows*, Notes on Numerical Fluid Mechanics, Vol. 18, Vieweg, 1987, pp. 1–40.
  27. C. A. Figueroa, I. E. Vignon-Clementel, K. E. Jansen, T. J. R. Hughes, C. A. Taylor, A coupled momentum method for modeling blood flow in three-dimensional deformable arteries, *Comp. Meth. Appl. Mech. Engng.* 195 (2006) 5685–5706.
  28. A. N. Brooks, T. J. R. Hughes, Streamline upwind / Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations, *Comp. Meth. Appl. Mech. Engng.* 32 (1982) 199–259.
  29. C. H. Whiting, K. E. Jansen, A stabilized finite element method for the incompressible Navier-Stokes equations using a hierarchical basis, *Int. J. Numer. Meth. Fluids* 35 (2001) 93–116.
  30. K. E. Jansen, C. H. Whiting, G. M. Hulbert, A generalized- $\alpha$  method for integrating the filtered Navier-Stokes equations with a stabilized finite element method, *Comp. Meth. Appl. Mech. Engng.* 190 (1999) 305–319.
  31. F. Shakib, <http://www.acusim.com>.
  32. J. Womersley, Method for the calculation of velocity, rate of flow and viscous drag in arteries when the pressure gradient is known, *J. Physiol. (London)* 127 (1955) 553–563.
  33. I. E. Vignon-Clementel, C. A. Figueroa, K. E. Jansen, C. A. Taylor, Outflow boundary conditions for three-dimensional finite element modeling of blood flow and pressure in arteries, *Comp. Meth. Appl. Mech. Engng.* 195 (2006) 3776–3796.
  34. T. J. R. Hughes, L. P. Franca, M. Mallet, A new finite element formulation for computational fluid dynamics: I. Symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics, *Comp. Meth. Appl. Mech. Engng.* 54 (1986) 223–234.
  35. C. H. Whiting, K. E. Jansen, S. Dey, Hierarchical basis in stabilized finite element methods for compressible flows, *Comp. Meth. Appl. Mech. Engng.* 192 (2003) 5167–5185.
  36. T. J. R. Hughes, L. P. Franca, M. Mallet, A new finite element formulation for computational fluid dynamics: IV. A discontinuity-capturing operator for multidimensional advective-diffusive systems., *Comp. Meth. Appl. Mech. Engng.* 58 (1986) 329–336.
  37. Y. Saad, M. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM Journal of Scientific and Statistical Computing* 7 (1986) 856–869.