Noname manuscript No.
(will be inserted by the editor)

Advances in Co–Volume Mesh Generation and Mesh Optimisation Techniques

Sean Walton · Oubay Hassan · Kenneth Morgan

Received: date / Accepted: date

Abstract This paper introduces developments in modified techniques for the generation of unstructured, non–uniform, dual orthogonal meshes which are suitable for use with co–volume solution schemes. Two new mesh generation techniques, a modified advancing front technique and an octree–Delaunay algorithm, are coupled with a mesh optimisation algorithm. When using a Delaunay–Voronoi dual, to construct mutually orthogonal meshes for co–volume schemes, it is essential to minimise the number of Delaunay elements which do not contain their Voronoi vertex. These new techniques provide an improvement over previous approaches, as they produce meshes in which the number of elements that do not contain their Voronoi vertex is reduced. In particular, it is found that the optimisation algorithm, which could be applied to any mesh cosmetics problem, is very effective, regardless of the quality of the initial mesh. This is illustrated by applying the proposed approach to a number of complex industrial aerospace geometries.

Keywords Co–volume · mesh generation · mesh optimisation · dual orthogonal meshes · marker and cell scheme · Yee scheme

S. Walton
College of Engineering, Swansea University Swansea SA2 8PP, Wales, UK
E-mail: s.p.walton@swansea.ac.uk

O. Hassan
College of Engineering, Swansea University Swansea SA2 8PP, Wales, UK
E-mail: o.hassan@swansea.ac.uk

K. Morgan
College of Engineering, Swansea University Swansea SA2 8PP, Wales, UK
E-mail: k.morgan@swansea.ac.uk

1 Introduction

Computational methods are widely employed in a variety of different application areas. For practical simulations, the requirement of modelling complex geometries means that unstructured mesh methods are particularly attractive, as fully automatic unstructured mesh generation procedures are now widely available. Popular algorithms implemented on unstructured meshes fall into the categories of finite volume and finite elements methods. These algorithms are, generally, low order and often require a significant computational resource to perform accurate simulations involving industrial geometries. In contrast, co–volume techniques exhibit a high degree of computational efficiency, in terms of both CPU and memory requirements, on structured dual orthogonal Cartesian meshes. The marker and cell (MAC) algorithm [3] and the Yee scheme [21] are examples of co–volume methods that have been widely employed for the solution of the Navier Stokes and Maxwell's equations respectively. A basic requirement for the successful implementation of the co–volume scheme is the existence of two, high quality, mutually orthogonal meshes. For an unstructured mesh implementation, the obvious dual mesh choice is the Delaunay–Voronoi diagram. Detailed mesh requirements for co–volume schemes are presented in section 1.1. Despite the fact that real progress has been achieved in unstructured mesh generation methods over the last two decades, generating suitable co–volume meshes in complex shaped domains is still an open problem. This is due to the difficulties encountered when attempting to generate sufficiently smooth, non–uniform, high quality dual meshes for such problems. Standard mesh generation methods are designed to create high quality Delaunay triangulations, but do not attempt to provide a high quality dual Voronoi mesh. Previous attempts at solving the problem of co–volume mesh generation are discussed in section 1.2. These techniques can only produce uniform meshes of suitable quality.

For the simulation of geometries that contain regions with high curvature and singularities, non–uniform meshes are often used to capture the complex variation of the solution field. In such cases, the quality of the generated elements depends upon the gradation of the spacing function. This paper presents a number of new techniques designed to generate non–uniform meshes for co-volume solvers. We start from a Delaunay triangulation of the boundary surface and split the generation of the domain into a region adjacent to the boundaries and a free space region. In the free space region, two approaches are introduced to generate meshes and compared to the standard automatic Delaunay sub division method. The first approach is based upon the recursive insertion of ideal lattice points, that locally satisfy the demands of the spacing distribution function, into the Delaunay generated mesh of the boundary points. In the second scheme, an octree–Delaunay algorithm is utilised to generate meshes with properties that are close to those of the ideal mesh. In the region adjacent to the boundary, we introduce a modified advancing front technique, in which points are located in such a way that ideal meshes can be recovered in the case of a uniform mesh. Finally, the quality of the

resulting mesh is improved by the use of a mesh optimisation scheme. The feasibility of generating suitable meshes in this manner, around increasingly complex geometries, is presented. To demonstrate the validity of the meshes, we present numerical examples of the scattering of electromagnetic waves by a complex 3D object. This example shows the efficiency and accuracy that can be achieved by a co–volume method utilising the proposed meshing scheme.

1.1 Co–Volume Mesh Requirements

Co–volume schemes were initially devised for structured grids and are applied to coupled system of equations such as the Maxwell equations and the Navier–Stokes equations. The unknowns are staggered in space, i.e some of the unknowns are evaluated at the centre of the cell and the other unknowns are evaluated on the edges of the cell. Finite difference approximation is then used and the solution is advanced in time in a staggered manner, with some of the unknowns evaluated at time n and the others evaluated at time $n + 1$. This can be interpreted as using two structured grids which are staggered and orthogonal. On structured grids, a primal mesh is first generated to cover the domain. The dual mesh is then constructed by joining the centre of each cell. This construction guarantees the required orthogonality for finite difference schemes and ensures that the quality of the primal and dual meshes are identical.

The generalisation of the method to unstructured meshes utilises the Delaunay mesh as the primal mesh and its Voronoi mesh as the dual [9]. The Voronoi vertices are the circumcentres of the corresponding Delaunay elements, which guarantees the required orthogonality. However, it is possible for the circumcentre to lie outside its corresponding element. In this case the staggering in space will not be guaranteed. In addition, to replicate the second order accuracy these schemes exhibit on structured grids, the Delaunay and Voronoi grids must intersect each other, i.e. the nodes of the dual mesh should coincide with the centroid of corresponding primal mesh and the dual mesh edges should pass through the centroids of the corresponding primal mesh faces [10]. In two and three dimensional meshes the Voronoi edge by definition intersect the Delaunay edge/face at its centroid, but this construction does not guarantee that the Delaunay face is a bisector of the corresponding Voronoi edge. Furthermore, the time step for an explicit scheme is directly proportional to the smallest edge of the primal and the dual meshes. While this length does not change in structured grids, in unstructured meshes, it is possible for Voronoi edges to vanish, i.e. for two or more elements to share the same circumcentre. Of these two requirements, ensuring that circumcentres lie inside their corresponding elements will be prioritised at the expense of mutual bisection. This is because elements with their circumcentre outside can cause stability problems if located in areas of high gradient in the solution field. Whereas a deviation from mutual bisection results in a local loss of second order accuracy.

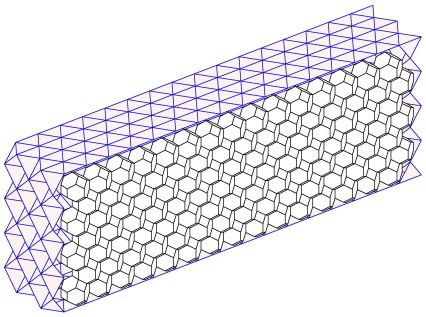


Fig. 1 Detail of a mesh of ideal tetrahedral elements, showing the surface Delaunay faces and the internal Voronoi cells.

Only a mesh made up of equilateral tetrahedra, in which all faces are equilateral triangles, guarantees the quality of the primal and dual meshes that is found in structured grids. However, in contrast to two dimensions, the equilateral tetrahedron is not a space filling element. Hence, a three dimensional analogue of a space filling ideal mesh will consists of equal non–equilateral tetrahedra. Each face in such a mesh will be an isosceles triangle, with one side of length l_D^{long} and two shorter sides of length $l_D^{\text{short}} = \sqrt{3}/2\, l_D^{\text{long}}$. Six such tetrahedra form a parallelepiped tiling the space, as illustrated in Figure 1. It can be shown that this configuration maximises the length of the minimum Voronoi edge for a fixed element size. All Voronoi edges have the same length $l_V \approx 0.38\,\delta$ where $\delta \equiv \langle l_D \rangle = (3\, l_D^{\text{long}} + 4\, l_D^{\text{short}})/7 \approx 0.92\, l_D^{\text{long}}$. This configuration guarantees that every node has a connectivity index of 14 and each vertex has a maximum angle of 70.5°.

However, the fact some of the element dihedral angles are equal to 90°, makes the element very sensitive to deformation, which become a barrier for the generation of non–uniform meshes. In the case of non–uniform meshes, in order to minimise the deviation of the circumcentre from the barycentre, the requirement that the dual edge has to cross the corresponding element face at the centroid has to be relaxed. This enables the use of the power diagram to locate a new circumcentre, based on a weight associated with each of the element nodes [19]. Furthermore, co–volume schemes are not restricted to using tetrahedral elements, which means that tetrahedral elements can be automatically merged into a polyhedron if the length of the dual edge is below a user specified tolerance. This will reduce the number of elements that have circumcentre located outside their elements and remove any short Voronoi edges that reduce the stability constraint of the scheme.

1.2 Existing Mesh Generation Techniques

To date, most mesh generation software is aimed at generating high quality primal meshes and pays no attention to the quality of the dual mesh. To ensure

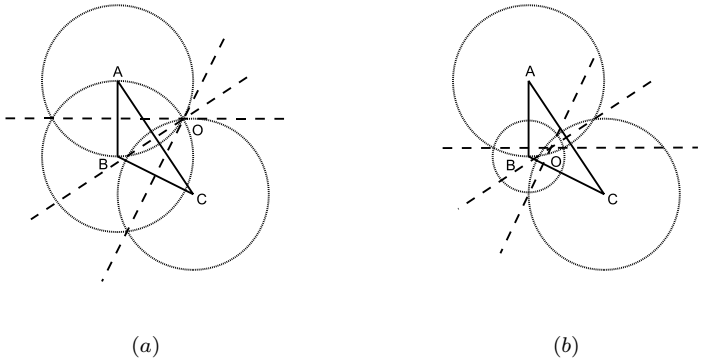
that the primal mesh and the dual mesh are staggered in space, it is essential to ensure that each Delaunay element contains its Voronoi vertex. A mesh which is staggered in space is termed well centred, as every Delaunay element contains its circumcentre [13]. In two dimensions, this can be achieved by utilising mesh optimisation techniques that are designed to eliminate obtuse triangles [10,11, 15]. However, the extension of these techniques into three dimensions is not trivial and does not guarantee the desired outcome.

To address the problem of generating suitable meshes for co–volume techniques, a stitching method was proposed in two dimensions [9]. In this approach, the problem of triangulating a domain of complicated shape is split into a set of relatively simple local triangulations. Each local mesh is constructed with properties which are close to those of an ideal mesh and these local meshes are combined, to form a consistent mesh, using a stitching algorithm. The quality of the stitched mesh is improved by the use of standard mesh cosmetics.

Although the stitching method was successful in generating two dimensional non–uniform meshes suitable for co–volume methods, its extension into three dimensions has proved to be troublesome. The method requires the layer by layer generation of high quality elements near the solid boundary, utilising a modified advancing front technique. Using a two dimensional stitching method, the required quality was maintained by gradually deforming an ideal equilateral mesh in free space. The three dimensional ideal meshes are very sensitive to grading and hence, satisfying spacing distribution functions that have more than a 10% linear variation is difficult [10].

To minimise the deviation of the circumcentre from the element centroid, the requirement that the dual edges have to cross the element faces at the centroid of the face must be relaxed. This enables the use of the power diagram, to locate a new circumcentre based on a weight associated with each of the element nodes [19]. The resulting weighted Voronoi diagram allows the movement of the Voronoi nodes while maintaining mutual orthogonality. The process of using the weighted Voronoi diagram is illustrated in Figure 2. An obtuse triangular element ABC, with its Voronoi node O initially sitting on its circumcentre outside the element is shown. The point O is located by creating circles of equal radius on each Delaunay node and finding the point of intersection of their common chords. By reducing the radius of the circle at B, the point of intersection is pulled into the element. The reduced radius at B moves the Voronoi nodes of the adjacent elements to maintain orthogonality. The weight at any node is defined as the change in radius at that point. The weights on each node are optimised as part of an optimisation process.

Co–volume schemes are not restricted to tetrahedral elements and adjacent tetrahedra can be automatically merged into a polyhedron, if the length of the dual edge is below a user specified tolerance. Typically, this tolerance is set to 10% of the local ideal Voronoi edge length, l_V, interpolated from the spacing distribution function. This will reduce the number of elements that have their circumcentres located outside the element and remove any short



(*a*) (*b*)

Fig. 2 The weighted Voronoi diagram. By intersecting the common chords of circles centred at each vertex we find the Voronoi vertex O. In (a) these circles have the same radius and therefore O lies on the circumcentre of the element. To pull O inside the element, the radius at B is reduced in (b).

Voronoi edges that reduce the stability constraints of the scheme. Examples of merged elements are shown in Figure 3.

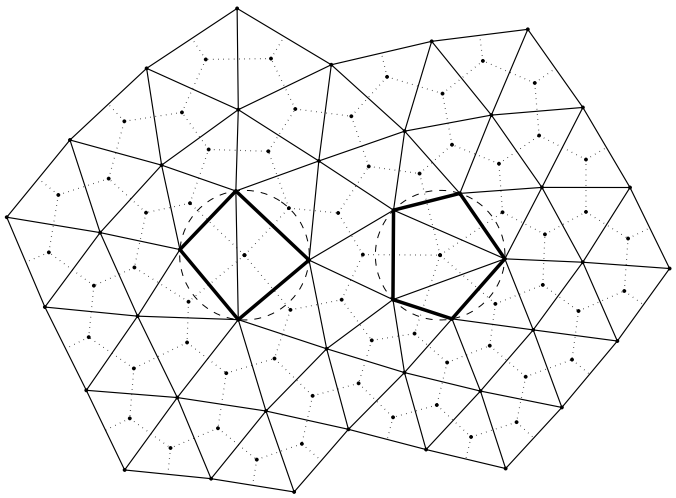


Fig. 3 Examples of merged elements, circled and edges plotted in bold, in a co–volume mesh.

2 Methodology

The central idea of this paper is to develop techniques specifically targeted at generating suitable meshes for co-volume schemes, therefore meeting the requirements of the solvers. In all cases the methods developed are designed to ensure the elements generated contain their circumcentre. This is done by examining and identifying patterns in the uniform ideal mesh described above.

The starting point of three dimensional volume mesh generation is a suitable surface triangulation that complies with the required mesh spacing function. In the current work, the spacing function is specified using a combination of background mesh, point, line and triangular sources [18]. Two dimensional surface meshing based on the advancing front technique is now well established. The generation of surface meshes that satisfy the requirement of the co–volume scheme can be routinely achieved.

The method adopted here for the generation of the three dimensional meshes is based upon the use of the Delaunay triangulation, with a modified point creation technique that enables the generation of elements that meet the quality required for co–volume methods. The point creation employed distinguishes between the near field region, that is close to the surface, and the free space region. The complete algorithm can be summarised as follows:

1. Generate a volume mesh by connecting the boundary points using the Delaunay criterion.
2. For the desired number of layers, create a near field set of points and insert them into the existing Delaunay mesh.
3. Create points in the free space and insert them into the Delaunay mesh
4. Apply mesh enhancement technique to improve the quality of the resulting mesh.

The initial volume mesh is a constrained Delaunay mesh of the boundary points that ensures the presence of all surface triangles. For the generation of the points in the near field region, a modified advancing front technique has been developed. Two methods have been considered for creating points within the free space region. The first method is based upon the generation of local lattice points, which resembles the distribution of an ideal mesh for a given spacing. The second method utilises an octree at the desired spacing to position nodes in the free space region.

2.1 Near Field Point Generation by a Modified Advancing Front Technique

Advancing front techniques are characterised by marching the volume mesh into empty space from the surface mesh by sequentially generating points and elements [7,6]. In the case of tetrahedral elements, this traditionally consists of placing a new node above each face on the surface to create an equilateral tetrahedron. However, in the ideal mesh shown in Figure 1 nodes lie above edges, not faces, of the elements below. The technique adopted here is to

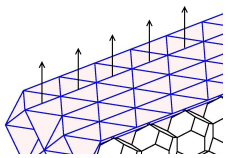


Fig. 4 Detail of a mesh of ideal tetrahedral elements, showing the surface Delaunay faces and the internal Voronoi cells. The arrows indicate where the next layer of nodes would be placed.

advance the generation of nodes from edges rather than from faces. This is done to recreate the structure of the ideal mesh for co–volume schemes.

To construct an additional layer on top of the ideal mesh shown in Figure 1, a single node advances from each pair of surface triangles. This is illustrated for a few edges in Figure 4. This means that, for a general surface, a method is required of identifying the set of edges from which nodes will be advanced. The edges can be found by pairing faces, such that each face only belongs to one pair. Pairing in this fashion is also encountered when merging triangular meshes into quadrilateral meshes. Remacle et al. [8] showed that this is a maximal matching problem, which can be solved using blossom type algorithms [2]. When this list of edges has been obtained, the next step is to decide where to place the node above each edge. VanderZee et al. [13] have carried out extensive work in investigating conditions for well centred elements. These are used as conditions to place nodes during the advancing front stage. Given a single face, the acceptable region, in which a node may be placed to create a well centred element, can be identified. This region is constructed by finding the reflection of the face through its own circumcentre, as shown in Figure 5(a). A prism is created by extruding the reflected face, infinitely in both directions, and part of the volume of this prism is cut out by the sphere whose equator is the circumcircle of the original non reflected face [13]. This is illustrated, in a side view, in Figure 5(b), where a point placed in the region R produces a well centred tetrahedron when connected with the remaining nodes.

With this in mind, a node located above an edge must sit within this region for both triangles connected to that edge. Assuming that the dihedral angle of the two faces allows the regions to cross, this position can be calculated as the mid point between two points found by advancing from the centroid of the two reflected elements. Using Figure 6 to illustrate the notation used, the region in which a node above an edge BC, which connects triangle ABC to BDC, may be located is determined as follows:

1. Calculate the inward normal of the triangles ABC and BDC, $\hat{\mathbf{n}}^{ABC}$ and $\hat{\mathbf{n}}^{BDC}$ respectively.
2. Construct the reflections, $A'B'C'$ and $B'D'C'$, of both triangles through their circumcentres.
3. Calculate the circumradius, R^{ABC} and R^{BDC}, of ABC and BDC respectively.

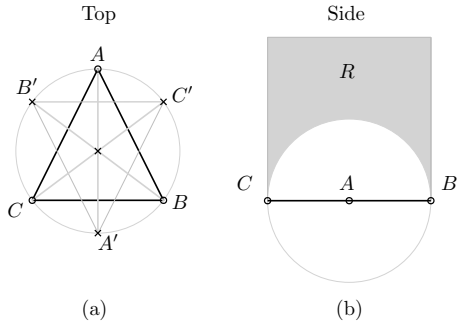


Fig. 5 (a) Given the face ABC the volume in which a point can be placed to construct a well centred tetrahedron is found as follows. Take the reflection of ABC through its own circumcentre, $A'B'C'$. The volume a point can be placed in is the prism, extruded from $A'B'C'$, with the sphere whose equator is the circumcircle of ABC cut out of it. (b) shows a side view of the resulting region R. (a) and (b) are not drawn to scale.

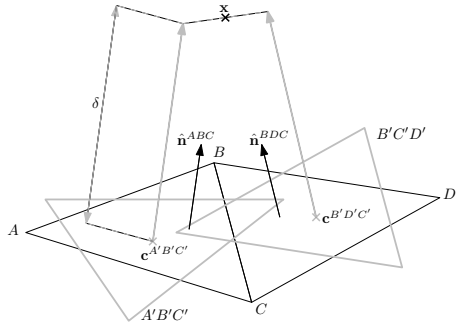


Fig. 6 Diagram showing the modified advancing front procedure for the edge BC.

4. Calculate the circumcetres, $\mathbf{c}^{A'B'C'}$ and $\mathbf{c}^{B'D'C'}$, of $A'B'C'$ and $B'D'C'$ respectively.
5. The distance to advance the front is calculated as $\delta = 1.1 \times \max(R^{ABC}, R^{BDC})$
6. The new node position is then given by

$$\mathbf{x} = 0.5 \times (\mathbf{c}^{ABC} + \mathbf{c}^{BDC} + \delta(\hat{\mathbf{n}}^{ABC} + \hat{\mathbf{n}}^{BDC})) \tag{1}$$

The process of determining which edges to advance from is as follows:

1. Establish the graph of the edges of the triangular surface mesh.
2. For each edge in this graph, permanently eliminate edges where the feasible regions of the adjacent faces do not cross.
3. Apply the blossom algorithm to maximise the number of paired elements in the resulting graph.
4. Store the common edge of paired elements into a list of edges from which to advance.

Any face ABC which is not present in the resulting graph, i.e. does not have an edge from which a point was advanced, is considered separately by

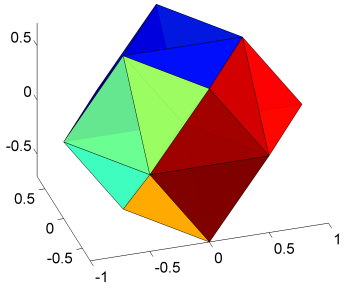


Fig. 7 A single point from the lattice of Figure 1, with its 14 neighbours lying either on the sphere of radius $\sqrt{3}/2$ or on the sphere of radius $1/\sqrt{2}$.

advancing, in the direction of the normal to ABC, from the circumcentre of its reflection ABC', by a distance $\delta = 1.1R^{ABC}$. This means every face should have a line of sight to an advancing node.

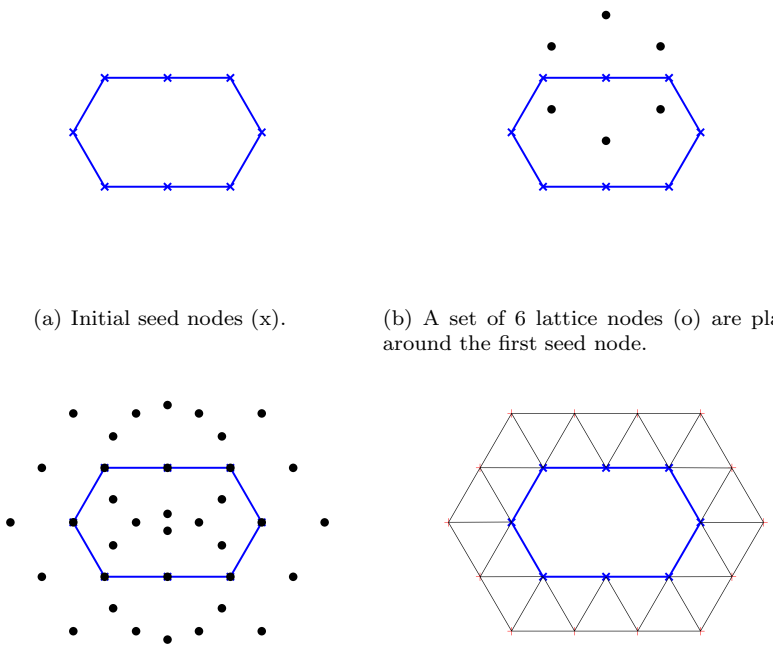
The points generated in this way, for a complete layer, are inserted into the constrained Delaunay triangulation. Points which lie close to each other, or to existing mesh points, are automatically rejected by the Delaunay kernel implementation presented in [18].

2.2 Free Space Lattice Point Generation

The motivation of utilising a lattice point generation technique [16] is to recreate, as closely as possible, the ideal mesh shown in Figure 1 for a non–uniform spacing function. In the ideal mesh, a repeating lattice can be identified, in which each node is connected to 14 neighbouring nodes and each tetrahedron, shown in Figure 7, in the lattice contains its circumcentre.

Figure 8 illustrates the process in two dimensions where the ideal mesh is of equilateral triangles, resulting in 6 lattice points. The technique starts with a set of seed nodes as shown in Figure 8(a). In the three dimensional case the seed nodes are not the boundary, but the set of nodes inserted using the modified advancing front method described above. With these seed nodes located and triangulated, the following steps are followed:

1. For each seed node:
   (a) Calculate the local spacing δ at the node.
   (b) Generate the lattice nodes around the seed node at the local spacing δ, illustrated in Figure 8(b).
   (c) For each of the new lattice nodes:
      i. If the node lies inside the computational domain and at a distance greater than δ_m from an existing boundary node, add the node to the list $\mathbf{P}_t$
      ii. Merge it with an existing node in $\mathbf{P}_t$, if the distance between them is less than δ_m.



(a) Initial seed nodes (x).

(b) A set of 6 lattice nodes (o) are placed around the first seed node.

(c) The set of nodes resulting from placing lattice nodes at each seed node.

(d) The nodes which were not deleted are now triangulated.

Fig. 8 A two dimensional illustration of the lattice point generation technique.

2. Insert the nodes in $\mathbf{P}_t$ into the mesh using Delaunay point insertion. This is shown in Figure 8(d).
3. Treat the nodes in $\mathbf{P}_t$ as seed nodes and repeat steps 1 and 2 to create $\mathbf{P}_{t+1}$. Keep repeating until a set of seed nodes results in a empty $\mathbf{P}_t$.

In the current implementation, the value of δ_m is taking to be 50% of the local desired spacing. Due to the large number of potential points in $\mathbf{P}_t$, the use of the Delaunay kernel alone to identify points that are close to existing points proved to be inefficient. A KD–tree library is employed for spatial searching. This enables searches for neighbours within a given range to be performed in $O(\log N)$ operations, where N denotes the total number of nodes [4]. Utilising a 3D local axis system for each seed point that has one of its axis aligned with the normal to the triangulation of the previous layer and a second axis aligned with the projection of one edge connected to the seed point onto the tangent plane at the point, produces the best quality local lattice configuration.

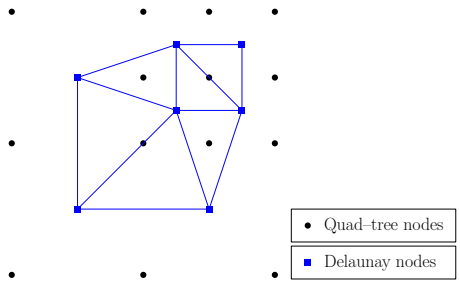


Fig. 9 Illustration of the octree point generation technique performed on a two dimensional quad–tree. The Delaunay nodes are triangulated and the quad-tree nodes are not.

2.3 Free Space Octree Point Generation

An alternative technique for creating a suitable mesh in the free space region is based on a combination of octree space subdivision and Delaunay point insertion. This algorithm first creates an octree that complies with the given spacing distribution function. This tree is then used as a template in which nodes are placed at the centroid of each octant. The Delaunay triangulation of those nodes results in a mesh of tetrahedra which can be merged to form alternating layers of cubes and pyramids with hexahedra. A similar approach is presented by Yang et al. [20]. They use the octree space division as the basis for a face centred cubic crystalline lattice structure. In their structure, points are not only placed at the centroid of each octant, but also at the vertices. We found that approach resulted in fewer suitable quality elements, for co–volume techniques, than the one presented here.

A two dimensional diagram illustrating the technique is shown in Figure 9. In two dimensions the quad–tree is constructed at the user specified spacing. Delaunay nodes are then inserted at the centroid of each quad and triangulated. In the figure, the two right angled triangles would be merged into a quadrilateral and the remaining triangles contain their circumcentres. Regardless of the spacing function specified the construction of the quad–tree restricts the gradation such that the structure of elements illustrated appears. Essentially the quad–tree gives us the building blocks to change element size without compromising quality, at the expense of compromising user specified spacing. In three dimensions the situation is complicated by the possible existence of hanging nodes at the centroid of octant faces. These hanging nodes do not themselves appear in the final set of mesh nodes, but the octant structure around hanging nodes in the octree results in elements which do not contain their circumcentres.

Figure 10 shows a cut through a three dimensional mesh created in the manner described above. The target spacing function was a single point source at the centre of the volume. At the end of the division process, the octree contains no hanging nodes at octant faces and, after merging, all the elements are well centred. Figure 11 shows a cut through a second mesh created in the

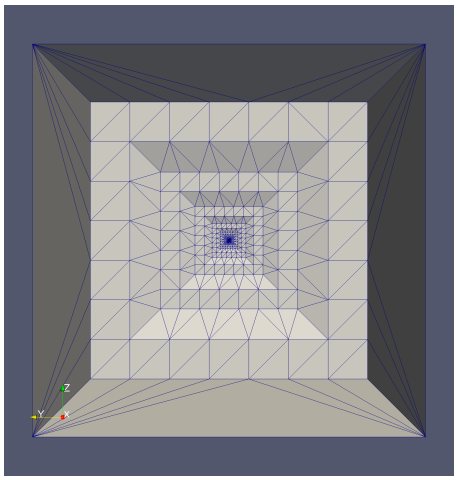


Fig. 10 A single point source with no boundary surface mesh. The mesh was generated using the octree Delaunay technique. The connections to the initial hull are also shown. After merging, all elements contain their circumcentres.

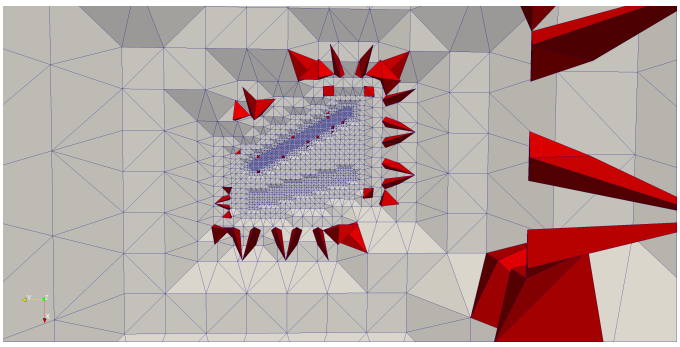


Fig. 11 A collection of sources with no boundary surface mesh. The mesh was generated using the octree Delaunay technique. The connections to the initial hull are also shown. The red solid elements do not contain their circumcentres.

same manner. The target spacing function is now a group of sources designed for meshing an ONERA M6 wing. In this case, hanging nodes exist in the octree, which leads to 0.4% non well centred elements in the final volume. It is possible to create completely well centred meshes by subdividing octants with hanging nodes. This has the disadvantage of increasing the number of elements which do not conform to the target spacing function. In some applications, this may be an option but, in many industrial applications, the resulting number of elements would be prohibitively large. It will be shown, however, that the mesh optimisation procedure is able to fix most of these elements.

2.4 Mesh Optimisation

Once the near field points are generated using the modified advancing front technique and free space points generated by one of the two other techniques, the mesh will be improved using optimisation techniques. Traditional mesh cosmetic techniques, such as edge splitting, edge collapsing and edge swapping, have been developed to improve the quality of Delaunay meshes. For co–volume schemes, a number of potential mesh cosmetics strategies exist. Lloyd's algorithm [5,1] has been shown to be ineffective in generating a completely well centred mesh, with around 10% of elements remaining non compliant, in the case of uniform meshes [19], and around 30%, in the case of non uniform meshes [15]. In an attempt to achieve a well centred mesh, VanderZee et al. [11] employed a minimisation approach, targeted directly at the problem. The effectiveness of this method has only been illustrated for simple three dimensional cases, such as cubes [12]. In this work, we apply an extension of the work presented by Walton et al. [15]. We direct the interested reader to that paper for the details of the optimisation algorithm. In the remainder of this section, we will briefly describe the algorithm, adding detail where this approach differs from the one presented previously.

2.4.1 Objectives of the Mesh Optimisation Procedure

The primary objective of the mesh optimisation presented here is to enhance the mesh by reducing the number of elements which do not contain their Voronoi vertex. When developing the mesh generation techniques, the aim was to produce elements which contained their circumcentre. This is because, at the generation stage, the Voronoi vertex coincides with the circumcentre. At the optimisation stage, we will be moving the Voronoi vertex using the power diagram, as described in section 1.2, such that it does not lie on the circumcentre. Co–volume techniques permit the use of polyhedral cells that satisfy the orthogonality condition. Voronoi edges that have a length close to zero will be removed by cell merging. The optimisation procedure must also take this into account. These requirements will be expressed in terms of objective function, which the optimisation algorithm will attempt to minimise.

2.4.2 The Objective Function

An obvious objective function to achieve our aim would be the distance between the Voronoi vertex and centroid of each element. Since by definition the centroid lies inside an element, moving the Voronoi vertex towards it will pull it inside. The key aim is to ensure the Voronoi vertex lies inside the element, without necessarily ensuring that it is located at the centroid. When this key aim is achieved, the objective function is set to zero. It is also set to zero for merged elements. Often, it is essential to prioritise the elements near boundaries, where the solution is expected to vary rapidly. An index is assigned to each element, based upon its distance from the boundary. This

index, which decreases with distance from the boundary, is used to scale the objective function.

The contribution of a particular element k to the objective function is defined as F_k, where

$$F_k = W_k L_k \|\mathbf{C}_k - \mathbf{V}_k\| \tag{2}$$

. Here, $\mathbf{C}_k$ and $\mathbf{V}_k$ are the locations of the centroid and Voronoi node of k respectively. W_k is a coefficient that takes the value zero, when $\mathbf{V}_k$ lies inside k or when k can be merged with an adjacent element, or one, otherwise. L_k is the index assigned to each element, based upon its distance from a boundary. The value of F_k may be regarded as a quality measure for element k.

2.4.3 The Optimisation Algorithm

The objective function constructed is non–smooth and non–differentiable. This is due to a combination of sudden jumps when Voronoi vertices enter elements or when elements are merged. Genetic type gradient–free optimisation algorithms possess the ability to effectively deal with such functions. The Modified Cuckoo Search (MCS) gradient–free optimisation technique [17] was developed with mesh optimisation in mind. The technique has been shown to be capable of creating completely well centred meshes in two dimensions [15].

The degrees of freedom in this mesh optimisation problem consist of the node positions and weights. This number can very quickly reach the order of millions, which motivates the separation of the problem into a series of local optimisations. The objective function is constructed by selecting the position and weight of a single node as degrees of freedom. The value of the objective function at a node is taken as the sum of F_k over each element, k, connected to that node, along with their neighbours to account for merging. This construction is illustrated in Figure 12. As a constraint, the fitness is assigned a large value, if one of these elements has a minimum dihedral angle less than 10 degrees. This ensures that flat and negative elements are avoided. This measure does not avoid tetrahedra, in the form of spires, which have poor aspect ratios. Spire elements do not pose a problem for co–volume solvers provided that the Voronoi vertex is pulled inside the element or the element itself is merged with an adjacent element.

The Modified Cuckoo Search has two key components, viz. exploration and exploitation. A population of agents, typically ten times the number of degrees of freedom, is generated. Each agent represents a potential set of node coordinates and weights. In the exploration step, a scaled random Lévy flight [14] is performed, from the locations of the group of agents with the worst values of F_k. These agents are replaced, by the new positions and values of F_k found after the random walk, and are called the discarded agents. In the exploitation step, a search is performed along a line between two random agents, selected from the group of agents with the best values of F_k. Agents in this elite group are only replaced if the newly found coordinates result in an improved value of F_k.

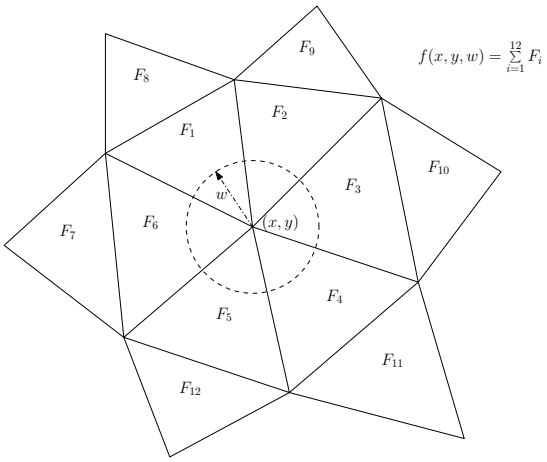


Fig. 12 A two dimensional illustration of how the objective function is constructed. Here the central node is to be optimised. The degrees of freedom are the node's coordinates (x, y) and weight w. The effected elements are drawn, the objective function $f(x, y, w)$ is then calculated by summing F_k over all these elements.

The order in which elements are optimised will be critical for success. There are two aims of the traversal strategy. Firstly, to ensure that the worst elements are optimised first and, secondly, to ensure that the system does not get trapped within a local minimum on subsequent iterations. Both aims are achieved using a binning scheme. The quality, F_k of each element is calculated and N_{bin} quality bins are created to group the elements. Treating the elements in order, starting with the worst quality bin, ensures that the worst elements are acted upon first. The order in which elements are treated within each bin is randomised, to avoid trapping the system into a local minimum by optimising elements in the same order on subsequent iterations. The number of bins is determined based on a minimum of 25 elements per bin and a maximum of 20 bins.

The full mesh optimisation procedure employed can be summarised as follows:

1. Perform edge splitting and edge collapse to remove any long or short edges that do not conform to the spacing distribution function. This will ensure that the mesh conforms to the spacing function after the near field and free space meshes are combined.
2. Perform an optimisation sweep
   (a) Identify all elements Ne_c with objective function not equal to zero.
   (b) Sort all elements with non–zero objective function, into N_{bin} bins, where $N_{bin} = \min(20, Ne_c/25)$.
   (c) Mark all nodes as unvisited.
   (d) For every bin, loop over the elements in the bin in a random manner. For each element, loop over the unvisited nodes and

   i. Generate the desired number of agents, i.e. 40 sets of coordinates and weights.
   ii. Employ the MCS to determine the coordinates of the node and the weight that minimise the objective function.
   iii. Mark the point as visited.
   iv. Compute the new objective function of all affected elements and adjust their bin number accordingly.
3. Perform swap diagonals to ensure the Delaunay criterion of maximising the minimum angle is met.
4. Return to step 2 if the user defined maximum number of iterations has not been reached.

The optimisation process described above is performed on a static connectivity. It is likely that, after node movement, a better quality connectivity might exist. At the end of each optimisation sweep, i.e. stage 3, the optimality of the connectivity is improved by swapping diagonals [18]. Elements with a minimum dihedral angle of less than 20 degrees are targeted for diagonal swapping, if an improvement can be made. The process is repeated until no further improvement is possible. In the examples presented in this paper, the complete procedure is iterated until either no change results in the mesh quality mesh or a user defined time limit is reached.

3 Results

To evaluate the effectiveness of the proposed procedures, the key measure is the percentage of elements that do not contain their Voronoi vertex. In addition, we will also present the distribution of the following quality measures:

- q_1: The minimum dihedral angle of each Delaunay element.
- q_2: The distance from the Delaunay element centroid to the Voronoi vertex, normalised by the inner radius of the element.
- q_3: The distance from the mid point of the Voronoi edge to the point on the edge which intersects the face, normalised by the local spacing.
- q_4: The distance from the intersection of the face by the Voronoi edge to the face centroid, normalised by the local spacing.
- q_5: The Voronoi edge lengths normalised by the local ideal Voronoi edge length, 0.38 times the mean Delaunay edge length of the element.

All meshes are produced using the constrained Delaunay triangulation [18] with four different point generation techniques:

- Automatic edge sub–division [18]. This technique will be referred to as Auto Delaunay (AD).
- Modified advancing front method for the first layer of elements followed by the automatic edge sub division. This technique is referred to as Auto Delaunay/Adv Front (ADAF).

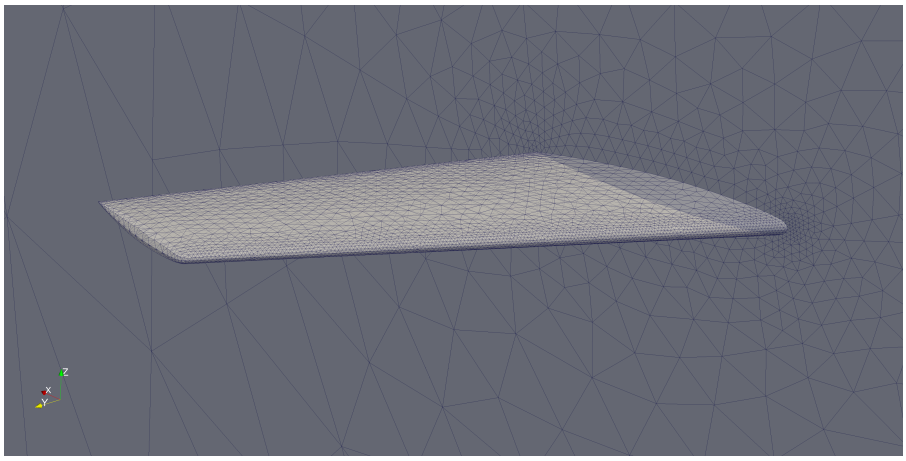
- Modified advancing front method for the first layer of elements followed by the lattice point insertion technique. This technique is referred to as Lattice/Adv Front (LAF).
- Modified advancing front method for the first layer of elements followed by the points created from the octree-Delaunay method. This technique is referred to as Octree/Adv Front (OAF).

In all cases, the boundary nodes are inserted using the Delaunay kernel and the boundary triangulation recovered. We have found that there is no advantage to generating more than one layer of elements using the advancing front technique. Subsequent nodes are inserted using the Delaunay kernel implementation presented in [18], which automatically rejects points which lie too close to the existing nodes in the mesh. The proposed mesh optimisation technique is then applied with a CPU time limit of 24 hours. To compare the proposed mesh optimisation technique to traditional methods, the percentage of elements not containing their Voronoi vertex is presented for meshes generated using automatic edge sub–division and the traditional cosmetics suite [18]. This is referred to as the traditional technique.

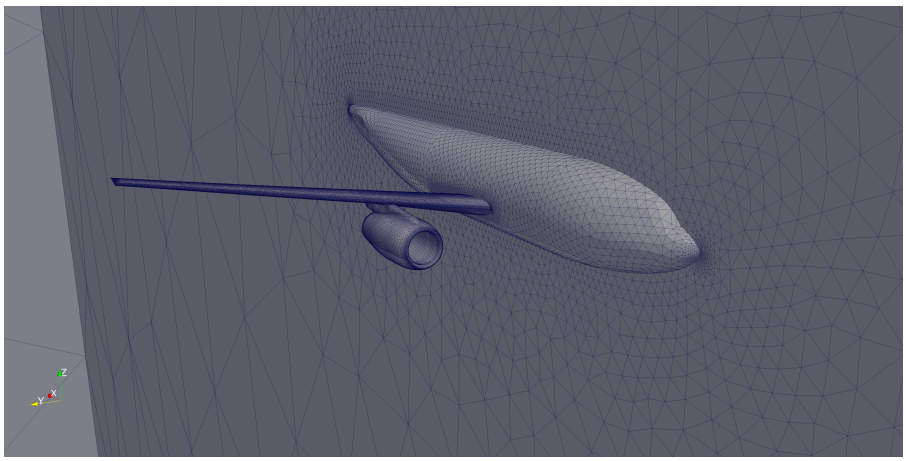
Three aerospace geometries, of increasing complexity, are considered. These are the ONERA M6 wing shown in Figure 13(a), the F6 aircraft geometry, with wing and installed nacelle, shown in Figure 13(b) and the full F16 aircraft geometry shown in Figure 13(c). It should be noted that the F16 stores are not attached to the aircraft surface, which creates gaps that are smaller than the prescribed local mesh spacing. The number of elements on each of the surface meshes is given in Table 1, along with the number of elements which do not contain their Voronoi vertices, for each generation method. In this table the percentage of boundary elements not containing their Voronoi vertices is also given.

3.1 ONERA M6

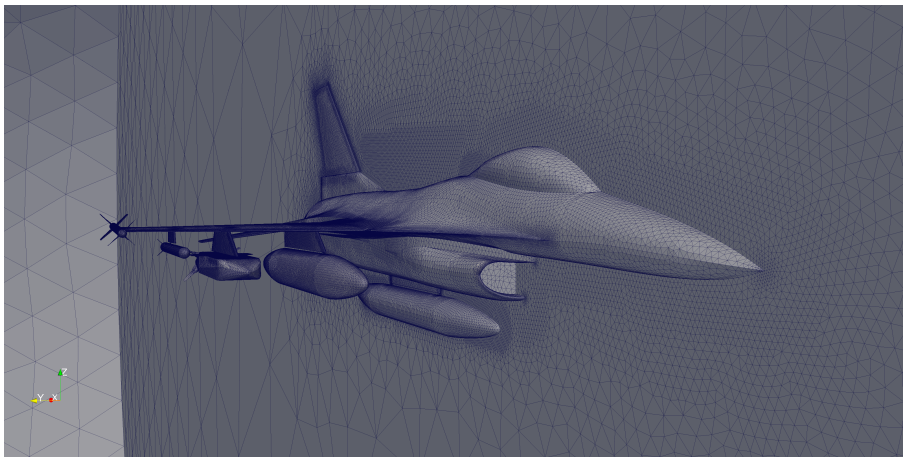
The ONERA M6 is the simplest example presented. Before optimisation, the most striking result shows the effectiveness of the modified advancing front technique introduced in this paper. Using the AD technique without the modified advancing front technique results in 36% of the boundary elements not containing their Voronoi vertices. This is reduced to 7.6% when the modified advancing front technique is used. The improved starting point remains important after the optimisation process has been applied. In 24 hours 400 iterations of the optimisation technique were performed. After optimisation, the ADAF technique has roughly half as many poor quality elements as the AD technique. Figures 14(a) and 14(b) show cuts of the optimised mesh at the trailing edge for AD and ADAF respectively. The structure of the mesh close to the boundary appears better using ADAF. Before optimisation LAF generates the best quality mesh close to the boundary. This is to be expected, since the technique will closely recreate the ideal mesh in the first few layers. The LAF technique does not perform as well in the volume with 42%



(a) ONERA M6 Wing



(b) F6

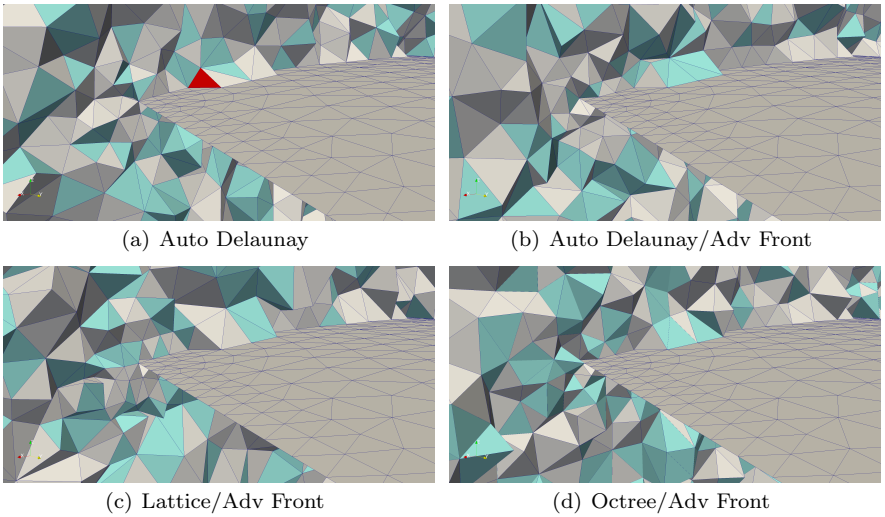


(c) F16

Fig. 13 The surface meshes for the three geometries considered.

Table 1 Summary of the number of elements with Voronoi vertices outside for each example and each mesh generation method. N_e refers to the number of elements.

		Surface	Volume	Before Proposed Optimisation		After Proposed Optimisation	
		N_e	N_e	Boundary Outside	Total Outside	Boundary Outside	Total Outside
M6	Traditional	8710	216759	38%	36%	N/A	N/A
	AD	8710	203834	36%	36%	1.1%	0.096%
	ADAF	8710	207458	7.6%	35%	0.51%	0.084%
	LAF	8710	234277	5.0%	42%	0.52%	0.096%
	OAF	8710	207650	7.6%	28%	0.49%	0.088%
F6	Traditional	82946	2537189	35%	34%	N/A	N/A
	AD	82946	2402578	37%	36%	1.16%	0.18%
	ADAF	82946	2454198	7.2%	34%	0.68%	0.16%
	LAF	82946	2404118	5.3%	40%	1.3%	0.37%
	OAF	82946	2446254	7.2%	27%	0.92%	0.35%
F16	Traditional	170104	5804032	40%	33%	N/A	N/A
	AD	170104	5518276	42%	34%	6.2%	2.3%
	ADAF	170104	5612485	12%	33%	3.9%	1.7%
	LAF	170104	4430551	9.3%	37%	3.5%	1.8%
	OAF	170104	5516251	12%	27%	4.2%	2.0%

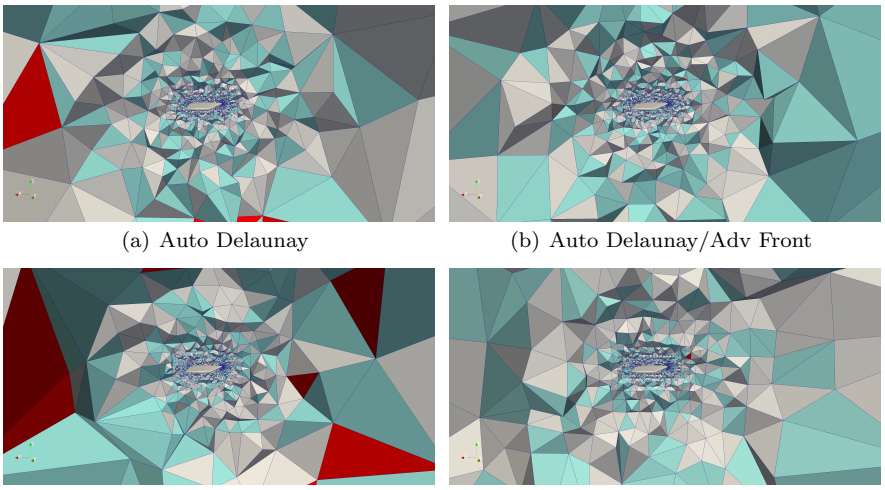


(a) Auto Delaunay (b) Auto Delaunay/Adv Front

(c) Lattice/Adv Front (d) Octree/Adv Front

Fig. 14 Comparison between different generation strategies for the ONERA M6 wing at the trailing edge. The coloured elements in the cuts have been merged and the solid red elements are the only elements which do not contain their Voronoi vertex.

of Voronoi vertices outside compared with 28% outside using OAF. This will be due to the advantages of LAF diminishing as the mesh grows outwards, whereas using OAF a mesh is generated over all space directly. This effect can be clearly seen when comparing Figure 15(c) to 15(d) which shows a view of a cut through meshes generated using LAF and OAF respectively. Another



(a) Auto Delaunay (b) Auto Delaunay/Adv Front

(c) Lattice/Adv Front (d) Octree/Adv Front

Fig. 15 Comparison between different generation strategies for the ONERA M6 wing showing the full domain. The coloured elements in the cuts have been merged and the solid red elements are the only elements which do not contain their Voronoi vertex.

important result is the effectiveness of the optimisation approach. Running optimisation almost eliminates any differences between the four techniques, reducing the percentage of poor quality elements by 3 orders of magnitude. The only notable exception is that running simply AD results in poorer elements close to the boundary. This highlights the importance of the modified advancing front technique. All of the new techniques significantly outperform the traditional generation techniques.

Figure 16 shows the secondary quality measures for each mesh in a histogram. In these plots, the first bar of the histograms represents the percentage of the quality measures that fall between the minimum and the first chosen value. The last bar of the histograms represents the percentage of the quality measures that fall between the last chosen value and the maximum. There is little difference between the techniques introduced in this paper, but there is a clear difference between these and the traditional technique. The mesh generated using the traditional technique has a better distribution of minimum dihedral angles. This is not surprising, since in many respects, this is a key quality measure traditional techniques aim to improve. It is the cost for focusing on co–volume mesh requirements. A quality measure which may seem to produce a counter intuitive result is q_4, the distance from the intersection of the face by the Voronoi edge to the face centroid, normalised by the local spacing. Theoretically, the quality measure q_4 should be zero for meshes that are generated using the Delaunay kernel. However, in the standard meshes, the deviation is due to boundary constraints and the traditional mesh cosmetics. Using the proposed generation technique, this quality measure is reduced by the introduction of the weight to pull the Voronoi vertices inside elements.

Large values for q_4 will result in reduced second order accuracy for co–volume schemes. This is an accepted cost which improves stability by pulling Voronoi vertices inside elements. The histogram shows that the traditional generation technique has lower values of q_4 but larger Voronoi-centroid distances. Finally, the Voronoi edge lengths are increased in the newly presented techniques which will have a positive effect on the minimum time step of the co–volume scheme.

3.2 F6

The geometry includes the full body of an aircraft with a wing mounted nacelle. The wing has a blunt trailing edge. Similar trends to the M6 can be observed. The modified advancing front technique significantly reduces the number of poor quality elements at the boundary. LAF does the best job of generating elements close to the boundary and OAF the best in the free space region. In 24 hours 45 iterations of the optimisation technique were performed. Optimisation reduces the differences between the new techniques. After optimisation, the best mesh is the ADAF. Figure 17 compares the meshes at the leading edge of the nacelle. The lattice structure in the LAF mesh can be seen in this region close to the boundary clearly in Figure 17(c). In the mesh generated using OAF, poor quality elements appear in the region where the near field points have been connected to the free space octree generated points. Figure 18 shows cuts through the meshes in the region around the wing. The differences between LAF and OAF can clearly be seen here. The remaining poor quality elements are near the trailing edge in OAF and in the free space in LAF. In all cases, the new techniques are orders of magnitude better than traditional methods.

The mesh statistics for the F6 are shown in the histogram in Figure 19. These show similar trends to the M6 example. One notable difference is that the mesh generated using ADAF has a noticeably better distribution of Voronoi–centroid distances.

3.3 F16

The F16 is the most complex example considered and is a challenging test for the techniques presented. Overall the trends are the same as in the other examples. Initially, LAF generates the best mesh close to the boundary and OAF the best in free space. In 24 hours 6 iterations of the optimisation technique were performed. After optimisation, the differences between the techniques are not significant. The new techniques produce significantly better meshes than traditional methods. The best mesh using new techniques has 1.7% total elements not containing their Voronoi vertices, compared with 33% using traditional techniques. Figure 20 compares the different techniques in the region close to the stores. The gaps, which are smaller than the prescribed spacing,

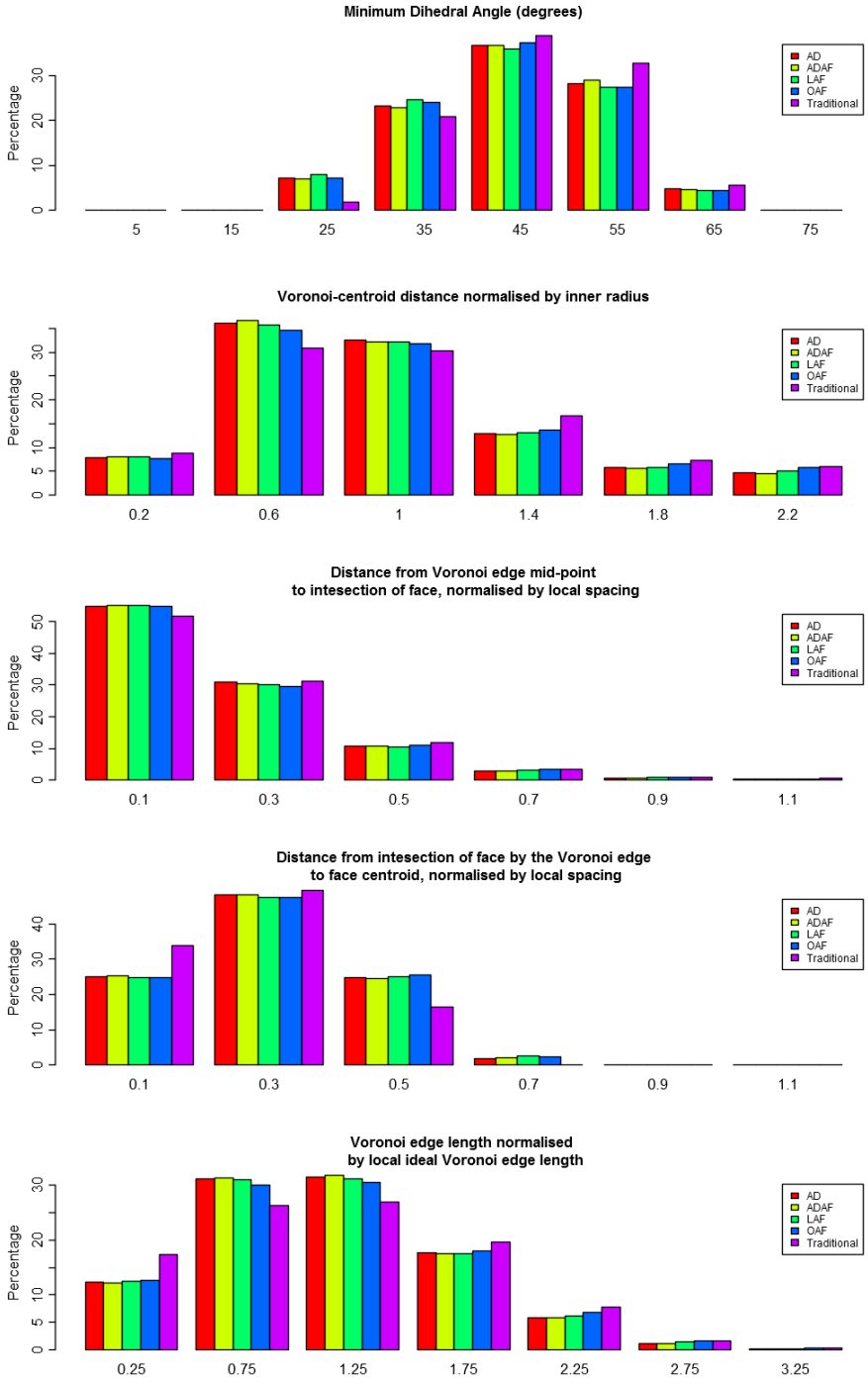
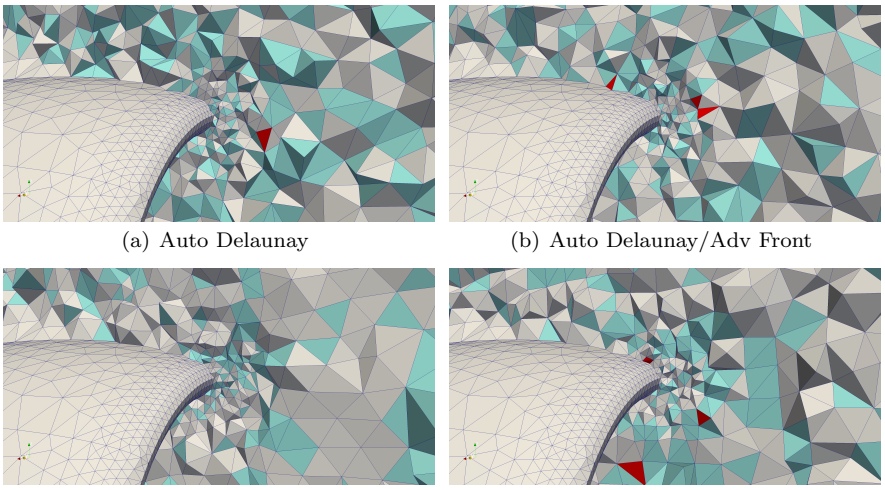


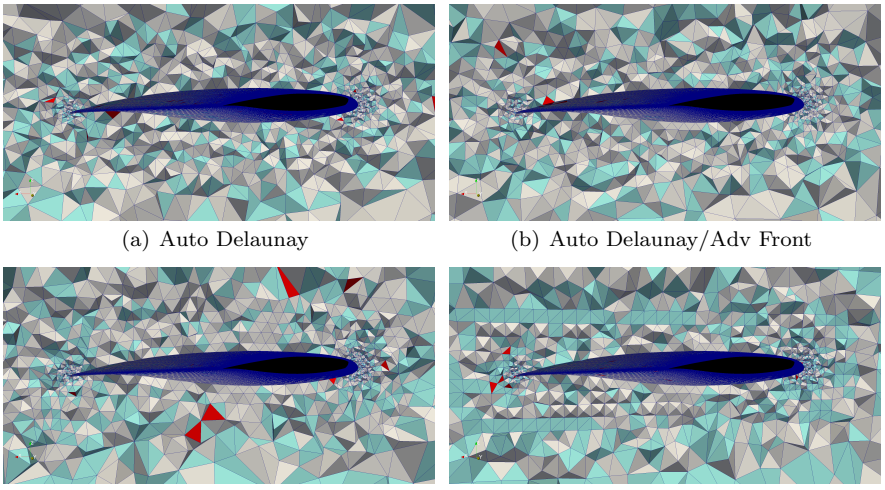
Fig. 16 Histogram showing mesh statistics for the M6.



(a) Auto Delaunay (b) Auto Delaunay/Adv Front

(c) Lattice/Adv Front (d) Octree/Adv Front

Fig. 17 Comparison between different generation strategies for the F6 geometry at the leading edge of the nacelle. The coloured elements in the cuts have been merged and the solid red elements are the only elements which do not contain their Voronoi vertex.



(a) Auto Delaunay (b) Auto Delaunay/Adv Front

(c) Lattice/Adv Front (d) Octree/Adv Front

Fig. 18 Comparison between different generation strategies for the F6 geometry cuts taken through the wing. The surface mesh is not plotted. The coloured elements in the cuts have been merged and the solid red elements are the only elements which do not contain their Voronoi vertex.

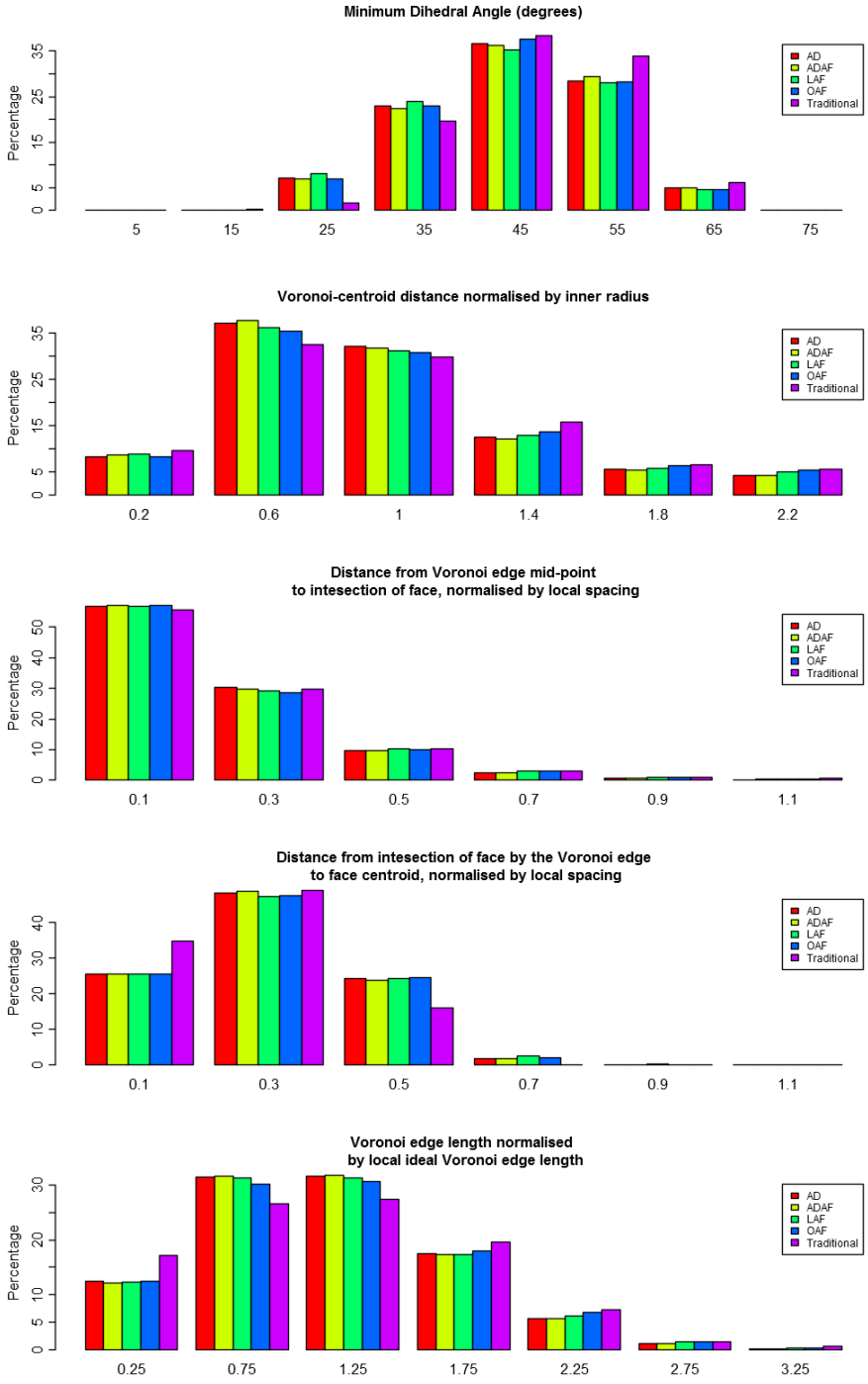
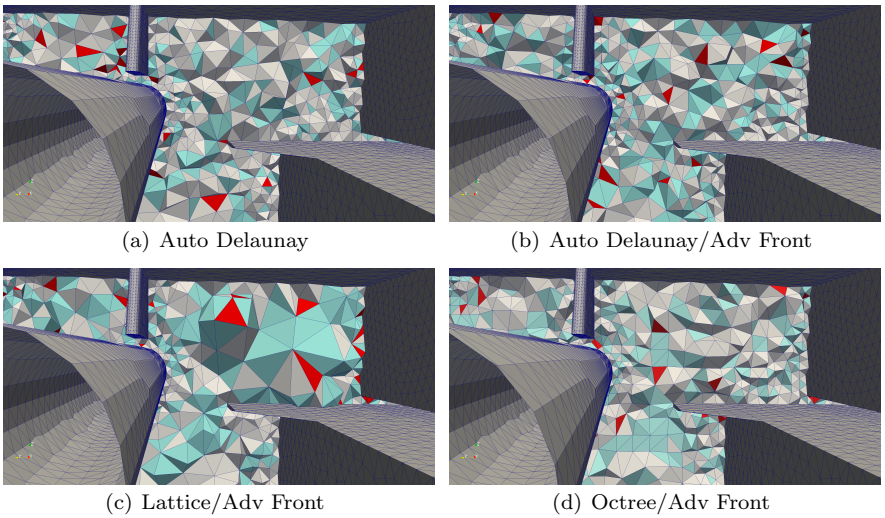


Fig. 19 Histogram showing mesh statistics for the F6.



(a) Auto Delaunay (b) Auto Delaunay/Adv Front

(c) Lattice/Adv Front (d) Octree/Adv Front

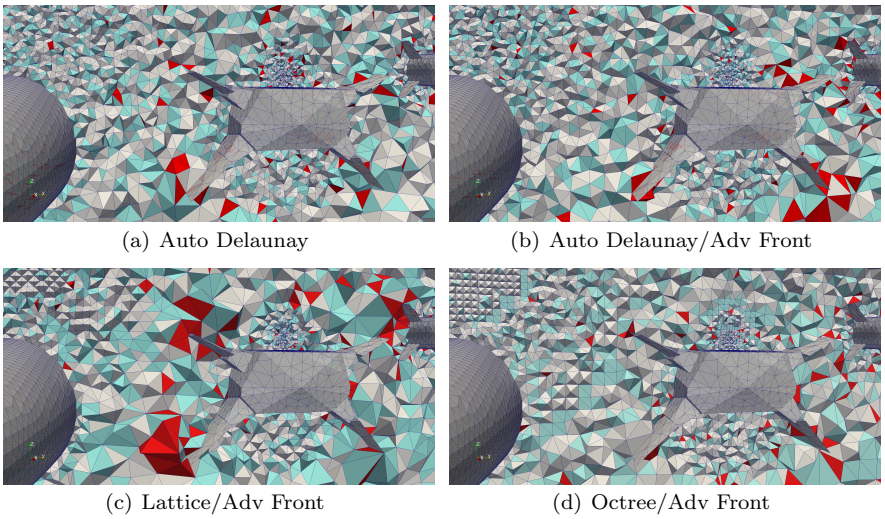
Fig. 20 Comparison between different generation strategies for the F16 zoomed in to show the gaps between the wing and stores. The coloured elements in the cuts have been merged and the solid red elements are the only elements which do not contain their Voronoi vertex.

between the wing and the stores create difficulties. The LAF technique has better elements directly in contact with the boundaries, but it struggles to generate good quality elements between them. In fact, in Figure 20(c), LAF does not seem to have generated enough points inside. This may indicate a need to adjust δ_m depending on the geometry, which would limit the application of this method to the general case. Figure 20(d) shows that OAF can handle this situation better.

Figure 21 shows the region behind a store, which has a complex fin configuration. The configuration is shown without the volume mesh in Figure 22. This surface results in volume elements with small dihedral angles. The plots in Figure 21 show how LAF is the most sensitive to this complex geometry. This is to be expected, since the mesh grows from the boundary in LAF. Figure 21(d) shows how OAF manages to generate good quality elements close to and far away from the surface. The mesh statistics for the F16, presented in Figure 23, show similar trends to previous examples.

3.4 Electromagnetic Wave Scattering Simulation

The aim of this work was to generate suitable unstructured meshes for co–volume solvers. To illustrate the effectiveness of the proposed strategies, a three–dimensional electromagnetic wave scattering simulation is considered. An unstructured co–volume solver for the Maxwell's equations was used, full details of which have been described in previous work [19].



(a) Auto Delaunay (b) Auto Delaunay/Adv Front

(c) Lattice/Adv Front (d) Octree/Adv Front

Fig. 21 Comparison between different generation strategies for the F16 zoomed in to show the complex fin configuration on the rear of the stores. The coloured elements in the cuts have been merged and the solid red elements are the only elements which do not contain their Voronoi vertex.

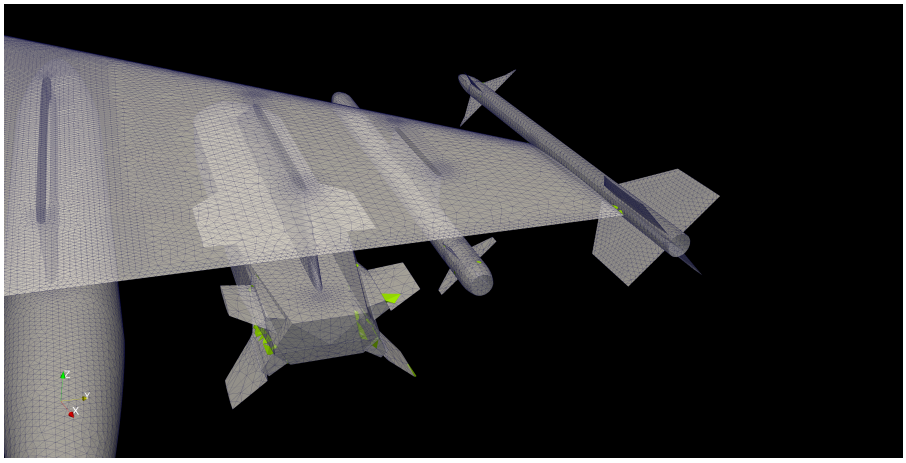


Fig. 22 Elements, plotted in green, with small dihedral angle caused by the complex F16 surface.

Interaction between a plane single frequency incident wave and a perfect electrical conductor (PEC) body with two pairs of rear fins is considered. A surface mesh of the geometry is shown in Figure 24.

The wave frequency is such that the length of the body is 6λ, where λ is the wavelength. The incident wave propagates in the (x, y) plane and the main axis of the body is aligned with the z axis of a standard Cartesian coordinate system. In all cases the simulation was run for 15 cycles of the

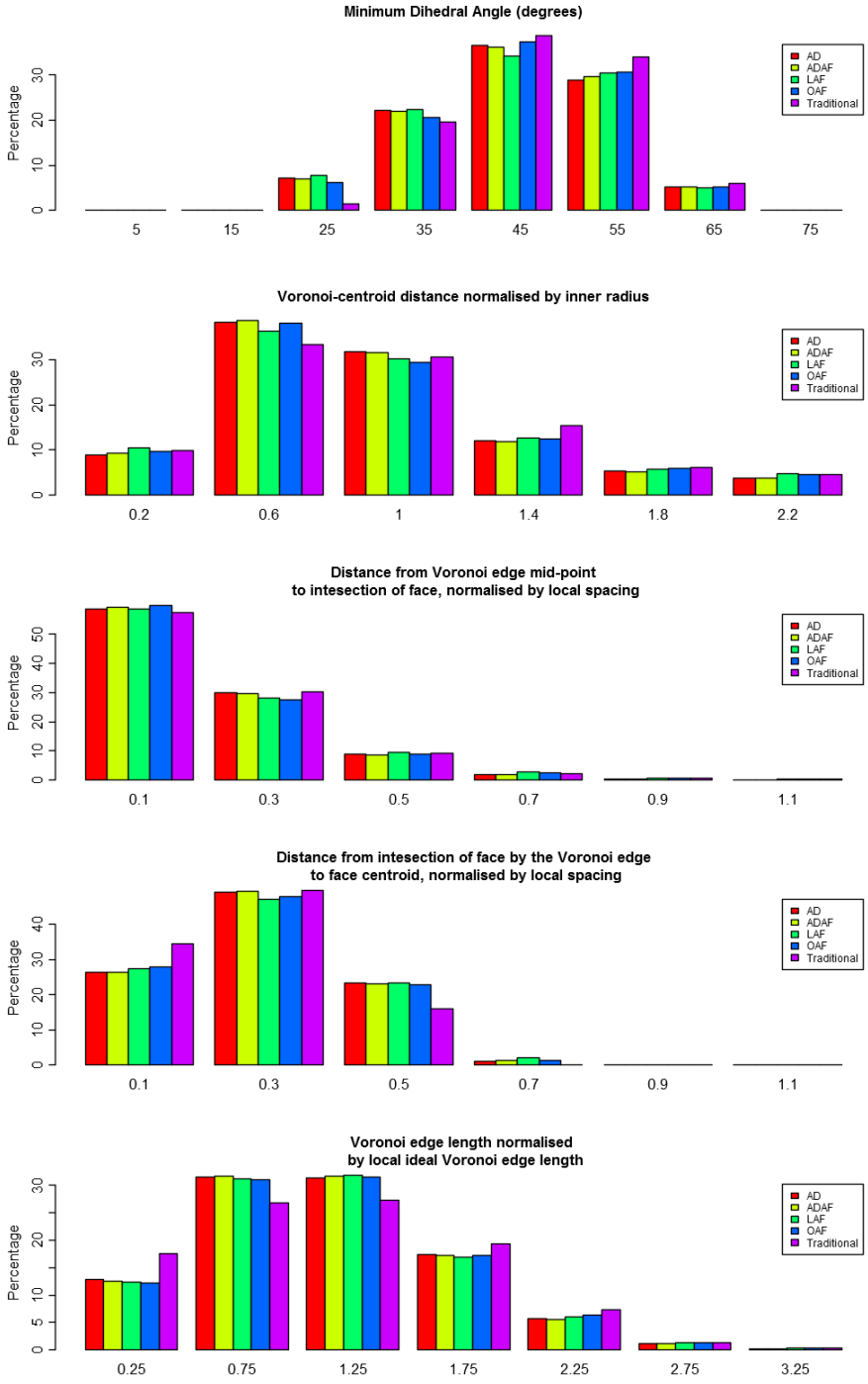


Fig. 23 Histogram showing mesh statistics for the F16.

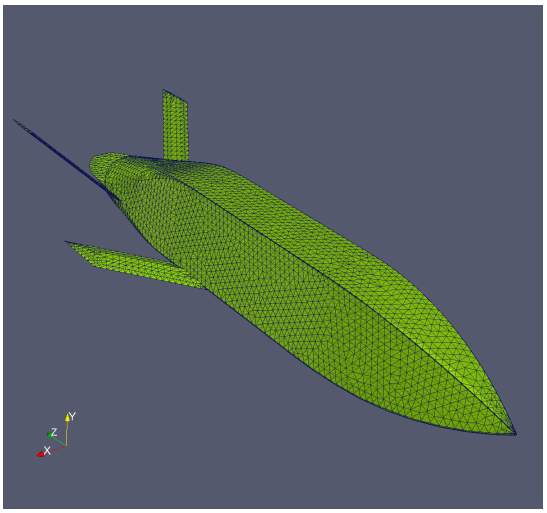


Fig. 24 The surface mesh of the PEC body in the electromagnetic wave scattering problem.

Table 2 Comparison of CPU time for the solution of scattering by a PEC object.

Generation method	Time Step	CPU Time per cycle
Xie et al. [19]	3.47×10^{-5}	148 minutes
Lattice/Adv Front	2.93×10^{-4}	17 minutes

plane wave, which was enough to provide a converged solution. The number of time steps required per cycle was determined by the time step size, which in turn is calculated using the minimum Delaunay and Voronoi edge lengths in the mesh [19].

Initially, to enable comparison with the method presented previously by Xie et al. [19], hybrid meshes with uniform target spacing functions were generated. An element size of $\lambda/15$ was selected. To create the hybrid meshes, a structured hexahedral mesh, extending 1.5λ beyond the boundary of the geometry, was generated. A hole, which contains the geometry, was cut out from the hexaheral mesh and extended to a distance of 0.4λ from the object. The domain bounded by the triangulated surface mesh of the object and the triangulated surface of the hole was triangulated in two different ways. In the first method, using the method described by Xie et al. [19], nodes were inserted from a uniform ideal tetrahedral mesh and improved using Lloyd's algorithm. In the second method, a mesh was generated using the lattice/Advancing Front technique and improved using the mesh optimisation presented in this paper. The mesh optimisation was limited to 30 cycles.

The findings are summarised in Table 2. Globally the mesh generated using the Xie et al. [19] technique has less elements with their Voronoi vertex outside. The prioritisation of boundary elements in the new technique is shown as

Table 3 Comparing the meshes generated for the PEC body with a uniform target spacing function using Lattice/Adv Front with varying time limits placed on the mesh optimisation.

Number of Optimisation Cycles and CPU Cost of optimisation	Percentage with Circumcentre Outside	Time Step	CPU Time per Solver Cycle
30 (24 hours)	0.8%	2.93×10^{-4}	16 mins
15 (12 hours)	0.8%	2.93×10^{-4}	16 mins
6 (6 hours)	1.2%	2.93×10^{-4}	16 mins
3 (3 hours)	2.3%	2.89×10^{-4}	19 mins
1 (1 hour)	7.1%	2.89×10^{-4}	Solution Diverged

the percentage of boundary elements with their Voronoi vertices outside is 0.04% compared with 0.4% using [19]. The increase in time step, resulting from increased minimum Voronoi edge length, from 3.47×10^{-5} to 2.93×10^{-4} has lead to the reduction in simulation CPU cost by a factor of 9. A study was performed in which the number of adaptive mesh optimisation cycles was reduced and the results are shown in Table 3. These results show that 3 cycles of the mesh optimisation were needed to generate a suitable mesh for the problem. The CPU cost of 3 cycles of optimisation was 3 hours, this effort resulted in a reduction of CPU cost per cycle from 148 minutes to 19 minutes. This reduction means the extra CPU cost from mesh generation is paid for in just 2 solver iterations. Figure 25 shows how the mesh quality changes as the number of cycles is increased. It is clear that there is a little improvement in quality measures beyond the third cycle. To show how the number of cycles affects the accuracy of the solution, the radar cross section (RCS), a parameter of practical interest in these problems [19], is plotted in Figure 26. The validated solution obtained by Xie at al. [19] on a uniform mesh is also plotted. The figure shows very little difference between the solutions.

The final example considered the previous geometry with a non–uniform spacing distribution function. Three sources were placed at the end of the PEC body to improve the resolution of the high curvature of the fins. A spacing of $\delta_1 = \lambda/30$, was specified in the proximity of the body. This was increased to $\delta_1 = \lambda/15$ at the surface of the hole of the ideal hexahedral mesh. Three cycles of mesh optimisation was performed, cuts through the generated mesh after optimisation are shown in Figure 27. In the final mesh, 0.23% of the elements globally, and 0.03% of the boundary connected elements, did not contain their Voronoi vertex. Figure 28 compares the solution obtained on the graded mesh, and on the uniform mesh using the present technique, with the solution of Xie et al. [19] on a uniform mesh. It is clear from this figure that there is very little difference between the three solutions.

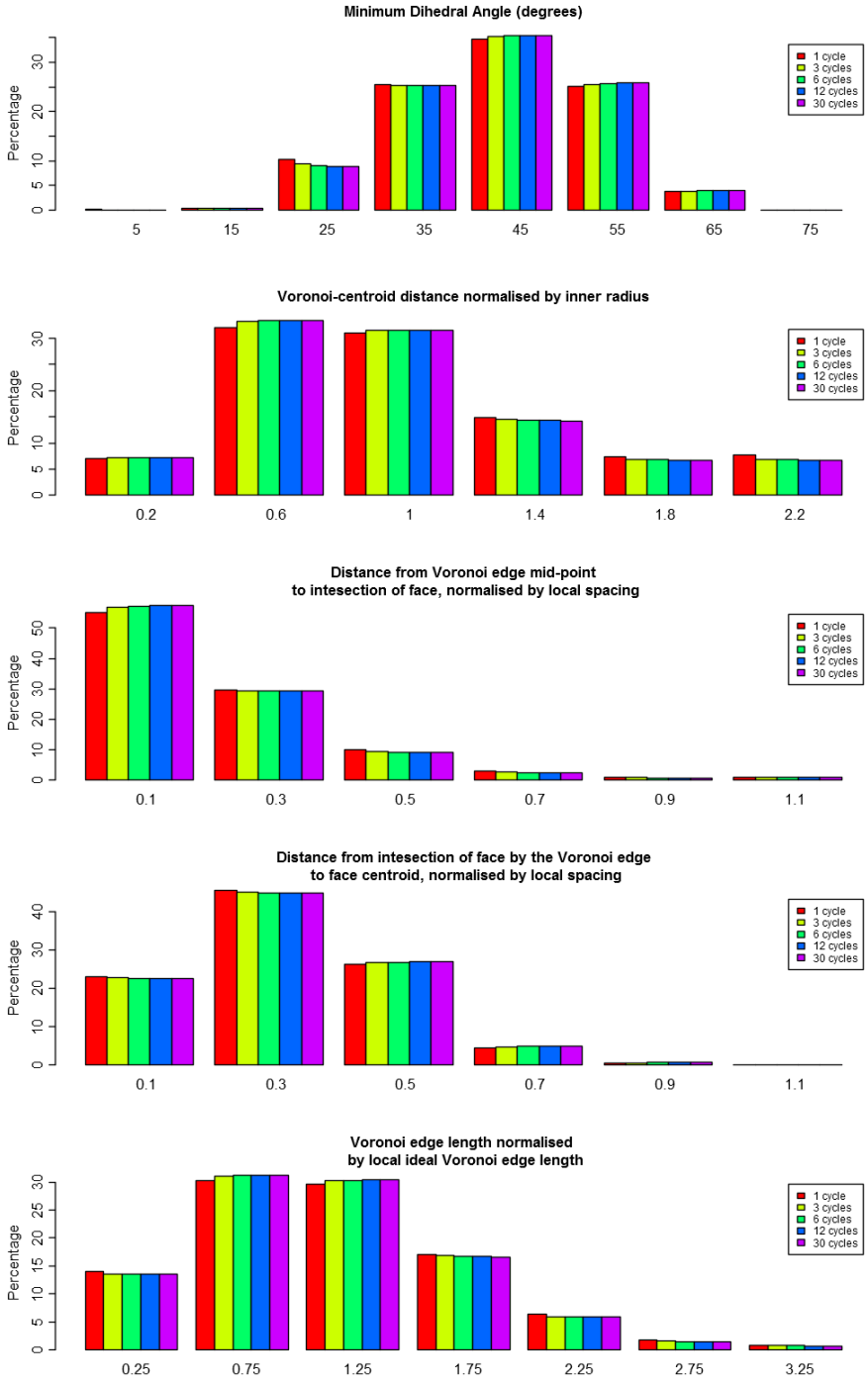


Fig. 25 Histogram showing how the mesh optimisation affects the mesh quality for the PEC body. Number of cycles refers to the number of cycles of optimisation.

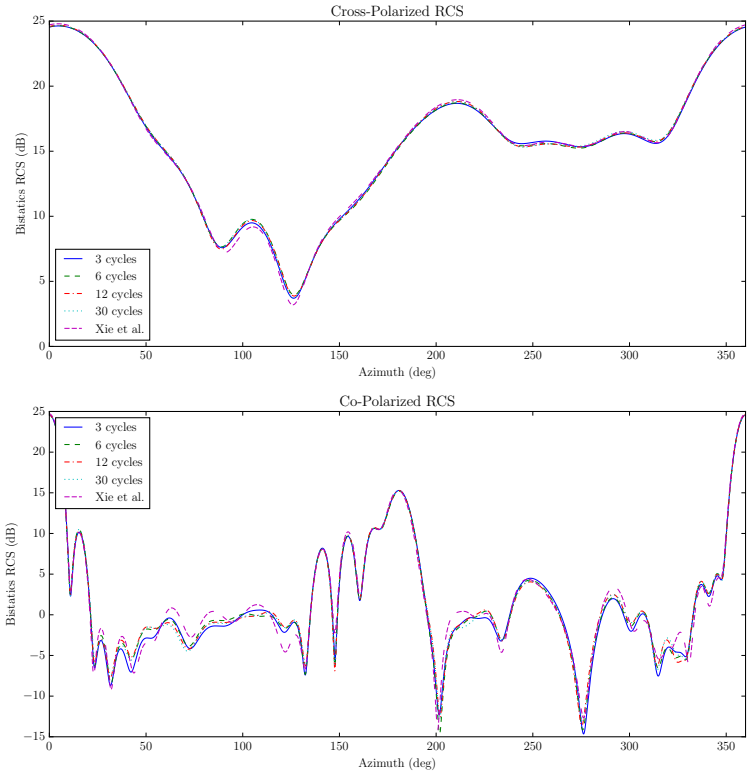


Fig. 26 Comparing the radar cross section for the electromagnetic wave scattering simulation, calculated on a mesh generated by the technique presented by Xie et al. [19] and the techniques proposed in this work, with varying numbers of optimisation cycles.

4 Conclusion

The aim was to determine the feasibility of generating meshes suitable for use with co–volume solution techniques. The primary measurement employed was the number of elements which do not contain their Voronoi vertex. A number of generation strategies were introduced, along with a mesh optimisation technique. These were specifically designed with this key measurement in mind. Three industrial aerospace configurations, of increasing geometrical complexity, were used as examples.

When comparing the different generation strategies, the results show that the Auto Delaunay/Adv Front strategy results in the best mesh in every case, after running the optimisation. Before optimisation, the Octree/Adv Front consistently results in the best mesh. We believe this is because the strategy

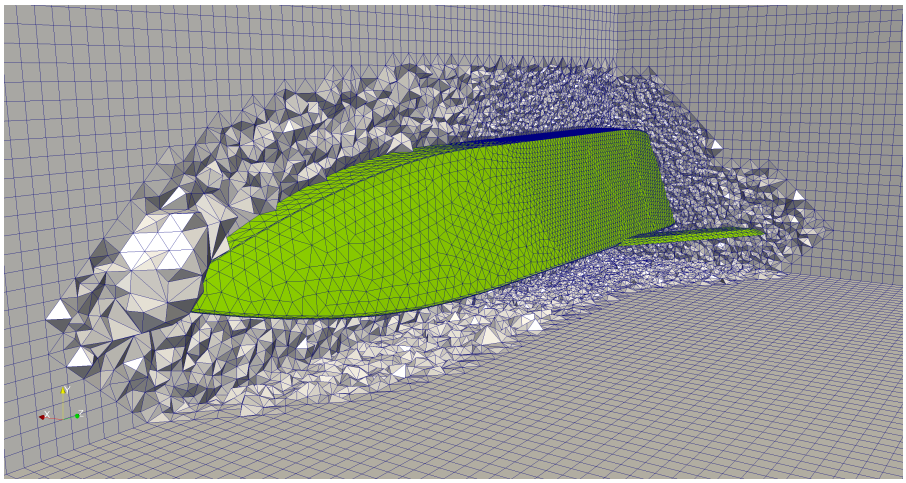


Fig. 27 Cuts through the graded volume mesh around the PEC body in the electromagnetic wave scattering problem. This was generated using the Lattice/Advancing Front technique and improved using the adaptive mesh optimisation presented in this paper.

is largely made up of a near ideal mesh, the octree. From an optimisation perspective, it is similar to starting the process close to a local minimum, from which it may be difficult to escape. A significant result from the test cases is the reduction in the number of boundary elements with their Voronoi vertices outside due to the modified advancing front technique. In the most complex example, the F16, previous methods resulted in 42% of the boundary elements not containing their Voronoi vertex, whereas using the new advancing front technique there are now just 12%.

The mesh optimisation algorithm effectively improved the quality of the meshes in all examples. This approach was able to reduce the number of elements with Voronoi vertices outside by at least an order of magnitude in all cases, sometimes more. This procedure eliminates almost any differences in mesh quality between the techniques. The distance from face centroids to the point at which the Voronoi vertex intersects the face is quite large in the more complex examples. This is due to using node weights to move the Voronoi vertex. Constructing a multi-objective objective function to account for this distance may result in improvements. Running the optimisation procedure is computationally expensive, roughly $O(N_e)$. We have shown that, when solving electromagnetics problems, the reduction in time step caused by the optimisation procedure more than pays for the cost of the optimisation. In the problem presented, the total computational cost from geometry to solution was reduced by an order of magnitude.

The examples presented here were deliberately selected as they were difficult, realistic examples. In each case, the percentage of elements not containing their Voronoi vertices was less than or equal to 2%. The results build confidence that it would be feasible to generate meshes for co–volume schemes,

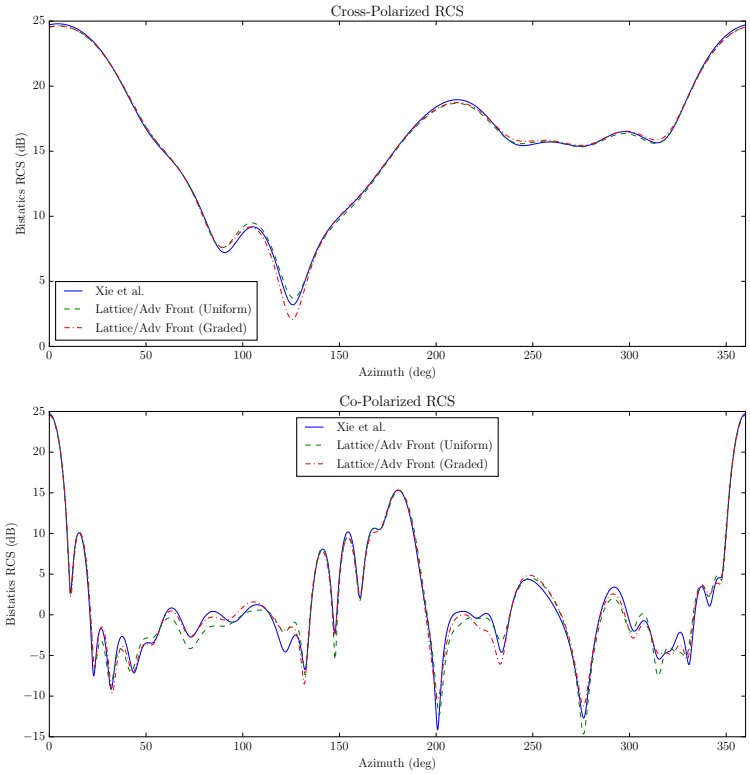


Fig. 28 Comparing the radar cross section for the electromagnetic wave scattering simulation, calculated on a mesh generated by the technique presented by Xie et al. [19] and the new techniques proposed in this work with both uniform and graded target spacing functions.

mainly due to the modified advancing front technique along with an effective adaptive mesh optimisation scheme. It is now necessary, and worthwhile, to develop three dimensional co–volume solvers in other areas, so that meshes generated by these procedures included here may be fully tested.

Acknowledgements The authors acknowledge the financial support for this work provided by the UK Engineering and Physical Sciences Research Council (EPSRC) under Research Grant EP/K000705.

References

1. Du, Q., Wang, D.: Tetrahedral mesh generation and optimization based on centroidal voronoi tessellations. International Journal for Numerical Methods in Engineering **56**,

1355–1373 (2003)
2. Edmonds, J.: Paths, trees, and flowers. Canadian Journal of Mathematics **17**, 449–467 (1965)
3. Harlow, F.H., Welch, J.E.: Numerical calculation of time–dependent viscous incompressible flow of fluid with free surface. The Physics of Fluids **8**, 2182–2189 (1965)
4. Kennel, M.: Kdtree 2: Fortran 95 and c++ software to efficiently search for near neighbors in a multi–dimensional euclidean space. ArXiv Physics e–prints (arXiv:physics/0408067v2) (2004)
5. Lloyd, S.: Least square quantization in the PCM. IEEE Transactions on Information Theory **28**, 129–137 (1982)
6. Löhner, R.: Progress in grid generation via the advancing front technique. Engineering with Computers **12**(3-4), 186–210 (1996). DOI 10.1007/BF01198734. URL http://dx.doi.org/10.1007/BF01198734
7. Peraire, J., Vahdati, M., Morgan, K., Zienkiewicz, O.C.: Adaptive remeshing for compressible flow computations. Journal of Computational Physics **72**, 449–466 (1987)
8. Remacle, J.F., Lambrechts, J., Seny, B., Marchandise, E., Johnen, A., Geuzainet, C.: Blossom-quad: A non-uniform quadrilateral mesh generator using a minimum-cost perfect-matching algorithm. International Journal for Numerical Methods in Engineering **89**(9), 1102–1119 (2012). DOI 10.1002/nme.3279. URL http://dx.doi.org/10.1002/nme.3279
9. Sazonov, I., Hassan, O., Morgan, K., Weatherill, N.P.: Smooth Delaunay–Voronoi dual meshes for co–volume integration schemes. In: P.P. Rebay (ed.) Proceedings of the 15th International Meshing Roundtable, pp. 529–541. Springer, Berlin (2006)
10. Sazonov, I., Wang, D., Hassan, O., Morgan, K., Weatherill, N.: A stitching method for the generation of unstructured meshes for use with co–volume solution techniques. Computer Methods in Applied Mechanics and Engineering **195**, 1826–1845 (2006)
11. VanderZee, E., Hirani, A., Guoy, D., Ramos, E.: Well-centered planar triangulation – an iterative approach. In: Proceedings of the 16th International Meshing Roundtable, pp. 121–138. SpringerLink (2008). Session 1B
12. VanderZee, E., Hirani, A., Zharnitsky, V., Guoy, D.: A dihedral acute triangulation of the cube. Computational Geometry **43**(5), 445–452 (2010)
13. Vanderzee, E., Hirani, A.N., Guoy, D., Zharnitsky, V., Ramos, E.A.: Geometric and combinatorial properties of well-centered triangulations in three and higher dimensions. Comput. Geom. Theory Appl. **46**(6), 700–724 (2013). DOI 10.1016/j.comgeo.2012.11.003. URL http://dx.doi.org/10.1016/j.comgeo.2012.11.003
14. Viswanathan, G.M.: Lévy flights and superdiffusion in the context of biological encounters and random searches. Physics of Life Reviews **5**, 133–150 (2008)
15. Walton, S., Hassan, O., Morgan, K.: Reduced order mesh optimisation using proper orthogonal decomposition and a modified cuckoo search. International Journal for Numerical Methods in Engineering **93**(5), 527–550 (2013). DOI 10.1002/nme.4400. URL http://dx.doi.org/10.1002/nme.4400
16. Walton, S., Hassan, O., Morgan, K.: Strategies for generating well centered tetrahedral meshes on industrial geometries. In: New Challenges in Grid Generation and Adaptivity for Scientific Computing, SEMA SIMAI Springer Series (in press)
17. Walton, S., Hassan, O., Morgan, K., Brown, M.R.: Modified cuckoo search: a new gradient free optimisation algorithm. Chaos, Solitons & Fractals **44**(9), 710–718 (2011)
18. Weatherill, N.P., Hassan, O.: Efficient three–dimensional Delaunay triangulation with automatic point creation and imposed boundary constraints. International Journal for Numerical Methods in Engineering **37**, 2005–2040 (1994)
19. Xie, Z.Q., Hassan, O., Morgan, K.: Tailoring unstructured meshes for use with a 3d time domain co-volume algorithm for computational electromagnetics. International Journal for Numerical Methods in Engineering **87**(1-5), 48–65 (2011). DOI 10.1002/nme.2970. URL http://dx.doi.org/10.1002/nme.2970
20. Yang, Y.J., Yong, J.H., Sun, J.G.: An algorithm for tetrahedral mesh generation based on conforming constrained delaunay tetrahedralzation. Computers & Graphics/Computing and Graphics **29**(4), 606–615 (2005)
21. Yee, K.: Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media. IEEE Transactions on Antennas and Propagation **14**, 302–307 (1966)