Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 203 (2017) 206–218

Procedia Engineering

www.elsevier.com/locate/procedia

26th International Meshing Roundtable, IMR26, 18-21 September 2017, Barcelona, Spain

# A framework for the generation of high-order curvilinear hybrid meshes for CFD simulations

Michael Turner[a], David Moxey[b], Joaquim Peiró[a,*], Mark Gammon[c], Claire R. Pollard[c], Henry Bucklow[c]

[a]*Department of Aeronautics, Imperial College London, South Kensington Campus, London SW7 2AZ, U.K.*
[b]*College of Engineering, Mathematics and Physical Sciences, University of Exeter,*
[c]*ITI Global, Cambridge, U.K.*

**Abstract**

We present a pipeline of state-of-the-art techniques for the generation of high-order meshes that contain highly stretched elements in viscous boundary layers, and are suitable for flow simulations at high Reynolds numbers. The pipeline uses CADfix to generate a medial object based decomposition of the domain, which wraps the wall boundaries with prismatic partitions. The use of medial object allows the prism height to be larger than is generally possible with advancing layer techniques. CADfix subsequently generates a hybrid straight-sided (or linear) mesh. A high-order mesh is then generated a posteriori using NekMesh, a high-order mesh generator within the Nektar++ framework. During the high-order mesh generation process, the CAD definition of the domain is interrogated; we describe the process for integrating the CADfix API as an alternative backend geometry engine for NekMesh, and discuss some of the implementation issues encountered. Finally, we illustrate the methodology using three geometries of increasing complexity: a wing tip, a simplified landing gear and an aircraft in cruise configuration.


Peer-review under responsibility of the scientific committee of the 26th International Meshing Roundtable.

*Keywords:* High-order curvilinear meshes; CAD interface; Medial-object domain decomposition; Isoparametric bounday-layer meshing.

## 1. Introduction

The generation of boundary conforming, valid, quality curvilinear high-order meshes, which incorporate stretched elements in the near-wall boundary-layer regions and are thus suitable for the simulation of high-Reynolds number flows of industrial interest, is a complex challenge. It is certainly one of the most significant obstacles preventing the wider uptake and development of high-order CFD simulations.

High-order mesh generation is mostly based on *a posteriori* approaches that deform a coarse linear mesh to accommodate the curvature at the boundary. The main challenge is to achieve valid meshes which are suitable for solver-based calculations. These approaches can be classified broadly into two types. In the first type, the mesh is treated as a solid body which is deformed to incorporate curvature at the boundary. These include linear elasticity

* Corresponding author.
*E-mail address:* j.peiro@imperial.ac.uk

1877-7058 
Peer-review under responsibility of the scientific committee of the 26th International Meshing Roundtable.
10.1016/j.proeng.2017.09.808

[1,2], non-linear hyperelasticity [3,4], thermo-elasticity [5] and the Winslow equations [6]. The second type obtains the high-order mesh as the minimum of a certain functional such as: scaled Jacobian distortion metrics [7,8], energies of surface deformation [9], and shape distortion metrics [10]. The generation of a high-order mesh using the *a posteriori* approach can be thought of consisting of three steps. The first step is to generate a coarse linear mesh. This could be accomplished by any state-of-the-art mesh generator, but the coarseness requirements will represent a significant challenge for some linear generators. The second step is to add extra nodes to curve the surface mesh entities to conform with the boundary of the domain, which is often given by a CAD representation. However, this step will inevitably create invalid, self-intersecting elements, often due to the presence of highly distorted, even non watertight, CAD definitions. The only remedy for this, particularly in the bottom-up approach we adopt, is to fix the CAD geometry. The final stage is to deform and curve the interior mesh to accommodate the boundary curvature whilst maintaining valid high quality elements, as described for instance in reference [11] and references therein. These *a posteriori* generators of high-order curvilinear meshes tend to have difficulties in ensuring the validity of the mesh if highly stretched elements are present in the linear mesh, as required for the efficient simulation of boundary-layer flows.

In this work we propose a semi-structured approach and show that difficulties in producing high-order curvilinear meshes can be significantly eased through the combination of appropriate state-of-the-art mesh generation technologies. To achieve this goal, the proposed approach combines two complementary mesh generation procedures.

The generation of the linear mesh is accomplished by the commercial software CADfix [12], which provides powerful CAD healing and modification tools, an interface, CFI, for handling CAD geometrical operations and queries, and a linear mesh generator based on the medial object approach to decompose the domain into so called partitions which, in turn can be discretized into structured or unstructured meshes. In particular, by properly designing the medial object partitioning one could obtain high-quality boundary-layer type meshes of triangular prisms near the wall surfaces.

The generation of the high-order mesh involves two steps. Starting with the straight-sided mesh, we incorporate additional points along the curves, surfaces and interior of the domain to obtain a curvilinear mesh of the required polynomial order and that is compliant with the CAD definition. This follows essentially the methodology proposed in reference [9] and it has been implemented in the open-source code NekMesh which is part of the Nektar++ spectral/hp element framework [13]. The next step is to generate a boundary-layer mesh of the desired resolution along the normal direction. We achieve this by generating a coarse high-order mesh within the medial object based partitions adjacent to the wall boundaries which is then subdivided along the normal direction using the isoparametric approach proposed in reference [14]. This approach is very flexible, modular, and permits defining a variety of resolutions from a base coarse high-order mesh that remains unchanged. This is a very convenient feature for mesh convergence studies.

These procedures will be described in more detail in the following sections. Section 2 describes the CAD interface that handles the geometrical queries during the generation of both linear and high-order meshes. Section 3 provides an overview of the medial object approach and discusses its application to the decomposition of the domain into near-field and far-field regions which are then subdivided into blocks that are discretized into a hybrid linear mesh. The generation of a high-order mesh from this linear mesh is described in Section 4. Finally, the methodology is applied to the generation of high-order meshes using three geometries of varying complexity and the meshes presented in Section 5.

## 2. CAD interface for geometrical queries

Robust CAD interaction is required for both linear and high-order meshing. Within the Nektar++ framework we have implemented a CAD engine that provides the small number of geometric enquiries required for mesh generation through a lightweight wrapper. The default wrapper of the system is currently built upon OpenCascade [15]. The use of the wrapper allows us to hide the vast complexity and size of OpenCascade from users and developers. Here we present an implementation of this wrapper based on CFI, the CAD interface of CADfix [12].

The use of CADfix, and its interface CFI, is motivated by the more stringent requirements on CAD quality for high-order meshing. CAD representations that may work very well within linear mesh generators, may not work for their high-order counterpart. For example, distortion levels in the surfaces, which might be perfectly acceptable for generating linear meshes, could induce poor quality or invalid elements in high-order meshes. Therefore access to

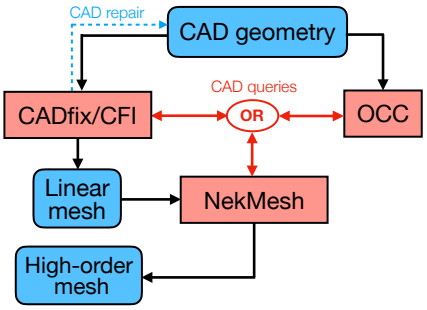



Fig. 1. Diagram depicting the meshing pipelines within NekMesh. A high-order mesh can be generated by Nekmesh along three possible paths. The first two will use a CAD geometry and either of the two CAD APIs: CFI (CADfix) or OCC (OpenCascade). In the third one, NekMesh uses CFI for the CAD geometry queries and a linear mesh generated by CADfix. This is the pipeline used here.

high quality CAD and CAD repair tools for poor quality CAD, along with a robust CAD interface, is vital to the creation of robust quality high-order meshing tools.

To integrate CFI as the CAD back-end of the Nektar++ CAD engine, we first set up a system which would allows us to interface to any CAD engine and created a CFI module to go along side the OpenCascade one. This method allows us to have both CAD engines within NekMesh. The use of either system is simply dependent on whether it is present on the given machine. From the point of view of the mesh generation algorithms within NekMesh, nothing has changed because all the differences between OpenCascade and CFI are below the CAD wrapper level. Therefore we were able to achieve an easy integration of CFI as the CAD back-end with all NekMesh modules and routines.

With the integration of CFI into NekMesh, combined with NekMesh's modular nature, we now have three possible meshing pipelines which are depicted in the flowchart of Fig. 1. NekMesh can construct the linear mesh followed by its high-order routines to produce a high-order mesh using either OpenCascade or CFI as the CAD back end. The use of CFI is advantageous over OpenCascade because of the ability to read directly from a CADfix file or an open CADfix session. Alternatively, a CADfix session can produce a linear mesh that NekMesh will read directly via CFI and process via the high-order routines. This is the method adopted here and discussed in the rest of this paper.

## 3. Linear mesh generation via the 3D medial object

The first step in our mesh generation process is to generate a linear mesh using CADfix. Although CADfix is a commercial tool, and its off-the-shelf capabilities are used for importing, preparing and interrogating geometry, the 3D medial object based partitioning and linear mesh generation utilises results from recent research projects [16]. In this work, the partitioning and coarse linear meshing have been adapted specifically to generate prismatic meshes suitable for promotion to high order. The medial object and partitioning capabilities of CADfix are not yet commercially available, and development work on these capabilities is ongoing.

The pipeline for generating the linear mesh within CADfix consists of several automatic, semi-automatic and manual tools. First, the geometry must be prepared, to define a valid domain, and repair any CAD defects. Next, we perform an automatic partitioning of the domain, using the 3D medial object, to decompose the domain. Finally, a coarse linear mesh is generated in each partition. Each step of this process has been adjusted to specifically generate linear meshes suitable for *a posteriori* high-order mesh generation, and these steps are described in more detail in the following sections.

### *3.1. Geometry preparation*

Commonly, the starting point for a CFD analysis is a CAD model which was not designed specifically for CFD. We require a *CFD-ready CAD geometry*: a definition of a fluid domain as a watertight CAD solid. However the starting geometry is often non-watertight, may have defects in the CAD geometry, and usually lacks a definition for the outer domain boundaries. CADfix can import CAD geometry from a wide range of design systems, and provides automatic, diagnostic-driven, and manual tools for repairing models with poor quality CAD geometry, constructing bounding

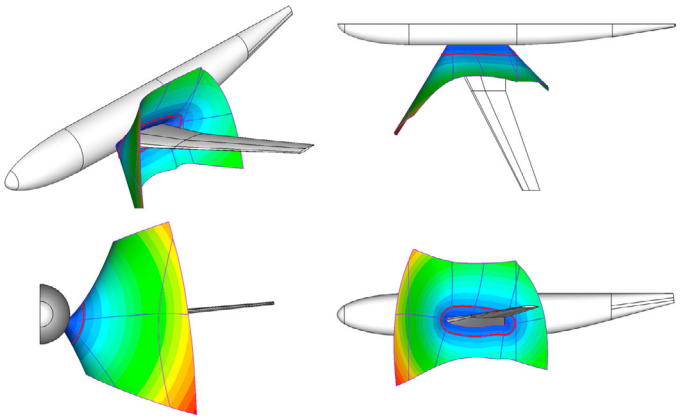


Fig. 2. Example of a 3D medial object for a fluid domain. The medial halos are the lines highlighted in red.

volumes, and establishing a well-connected, watertight model. The medial object algorithm also has requirements on the quality of the input geometry. Large edge-face and vertex-face gaps, and faces with very sharp corners, need to be corrected before meshing can proceed. As the partitioning and linear mesh generation process respect the CAD topology, it is also advisable to remove excessively short edges, and narrow sliver faces. These requirements are not dissimilar to those imposed by standard surface and volume meshing algorithms, and can typically be detected and fixed automatically.

### *3.2. 3D medial object*

The medial axis was introduced by Blum [17] as a method for analyzing shapes. For a fluid domain, it can be defined as the set of all points in the domain which have more than one closest point on the boundary of the domain. If these points are taken together with their distance to the domain boundary (the *medial radius*), they form a complete description of the flow domain. We compute the medial axis as a non-manifold CAD object, with relationships stored between the components of the medial object, and the defining components of the domain boundary. See Fig. 2 for an example 3D medial object of a fluid domain. Robustly computing the 3D medial object has been a long-standing challenge for the CAE community, as it has significant applications in structured meshing, mid-surfacing, and feature recognition, as well as automatic partitioning. Our algorithm is based on a domain Delaunay triangulation [18], and recent developments [19] allow it to work robustly on a wide range of production CAD models, or the air volume around such models.

### *3.3. Partitioning using the 3D medial object*

We construct our partitions by first computing the 3D medial object, and use this to robustly generate an offset surface, or *shell*, from the boundaries of the fluid domain. The medial object is used to find lines where simply offsetting the CAD faces would cause the shell to self-intersect, known as *medial halos* (Fig. 2). The shell (Fig. 3) divides the fluid domain into two partitions: one near-field partition close to the boundary, and one far-field partition. The near-field partition is subsequently divided into multiple smaller partitions, by dividing along feature lines from the CAD model.

Where the fluid domain has a sharp concave edge or corner (for example, at a wing/fuselage junction), flows will occur with potentially large velocity gradients in two or three directions. The ideal mesh in these areas therefore requires elements aligned with these principal directions. There are several options available within the medial object methodology to achieve a suitable mesh.

The medial object and medial halos can be used to resolve concavities in wing root junctions, allowing for better mesh alignment when using hexahedral meshing. This leads to an H-type topology, similar to those constructed with a structured multiblock system, which is illustrated in Fig. 4(a). However, this topology is not ideal for the

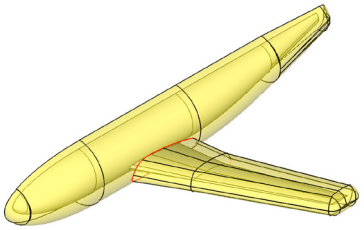


Fig. 3. A ”shell” around the aircraft geometry that divides the domain into two partitions: near-field (close to body) and far-field (away from body). The near-field partition will be used to generate a prismatic boundary-layer mesh and a tetrahedral mesh is generated in the far-field partition.

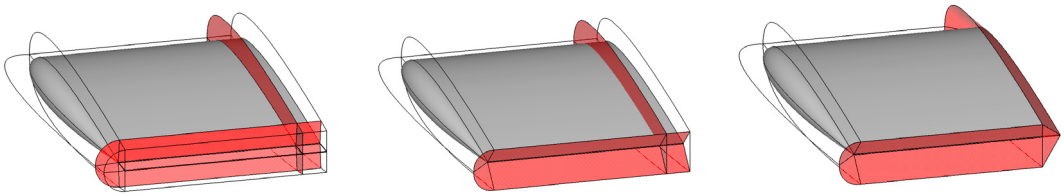



Fig. 4. An illustration of the different shell structures designed to create: (a) H-type, (b) C-type, and (c) O-type topologies. The O-type topology has been adopted here to generate a prismatic boundary-layer linear mesh.

highly stretched meshes we need for high-order meshing. We can instead move to a C-type topology without squaring off as shown in Fig. 4(b). This style of blocking removes the need for hexahedral elements in the trailing edge, replacing them with prismatic elements, but this topology still needs hexahedral elements in the wing root junction block to produce a blocking that can be meshed. Therefore, for the purposes of generating the highly stretched meshes required to simulate near-wall flows, a different partition structure has been specifically designed to create a O-type topology, as shown in Fig. 4(c), to allow a structured prism dominant linear mesh, removing the need for hexahedral meshing in junction regions. This is the topology adopted in the following to generate linear boundary-layer meshes.

### *3.4. Generation of the linear mesh*

The partitioning of the flow domain has been done with particular mesh styles in mind. The O-type near field shell topology allows swept meshing to be completed with a prismatic mesh style in all partitions, with one element generated through the thickness in both directions. Finally, the far field is suitable for tetrahedral meshing which will interface exactly with the triangle swept faces of the prismatic mesh.

To ensure a fully conformal mesh between partitions, we follow a bottom-up mesh generation process. First the lines of the partitions are meshed, then the faces are meshed with elements conforming to the lines, and finally the partitions themselves are volume meshed with elements conforming to the faces. When a structured junction partition is present, the line meshes must be *balanced* to satisfy the rules imposed by a structured mesh style; this is solved as an integer programming problem [20], and solved using an off-the-shelf solver [21]. To ensure good quality in the final mesh, the line meshes are also aligned, by performing a least-squares optimisation to reduce skew between line nodes.

The template faces for the swept partitions are meshed with a Delaunay triangulation. Sizing for the surface meshes is calculated based on a simple turn angle of 30 degrees, to generate a coarse linear mesh. These are then swept into prismatic elements using the CADfix sweep mesher. Once the mesh is completed, a mesh quality test is run to ensure all of the elements produced during the linear mesh stage are not inverted.

## 4. High-order meshing

The *a posteriori* generation of a high-order mesh from a linear mesh proceeds in a bottom-up fashion following the ideas proposed in reference [9]. The additional points required for the high-order polynomial discretization are incorporated sequentially first along the curves, then on the surfaces of the CAD geometry and, finally, in the interior of the domain. The generation of points along the curves is essentially the one proposed in reference [9], the following sections describe the improvements that we have incorporated into the methodology to achieve the type of meshes sought in this work.

### *4.1. Surface optimisation*

High-order surface optimisation to overcome CAD distortion is the key stage in the generation of a high-order mesh. We know that small inaccuracies in the representation of the geometric boundary have a large impact on the flow solution. These inaccuracies include: highly distorted surface elements, i.e. very high curvature in the mesh entities; mesh nodes being a significant distance from the true CAD surface; and under-representation of the geometric curvature due to using a too low polynomial order with too large a element. If the vertex locations of the linear surface mesh are taken to be fixed, producing a high-order surface can be accomplished simply by using an affine mapping of the triangle in the 2D parameter plane to the reference triangle of a high-order element. This can then be used to locate the new high-order nodes in the parameter space, which are then projected into 3D using the CAD engine. However, this means that the high-order triangles will inherit the distortion of the CAD surface, lowering the quality of the mesh and in some cases causing invalid elements. To address this issue, NekMesh is equipped with a method to take the high-order surface mesh made using the affine mapping approach and optimise the location of the high-order nodes to reduce distortion. This is done by modelling the mesh entities as spring networks and minimising the spring energy, in a similar approach to the work of [9].

First we optimise mesh edges that lie on curves; then edges that lie on surfaces; and finally interior triangle faces that lie on CAD surfaces. In the first case (edges on CAD curves), the problem is a 1D optimisation of spring system in the curve's parameter space The vertices in the linear mesh are considered to be fixed.

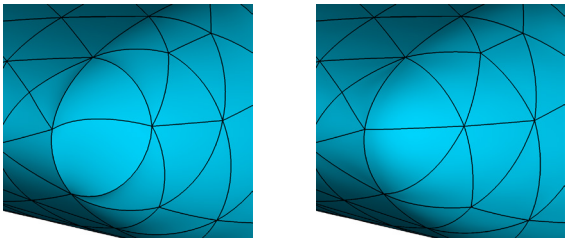
Performing the optimisation of the edges which lie on the CAD surfaces follows a very similar procedure but is formulated in the 2D parameter plane. This procedure reduces the distortion found in the high-order edges by minimising the length of the edge; that is, the optimised high-order edge will lie approximately on the geodesic between the two end points on the surface.

The procedure for optimising the location of face interior nodes requires a slightly alternative approach. The system is considered as a set of freely movable nodes, consisting of those nodes lying on the interior of the triangle, and a set of fixed nodes which lie on the edges. Each of the free nodes is connected to a system of six surrounding nodes by springs, and this is the system which is minimised. To optimise the energy of the system a bounded version of the BFGS algorithm is used. This bounding is necessary due to the limits of the parameter space in the CAD entities [22]. One of the primary advantages of this approach is the availability and low cost of computing analytical gradients.

Fig. 5 shows the effectiveness of this optimisation procedure. The left-hand figure shows the surface mesh before optimisation, and the right-hand figure after optimisation of the spring networks. In this case, the highly distorted CAD surface of the rounded leading edge of a wingtip causes suboptimal surface mesh generation. The figure clearly shows that the high-order triangles are deformed under the linear mapping. However, when this optimisation procedure is performed, the mesh edges approximate geodesic lines better and the resulting surface mesh is smoother.

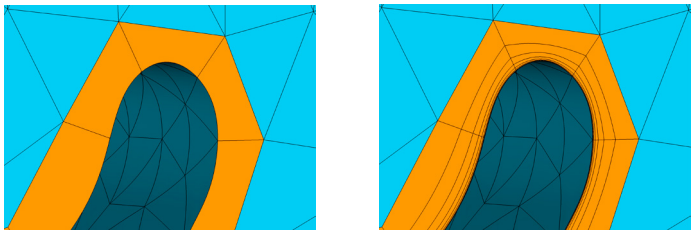
### *4.2. Boundary layer meshing*

Efficient viscous simulations at high Reynolds numbers often calls for the use of highly stretched elements with high aspect ratios of 100:1 and above in the wall-normal direction to resolve the large flow gradients in the boundary layer. This requirement poses a significant challenge for high-order mesh generation since imposing surface curvature on thin elements will almost certainly yield self-intersecting elements in regions of high curvature. To deal with this issue we adopt the isoparametric approach [14] to the linear meshes produced via the medial object to generate high-order highly stretched boundary layer meshes.



(a) (b)

Fig. 5. The high-order surface created using the affine mapping with (b) and without (a) optimisation in a region of high distortion in the CAD surface.



(a) (b)

Fig. 6. Isoparametric approach: (a) macro high-order prismatic elements in the near wall region; (b) subdivision along the normal to produce elements of high aspect ratio.

Firstly, a macro boundary-layer mesh of prisms is produced by the medial object method in the near-field region, i.e. near the walls. The far-field partition is discretized into tetrahedra. The macro boundary-layer mesh consist of a single layer of prisms. The use of 3D medial object partitioning to generate this prism layer allows the prism height to be much greater than can be achieved by most commercial mesh generators.

The generation of the high-order surface will curve one of the triangular faces of the prism. To reduce the likelihood of generating invalid elements, it is important to select a thickness of the shell, or equivalently a height of the prisms, that amply accommodates the surface curvature. Nevertheless, in those rare occasions where invalid elements exist, the domain interior deformation could be used to improve the quality of the mesh, as described in the next section. The high-order boundary-layer meshes presented here have not required it.

Now, since the height of the macro-layer can readily accommodate the curvature of the surface, the prisms in the boundary-layer mesh can be transformed into valid high-order elements. The next step in volume generation is to split these high-order prisms using the isoparametric approach [14]. The basic idea is to take advantage of the presence of a bijective mapping between a reference element and the physical space to introduce subdivisions, according to a user-defined criterion, of the reference element along the height to generate layers along the normal in the physical space. This allows for the creation of very thin boundary layer elements without self intersection if the mapping satisfies certain restrictions which are discussed in detail in reference [14]. The generation of the linear macro-prism layer represents one of the newest features of the proposed approach. The generation of the boundary-layer mesh by this approach is illustrated in Figure 6 which shows a boundary layer region of macro-prisms split to produce highly curved valid boundary-layer elements with very high aspect ratios.

### *4.3. Volume meshing*

Incorporating the curvature of the CAD surfaces onto the high-order surface triangulation produces high-order elements in the interior of the volume with curved faces and edges. Sometimes, this leads to self-intersection in the elements that makes them unsuitable for computation. For the geometries and polynomial orders presented here, the "thick" prismatic boundary-layer meshes generated via the medial object are able to accommodate curvature without producing any invalid elements. The introduction of curvature in the far-field region is not an issue since the far-field boundary is chosen to be convex. The placement of additional nodes required for the polynomial representation of the

high-order elements is accomplished by means of a standard mapping between a reference element and the physical element which accounts for the presence of curvature on its faces and edges lying on the CAD definition, whilst the other edges are straight and their faces planar.

However, should any invalid elements arise through the introduction of CAD surfaces curvature in more complex cases than those presented here, they could be made valid and their quality improved through the use of the variational approach for mesh enhancement and untangling proposed in reference [11], which is available in NekMesh generation pipeline. The method is very efficient and could also be used to improve the quality of valid high-order meshes, but it has made little difference to the meshes produced by the method we propose since their quality is very high.

## 5. Examples of Application

This section presents an illustration of the proposed mesh generation methodology and the high-order meshes it produces using three geometries proposed for CFD validation: a NACA wing tip, the Boeing rudimentary landing gear and the NASA Common Research Model. These are described in the following sections and, for reference, all the corresponding high-order meshes have been generated using a polynomial order $P = 4$.

### *5.1. NACA0012 wing tip*

To illustrate the various steps of the previously described pipeline for constructing a high-order mesh, we will employ a simple geometrical domain that consists of an unswept wing of rectangular planform composed of NACA0012 aerofoil sections and a round tip, essentially a wing tip, enclosed in a rectangular box. This geometry is also of aerodynamic interest as a case study of vortex roll-up proposed and experimentally measured by Chow et al. [23] which has been used in CFD validation studies, see for instance [24].

We proceed first with the calculation of the medial object followed by decomposition of the computational domain. As a way of illustration, Fig. 7(a) depicts a medial-object interface at the junction between the wing and the symmetry plane, and Fig. 7(b) shows the partitions in the boundary-layer region around the wing.

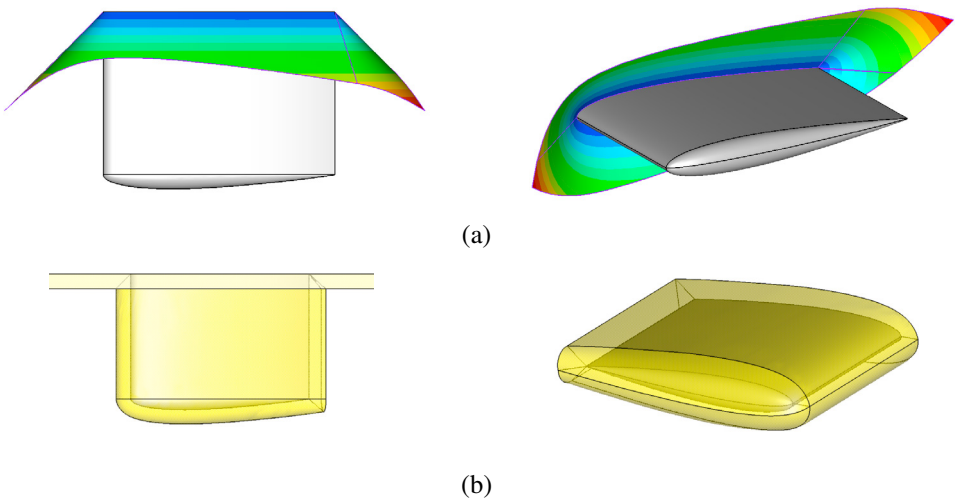



Fig. 7. Medial object decomposition for the NACA wing tip: (a) interface of the medial object near junction between the wing and the symmetry plane, and (b) partitions containing the boundary-layer region.

The near-field region is discretized into 150 000 triangular prisms and the far-field region into 40 000 tetrahedra. An indication of the mesh resolution used is given in Fig. 8 which shows the curved edges of the high-order surface mesh.

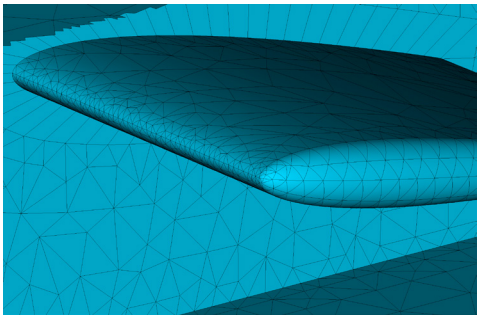
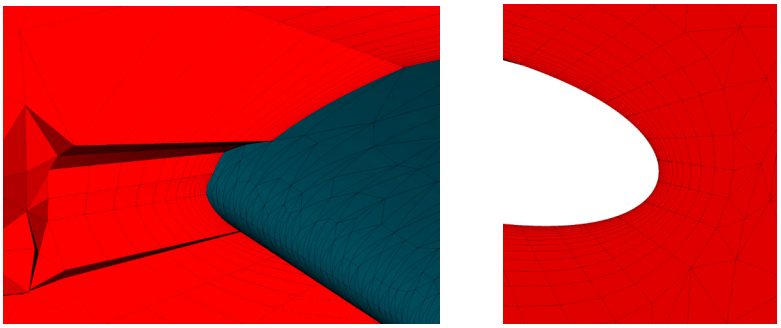


Fig. 8. High-order surface triangulation of the NACA wing tip. The edges of the mesh are curved to conform to the geometry, but the mesh nodes are not displayed to avoid cluttering the view.

The prismatic mesh in the near-field is then split into 15 layers to generate a boundary-layer mesh with highly stretched elements. A close-up view of this mesh is shown in Fig. 9, where only the curved edges are drawn for clarity.



(a) (b)

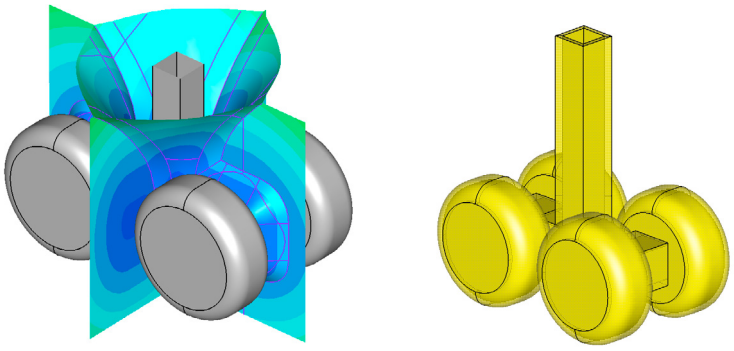
Fig. 9. NACA wing tip: close-up of the curvilinear boundary-layer mesh near the leading edge (a), and in the symmetry plane (b). The interior points are not shown.

### *5.2. Boeing rudimentary landing gear*

We consider now a four-wheel "rudimentary" landing gear (RLG) truck geometry which was designed by Philipe Spalart at Boeing [25] for public-domain research to serve as a benchmark problem for airframe noise computations. The RLG model has a relatively simple topology with four main components, the vertical post, truck, wheels and the adapter. These components are made up of curves and surfaces defined through analytical expressions. Here we have used an alternative definition via NURBS that represent them exactly.

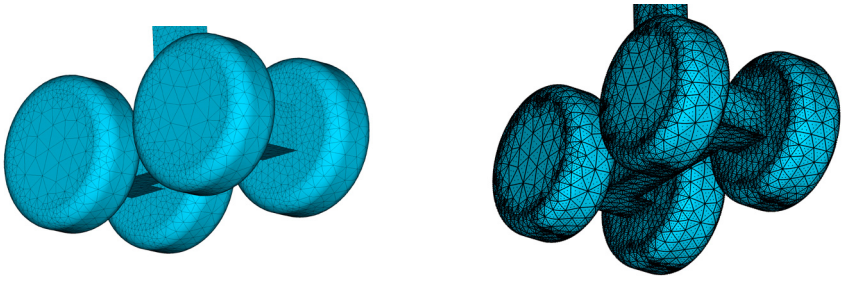
The medial object designed to achieve an O-type topology around the wheels for this configuration is shown in Fig. 10(a). This topology permits the subsequent decomposition of the domain into blocks as depicted in Fig. 10(b).

These blocks are used to generate a boundary-layer mesh with 9 600 triangular prismatic elements. This mesh contains a single layer of these elements. The rest of the domain is discretized into 80 000 tetrahedra. Fig. 11 shows an enlarged view of the high-order surface mesh in the vicinity of the wheels. The coarse boundary-layer mesh is subsequently split into 10 layers along the surface normal, to produce a much finer resolution in that direction. This is illustrated in Fig. 12 which shows a close-up of both the linear and high-order meshes near the shoulder of the wheel.



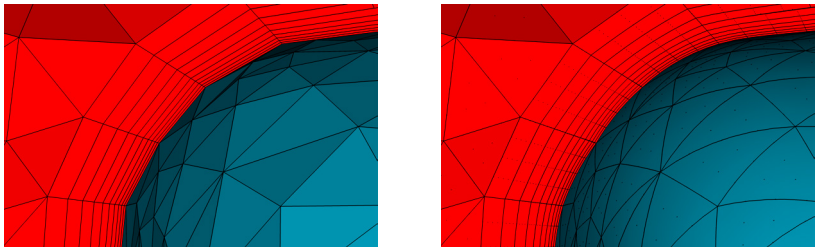
(a) (b)

Fig. 10. Boeing RLG geometry: (a) medial object, and (b) block decomposition.



(a) (b)

Fig. 11. Boeing RLG high-order surface mesh: (a) curved edges; (b) incorporating mesh nodes ($P = 4$).



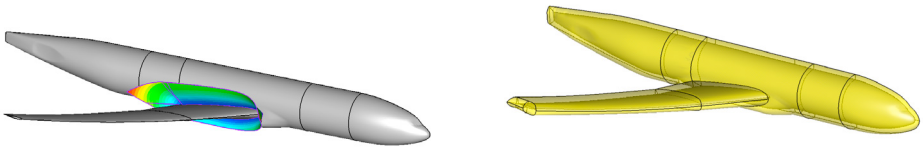
(a) (b)

Fig. 12. Enlargement of boundary layer mesh near the shoulder of the Boeing RLG wheel: (a) straight-sided mesh; (b) high-order mesh.

### *5.3. NASA Common Research Model*

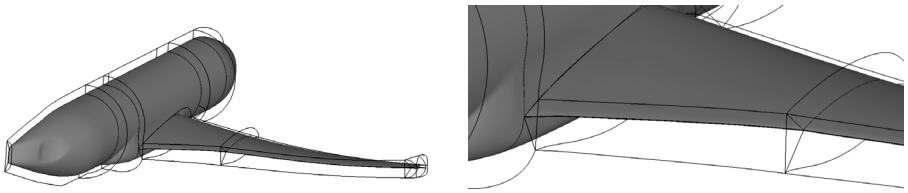
The Common Research Model (CRM) is a conventional configuration designed by NASA [26] to produce a database of experimental results for CFD validation. We have chosen the wing/body alone configuration of the five available. This configuration consists of a fuselage with a maximum radius of about 0.17m, and a 35 degrees backward-swept wing with a span of 1.60m, and an aspect ratio of 9. The original definition of the CAD geometry in STEP format can be found in the repository [27]. Further description of the NASA common research model and its characteristics can be found in [26] and the repository's webpage.

Unlike the previous geometries, the first step of the process was to use CADfix to thoroughly clean the CAD geometry and fix a number of inconsistencies and severe distortions present to ensure it was useable for both the generation of the medial object and the high-order mesh. The customary medial object interface at the wing-fuselage



(a) (b)

Fig. 13. NASA CRM medial object: (a) interface of the medial object at the wing-fuselage junction, and (b) partitions in the near-field region.



(a) (b)

Fig. 14. A wireframe representing the edges of the partitions in the near-field region: (a) global view of blocks around wing and fuselage, and (b) close-up near the wing.

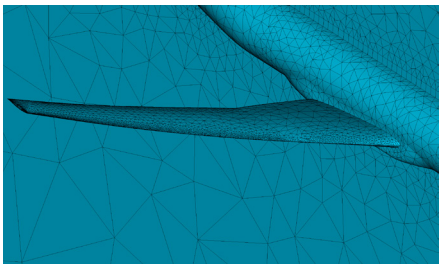
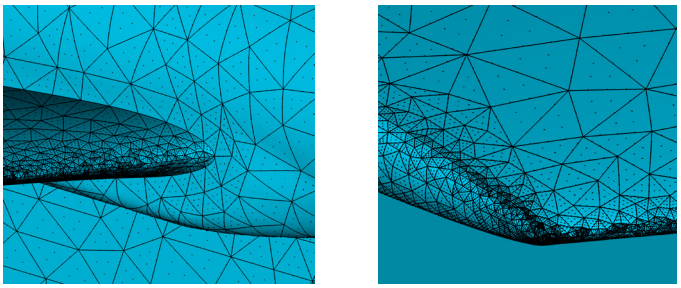


Fig. 15. CRM high-order surface mesh overview. Only the curved edges of the mesh are shown.

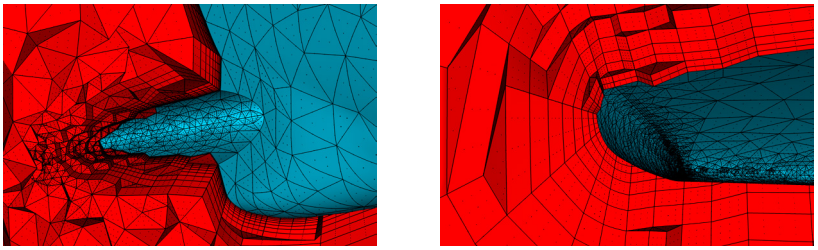


(a) (b)

Fig. 16. CRM high-order surface mesh enlargement near: (a) leading edge at the wing-fuselage junction, and (b) the wing tip leading edge.

junction and the block partitioning near the aircraft are depicted in Figs. (a) and (b), respectively. Fig. 14 aims at providing a better illustration of the blocks in the near-field region through a wireframe representing the edges of the partitions in that region.

The initial coarse, single-layer, mesh of the near-field partition consisted of 20 000 triangular prisms. Fig. 15 shows the curved edges of the CRM high-order surface mesh. A more detailed view of the CRM high-order surface mesh, including interior points, is given in Fig. 16 which includes close-ups near leading edge at the wing-fuselage junction, and at the wing tip leading edge.



(a) (b)

Fig. 17. Close-up of the CRM boundary layer mesh in the regions adjacent to: (a) the wing-fuselage junction, and (b) the wing tip.

Finally, the coarse boundary-layer mesh is split into 10 layers. Fig. 17 shows enlargements of that mesh in the regions adjacent to the wing-fuselage junction, and the wing tip.

## 6. Conclusions

We have addressed the challenge of generating high-order meshes with high-aspect elements to efficiently simulate boundary-layer flows through the use of a novel mesh generation pipeline. This pipeline combines a mesh generator for hybrid linear, or straight-sided, meshes and an *a posteriori* high-order mesher. The linear mesh generator is based on a medial-object approach for domain decomposition into near-field and far-field regions. The linear mesh generator is based on a medial-object approach for domain decomposition into near-field and far-field regions, currently under development within CADfix. The parameters of the decomposition have been tuned to facilitate the generation of prismatic elements in the near field. The use of medial object allows this prism layer to be much taller than can be achieved using most commercially available mesh generators. The near-field region is discretized into prismatic elements, and tetrahedra are used in the near-field region. One of the contributions of the paper is the design of an O-type topology for the near-field region that facilitates its discretization into prismatic elements. To generate the high-order mesh the prismatic elements are curved to conform to the boundary. This is accomplished through CFI, the CADfix interface for geometry queries. The prismatic high-order mesh is then split using the isoparametric approach. Finally additional points are incorporated in the interior and mesh deformation could be used to improve the mesh. This was not required for the examples presented here.

In general we have found that the robustness and capability of the method we use for high-order algorithms is highly dependent on the quality of the linear mesh, of the CAD engine, and also of the input CAD itself. Our experience indicates that CAD and linear meshing technologies widely perceived to be more than adequate in academic and industrial settings can, in practice, be completely impractical for high-order meshing purposes. This is mainly due to the coarseness requirements for the linear mesh and the higher sensitivity of high-order algorithms to distortions in the mappings defining the CAD curves and surfaces. Through the use of CADfix, and its interface CFI, for fixing CAD information, generating appropriate linear meshes and facilitating the interface for the geometric enquiries required for high-order mesh generation, we have been able to greatly improve our rate of success at producing high-order meshes with complex geometries.

Although the chosen O-type medial object decomposition is simpler to obtain than a general multi-block decomposition, it is sufficiently flexible to deal with reasonably complex geometries using a relatively high polynomial order ($P = 5$) and without the need to employ mesh deformation to improve mesh quality. We anticipate however that configurations of increased complexity will require a slightly more general approach. One avenue of development of the methodology is to incorporate H-type regions at the boundary layer. This will allow the use of hexahedral elements in regions like the wing-fuselage junction and will result in a higher quality boundary-layer mesh. In the isoparametric context, this will require to split the hexahedra in two directions simultaneously. This is not part of the current implementation, but it can be extended to incorporate such splitting.

Another useful addition to the methodology would be to incorporate curvature on the elemental faces belonging to the interface between the near-field and far-field regions. This should be possible since CADfix represent these as CAD surfaces and we could apply the high-order surface mesh optimization to them. This would produce a higher

quality boundary-layer mesh. It is also likely to induce invalid elements in the tetrahedral mesh in the far-field region, but these could be untangled and their quality improved through the variational approach [11].

## Acknowledgements

Mike Turner acknowledges funding from Airbus and EPSRC under an industrial CASE studentship. David Moxey acknowledges support from the EU Horizon 2020 project ExaFLOW (grant 671571) and the PRISM project under EPSRC grant EP/L000407/1.

## References

[1] Z. Xie, R. Sevilla, O. Hassan, K. Morgan, The generation of arbitrary order curved meshes for 3D finite element analysis, Computational Mechanics 51 (2013) 361–374.
[2] R. Hartmann, T. Leicht, Generation of unstructured curvilinear grids and high-order discontinuous Galerkin discretization applied to a 3D high-lift configuration, International Journal for Numerical Methods in Fluids 82 (2016) 316–333.
[3] P.-O. Persson, J. Peraire, Curved mesh generation and mesh refinement using Lagrangian solid mechanics, in: 47th AIAA Aerospace Sciences Meeting and Exhibit, Orlando (FL), USA, 2009. AIAA paper 2009–949.
[4] R. Poya, R. Sevilla, A. J. Gil, A unified approach for a posteriori high-order curved mesh generation using solid mechanics, Computational Mechanics 58 (2016) 457–490.
[5] D. Moxey, D. Ekelschot, Ü. Keskin, S. Sherwin, J. Peiró, A thermo-elastic analogy for high-order curvilinear meshing with control of mesh validity and quality, Procedia Engineering 82 (2014) 127–135.
[6] M. Fortunato, P.-O. Persson, High-order unstructured curved mesh generation using the Winslow equations, Journal of Computational Physics 307 (2016) 1–14.
[7] S. Dey, R. O'Bara, M. Shephard, Curvilinear mesh generation in 3D, in: Proceedings of the 8th International Meshing Roundtable, 1999. South Lake Tahoe, California.
[8] T. Toulorge, C. Geuzaine, J.-F. Remacle, J. Lambrechts, Robust untangling of curvilinear meshes, Journal of Computational Physics 254 (2013) 8–26.
[9] S. Sherwin, J. Peiró, Mesh generation in curvilinear domains using high-order elements, International Journal for Numerical Methods in Engineering 53 (2002) 207–223.
[10] A. Gargallo-Peiró, X. Roca, J. Peraire, J. Sarrate, Defining quality measures for validation and generation of high-order tetrahedral meshes, in: Proceedings of the 22nd International Meshing Roundtable, Springer International Publishing, 2014, pp. 109–126.
[11] M. Turner, J. Peiró, D. Moxey, Curvilinear mesh generation using a variational framework, 2017. Accepted for publication in Comput. Aided Design.
[12] ITI-Global, CADfix: CAD translation, healing, repair, and transformation, 2017.
[13] C. D. Cantwell, D. Moxey, A. Comerford, A. Bolis, G. Rocco, G. Mengaldo, D. de Grazia, S. Yakovlev, J.-E. Lombard, D. Ekelschot, B. Jordi, H. Xu, Y. Mohamied, C. Eskilsson, B. Nelson, P. Vos, C. Biotto, R. M. Kirby, S. J. Sherwin, Nektar++: An open-source spectral/hp element framework, Computer Physics Communications 192 (2015) 205–219.
[14] D. Moxey, M. D. Green, S. J. Sherwin, J. Peiró, An isoparametric approach to high-order curvilinear boundary-layer meshing, Comput. Meth. Appl. Mech. Eng. 283 (2015) 636–650.
[15] OPEN CASCADE, https://www.opencascade.com, Accessed May 2017.
[16] J. H. Bucklow, R. Fairey, M. R. Gammon, An automated workflow for high quality CFD meshing using the 3D medial object, in: 23rd AIAA Computational Fluid Dynamics Conference, Denver, Colorado, 2017. AIAA 2017-3454.
[17] H. Blum, A transformation for extracting new descriptors of shape, Models for the Perception of Speech and Visual Form 5 (1967) 362–380.
[18] D. Sheehy, Medial surface computation using a domain Delaunay triangulation, Ph.D. thesis, Queen's University of Belfast, 1994.
[19] H. Bucklow, 3D medial object computation using a domain Delaunay triangulation, 2014. In Medial Object Technology Workshop.
[20] T. Tam, C. G. Armstrong, Finite element mesh control by integer programming, International Journal for Numerical Methods in Engineering 36 (1993) 2581–2605.
[21] Software package `lpsolve`, http://lpsolve.sourceforge.net, 2017. Ver. 5.5.
[22] R. H. Byrd, P. Lu, J. Nocedal, C. Zhu, A limited memory algorithm for bound constrained optimisation, SIAM Journal on Scientific Computing 16 (1995) 1190–1208.
[23] S. Chow, G. Zilliac, P. Bradshaw, Mean and turbulence measurements in the near field of a wingtip vortex, AIAA Journal 35 (1997) 1561–1567.
[24] J.-E. W. Lombard, D. Moxey, S. J. Sherwin, J. F. A. Hoessler, S. Dhandapani, M. J. Taylor, Implicit large-eddy simulation of a wingtip vortex, AIAA Journal 54 (2016) 506–518.
[25] P. Spalart, K. Mejia, Analysis of experimental and numerical studies of the rudimentary landing gear, in: Proceedings of the 49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, Orlando (FL), USA, 2011. AIAA 2011–355.
[26] J. C. Vassberg, M. A. DeHaan, S. M. Rivers, R. A. Wahls, Development of a common research model for applied CFD validation studies, in: 26th AIAA Applied Aerodynamics Conference, Honolulu, Hawaii, 2008. AIAA-2008-6919.
[27] Common Research Model geometry repository, https://commonresearchmodel.larc.nasa.gov, Accessed May 2017.