Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 00 (2014) 000–000

Procedia Engineering

www.elsevier.com/locate/procedia

23rd International Meshing Roundtable (IMR23)

Face-centred Voronoi refinement for surface mesh generation

Darren Engwirda[a,**], David Ivers[a]

[a] *School of Mathematics and Statistics F07, University of Sydney, NSW 2006, Australia*

Abstract

A Frontal-Delaunay surface meshing algorithm for closed 2-manifolds embedded in $\mathbb{R}^3$ is presented. This new algorithm is an extension of existing restricted Delaunay-refinement techniques, in which the point-placement scheme is modified to improve element quality in the presence of mesh size constraints. Specifically, it is shown that the use of *off-centre* Steiner vertices, positioned along facets in the associated Voronoi diagram, typically leads to an improvement in the shape- and size-quality of the resulting surface tessellation. The new method can be viewed as a hybridisation of conventional Delaunay-refinement and advancing-front techniques, in which new vertices are positioned to satisfy both element size and shape constraints. It is shown that by restricting point-placement to faces of the associated Voronoi diagram, the new Frontal-Delaunay algorithm maintains many of the theoretical guarantees commonly associated with conventional restricted Delaunay-refinement techniques. The performance of the new Frontal-Delaunay scheme is investigated experimentally, via a series of comparative studies designed to contrast the performance of the new algorithm with a typical Delaunay-refinement technique. It is shown that the new Frontal-Delaunay algorithm inherits many of the benefits of both Delaunay-refinement and advancing-front type methods, typically leading to the construction of very high quality triangulations in practice. Experiments are conducted using a range of complex benchmarks, verifying the robustness and practical performance of the proposed scheme.

Keywords: Surface mesh generation, Delaunay-refinement, Frontal-Delaunay, off-centres

1. Introduction

Surface mesh generation is a core element of numerous computational modelling and simulation tasks, including many forms of computational engineering and numerical modelling, a range of problems in computer graphics and animation, and various data visualisation applications. Given a geometric domain described by a bounding surface Σ embedded in $\mathbb{R}^3$, the surface *triangulation* problem consists of tessellating Σ into a *mesh* of non-overlapping triangular elements, such that all geometric and user-defined constraints are satisfied. While each use-case contributes its own set of specific considerations, it is typical to require that the surface tessellation: (i) consist of elements of high shape-quality, (ii) provide a good geometric and topological approximation to the underlying surface Σ, and (iii) satisfy a set of user-specified element size constraints. While various strategies have been developed to solve the surface meshing

[*] Corresponding author. Tel.: +1-212-678-5573.
[**] Current address: Department of Earth, Atmospheric and Planetary Sciences, Room 54-1517, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139-4307.
E-mail address: d.engwirda@maths.usyd.edu.au; engwirda@mit.edu (Darren Engwirda).

problem in the past, a new algorithm is developed in this study, designed to improve the quality of the resulting triangulations.

1.1. Related Work

Mesh generation is a broad and evolving area of research. A number of algorithms have been developed to tackle various meshing tasks, including *grid-overlay* methods [1,2] in which a mesh is generated by warping and refining a semi-structured background grid, *advancing-front* techniques [3,4,5,6] which grow a mesh incrementally in a layer-wise fashion, and *Delaunay-based* strategies [7,8,9,10,11,12,13,14,15,16] which incrementally refine an initially coarse Delaunay triangulation.

The Delaunay triangulation [17] is imbued with a number of attractive theoretical properties. In the context of mesh generation, for triangulations in $\mathbb{R}^2$ and surface tessellations embedded in $\mathbb{R}^3$, it is known that such structures are topologically optimal [18], maximising the worst-case element quality in the resulting mesh. The reader is referred to [18] for additional details. As the name suggests, Delaunay-refinement schemes are based on the incremental *refinement* of an initially coarse bounding tessellation. At each step of the algorithm, any elements that violate a set of geometric or user-defined constraints are identified and the worst offending element is *eliminated*. This is achieved through the insertion of an additional *Steiner* vertex at the refinement point of the element – either the circumcentre of the element itself, or a point on an adjacent segment of the bounding geometry. The original planar Delaunay-refinement methods of Chew [7] and Ruppert [8,9] have since been extended to support volumetric tessellations [11] and surface triangulations [14,15,13].

Delaunay-based methods have previously been applied to the surface meshing problem, via the so-called *restricted* Delaunay tessellation of Edelsbrunner and Shah [19]. Given a surface Σ embedded in $\mathbb{R}^3$ and a set of points $X \in \Sigma$, the restricted Delaunay surface triangulation $\mathrm{Del}\,|_\Sigma(X)$ is a sub-complex of the full-dimensional Delaunay tessellation $\mathrm{Del}(X)$, containing the 2-faces $f \in \mathrm{Del}(X)$ that approximate the underlying surface. Boissonnat and Oudot [14,15] have shown that when the point-wise sampling X is sufficiently *dense* and *well-distributed*, the restricted Delaunay triangulation $\mathrm{Del}\,|_\Sigma(X)$ is an accurate piecewise approximation of Σ, incorporating several theoretical guarantees of fidelity. Given such behaviour, a Delaunay-refinement algorithm for surface mesh generation proceeds as before, with an initial coarse restricted Delaunay triangulation, induced by a sparse sampling $X \in \Sigma$, incrementally refined via the introduction of new Steiner vertices positioned on the surface Σ. This process is discussed in detail in Section 2.

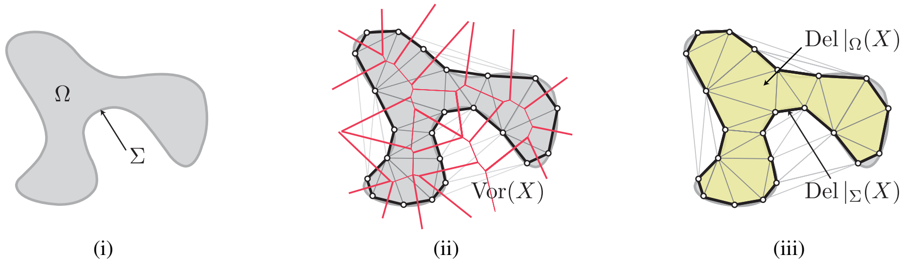
Advancing-front techniques are an alternative approach to mesh generation, in which elements are created one-by-one, starting from a set of *frontal* facets that are initialised on the boundary of the domain to be meshed. The mesh is *marched* inwards layer-by-layer, with new elements created by carefully positioning vertices adjacent to the facets in the frontal set. Following the placement of new elements, the set of frontal facets is updated to incorporate the updated mesh topology. Elements are added incrementally in this fashion until a complete tessellation of the domain is obtained. Such methods are applicable to surface meshing problems [4,5,6], although several non-trivial issues of robustness are known to exist [6]. Fundamentally, advancing-front techniques are *heuristic* in nature, and do not typically offer theoretical guarantees of convergence or bounds on element quality.

While the theoretical guarantees associated with Delaunay-refinement schemes make them a good choice for practical mesh generation, it is known that optimised advancing-front methods often produce meshes with superior element quality characteristics in practice. Such considerations often lead to the adoption of frontal-type schemes in preference to Delaunay-based approaches, despite issues of theoretical robustness. Such choices appear to be especially prevalent in the context of numerical modelling, including problems in fluid dynamics and/or structural analysis, where improvements in mesh quality can significantly enhance the performance of the underlying numerical scheme.

2. Restricted Delaunay-refinement Techniques

Delaunay-refinement algorithms for surface meshing operate by incrementally adding new Steiner vertices to an initially coarse restricted Delaunay triangulation of the underlying surface. Contrary to typical planar algorithms [7,8,9], refinement schemes for surface meshing are designed not only to ensure that the resulting mesh satisfies element shape and size constraints, but that the geometry and topology of the mesh itself is an accurate piecewise approximation of the underlying surface.

Fig. 1. Restricted tessellations for a curved domain in $\mathbb{R}^2$, showing (i) the curve Σ and enclosed area Ω, (ii) the Delaunay tessellation $\mathrm{Del}(X)$ and Voronoi diagram $\mathrm{Vor}(X)$, and (iii) the restricted curve and area tessellations $\mathrm{Del}\,|_\Sigma(X)$ and $\mathrm{Del}\,|_\Omega(X)$.



Before discussing the details of Delaunay-refinement algorithms for surface meshing problems, a number of important theoretical concepts are introduced:

Definition 2.1 (restricted Delaunay tessellation)**.** Let Σ be a closed 2-manifold embedded in $\mathbb{R}^3$, enclosing a volume $\Omega \subset \mathbb{R}^3$. Let $\mathrm{Del}(X)$ be a full-dimensional Delaunay tessellation of a point-wise sample $X \subseteq \Sigma$ and $\mathrm{Vor}(X)$ be the Voronoi diagram associated with $\mathrm{Del}(X)$. The *restricted Delaunay surface triangulation* $\mathrm{Del}\,|_\Sigma(X)$ is a sub-complex of $\mathrm{Del}(X)$ including any 2-face $f \in \mathrm{Del}(X)$ associated with an edge $\mathbf{v}_f \in \mathrm{Vor}(X)$ such that $\mathbf{v}_f \bigcap \Sigma \neq \emptyset$. The *restricted Delaunay volume triangulation* $\mathrm{Del}\,|_\Omega(X)$ is a sub-complex of $\mathrm{Del}(X)$ including any 3-simplex $\tau \in \mathrm{Del}(X)$ associated with an *internal* circumcentre $\mathbf{c} \in \Omega$.

It has been shown by a number of authors, including Cheng, Dey, Levine, Ramos and Ray [20,13,21] and Boissonnat and Oudot [14,15] that given a *sufficiently dense* point-wise sampling of the surface $X \in \Sigma$ the restricted Delaunay tessellation $\mathrm{Del}\,|_\Sigma(X)$ is guaranteed to be both geometrically and topologically representative of the underlying surface Σ. Specifically, it is known that, when the restricted tessellation is a so-called *loose γ-sample* [14,15], $\mathrm{Del}\,|_\Sigma(X)$ is homeomorphic to the surface Σ, the Hausdorff distance $\mathrm{H}(\mathrm{Del}\,|_\Sigma(X), \Sigma)$ is small and that $\mathrm{Del}\,|_\Sigma(X)$ provides a good piecewise approximation of the geometrical properties of Σ, including its normals, area and curvature. Similar theoretical guarantees extend to the associated restricted volume tessellation $\mathrm{Del}\,|_\Omega(X)$. The properties of restricted Delaunay tessellations are well documented in the literature, and a full account is not given here. The reader is referred to the detailed expositions presented in [14,15] and [18] for further details and proofs.

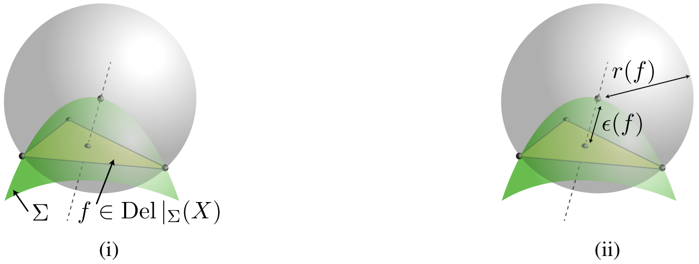
Definition 2.2 (surface Delaunay ball)**.** Let $\mathrm{Del}\,|_\Sigma(X)$ be a restricted Delaunay triangulation of a 2-manifold Σ. Given a simplex $f \in \mathrm{Del}\,|_\Sigma(X)$, any circumball of f centred on the surface Σ is a *surface Delaunay ball* of f, denoted $\mathrm{SDB}(f)$.

Surface Delaunay balls are centred at the intersection of the Voronoi diagram $\mathrm{Vor}(X)$ with the surface Σ. Specifically, each 2-face $f \in \mathrm{Del}\,|_\Sigma(X)$ is associated with an orthogonal edge in the Voronoi diagram $\mathbf{v}_f \in \mathrm{Vor}(X)$, which is guaranteed, by definition, to intersect with the surface Σ *at least* once. In some cases, especially when the sampling $X \in \Sigma$ is relatively coarse, there may be multiple surface balls associated with any given face $f \in \mathrm{Del}\,|_\Sigma(X)$. In such cases, it is typical to consider only the largest surface ball. See Figure 2 for an illustration of the surface Delaunay ball for a general facet f.

Definition 2.3 (surface discretisation error)**.** Let $\mathrm{Del}\,|_\Sigma(X)$ be a restricted Delaunay surface triangulation of a 2-manifold Σ. Given a simplex $f \in \mathrm{Del}\,|_\Sigma(X)$, the *surface discretisation error* $\epsilon(f)$ is the Euclidean distance between the centres of the largest surface Delaunay ball of f and its diametric ball.

The surface discretisation error is a measure of how well the restricted triangulation $\mathrm{Del}\,|_\Sigma(X)$ approximates the underlying surface Σ geometrically. Considering a face $f \in \mathrm{Del}\,|_\Sigma(X)$, the surface discretisation error $\epsilon(f)$ is a measure of the distance between the face f and the furthest associated point on the surface Σ. This measure can be thought of as a one-sided discrete Hausdorff distance, $\epsilon(f) = \mathrm{H}(\mathrm{Del}\,|_\Sigma(X), \Sigma)$, defined from a set of representative points on the triangulation $\mathrm{Del}\,|_\Sigma(X)$ to the surface Σ. Clearly, if the triangulation $\mathrm{Del}\,|_\Sigma(X)$ is a good piecewise representation of the surface Σ, the surface discretisation error $\epsilon(f)$ should be small. See Figure 2 for a description of the surface discretisation error for a facet $f \in \mathrm{Del}\,|_\Sigma(X)$.

Fig. 2. The surface Delaunay ball for a restricted 2-face $f \in \mathrm{Del}\,|_\Sigma(X)$, showing (i) placement of the surface ball at the intersection of the dual edge $\mathbf{v}_f \in \mathrm{Vor}(X)$ and the surface Σ, and (ii) associated radius $r(f)$ and surface discretisation error $\epsilon(f)$.



(i)

(ii)

Definition 2.4 (radius-edge ratio)**.** The *radius-edge* ratio $\rho(\tau)$ for a d-simplex τ is $R/\|\mathbf{e}_{\mathrm{min}}\|$, where R is the radius of the diametric ball of τ and $\|\mathbf{e}_{\mathrm{min}}\|$ is the length of its shortest edge.

The radius-edge ratio is a measure of element shape-quality, and achieves a minimum for equilateral elements, such that $\rho(\tau) = 1/\sqrt{3}$, increasing toward $+\infty$ as elements tend toward degeneracy. For 2-simplexes, the radius-edge ratio is *robust* and can be related to the minimum plane angle θ_{min}, such that $\rho(\tau) = \frac{1}{2}(\sin(\theta_{\mathrm{min}}))^{-1}$. Due to the summation of angles in a triangle, given a minimum angle θ_{min} the largest angle θ_{max} is clearly bounded, such that $\theta_{\mathrm{max}} \leq 180^\circ - 2\theta_{\mathrm{min}}$.

Definition 2.5 (area-length ratio)**.** The *area-length* ratio $a(\tau)$ for a d-simplex τ is $A/\|\mathbf{e}_{\mathrm{rms}}\|^d$, where A is the signed area of τ and $\|\mathbf{e}_{\mathrm{rms}}\|$ is the root-mean-square of its edge lengths.

The area-length ratio is a *robust* measure of element shape-quality, and is typically normalised to achieve a score of $+1$ for ideal elements. As an element is distorted toward degeneracy $a(\tau) \to 0$, with fully inverted elements achieving a score of -1. In contrast to the radius-edge ratio, higher-dimensional generalisations of the area-length metrics are known to be *robust* to all element configurations, and, as a consequence, the area-length measure is widely used as a shape-quality metric in a variety of meshing applications. The area-length measure is used throughout the current study to catalogue the shape-quality of the various surface triangulations.

2.1. An Existing Algorithm

The development of provably-good Delaunay-refinement schemes for surface-based mesh generation is an ongoing area of research. An algorithm for the meshing of closed 2-manifolds embedded in $\mathbb{R}^3$ is presented, adapted largely from the methods presented by Boissonnat and Oudot in [14,15]. This method is largely equivalent to the CGALMESH algorithm, available as part of the CGAL package, and summarised by Jamin, Alliez, Yvinec and Boissonnat in [16]. A similar algorithm is also outlined by Cheng, Dey and Shewchuk in [18]. The algorithm presented in this section is referred to as the 'conventional' Delaunay-refinement approach, due to its direct use of circumcentre-based Steiner vertices.

In this study, attention is restricted to the development of so-called *remeshing* operations, in which the underlying surface Σ is specified as an existing manifold triangular complex $\mathcal{P}$. While this constraint may seem overly restrictive, it is important to recognise that the use of such triangulated surface representations is widespread within the computational modelling communities, associated with applications such as laser scanning, terrain modelling and some computer drawing applications. Furthermore, remeshing operations are a necessary task in many practical cases, with triangulated surface models often incorporating a number of undesirable defects, including elements of low shape quality and/or regions of unsuitable resolution. The remeshing algorithms presented in this study are designed to remedy these situations, generating new high-quality surface meshes that adhere to user defined sizing constraints. Additionally, the algorithms presented in this study require that the underlying surfaces are sufficiently *smooth*, without sharp ridges or corners. Extensions to more general classes of surfaces, including parametric and/or implicit representations and those including sharp features and constraints, is left for future investigation.

Following the approach described by Jamin et al. in [16], the Delaunay-refinement algorithm takes as input a surface domain, described by a closed 2-manifold Σ, an upper bound on the allowable element radius-edge ratio

Algorithm 2.1 Restricted Surface Delaunay-refinement

```
function DelaunaySurface($\Sigma, \Omega, \bar{\rho}, \bar{\epsilon}, h(\mathbf{x}), \mathcal{T}|_\Sigma, \mathcal{T}|_\Omega$)
    Form an initial sampling $X \in \Sigma$ such that $X$ is well-distributed on $\Sigma$.
    Form Delaunay tessellation $\mathrm{Del}(X)$.
    Form restricted objects $\mathrm{Del}|_\Sigma(X)$ and $\mathrm{Del}|_\Omega(X)$.
    Enqueue all restricted 2-simplexes $Q|_\Sigma \leftarrow f \in \mathrm{Del}|_\Sigma(X)$. A 2-simplex $f$ is enqueued if BadSimplex($f$) returns true.
    while ($Q|_\Sigma \neq \emptyset$) do                ▷ {main refinement sweeps}
        Call RefineSimplex($f \leftarrow Q|_\Sigma$)
        Update the restricted Delaunay tessellations $\mathrm{Del}|_\Sigma(X)$ and $\mathrm{Del}|_\Omega(X)$.
        Update $Q|_\Sigma$ to reflect changes to $\mathrm{Del}|_\Sigma(X)$.
    end while
    return $\mathcal{T}|_\Sigma = \mathrm{Del}|_\Sigma(X)$ and $\mathcal{T}|_\Omega = \mathrm{Del}|_\Omega(X)$
end function
```

```
function RefineSimplex($f$)                ▷ {surface refinement}
    Call SurfaceDelaunayBall($f$, $\mathrm{B}(\mathbf{c}, r)_{\max}$).
    Form new Steiner vertex $\mathbf{p}$ about $\mathrm{B}(\mathbf{c}, r)_{\max}$.
    Insert Steiner vertex $X \leftarrow \mathbf{p}$, update $\mathrm{Del}(X) \leftarrow X$.
end function
```

```
function SurfaceDelaunayBall($f$)                ▷ {surface ball}
    Form Voronoi edge $\mathbf{v}_f$ orthogonal to 2-simplex $f$.
    Form the set of associated surface Delaunay balls $\mathrm{B}(\mathbf{c}, r)_i$ for the restricted 2-simplex $f \in \mathrm{Del}|_\Sigma(X)$. Balls are centred about the set of surface intersections $\mathbf{v}_f \bigcap \Sigma \neq \emptyset$.
    return $\mathrm{B}(\mathbf{c}, r)_{\max}$, where $r_{\max}$ is maximal.
end function
```

```
function BadSimplex($f$)                ▷ {termination criteria}
    return $\rho(f) > \bar{\rho}$ or $\epsilon(f) > \bar{\epsilon}$ or $h(f) > h(\mathbf{x}_f)$
end function
```

$\bar{\rho}$, a mesh size function $h(\mathbf{x})$ defined at all points on the surface Σ and an upper bound on the allowable surface discretisation error $\bar{\epsilon}$. The input surface Σ encloses a bounded volume Ω. The algorithm returns a triangulation $\mathcal{T}|_\Sigma$ of the surface Σ, where $\mathcal{T}|_\Sigma$ is a restricted Delaunay surface triangulation of a point-wise sampling $X \in \Sigma$, such that $\mathcal{T}|_\Sigma = \mathrm{Del}|_\Sigma(X)$. As a by-product, the algorithm also returns a coarse triangulation $\mathcal{T}|_\Omega$ of the enclosed volume Ω, where $\mathcal{T}|_\Omega$ is a restricted Delaunay volume triangulation $\mathcal{T}|_\Omega = \mathrm{Del}|_\Omega(X)$. Both $\mathrm{Del}|_\Sigma(X)$ and $\mathrm{Del}|_\Omega(X)$ are subcomplexes of the full-dimensional Delaunay tessellation $\mathrm{Del}(X)$. Note that $\mathrm{Del}|_\Sigma(X)$ is a triangular complex, while $\mathrm{Del}|_\Omega(X)$ and $\mathrm{Del}(X)$ are tetrahedral complexes. The method is summarised in Algorithm 2.1.

The Delaunay-refinement algorithm guarantees, firstly, that all elements in the output surface triangulation $f \in \mathcal{T}|_\Sigma$ satisfy element shape constraints, $\rho(f) \leq \bar{\rho}$, element size constraints $h(f) \leq h(\mathbf{x}_f)$ and surface discretisation bounds $\epsilon(f) \leq \bar{\epsilon}$. Furthermore, for sufficiently small mesh size functions $h(\mathbf{x})$, the surface triangulation $\mathcal{T}|_\Sigma$ is guaranteed to be a *good* piecewise approximation of the underlying surface Σ, exhibiting both geometrical and topological convergence as $h(\mathbf{x}) \rightarrow 0$, consistent with the characteristics of restricted Delaunay tessellations outlined in Definition 2.1.

The Delaunay-refinement algorithm begins by creating an initial point-wise sampling of the surface $X \in \Sigma$. Exploiting the discrete representation available for Σ, the initial sampling is obtained as a *well-distributed* subset of the existing vertices $Y \in \mathcal{P}$, where $\mathcal{P}$ is the polyhedral representation of the surface Σ. In the next step, the initial triangulation objects are formed. In this study, the full-dimensional Delaunay tessellation, $\mathrm{Del}(X)$, is built using an incremental Delaunay triangulation algorithm, based on the Bowyer-Watson technique [22]. The restricted surface and volumetric triangulations, $\mathrm{Del}|_\Sigma(X)$ and $\mathrm{Del}|_\Omega(X)$, are derived from $\mathrm{Del}(X)$ by explicitly testing for intersections between the associated Voronoi diagram $\mathrm{Vor}(X)$ and the surface Σ. These queries are computed efficiently by storing the surface definition $\mathcal{P}$ in an aabb-tree [23]. The main loop of the algorithm proceeds to incrementally refine any 2-faces $f \in \mathrm{Del}|_\Sigma(X)$ that violate either the radius-edge, element size or surface discretisation requirements. The refinement process is priority scheduled, with triangles $f \in \mathrm{Del}|_\Sigma(X)$ ordered according to their radius-edge ratios $\rho(f)$, ensuring that the element with the *worst* ratio is refined at each iteration. Individual elements are refined based on their surface Delaunay balls, with a triangle $f \in \mathrm{Del}(X)$ eliminated by inserting the centre of the largest ball $\mathrm{B}(\mathbf{c}, r)_{\max} = \mathrm{SDB}(f)$ into the tessellation $\mathrm{Del}(X)$. This process is a direct generalisation of the circumcentre-based insertion method associated with Ruppert's algorithm for planar domains [8,9]. As a consequence of changes to the full-dimensional tessellation following the insertion of a new Steiner vertex, corresponding updates to the restricted triangulations $\mathrm{Del}|_\Sigma(X)$ and $\mathrm{Del}|_\Omega(X)$ are instigated, ensuring that all tessellation objects remain valid throughout the refinement process. The Delaunay-refinement algorithm terminates when all 2-faces $f \in \mathrm{Del}|_\Sigma(X)$ satisfy all radius-edge, size and surface discretisation thresholds, such that[1] $\rho(f) \leq \bar{\rho}$, $h(f) \leq \alpha h(\mathbf{x}_f)$ and $\epsilon(f) \leq \bar{\epsilon}$, respectively, where the element size $h(f)$ is proportional is the radius of the associated surface ball $\mathrm{B}(\mathbf{c}, r)_{\max} = \mathrm{SDB}(f)$, such that[2] $h(f) = \sqrt{3}\, r$, and the target size $h(\mathbf{x}_f)$ is sampled at the centre of the surface ball $\mathbf{c}$.

[1] The coefficient $\alpha = 4/3$, to ensure that the mean element size does not, on average, undershoot the desired target size $h(\mathbf{x}_f)$.

[2] The coefficient $\sqrt{3}$ represents the mapping between the edge length and diametric ball radius for an equilateral element. Such scaling ensures that size constraints are applied with respect to mean edge length.

3. Restricted Frontal-Delaunay Methods

Frontal-Delaunay algorithms are a hybridisation of advancing-front and Delaunay-refinement techniques, in which a Delaunay triangulation is used to define the topology of a mesh while new Steiner vertices are inserted in a manner consistent with advancing-front methodologies. In practice, such techniques have been observed to produce very high-quality meshes, inheriting the smooth, semi-structured vertex placement of pure advancing-front methods and the optimal mesh topology of Delaunay-based approaches. While Frontal-Delaunay methods have previously been used by a range of authors in the context of planar, volumetric and parametric surface meshing, including, for example studies by Üngör and Erten [24], Rebay [25], Mavriplis [26], Frey, Borouchaki and George [27], and Remacle, Henrotte, Carrier-Baudouin, Béchet, Marchandise, Geuzaine and Mouton [28], the authors are not aware of any previous investigations describing the application of such techniques to the surface meshing problem directly. The conventional advancing-front method, on the other hand, has been generalised to support surface meshing, as, for example, outlined in studies by Rypl [29,5] and Schreiner, Scheidegger, Fleishman and Silva [6,30].

In these previous studies, the conventional planar advancing-front methodology is directly extended to support surface operations. Meshing proceeds with the incremental introduction of a well distributed set of vertices X positioned on the surface $X \in \Sigma$. It should be noted that, contrary to the restricted Delaunay techniques introduced in previous sections, advancing-front methods typically maintain a partial manifold triangular complex $\mathcal{T}|_{\Sigma}$ only – they do not construct a full-dimensional tessellation $\mathcal{T}|_{\Omega}$. It should also be noted that, in addition to the usual limitations associated with advancing-front strategies, serious issues of robustness often afflict these techniques in practice due to the difficulties associated with the reliable evaluation of the requisite geometric intersection and overlap predicates for sets of non-planar elements. Additionally, for highly curved and/or poorly separated surface definitions it is known to be difficult to ensure the topological correctness of the output mesh $\mathcal{T}|_{\Sigma}$. Schreiner et al. [6,30] introduce a number of heuristic techniques in an effort to overcome these difficulties.

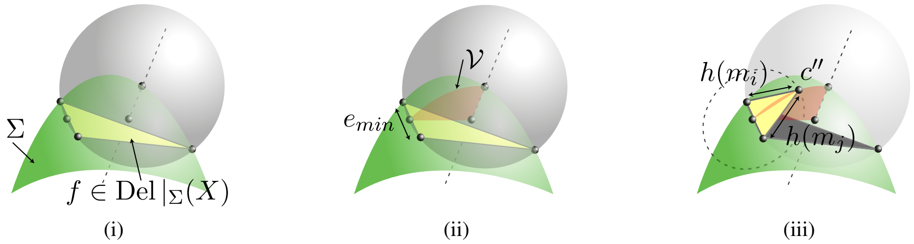
3.1. Off-centres

The new Frontal-Delaunay algorithm presented in this study is primarily based on ideas introduced by Rebay, who, in [25], developed a *planar* Frontal-Delaunay algorithm in which new vertices are positioned along edges of the associated Voronoi diagram. Rebay showed that new vertices can be positioned on $\text{Vor}(X)$ according to an *a priori* mesh size function $h(\mathbf{x})$, a strategy broadly consistent with conventional advancing-front techniques. While his algorithm maintains a Delaunay triangulation $\mathcal{T} = \text{Del}(X)$ of the current vertex set X, it is still fundamentally an advancing-front scheme – bereft of guarantees on the element shape quality $\rho(\tau)$. Rebay reported that his scheme produced high-quality output, typically outperforming conventional Delaunay-refinement in practice.

Üngör and Erten have approached the problem from the flip-side, introducing the notion of *generalised* Steiner vertices for planar Delaunay-refinement methods. Their 'off-centre' vertices are points lying along edges in the associated Voronoi diagram, as per Rebay. In [31], Üngor showed that when refining a triangle τ, its off-centre should be positioned, if possible, such that the new triangle σ adjacent to the shortest edge in τ satisfies the shape-quality constraint $\rho(\sigma) \leq \bar{\rho}$. Importantly, Üngör demonstrated that such a strategy typically leads to an improvement in the performance of Delaunay-refinement in practice, reducing the size of the output $|\mathcal{T}|$. Üngör extended the guarantees derived for Ruppert's Delaunay-refinement algorithm to his off-centre technique and showed that such bounds are typically improved upon in practice. Off-centre refinement is based on Ruppert's Delaunay-refinement framework directly, involving modifications to the procedure used to refine low-quality triangles only.

In this study, a generalisation of the ideas introduced by Rebay and Üngör is formulated for the surface meshing case – using off-centre Steiner vertices to simulate the vertex placement strategy of a conventional advancing-front approach, while also preserving the framework of a Delaunay-refinement technique. The aim of such a strategy is to recover the high element qualities and smooth, semi-structured point-placement generated by frontal methods, while inheriting the theoretical guarantees of Delaunay-refinement methods. Advancing-front algorithms typically incorporate a mesh size function $h(\mathbf{x})$, a function $f : \mathbb{R}^3 \to \mathbb{R}^+$ defined over the domain to be meshed, where $h(\mathbf{x})$ represents the desired edge length $||\mathbf{e}||$ at any point $\mathbf{x} \in \Sigma$. This mesh size function typically incorporates size constraints dictated by the both the user and the geometry of the domain to be meshed. The construction of appropriate mesh size functions will be discussed in subsequent sections, but for now, it suffices to note that $h(\mathbf{x})$ is a g-Lipschitz function defined at all points on Σ.

Fig. 3. Placement of an off-centre vertex, showing (i) the surface ball associated with a facet $f \in \mathrm{Del}\,|_{\Sigma}(X)$, (ii) the plane $\mathcal{V}$ aligned with the local facet of $\mathrm{Vor}(X)$ associated with the short edge $\mathbf{e}_{\min} \in f$, (iii) placement of the size-optimal point $\mathbf{c}''$ to satisfy the size constraints $h(\mathbf{m}_i)$ and $h(\mathbf{m}_j)$.



The proposed Frontal-Delaunay algorithm developed in this study is an extension of the restricted Delaunay-refinement algorithm presented in Section 2, modified to use off-centre rather than circumcentre-based refinement strategies. The basic framework is consistent with the Delaunay-refinement algorithm described previously, in which an initially coarse restricted Delaunay triangulation of a surface Σ is refined through the introduction of additional Steiner vertices $X \in \Sigma$ until all constraints are satisfied. A restricted surface triangulation $\mathcal{T}|_{\Sigma} = \mathrm{Del}\,|_{\Sigma}(X)$ is constructed as a sub-complex of the full-dimensional tessellation $\mathrm{Del}(X)$ and a coarse restricted volumetric tessellation $\mathcal{T}|_{\Omega} = \mathrm{Del}\,|_{\Omega}(X)$ is also available as a by-product. The constraints satisfied by the Frontal-Delaunay algorithm are identical to those incorporated in the restricted Delaunay-refinement scheme, with upper bounds on the radius-edge ratio $\bar{\rho}$, surface discretisation error $\bar{\epsilon}$ and element size $h\,(\mathbf{x}_f)$ all required to be satisfied for convergence. See Algorithm 2.1 for a detailed outline of the method.

3.2. Point-placement Strategy

Given a surface facet $f \in \mathrm{Del}\,|_{\Sigma}(X)$ marked for refinement, the new Steiner vertex introduced to eliminate f is an off-centre, constructed based on local size and shape constraints. Adopting the *generalised* off-centre framework introduced by Üngör, the 'ideal' location of the off-centre $\mathbf{c}$ for the given element f is based on a consideration of the isosceles triangle σ formed about the short edge $\mathbf{e}_{\min} \in f$. The placement of a new point $\mathbf{c}''$ is considered, designed to ensure that σ satisfies local size constraints. The final off-centre $\mathbf{c}$ is a modification of $\mathbf{c}''$, such that there is a reduction to the standard circumcentre-based scheme in limiting cases. The off-centres introduced in this study involve the placement of two distinct kinds of Steiner vertices. Type I vertices are equivalent to conventional element circumcentres, and are used to satisfy local element shape constraints. Type II vertices are the *size-optimal* points $\mathbf{c}''$, designed to satisfy local sizing constraints.

Considering the placement of a Type II Steiner vertex for a 2-simplex $f \in \mathrm{Del}\,|_{\Sigma}(X)$, the *size-optimal* point $\mathbf{c}''$ is positioned at an intersection of the surface Σ and a plane $\mathcal{V}$, where $\mathcal{V}$ is aligned with the local facet of the Voronoi diagram $\mathrm{Vor}(X)$ associated with the short edge $\mathbf{e}_{\min} \in f$. The plane is positioned such that it passes through three local points on $\mathrm{Vor}(X)$: the midpoint of the short edge $\mathbf{e}_{\min} \in f$, the centre of the diametric ball of f and the centre of the surface Delaunay ball $\mathrm{B}(\mathbf{c}, r)_{\max} = \mathrm{SDB}(f)$. The vertex $\mathbf{c}''$ is positioned such that the size of the new triangle $h(\sigma)$ satisfies local constraints. Specifically, the altitude of the triangle is calculated to ensure that the two new edges of σ are not too long, such that $\|\mathbf{e}_1\| \simeq h(\mathbf{m}_1)$ and $\|\mathbf{e}_2\| \simeq h(\mathbf{m}_2)$, where the $\mathbf{m}_i$'s are the edge midpoints. Each constraint is solved for an associated altitude $a_i'' = (h(\mathbf{m}_i)^2 - \|\frac{1}{2}\mathbf{e}_{\min}\|^2)^{\frac{1}{2}}$. The diametric ball of the edge $\mathbf{e}_{\min}$ is required to be empty, imposing a limit on the corresponding altitudes, such that $\bar{a}_i'' = \max\,(a_i'', \|\frac{1}{2}\mathbf{e}_{\min}\|)$. This condition ensures that $\mathbf{c}''$ is always sufficiently far from $\mathbf{e}_{\min}$, even when the local size function $h(\mathbf{x})$ is much smaller than $\|\mathbf{e}_{\min}\|$. Given the altitudes, the position of the size-optimal point $\mathbf{c}''$ is calculated by computing the intersection of the surface Σ with a circle of radius $\min\,(\bar{a}_1'', \bar{a}_2'')$, centred at the midpoint of the short edge $\mathbf{e}_{\min} \in f$ and inscribed on the plane $\mathcal{V}$. In the case of multiple intersections, the candidate point $\mathbf{c}_i''$ of closest alignment to the *frontal* direction vector $\mathbf{d}_f$ is selected. Specifically, the point $\mathbf{c}_i''$ that maximises the scalar product $(\mathbf{c}_i'' - \mathbf{x}_{\mathrm{mid}}) \cdot \mathbf{d}_f$ is chosen, where $\mathbf{x}_{\mathrm{mid}}$ is the midpoint of the short edge $\mathbf{e}_{\min} \in f$ and the frontal direction vector $\mathbf{d}_f$ is taken from the midpoint $\mathbf{x}_{\mathrm{mid}}$ to the centre of the surface ball $\mathrm{B}(\mathbf{c}, r)_{\max} = \mathrm{SDB}(f)$.

Note that, for non-uniform $h(\mathbf{x})$, the expressions for the point $\mathbf{c}''$ are non-linear, with the altitudes a_i'' depending on the evaluation of the mesh size function at the edge midpoints $h(\mathbf{m}_i)$ and visa-versa. In practice, since $h(\mathbf{x})$ is guaranteed to be Lipschitz smooth, a simple iterative predictor-corrector procedure is sufficient to solve these expressions approximately. The positioning of size-optimal Type II Steiner vertices is illustrated in Figure 3.

Given the point $\mathbf{c}''$, the position of the final off-centre $\mathbf{c}$ for the 2-simplex f is determined. The point $\mathbf{c}$ is selected according to a *worst-case* argument, setting $\mathbf{c} = \mathbf{c}''$ if $\mathbf{c}''$ lies no further from $\mathbf{x}_{\text{mid}}$ than the centre of the surface Delaunay ball $\text{B}(\mathbf{c}, r)_{\max} = \text{SDB}(f)$. The off-centre is simply set to the centre $\mathbf{c}_{\max}$ otherwise. This procedure ensures that the new point placement strategy reduces to a 'conventional' circumcentre-based scheme in limiting cases, with $\mathbf{c}_{\max}$ equivalent to the standard Type I Steiner vertices used in the restricted Delaunay-refinement algorithm presented in Section 2.

3.3. Discussion

While the new point-placement strategy offers increased geometric flexibility when positioning Steiner vertices compared to conventional circumcentre-based refinement schemes, it is still similarly derived by considering fundamental properties associated with the underlying Voronoi diagram. Importantly, by constraining Steiner vertices to the facets of $\text{Vor}(X)$ it is ensured that the distribution of mesh vertices remains *well-separated* throughout the refinement process. For the sake of brevity, a full proof of termination or correctness for the new method is not included here, but it is important to note that constraints on element radius-edge ratios $\rho(f)$, element size $h(\mathbf{x}_f)$ and surface discretisation error $\epsilon(f)$ are satisfied *by definition*, provided that termination is achieved in practice. The development of a suitable theoretical model for the new Frontal-Delaunay algorithm is the subject of a forthcoming publication [32].

4. Mesh Size Functions

The construction of high-quality mesh size functions is an important aspect of both the restricted Delaunay-refinement and Frontal-Delaunay algorithms presented in Sections 2 and 3. A good mesh size function $h(\mathbf{x})$ incorporates sizing constraints imposed by both the user and the geometry of the domain to be meshed. These contributions can be considered via two separate size functions, where $h_u(\mathbf{x})$ represents user-defined sizing information and $h_g(\mathbf{x})$ encapsulates geometry driven constraints. In this study, it is required that both $h_u(\mathbf{x})$ and $h_g(\mathbf{x})$ be piecewise linear functions defined on a supporting tetrahedral complex $\mathcal{S}$. Construction of appropriate user-defined functions $h_u(\mathbf{x})$ is highly problem dependent and detailed formulations are not considered here.

Based on the work of Boissonnat and Oudot [14,15], it is known that the geometric mesh size function $h_g(\mathbf{x})$ must be sufficiently small relative to the *local-feature-size*, denoted $\text{lfs}(\mathbf{x})$, induced by the bounding surface Σ to ensure that the resulting surface triangulation $\text{Del}\,|_{\Sigma}(X)$ is both topologically and geometrically correct. Specifically, to guarantee that the resulting tessellation is a so-called *loose γ-sample*, it is required that $h_g(\mathbf{x}) \leq 0.08\,\text{lfs}(\mathbf{x})$, where $\text{lfs}(\mathbf{x})$ is the distance to the *medial-axis* of the bounded domain Ω. In practice, such constraints are typically found to be overly restrictive [18], allowing for a relaxation of the coefficient $\gamma = 0.08$. The local-feature-size can be expressed as the distance from any point $\mathbf{x}_i \in \Sigma$ to the closest point on the *medial-axis* of the domain. Given that Σ is represented as an underlying triangulation $\mathcal{P}$ in this work, a discrete approximation to $\text{lfs}(\mathbf{x})$ can be calculated directly via the method of Amenta and Bern [33,34,35]. For the sake of brevity, we do not describe these methods in full here, but simply note that they provide an efficient means to estimate $\text{lfs}(\mathbf{x})$ at the vertices of the supporting complex $\mathcal{S}$.

Given an estimate of $\text{lfs}(\mathbf{x})$ on the boundary of $\mathcal{S}$, it is possible to exact a degree of user-defined control on the resulting size function via a Lipschitz smoothing process. Following the work of Persson [36], a *gradient-limited* function $\tilde{h}(\mathbf{x})$ can be constructed by limiting the variation in $h(\mathbf{x})$ over the elements of $\mathcal{S}$. In this study, a scalar smoothing parameter $g \in \mathbb{R}^+$ is used to globally, and isotropically limit variation, such that $\tilde{h}(\mathbf{x}_i) \leq \tilde{h}(\mathbf{x}_j) + g\,\|\mathbf{x}_i - \mathbf{x}_j\|$ for all vertex pairs $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{S}$. The gradient-limited size function $\tilde{h}(\mathbf{x})$ becomes more uniform as $g \to 0$.

5. Results & Discussions

The performance of the Delaunay-refinement and Frontal-Delaunay surface meshing algorithms presented in Sections 2 and 3 was investigated experimentally, with both techniques used to mesh a series of benchmark problems.

Both the Frontal-Delaunay and Delaunay-refinement algorithms were implemented, allowing the performance and output of the two algorithms to be compared side-by-side. Due to similarities in the overall structure, a common code-base was used, with the algorithms differing only in the type of Steiner vertices inserted, as per the discussions outlined in Section 3. Both algorithms were implemented in C++ and compiled as 64-bit executables.

The Delaunay-refinement and Frontal-Delaunay algorithms were used to mesh a series of five benchmark problems, including three test-cases presented in Figure 4 in which uniform size constraints $h(\mathbf{x}) = \alpha$ were specified, and two detailed comparative studies presented in Figures 5 and 6 designed to assess the impact of graded size constraints. The maximum radius-edge ratio was held constant across all benchmark problems, such that $\theta_{\min} \simeq 29^{\circ}$. Additionally, a non-uniform surface discretisation threshold was enforced, setting $\bar{\epsilon} = \beta h(\mathbf{x})$, with $\beta = {}^{1}/_{10}$. For all test problems, detailed statistics on element quality are presented, including histograms of element *area-length* $a(f)$, *plane-angle*[3] $\theta(f)$ and *relative-length*[4] $\|\mathbf{e}\|\,(h(\mathbf{x}_e))^{-1}$ distributions. Histograms further highlight the minimum and mean area-length quality metrics, the worst-case plane angle bounds, $\theta_{\min}$ and $\theta_{\max}$ and the mean relative edge-length. Both algorithms successfully meshed the full set of benchmark problems, demonstrating that the new Frontal-Delaunay algorithm is capable of generating high quality surface triangulations, satisfying constraints on element shape-quality $\rho(f)$, element size $h\,(\mathbf{x}_f)$ and surface discretisation error $\epsilon(f)$, consistent with a conventional Delaunay-refinement technique.

Results for the uniform meshing test-cases are presented in Figure 4, where a small constant mesh size was imposed globally, setting $h(\mathbf{x}) = \alpha$, where α was chosen to be 3% of the maximum bounding-box dimension associated with each model. Analysis of these results show that both the Frontal-Delaunay and Delaunay-refinement algorithms generate high-quality surface meshes for all test cases – satisfying the required element shape-quality, size and surface error thresholds. Focusing on the distribution of element shape-quality explicitly, it is clear that the Frontal-Delaunay algorithm achieves significantly better results, producing meshes with higher mean area-length ratios in all cases. Additionally, the distribution of $\theta(f)$ shows much tighter clustering about 60°, again implying significant improvements in mean element shape. Analysis of the relative-length distribution shows that meshes generated using the Frontal-Delaunay algorithm tightly conform to the imposed mesh size constraints, with a tight clustering of $\|\mathbf{e}\|\,(h(\mathbf{x}_e))^{-1}$ about 1. In contrast, output generated using the Delaunay-refinement scheme is seen to incorporate significant sizing 'error', typified by broad distributions of relative-length, straddling $\|\mathbf{e}\|\,(h(\mathbf{x}_e))^{-1} \simeq 1$. A detailed visual inspection of the results presented in Figure 4 show a marked increase in both the uniformity and sub-structure in the output associated with the Frontal-Delaunay approach.

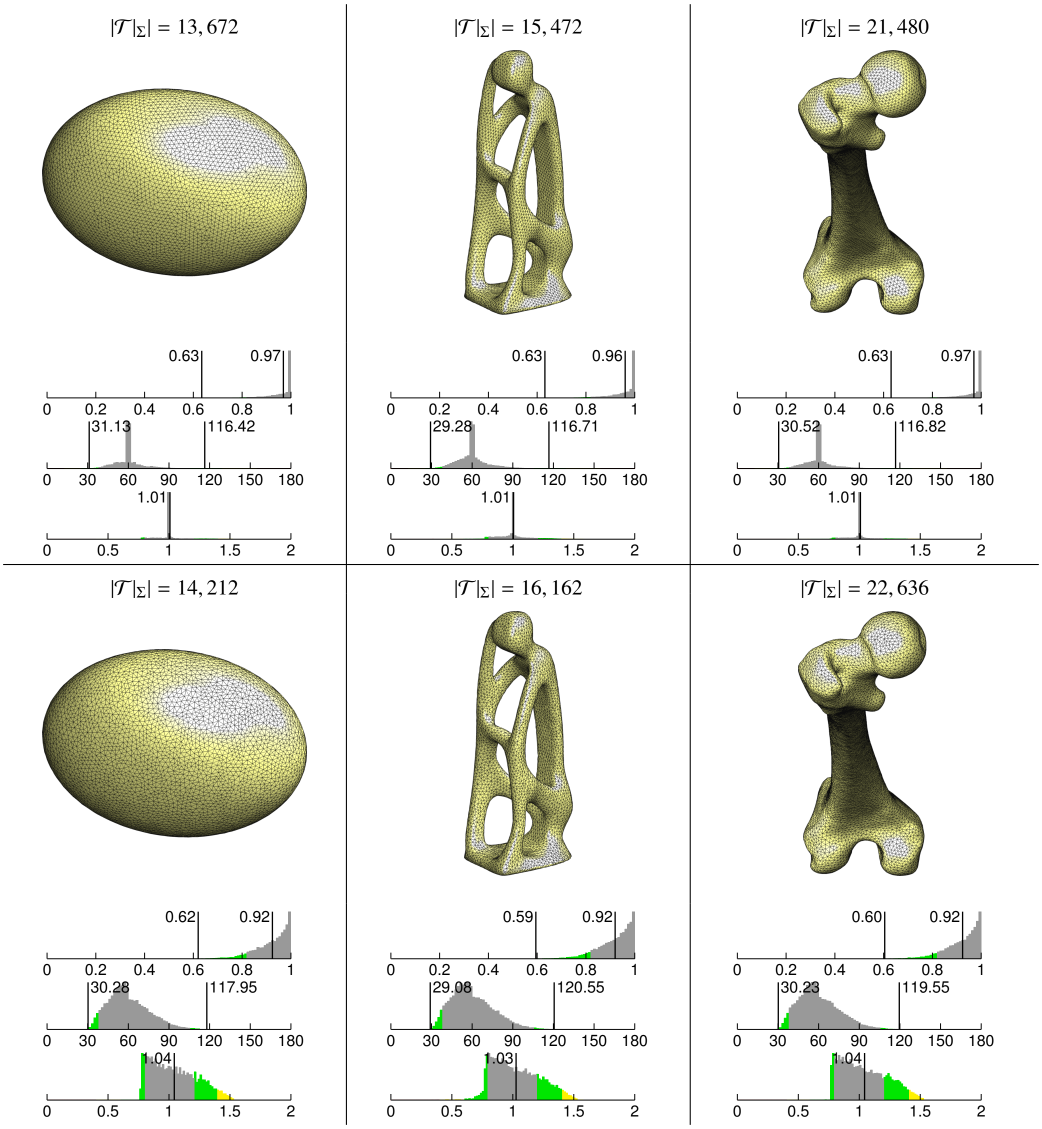
A sequence of meshes for the BUNNY and VENUS problems are presented in Figures 5 and 6, examining the impact of graded size constraints on algorithm performance. A set of meshes were generated for both test cases using low, medium and high resolution settings, where the associated mesh size functions $h(\mathbf{x})$ were built using increasingly stringent gradient limits, such that $g_i \in \{{}^{3}/_{10}, {}^{2}/_{10}, {}^{1}/_{10}\}$ respectively. Analysis of Figures 5 and 6 show that both the Frontal-Delaunay and Delaunay-refinement algorithms generate high-quality meshes for all test cases – satisfying the required element quality, size and surface error thresholds. Analysis of the distributions of $a(f)$ and $\theta(f)$ again show that the Frontal-Delaunay algorithm consistently outperforms the Delaunay-refinement scheme, generating meshes with higher mean area-length ratios in all cases. Furthermore, it is evident that the quality of meshes generated using the Frontal-Delaunay algorithm improves as $h(\mathbf{x}) \rightarrow 0$, as indicated by the increase in mean $a(f)$ and the narrowing of the $\theta(f)$ distribution about 60.0°. In contrast, similar analysis shows that output generated via the Delaunay-refinement technique is essentially independent of sizing constraints, with distributions of $a(f)$ and $\theta(f)$ showing little variation with $h(\mathbf{x})$. Visually, the enhanced quality of the meshes generated using the Frontal-Delaunay algorithm is again evident, with all test cases showing a marked increase in both smoothness and sub-structure. Given the tight bounds on radius-edge ratios in these tests, the Delaunay-refinement algorithm appears to achieve a maximum mean area-length ratio $\overline{a}(f) \simeq 0.92$. The Frontal-Delaunay algorithm is seen to generate consistently better output, with mean area-length metrics reaching $\overline{a}(f) \simeq 0.96$ at high resolution settings.

Consistent with previous results, analysis of the distribution of relative-length ratios show that the Frontal-Delaunay algorithm tightly conforms to the imposed non-uniform sizing constraints, with $\|\mathbf{e}\|\,(h(\mathbf{x}_e))^{-1}$ tightly clustered about 1. Furthermore, like the element shape-quality results discussed previously, it can be seen that the distribution of

[3] The *plane-angle* distribution $\theta(f)$ includes the three angles associated with each surface facet $f \in \mathrm{Del}\,|_{\Sigma}(X)$.

[4] The *relative-length* distribution $\|\mathbf{e}\|\,(h(\mathbf{x}_e))^{-1}$ is a measure of size-function conformance, expressing the ratio of actual-to-desired edge length for all edges $e \in \mathrm{Del}\,|_{\Sigma}(X)$, where the mesh size function $h(\mathbf{x}_e)$ is sampled at the edge midpoints $\mathbf{x}_e$.

Fig. 4. Triangulations for the ELLIPSOID, WOODTHINKER and FEMUR problems, showing output for the Frontal-Delaunay (upper) and Delaunay-refinement (lower) algorithms. A uniform mesh size function $h(\mathbf{x}) = \alpha$ is imposed, where $\alpha \in \mathbb{R}^+$. Element shape-quality is constrained such that $\theta_{\min} \geq 29°$. The surface discretisation threshold is scaled to the local sizing constraints, such that $\bar{\epsilon} = (1/10)\, h(\mathbf{x})$. Element counts $|\mathcal{T}|_\Sigma|$ are included for each case. Normalised histograms of element area-length ratio $a(f)$, plane-angle $\theta(f)$ and relative-length $||\mathbf{e}||\,(h(\mathbf{x}_e))^{-1}$ are illustrated.

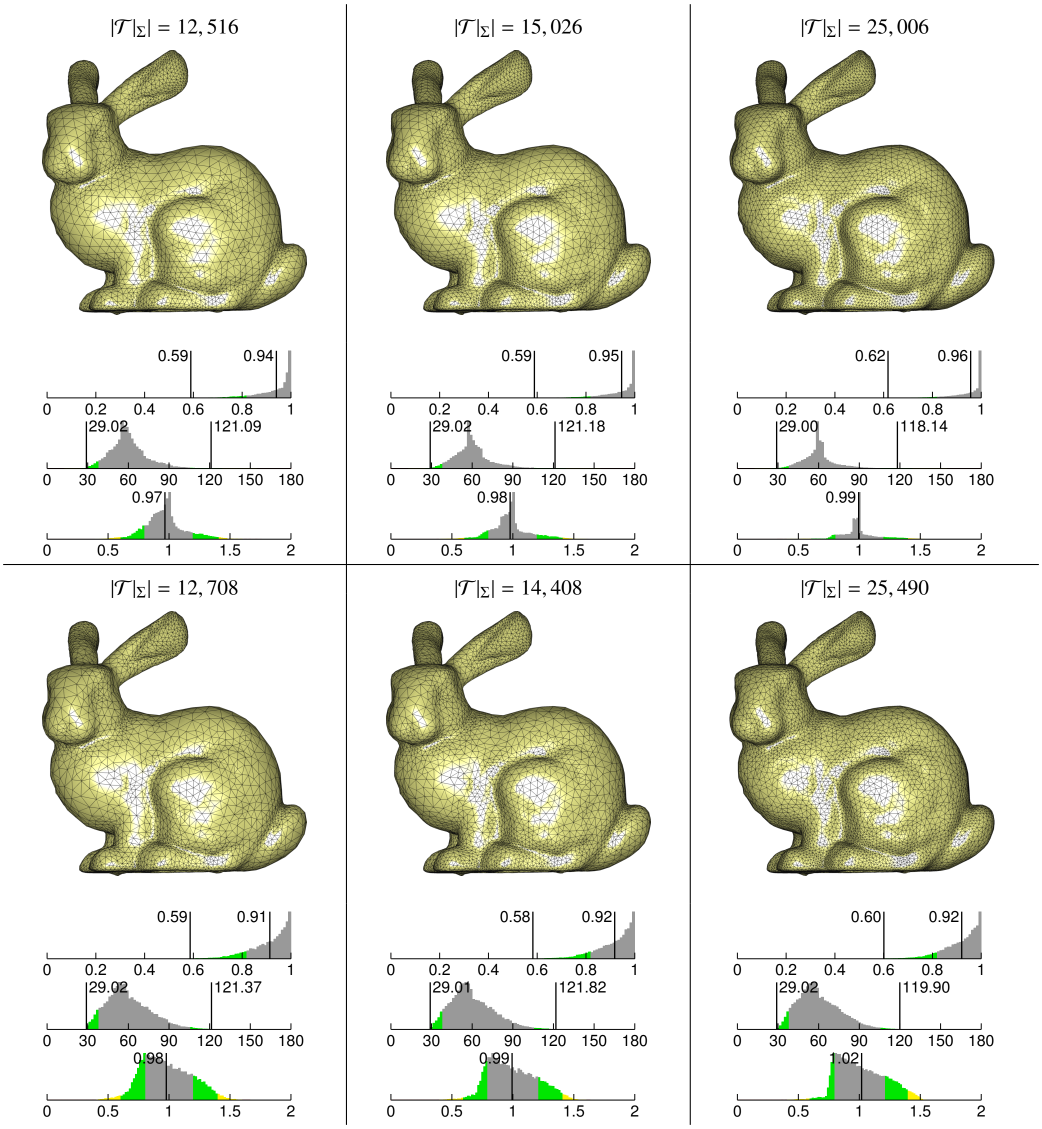


relative-length improves as $h(\mathbf{x}) \rightarrow 0$, as indicated by the narrowing of the distribution about $||\mathbf{e}||\,(h(\mathbf{x}_e))^{-1} = 1$. In contrast, the Delaunay-refinement scheme is again shown to be associated with significant sizing error, as illustrated by broad distributions of relative-length straddling $||\mathbf{e}||\,(h(\mathbf{x}_e))^{-1} \simeq 1$. Such behaviour appears to be largely independent of the characteristics of the imposed mesh size constraints.

6. Conclusions

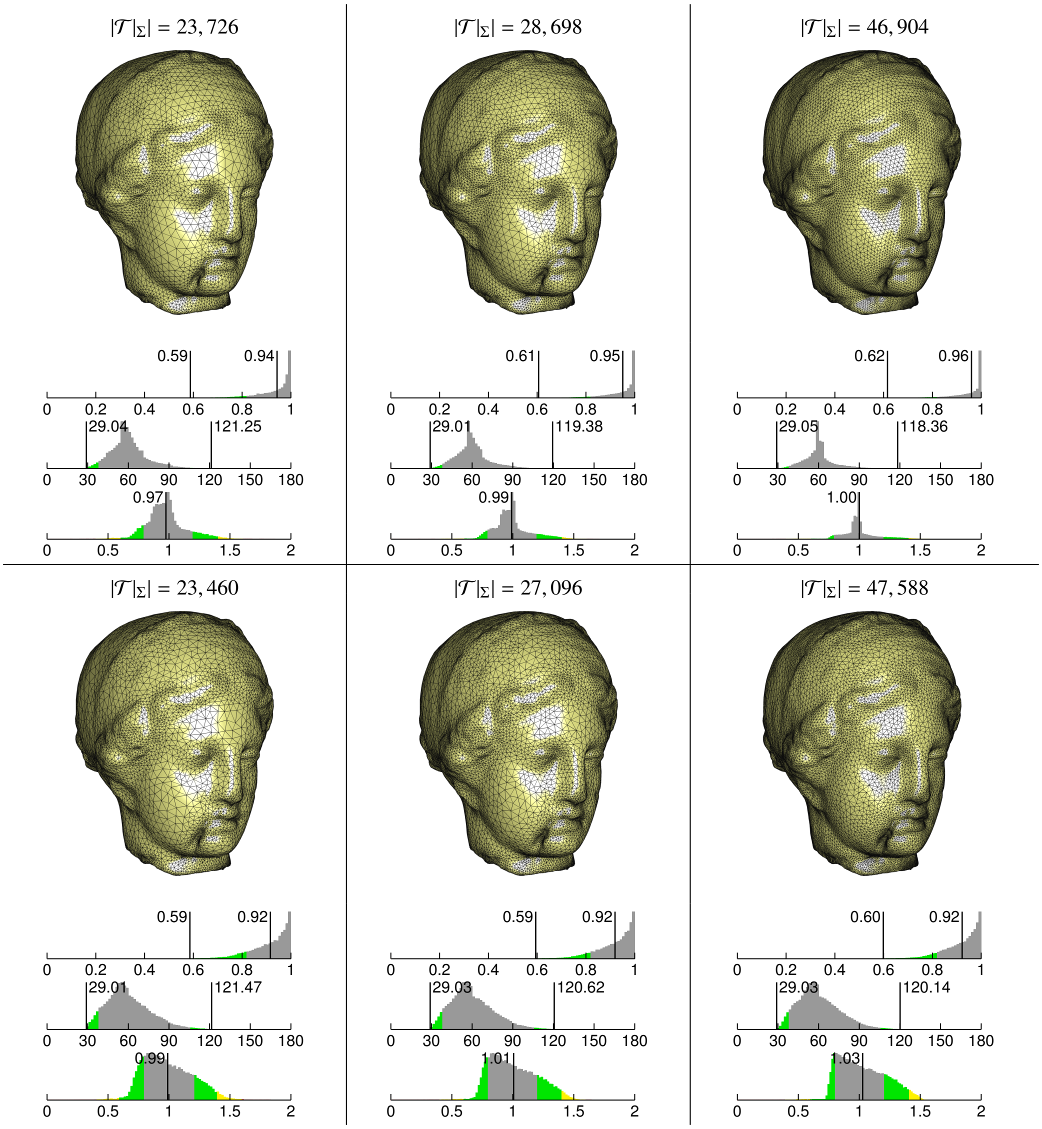
A new Frontal-Delaunay algorithm has been presented, designed to triangulate closed 2-manifold domains embedded in $\mathbb{R}^3$. The new algorithm is based on the so-called *restricted* Delaunay triangulation, in which a surface mesh $\mathrm{Del}\,|_\Sigma(X)$, conforming to an underlying surface Σ, is constructed as a subset of a full-dimensional Delaunay tessella-

Fig. 5. Size-driven meshing for the BUNNY problem, showing output for the Frontal-Delaunay (upper) and Delaunay-refinement (lower) algorithms. Mesh size functions are built with increasingly stringent gradient limits, $g_i \in \{3/10, 2/10, 1/10\}$ from left to right. Element shape-quality is constrained such that $\theta_{\min} \geq 29°$. The surface discretisation threshold is scaled to the local sizing constraints, such that $\bar{\epsilon} = (1/10)\, h(\mathbf{x})$. Element counts $|\mathcal{T}|_\Sigma|$ are included for each case. Normalised histograms of element area-length ratio $a(f)$, plane-angle $\theta(f)$ and relative-length $\|\mathbf{e}\|\,(h(\mathbf{x}_e))^{-1}$ are illustrated.



tion Del(X). The new restricted Frontal-Delaunay algorithm is based on a generalisation of the work of Rebay [25] and Üngör [31] for planar problems, in which generalised Steiner vertices are inserted along edges in the Voronoi diagram. This work has been extended to support surface meshing operations through the development of a new point-placement strategy that positions vertices on facets in the underlying Voronoi complex. This new scheme allows for the insertion of both size- and shape-optimal Steiner vertices, leading to a hybrid approach that combines many of the advantages of conventional advancing-front and Delaunay-refinement techniques. A series of comparative experimental studies confirm the effectiveness of this new approach in practice, demonstrating that an improvement in element quality and a reduction in output size is typically achieved, compared to an existing Delaunay-refinement scheme. Importantly, it has also been demonstrated that the new Frontal-Delaunay algorithm satisfies the same set of constraints as conventional Delaunay-refinement approaches, adhering to limits on element radius-edge ratios, edge

Fig. 6. Size-driven meshing for the VENUS problem, showing output for the Frontal-Delaunay (upper) and Delaunay-refinement (lower) algorithms. Mesh size functions are built with increasingly stringent gradient limits, $g_i \in \{3/10, 2/10, 1/10\}$ from left to right. Element shape-quality is constrained such that $\theta_{\min} \geq 29°$. The surface discretisation threshold is scaled to the local sizing constraints, such that $\bar{\epsilon} = (1/10)\, h(\mathbf{x})$. Element counts $|\mathcal{T}|_\Sigma|$ are included for each case. Normalised histograms of element area-length ratio $a(f)$, plane-angle $\theta(f)$ and relative-length $\|\mathbf{e}\|\,(h(\mathbf{x}_e))^{-1}$ are illustrated.



length and surface discretisation error. Results show that the new algorithm is an effective hybridisation of existing mesh generation techniques, combining the high element quality and mesh sub-structure of advancing-front techniques with the theoretical guarantees of Delaunay-refinement schemes. It is expected that applications requiring high mesh quality, including problems in computational fluid dynamics and/or structural analysis, may benefit from the new Frontal-Delaunay technique. Future work should incorporate support for an extended class of surface definitions, including domains containing sharp features.

Acknowledgements. This work was carried out at the University of Sydney with the support of an Australian Postgraduate Award. The authors also wish to thank the anonymous reviewers for their helpful comments and feedback.

References

[1] M. Bern, D. Eppstein, J. Gilbert, Provably Good Mesh Generation, J. Comput. Syst. Sci 48 (1990) 231–241.

[2] S. A. Mitchell, S. A. Vavasis, Quality Mesh Generation in Three-dimensions., in: Symposium on Computational Geometry, 1992, pp. 212–221. URL: `http://dblp.uni-trier.de/db/conf/compgeom/compgeom92.html#MitchellV92`.

[3] J. Peraire, J. Peiró, K. Morgan, Advancing-Front Grid Generation, in: J. F. Thompson, B. K. Soni, N. P. Weatherill (Eds.), Handbook of Grid Generation, Taylor & Francis, 1998. URL: `http://books.google.com.au/books?id=ImaDT6ijKq4C`.

[4] J. Schöberl, NETGEN: An Advancing Front 2D/3D Mesh Generator based on Abstract Rules, Computing and Visualization in Science 1 (1997) 41–52.

[5] D. Rypl, Approaches to Discretization of 3D Surfaces, in: CTU Reports, volume 7, CTU Publishing House, Prague, Czech Republic, 2003.

[6] J. Schreiner, C. E. Scheidegger, S. Fleishman, C. T. Silva, Direct (Re)Meshing for Efficient Surface Processing, Computer Graphics Forum 25 (2006) 527–536.

[7] L. P. Chew, Guaranteed-quality Triangular Meshes, Technical Report, Cornell University, Department of Computer Science, Ithaca, New York, 1989.

[8] J. Ruppert, A New and Simple Algorithm for Quality 2-dimensional Mesh Generation, in: Proceedings of the fourth annual ACM-SIAM Symposium on Discrete algorithms, SODA '93, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1993, pp. 83–92. URL: `http://dl.acm.org/citation.cfm?id=313559.313615`.

[9] J. Ruppert, A Delaunay Refinement Algorithm for Quality 2-Dimensional Mesh Generation, Journal of Algorithms 18 (1995) 548 – 585.

[10] J. R. Shewchuk, Delaunay Refinement Mesh Generation, Ph.D. thesis, Pittsburg, Pennsylvania, 1997.

[11] J. R. Shewchuk, Tetrahedral Mesh Generation by Delaunay Refinement, in: Proceedings of the fourteenth annual symposium on Computational geometry, SCG '98, ACM, New York, NY, USA, 1998, pp. 86–95. URL: `http://doi.acm.org/10.1145/276884.276894`. doi:`10.1145/276884.276894`.

[12] S. Cheng, T. Dey, Quality Meshing with Weighted Delaunay Refinement, SIAM Journal on Computing 33 (2003) 69–93.

[13] S. W. Cheng, T. K. Dey, E. A. Ramos, Delaunay Refinement for Piecewise Smooth Complexes, Discrete & Computational Geometry 43 (2010) 121–166.

[14] J. D. Boissonnat, S. Oudot, Provably Good Surface Sampling and Approximation, in: ACM International Conference Proceeding Series, volume 43, 2003, pp. 9–18.

[15] J. D. Boissonnat, S. Oudot, Provably Good Sampling and Meshing of Surfaces, Graphical Models 67 (2005) 405–451.

[16] C. Jamin, P. Alliez, M. Yvinec, J. D. Boissonnat, CGALmesh: A Generic Framework for Delaunay Mesh Generation, Technical Report, INRIA, 2013.

[17] B. Delaunay, Sur la sphere vide, Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk 7 (1934) 1–2.

[18] S. W. Cheng, T. K. Dey, J. R. Shewchuk, Delauay Mesh Generation, Taylor & Francis, New York, 2013.

[19] H. Edelsbrunner, N. R. Shah, Triangulating Topological Spaces, International Journal of Computational Geometry & Applications 7 (1997) 365–378.

[20] S. W. Cheng, T. K. Dey, J. A. Levine, A Practical Delaunay Meshing Algorithm for a Large Class of Domains*, in: M. Brewer, D. Marcum (Eds.), Proceedings of the 16th International Meshing Roundtable, Springer Berlin Heidelberg, 2008, pp. 477–494. URL: `http://dx.doi.org/10.1007/978-3-540-75103-8_27`. doi:`10.1007/978-3-540-75103-8_27`.

[21] S. W. Cheng, T. K. Dey, E. A. Ramos, T. Ray, Sampling and Meshing a Surface with Guaranteed Topology and Geometry, SIAM journal on computing 37 (2007) 1199–1227.

[22] A. Bowyer, Computing Dirichlet Tessellations, The Computer Journal 24 (1981) 162–166.

[23] P. Alliez, S. Tayeb, C. Wormser, AABB Tree, Technical Report, 2009.

[24] H. Erten, A. Üngör, Quality Triangulations with Locally Optimal Steiner Points, SIAM J. Sci. Comp. 31 (2009) 2103–2130.

[25] S. Rebay, Efficient Unstructured Mesh Generation by Means of Delaunay Triangulation and Bowyer-Watson Algorithm, Journal of Computational Physics 106 (1993) 125 – 138.

[26] D. J. Mavriplis, An Advancing Front Delaunay Triangulation Algorithm Designed for Robustness, Journal of Computational Physics 117 (1995) 90 – 101.

[27] P. J. Frey, H. Borouchaki, P. L. George, 3D Delaunay Mesh Generation Coupled with an Advancing-Front Approach, Computer Methods in Applied Mechanics and Engineering 157 (1998) 115 – 131.

[28] J. F. Remacle, F. Henrotte, T. Carrier-Baudouin, E. Béchet, E. Marchandise, C. Geuzaine, T. Mouton, A Frontal-Delaunay Quad-Mesh Generator using the $\mathbb{L}^{\infty}$-norm, International Journal for Numerical Methods in Engineering 94 (2013) 494–512.

[29] D. Rypl, Sequential and Parallel Generation of Unstructured 3D Meshes, Ph.D. thesis, CTU in Prague, 1998.

[30] J. Schreiner, C. Scheiclegger, C. Silva, High-Quality Extraction of Isosurfaces from Regular and Irregular Grids, Visualization and Computer Graphics, IEEE Transactions on 12 (2006) 1205–1212.

[31] A. Üngör, Off-centers: A New Type of Steiner Points for Computing Size-optimal Guaranteed-quality Delaunay Triangulations, Computational Geometry: Theory and Applications 42 (2009) 109–118.

[32] D. Engwirda, D. Ivers, Face-centred Voronoi refinement for restricted Delaunay surface mesh generation (in preparation).

[33] N. Amenta, M. Bern, Surface Reconstruction by Voronoi Filtering, Discrete & Computational Geometry 22 (1999) 481–504.

[34] N. Amenta, S. Choi, R. K. Kolluri, The power crust, unions of balls, and the medial axis transform, Computational Geometry 19 (2001) 127–153.

[35] T. K. Dey, W. Zhao, Approximating the medial axis from the voronoi diagram with a convergence guarantee, Algorithmica 38 (2004) 179–200.

[36] P.-O. Persson, Mesh Size Functions for Implicit Geometries and PDE-based Gradient Limiting, Engineering with Computers 22 (2006) 95–109.