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)

New techniques for enhanced medial axis based decompositions in 2-D

Harold J. Fogg[a], Cecil G. Armstrong[a], Trevor T. Robinson[a,*]

[a]*School of Mechanical and Aerospace Engineering, Queen's University Belfast, BT9 5AH, N. Ireland,*

Abstract

New techniques are presented for using the medial axis to generate high quality decompositions for generating block-structured meshes with well-placed mesh singularities away from the surface boundaries. Established medial axis based meshing algorithms are highly effective for some geometries, but in general, they do not produce the most favourable decompositions, particularly when there are geometry concavities. This new approach uses both the topological and geometric information in the medial axis to establish a valid and effective arrangement of mesh singularities for any 2-D surface. It deals with concavities effectively and finds solutions that are most appropriate to the geometric shapes. Methods for directly constructing the corresponding decompositions are also put forward.
© 2014 The Authors. Published by Elsevier Ltd.
Peer-review under responsibility of organizing committee of the 23rd International Meshing Roundtable (IMR23).

Keywords: medial axis; multiblock decomposition; block-structured; quad mesh.

1. Introduction

An assessment of the relative performances of multiblock structured meshes of turbomachinery that were generated by medial axis based methods, Cartesian grid based methods and by manual interactive work was made in a recent paper [1]. Adjoint error analyses in CFD simulations showed that the medial axis based methods generated higher quality meshes, and this can be attributed to their boundary-sympathetic block configurations. But despite their relative effectivenesses, there are still some obvious improvements to be made, such as more natural treatment of concavities and responsiveness to geometric shapes. Addressing these limitations by employing the neglected geometric information in the medial axis is the objective of this work.

1.1. Medial axis definition

For a 2-D surface, the *medial axis* is defined as the set of points where each point, $\hat{\mathbf{p}}$, has a centred inscribed circle, U, that touches the boundary entities more than once. Distinct parts of the medial axis are characterised by the number of touching points of $U(\hat{\mathbf{p}})$.

Medial edges connect subsets of the points with two touching points and they usually make up the majority of the medial axis. The radius of $U(\hat{\mathbf{p}})$ is called the *medial radius*, r_m, and the subtended angle between the radii of $U(\hat{\mathbf{p}})$

*Trevor T. Robinson, School of Mechanical and Aerospace Engineering, Queen's University Belfast, BT9 5AH, N. Ireland.
E-mail address: t.robinson@qub.ac.uk

1877-7058 © 2014 The Authors. Published by Elsevier Ltd.
Peer-review under responsibility of organizing committee of the 23rd International Meshing Roundtable (IMR23).

to the touching points is called the *medial angle*, θ_m. A special type of medial edge feature that frequently occurs is termination at a convex corner where r_m shrinks to zero. The corresponding medial edge is called a *flare*. These are all illustrated in Fig. 1.

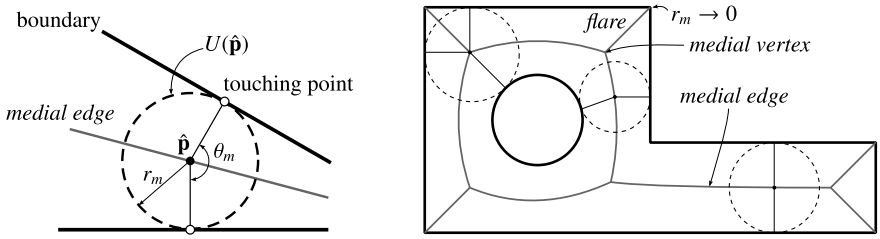


Fig. 1: Medial edge features and terminology (left) and medial axis for simple surface (right).

Medial vertices represent the meeting points of medial edges as $U(\hat{\mathbf{p}})$ transitions from one boundary pair to another. In standard cases three touching points are involved and in special cases there are more than three. There are also degenerate cases such as when $U(\hat{\mathbf{p}})$ is in finite contact with the boundary, or in a curvature contact situation such as at the end of an ellipse when the curvature of $U(\hat{\mathbf{p}})$ equals that of the boundary, resulting in a single touching point belonging to a medial vertex with θ_m equal to zero (cf. Fig. 9).

1.2. Review of established medial axis based decomposition algorithms

Quad meshing by medial axis decomposition was first conceived by Nackman and Srinivasan [2]. They noticed that the medial axis presented a well-defined natural decomposition of the surface and used it without alterations to split the surface into blocks. Some additional rules for merging the produced blocks were devised to enhance the suitability of the decomposition.

An improved method was proposed by Tam and Armstrong [3] that produces better quality decompositions. They developed a strategy for using a Constrained Delaunay Triangulation (CDT) of a set of boundary points, whose circumcenters approximate the medial axis to an acceptable accuracy, to assemble 'shape molecules' characterising a mesh topology. By inserting cuts between the medial vertices the shape molecules are reduced to 'shape atoms' which are topological 4-, 5- and 6-sided polygons. Each of these are meshable by midpoint subdivision and an integer programming method can be used to find allowable division numbers on the subblock edges [4,5]. An augmented algorithm with this approach has apparently been implemented in the commercial software of Abaqus [6], offering automatic high quality meshing capability.

More recently, Rigby [7] proposed an algorithm called 'TopMaker'. It consists of a few simple rules for creating suitable cuts at medial vertices and along the medial axis and it is surprisingly effective in generating pleasing decompositions for a certain class of geometries.

Other related methods include [8] which is less general and has rules specifically tailored for managing circular holes. The 'd-MAT' method used by Xia and Tucker [1,9] has similar rules to the TopMaker method and produces comparable decompositions.

1.2.1. Weaknesses of TopMaker and Tam and Armstrong (T&A) methods

In comparison to other widely used general quad meshing algorithms, such as paving [10] or cartesian grid methods [11], the TopMaker and T&A algorithms can usually generate superior block-structured meshes; they have fewer mesh singularities (defined in Sec. 2.2) and simpler topologies than paved meshes and have high quality elements on the boundaries unlike Cartesian meshes. However, they are in fact only effective for certain types of surfaces.

Neither handles concavities particularly well. The TopMaker algorithm makes the assumption that all corners are flat except those connected to flares which have one element. Using the definitions given in Sec. 3 (Fig. 6 (right)), these translate as $n_c = 2$ and $n_c = 1$ type corners respectively. Whilst this makes the decision making in the algorithm very straightforward, it doesn't always lead to serviceable decompositions. Consider the decomposition shown in Fig. 2 (top right). The mesh is forced to wrap around the trailing edges which leads to highly distorted blocks. Therefore, in general, acceptable quality decompositions are only created for geometries with corner angles $<\sim 5\pi/4$.

The T&A method performs slightly better with concavities, as supported by Fig. 2 (bottom right). In the first stage the surface is split by the best candidate CDT edges that most closely fit with the assigned mesh patterns at the corners, which are chosen based on corner angles. The splitting edges for concavity removal are shown in blue in Fig. 2 (bottom left). This strategy can at least produce $n_c = 3$ and 4 type corner patterns but it often does not lead to the best overall mesh topology. For example, the method always creates a split between concave vertices that share an inscribed circle and this results in the sub-optimal decomposition shown in Fig. 3 (centre) for the *indents* surface. But the most obvious and best quality decomposition is shown in Fig. 3 (right).

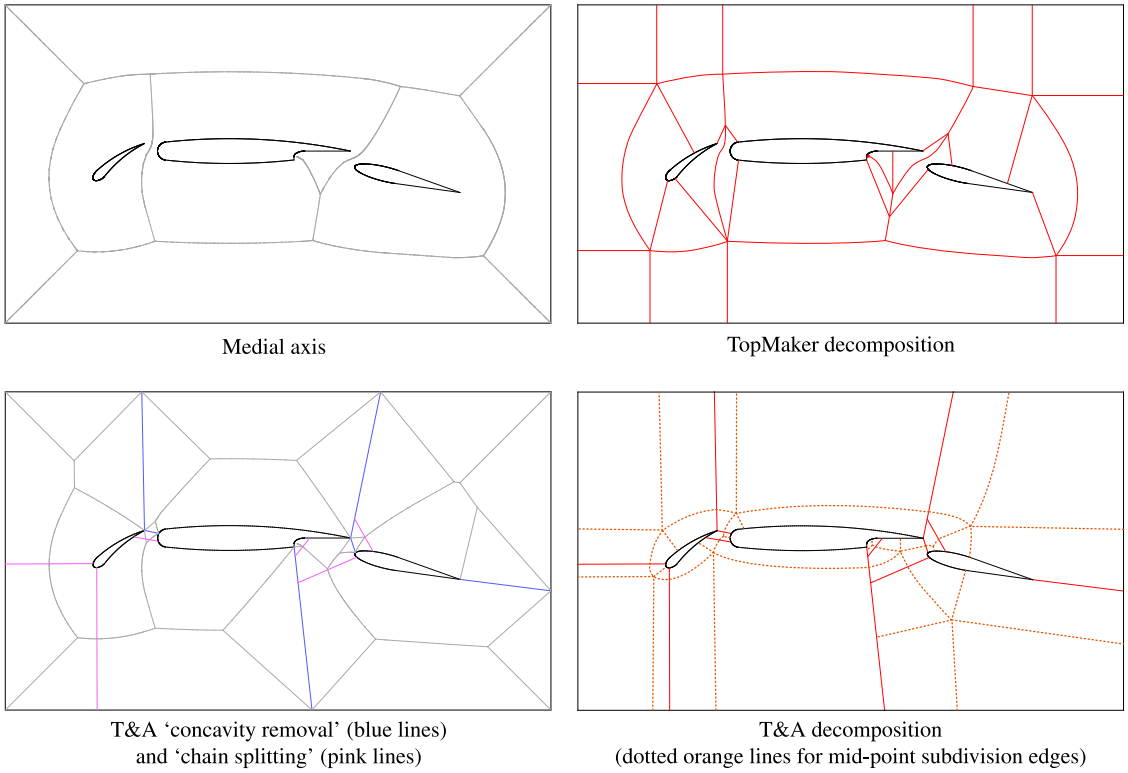


Fig. 2: *Multi-element aerofoil* surface decomposed by TopMaker and T&A algorithms.

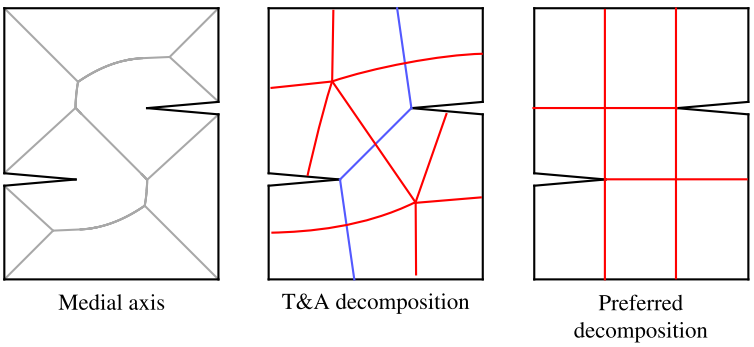


Fig. 3: *Indents* surface decomposed by T&A algorithm and its preferred decomposition.

Another limitation is the almost exclusive use of topological information in the medial axis and the disregard of important embedded geometrical shape information. For example, each of the three surfaces shown in Fig. 4 (top row) are the same as far as the TopMaker and T&A algorithms are concerned because the medial axis topologies are

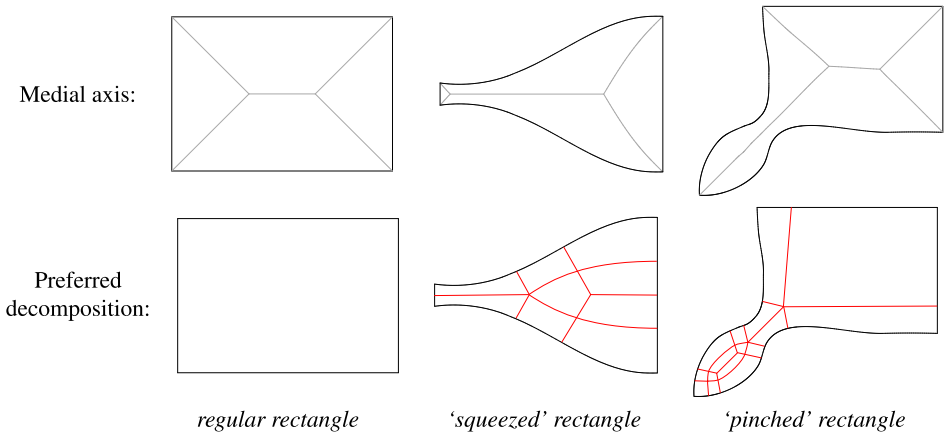


Fig. 4: Three surfaces with the identical medial axis topology but quite different geometric shapes (top row) and suggested good quality decompositions (bottom row).

identical. Thus, they are all treated as rectangular blocks for which a regular grid can be used in their interiors. This is appropriate for the first surface but a regular grid mapped to the second and third will be highly distorted. Preferably, they would be decomposed, or more specifically, the block topology solutions should contain some positive-negative singularity pairs giving the block structure sketched in Fig. 4 (bottom row – middle and right).

1.3. Contributions

Herin, a new approach is described which combines the medial axis based decomposition algorithms and the frame field decomposition approaches [12–14]. It makes use of the valuable and hitherto neglected geometric information in the medial axis, namely the medial angle, which leads to the development of a simple well-reasoned method for finding an effective mesh topology. The method can find a suitable configuration of mesh singularities on the medial axis which is essentially the critical step in a mesh generation problem (briefly substantiated in the next section). It contains a general way of handling concavities and it is precisely responsive to important geometric shapes so that the preferred solutions shown in Fig. 3 (right) and Fig. 4 (bottom row) are found automatically.

2. Theoretical properties of quad meshes

2.1. Introduction

The new techniques are reasoned in terms of a precise theoretical description of unstructured quad meshes with infinitesimal properties that was described by Bunin [15]. The theory gives insight into the fundamental properties of quad meshes and explains the crucial role played by mesh singularities, which are nodes where the regular grid connectivities are disrupted.

2.2. Overview of the theory

The key results of Bunin's theory are:

- The governing equation of an orthogonal mesh is

$$\Delta_s \phi = K + \sum_{i=1}^{N} k_i \frac{\pi}{2} \delta_{\mathbf{p}_i}, \quad k_i \in \mathbb{Z} \geq -4. \tag{1}$$

where Δ_s is the Laplace-Beltrami operator, ϕ is a scalar field controlling the mesh, K is the Gaussian curvature, k_i is the type of the singularity and $\delta_{\mathbf{p}_i}$ is a Dirac delta function centred at $\mathbf{p}_i$. This is a Poisson equation and it describes stationary heat conduction amongst other physical phenomena.

- The type of a singularity is given by the integer k which essentially denotes the number of additional elements above four at a node.
- The variable ϕ corresponds to $-\ln h$, where h is the edge length of an infinitesimal square element.
- The curvatures of the mesh edges are related to the variation in the ϕ-field by

$$\kappa_g = \frac{\partial \phi}{\partial e} \equiv \langle \nabla_s \phi, \mathbf{e} \rangle, \tag{2}$$

where κ_g denotes the geodesic curvature of the edge and its intrinsic normal, $\mathbf{e}$, is defined by $\mathbf{e} = \mathbf{n} \times \mathbf{t}$ where $\mathbf{n}$ and $\mathbf{t}$ are the surface normal and the tangent vector respectively.

- A cross-field describing the mesh directionality (see [12,13,16–18]) is closely related to the ϕ-field. The total change in angle of a cross along a curve, α, (in a parallel-transport sense) is given by

$$\Delta\theta = \int_\alpha \frac{\partial \phi}{\partial e} ds. \tag{3}$$

With regard to the problem of mesh generation, a continuum description of a mesh can be obtained by solving a heat conduction type problem (by FEM for example) once a valid arrangement of mesh singularities have been specified. However, finding the positions of mesh singularities is non-trivial, indeed in the context of this theory it is in the class of Inverse Poisson problems which are known to be ill-posed and difficult to solve effectively [19].

3. Resolving mesh singularities using the medial axis

3.1. Optimum mesh-flow indicated by the medial angle

The medial axis supplies a link between nearby boundaries and provides a means to assess the geometry locally. An intuitive idea is that the relative orientations of the boundaries indicate preferred patterns of the mesh in the intermediate areas. It is convenient to use a cross-field representation of the mesh to demonstrate this.

Given two points with assigned crosses, the total change in angle of any cross-field between them is unique up to a multiple of $\pi/2$. The multiplying integer, call it n, essentially specifies the number of $\pi/2$ turns made by a cross moving between the points. With reference to Bunin's continuum theory, the total flux through a curve linking the points is given by

$$\Phi_\alpha \equiv \int_\alpha \frac{\partial \phi}{\partial e} ds = \int_\alpha \kappa_g \, ds + \theta_b - \theta_a - n\frac{\pi}{2}, \tag{4}$$

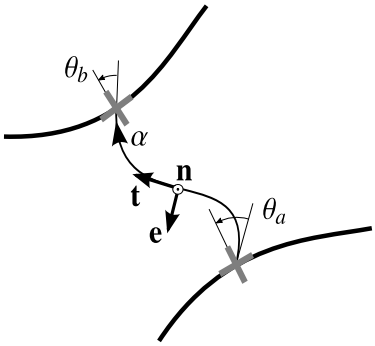
with the symbol definitions as indicated in Fig. 5 (left). If smooth regular cross-fields are preferred then the total flux should be minimised and the optimum integer for n can be computed by

$$n|_{\text{optimum}} = \text{round}\left(\frac{\int_\alpha \kappa_g \, ds + \theta_b - \theta_a}{\pi/2} \right). \tag{5}$$

For the particular case of a planar surface with the curve α composed of the medial radii (traversed so that κ_g is positive), then $\int_\alpha \kappa_g \, ds = \pi - \theta_m$. If the touching points are on boundary edges then θ_a and θ_b are zero, therefore

$$n|_{\text{optimum}} = \text{round}\left(\frac{\pi - \theta_m}{\pi/2} \right). \tag{6}$$

The three basic mesh-flow types between boundary edges linked by the medial axis and their preferred medial angle ranges are summarised in the table in Fig. 5 (right). If the touching points are on concave vertices θ_a and θ_b are not zero in general. To determine the optimum mesh-flow type in these cases the cross directions from which θ_a and θ_b



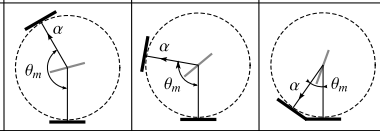
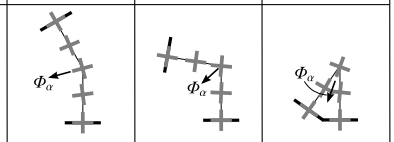
θ_m range	$[\pi, 3\pi/4)$	$[3\pi/4, \pi/4)$	$[\pi/4, 0]$
$n\|_{\text{optimum}}$	0	1	2
$\Phi_\alpha\|_{\text{optimum}}$	$\pi - \theta_m$	$\pi/2 - \theta_m$	$-\theta_m$
Medial axis appearance			
Cross-field behaviour			
Colour			

Fig. 5: Curve between two boundary-aligned crosses of nearby boundaries (left) and a table summarising the optimum mesh-flow types for medial angle ranges (right).

are measured are chosen so that their values are minimised and the non-zero entry $\theta_b - \theta_a$ is included in the calculation of Φ_α. Alternatively, the medial radii directions can be adjusted by θ_a and θ_b before θ_m is measured as shown in Fig. 6 (left).

In a similar way, the optimum mesh patterns at boundary corners can be decided by minimising the associated total flux radiating from the corner in the ϕ-field. Bunin gives an analytic expression for the strength of a point source necessary at a corner which involves the corner angle, θ_c, and an integer n_c that essential defines how many elements occur there. Choosing n_c to minimise the strength gives the optimum ranges shown in Fig. 6 (right). Thus, appropriate

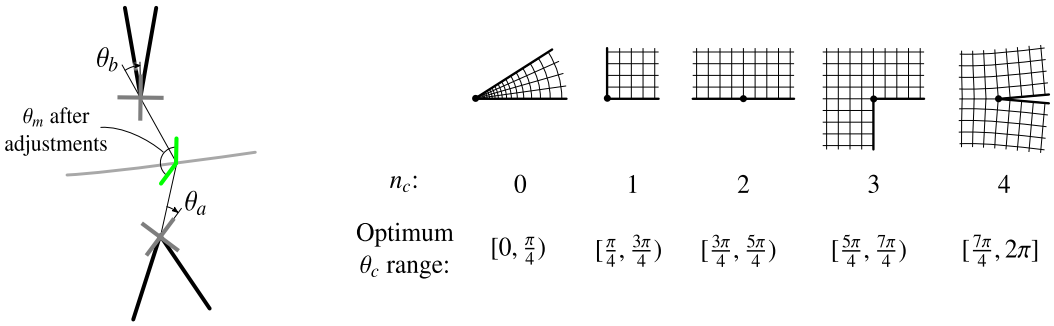


Fig. 6: Adjustments at concave vertices (left) and optimum mesh patterns at corners (right).

assigned crosses at corner vertices have either one direction along the corner bisector ($n_c = 0, 2, 4$) or one offset by $\pi/4$ from the bisector ($n_c = 1, 3$).

3.2. Identifying mesh singularities

If the optimum mesh-flow behaviour is selected for every position of the medial axis the positions of mesh singularities are essentially specified. A general method for identifying these positions involves a flux balance calculation. Supposing the surface is sliced into a series of thin regions by medial radii as shown in Fig. 7 and 8. Treating each sliver as a control area, a flux balance analysis can be carried out using the medial angle-dependent values for the total fluxes across pairs of medial radii as described above. The total fluxes across the small boundary segments are given by the differences in angle of the adjacent boundary-aligned crosses (which should be small). The flux residual must equal $k\pi/2$ where the integer k denotes the type of singularity present. Usually only simple singularities occur, *i.e.* $k = \pm 1$ or 'positive' and 'negative' singularities. These are marked by blue and red dots respectively in the figures.

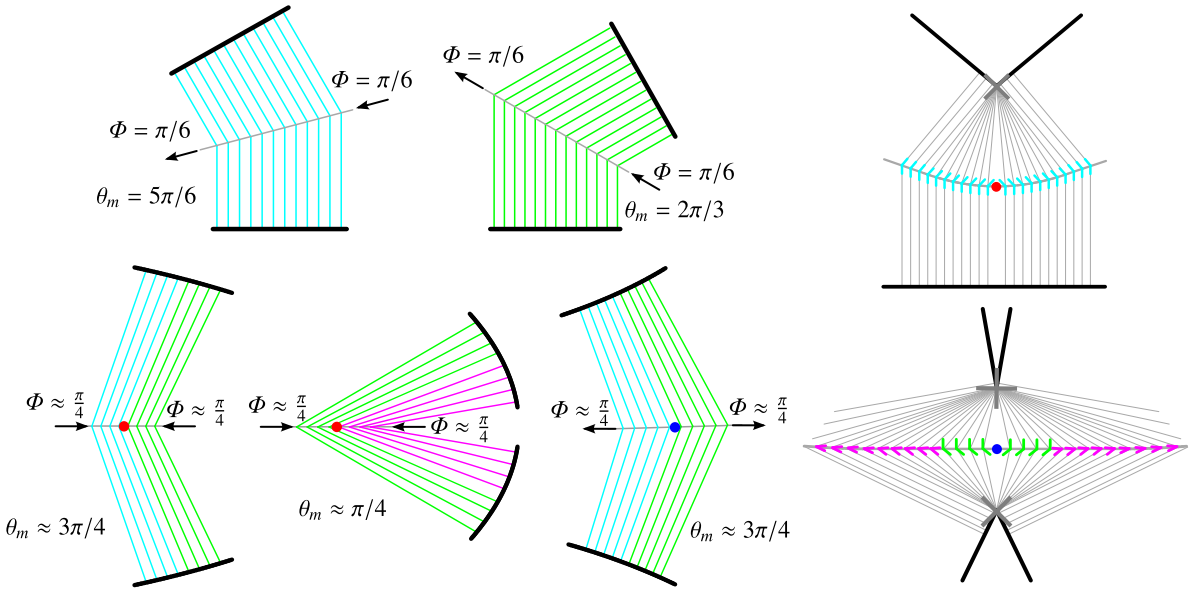


Fig. 7: Examples of flux balancing along medial edges.

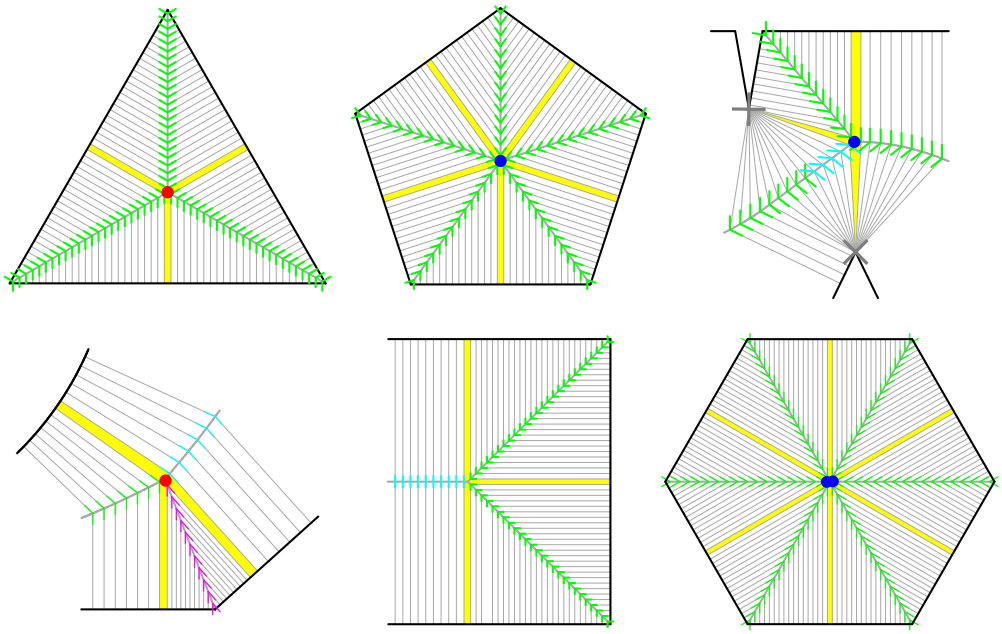


Fig. 8: Examples of flux balancing at medial vertices (highlighted in yellow).

In some special cases higher-order singularities can occur, an example of which is at the central medial vertex of a regular hexagon as shown in Fig. 8 (bottom right) where a $k = 2$ type singularity occurs, and it is identified in a general fashion.

At a medial vertex where the touching circle has a finite contact with a boundary there are a range of medial angle values associated with the medial vertex. These features can be equated to the limiting case where the boundary curve

tends to a circular arc from an elliptical arc. Thus, the singularity type at the medial vertex is given by

$$k = -\text{floor}\left(\frac{\max(\theta_m)}{\pi/2}\right). \tag{7}$$

For example, a semi-circular end of a rectangular block has a corresponding medial vertex with $\max(\theta_m) = \pi$, therefore a $k = -2$ singularity is placed on the medial vertex. This is illustrated in Fig. 9.

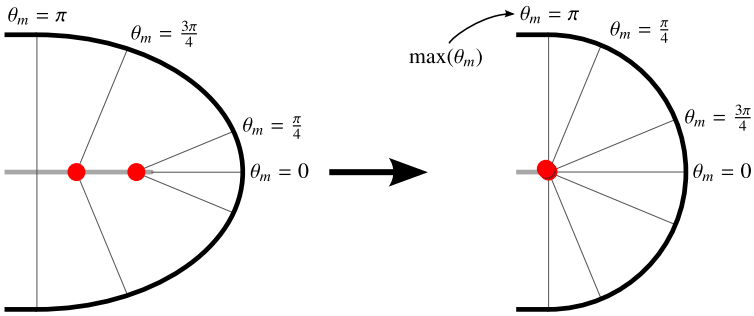


Fig. 9: Treatment of finite contact regions. A circular arc is equivalenced to an elliptical arc whose curvature tends towards uniformity.

3.3. Algorithm

A basic algorithm was developed to use the principles described to find an effective arrangement of mesh singularities. Its general steps are outlined below. It was implemented in a Python program using the CADfix API [20].

Algorithm 1. Find mesh singularities using medial axis

1. Generate the medial axis for the surface. CADfix's [20] CDT based medial axis generation tool is used for this.
2. For each medial edge assemble a sequence of analysis positions between the start and end points of the edge. A suitable interval size will capture the smooth change in θ_m and ensure that a single critical switch point of the optimum mesh-flow is contained in the interval.
3. Perform a total flux balance calculation for each sliver region bounded between the interval positions on medial edges and medial vertices, hence determine appropriate singularity positions.

The essential output of the algorithm is a list of singularity positions and types. Line geometry is also created purely to illustrate the method which is shown in the next section.

3.4. Results

Depictions of the algorithm results for geometries that were previously seen in Figs. 2–4 are shown in Fig. 10. Two images are included for each example, one showing the medial axis and artefacts representing the solution process, and the other showing a crudely defined cross-field. When creating the crosses their overall turning behaviour over the medial radii is taken from the assigned mesh-flow type and a smooth and even rate of change of angle is assumed.

The results match the 'preferred decomposition' topologies in each case. The additional positive and negative singularities appear in their natural positions in the *'squeezed'* and *'pinched' rectangle* surfaces. And, the concavities in the *multi-element aerofoil* and *indents* surfaces are satisfyingly managed.

One approach to generating a multiblock decomposition is to trace the separatrices of a cross-field. This has been explored in [13,21,22] and robust and effective cross-field initialisation and streamline tracing on a triangulation of the surface have been developed. In Fig. 11 a decomposition of the *multi-element aerofoil* surface is shown that was

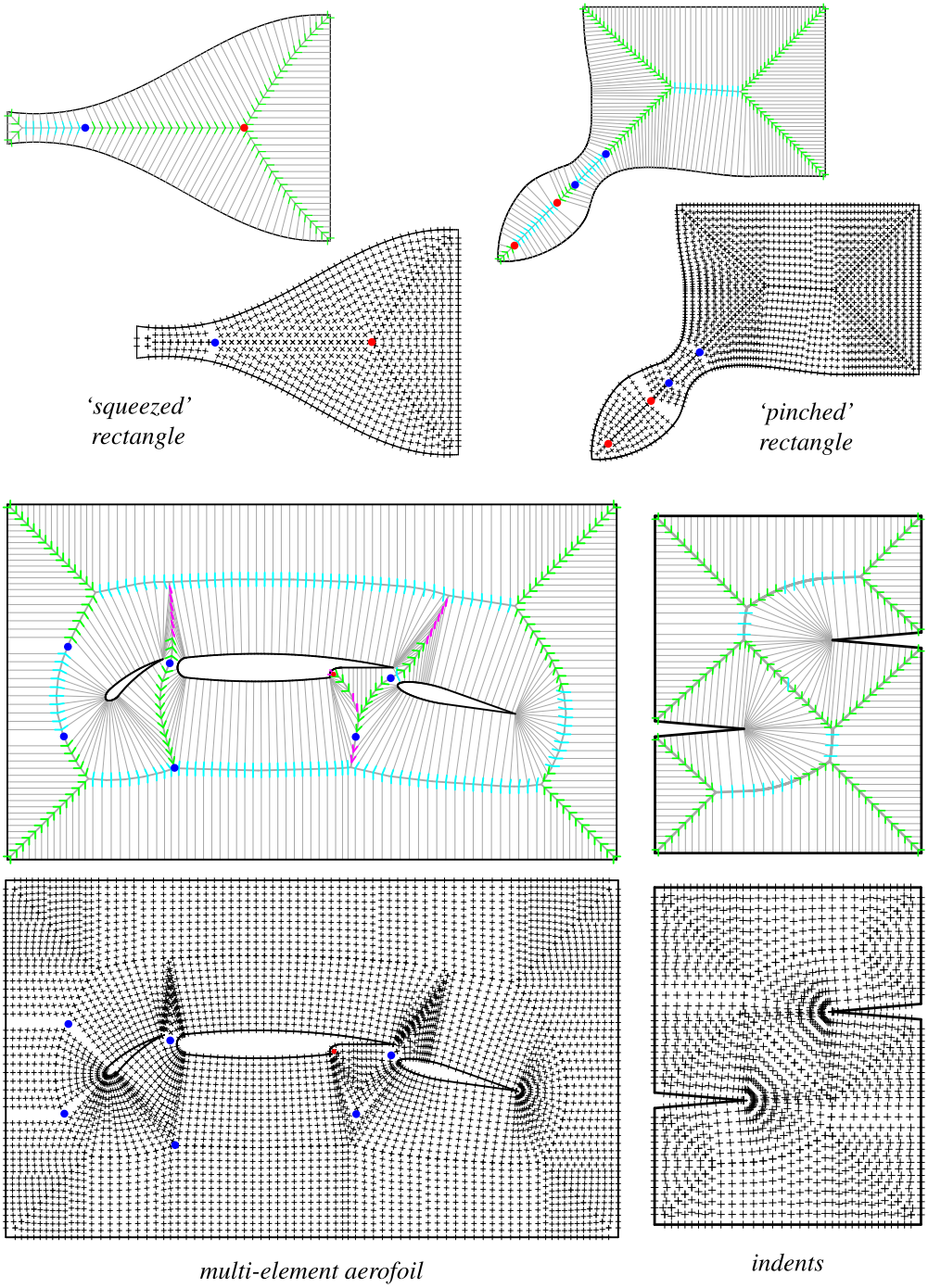


Fig. 10: Example results of singularity identification algorithm.

created by the method of [22] with corresponding mesh singularities and cross-field (but more exactly defined) to those in Fig. 10. In principle, these methods could be applied in the present method (with some modifications to clean up the cross-fields) but it would eliminate the niche that the medial axis decomposition methods fit into. Medial axis based methods should be 'lightweight', cheap, fast and robust, otherwise they offer no advantage over more sophisticated methods [21,23,24] that can handle more general problems with target size fields. Thus, undemanding

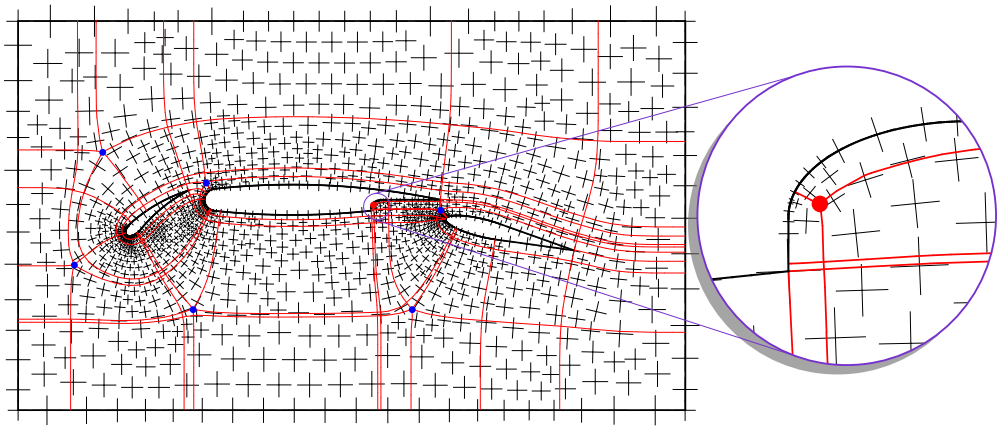


Fig. 11: Multiblock decomposition of *multi-element aerofoil* by [22].

techniques are called for to split the geometry using the medial axis in accordance with the mesh details obtained by the presented techniques, in an analogous manner to the T&A and TopMaker methods, but generating better quality decompositions.

4. Proposed decomposition procedure

Three agendas for choosing splits are:

1. to isolate singularities into separate blocks, except those closer than a minimum distance (a fraction of the local r_m, say),
2. to create splits from concave corner vertices in agreement with the assigned mesh patterns,
3. and to remove holes so that all blocks are topological discs.

4.1. Splitting edges

The most straightforward tactic for creating splits is to use the medial radii spanning between the touching points coupled by the medial axis. The two standard types are those made by pairs of medial radii between boundary edges where the adjoining convex corner is designated a $n_c = 2$ or $n_c = 1$ type as shown in Fig. 12 (left). The first choice is the $n_c = 2$ type split because it produces logically[1] convex sub-regions, whereas a $n_c = 1$ type split results in a concave $n_c = 3$ type corner.

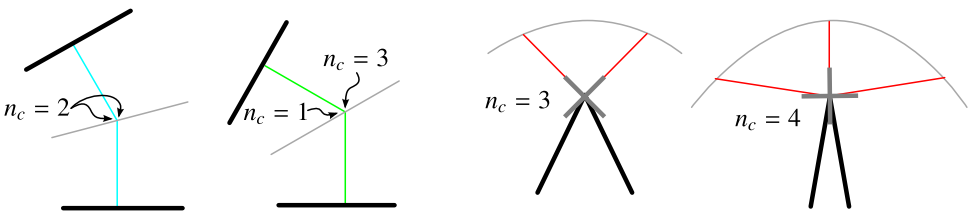


Fig. 12: Splitting edges between boundary edges (left) and from concave vertices (right).

[1] *i.e.* treating curvilinear grid edges as straight lines.

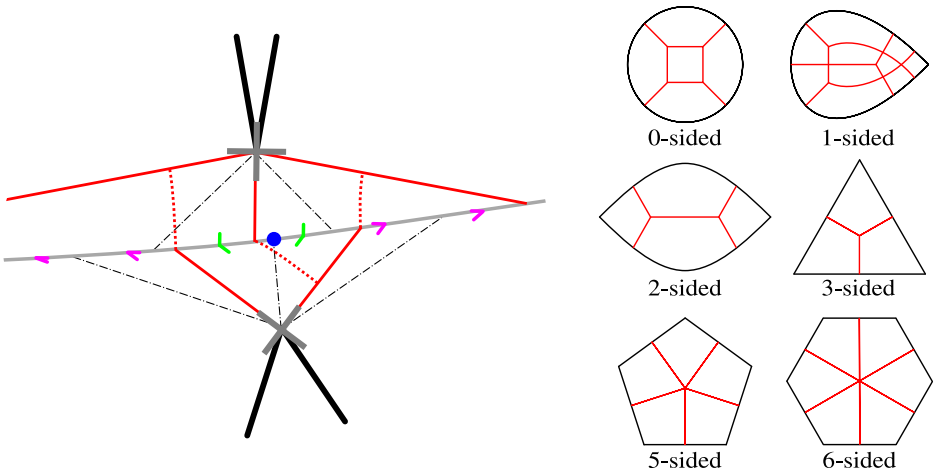


Fig. 13: Creating splitting edges between opposite concave vertices (left) and mesh templates for simple primitives (right).

For a logically concave vertex (*i.e.* $n_c = 3$ or 4) there are a discrete number of medial radii connected to the vertex that are allowed to be used as part of a split, as shown in Fig. 12 (right). The most complex scenario is when two concave vertices are on opposite sides of the medial axis. In these cases, there are no legitimate splits composed of pairs of medial radii generally. A simple work-around is to join a splitting edge of one orthogonally to the appropriate opposite splitting edge as shown in Fig. 13 (left).

4.2. Algorithm

The essential steps in the proposed decomposition procedure are itemised below.

Algorithm 2. Decompose surface

1. Create splits from concavities.
2. Create splits to isolate singularities.
3. Check if all decomposed subregions are topological discs. If not, add extra splits.
4. Apply additional splits to remove logically non-convex corners from subregions.

The final product is a decomposed surface where each block is identifiable as a logical m-sided polygon and standard mesh templates (seen in Fig. 13 (right)) can be applied to each. These are applicable to blocks that have m clear-cut $n_c = 1$ type corners and the rest of the type $n_c = 2$.

In the first step, splits are created from $n_c = 3$ and $n_c = 4$ type concave vertices. A single split is sufficient for both, taking the central one of the three candidates for $n_c = 4$ type vertices and either of the two for $n_c = 3$ type vertices. Whenever there are multiple possibilities for splits they can be ranked; $n_c = 2$ type splits are always preferred over $n_c = 1$ and subtended angles closer to π and $\pi/2$ are liked better for each. Next, suitable splits are searched for to separate singularities that are above the local threshold distance, again opting for the highest ranked candidate split. Usually the first two steps will decompose the surface into topological disc blocks. If that is not the case then suitable additional splits should be made. Finally, splits are needed to further decompose logically non-convex subregions so that the standard mesh templates can be applied. This can be achieved by projecting an edge from a concave vertex to a nearby logically perpendicular block boundary or to an added nearby logically perpendicular $n_c = 2$ type split.

Examples of the decompositions possible from this approach are shown in Fig. 14. The mid-point subdivision edges of the mesh templates are sketched in dotted lines.

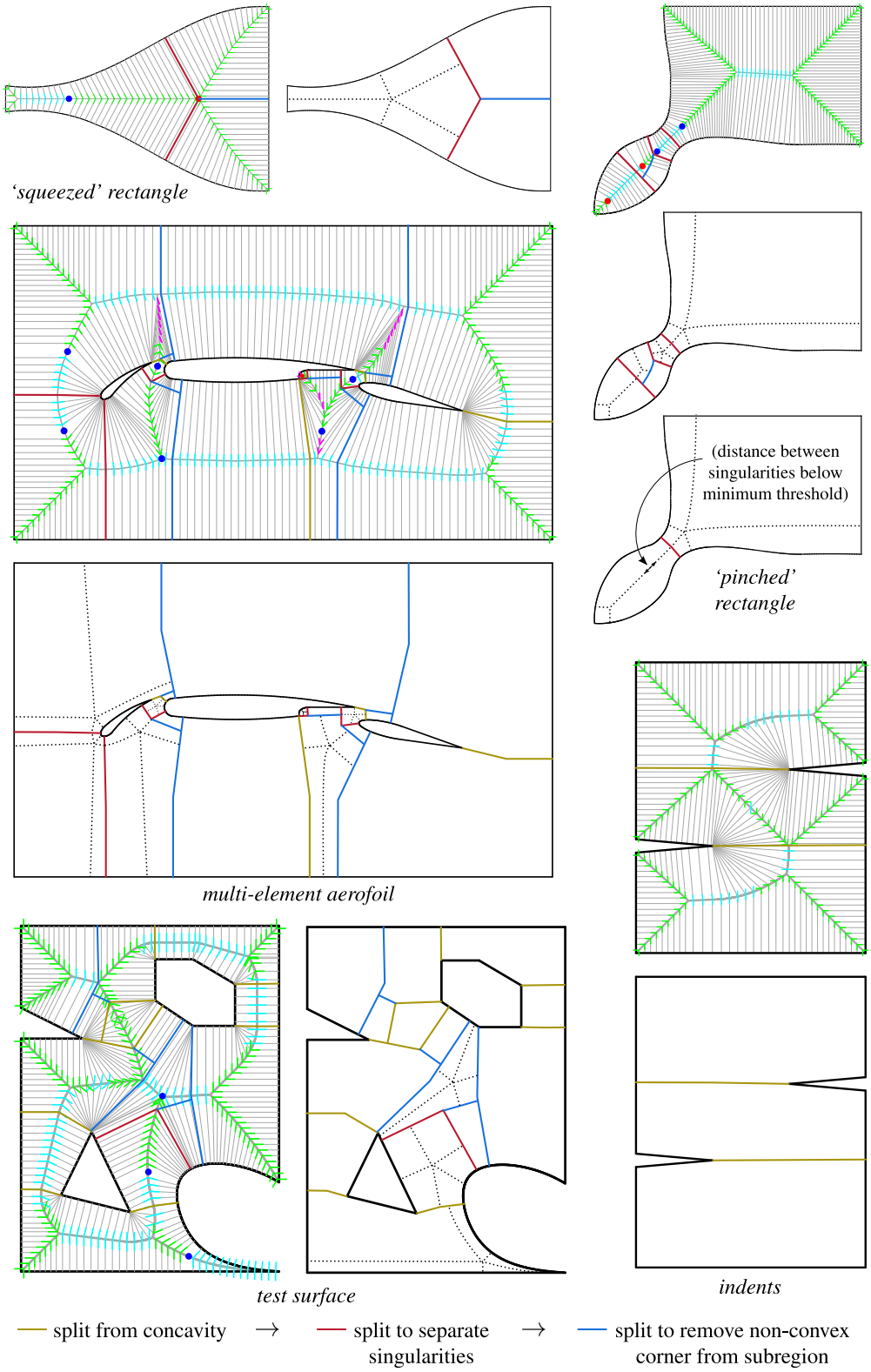


Fig. 14: Illustrative results of decomposition procedure.

5. Conclusion

A novel approach has been described for using the medial axis to first identify an effective configuration of mesh singularities, and secondly, to decompose the surface for block-structured meshing. The weaknesses of existing medial axis based methods due to assumed mesh behaviour in relation to the medial axis which is inappropriate around concavities are avoided. The proposed methods offer more capable handling of 2-D surfaces of all types for generating high-quality decompositions. And yet, the increased generality does not entail a large increase in complexity.

Future work includes refining the implementation details of the singularity identification program and putting into effect the proposed decomposition algorithm.

Acknowledgements

This work was sponsored by the Aircraft Research Association (ARA). The authors would like to thank Transcendata for their support.

References

[1] Z. Ali, P. G. Tucker, Multiblock structured mesh generation for turbomachinery flows, in: J. Sarrate, M. Staten (Eds.), Proceedings of the 22nd International Meshing Roundtable, Springer International Publishing, 2013, pp. 165–182.

[2] L. Nackman, V. Srinivasan, Method of generating finite elements using the symmetric axis transform, 1989. URL: `http://www.google.co.uk/patents/US4797842`, US Patent 4,797,842.

[3] T. K. H. Tam, C. G. Armstrong, 2-D finite element mesh generation by medial axis subdivision, Advances in Engineering Software 13 (1991) 313–324.

[4] T. K. H. Tam, C. G. Armstrong, Finite element mesh control by integer programming, International Journal for Numerical Methods in Engineering 36 (1993) 2581–2605.

[5] T. Li, R. McKeag, C. Armstrong, Hexahedral meshing using midpoint subdivision and integer programming, Computer Methods in Applied Mechanics and Engineering 124 (1995) 171 – 193.

[6] Abaqus FEA, SIMULIA website. Dassault Systemes., `http://www.3ds.com/products-services/simulia/portfolio/abaqus/`. (Accessed: 12/6/2013).

[7] D. Rigby, TopMaker: A Technique for Automatic Multi-Block Topology Generation Using the Medial Axis, Technical Report NASA/CR–213044, 2004.

[8] D. Guoy, J. Erickson, Automatic Blocking Scheme for Structured Meshing in 2D Multiphase Flow Simulation, in: Proceedings, 13th International Meshing Roundtable, 2004-3765C, Williamsburg, VA, 2004, pp. 121–132.

[9] H. Xia, P. G. Tucker, Fast equal and biased distance fields for medial axis transform with meshing in mind, Applied Mathematical Modelling 35 (2011) 5804 – 5819.

[10] T. D. Blacker, M. B. Stephenson, Paving: A new approach to automated quadrilateral mesh generation, International Journal for Numerical Methods in Engineering 32 (1991) 811–847.

[11] R. Schneiders, A grid-based algorithm for the generation of hexahedral element meshes, Engineering with Computers 12 (1996) 168–177.

[12] M. Nieser, U. Reitebuch, K. Polthier, Cubecover parameterization of 3D volumes, Computer Graphics Forum 30 (2011) 1397–1406.

[13] N. Kowalski, F. Ledoux, P. Frey, A pde based approach to multidomain partitioning and quadrilateral meshing, in: X. Jiao, J. Weill (Eds.), Proceedings of the 21st International Meshing Roundtable, Springer Berlin Heidelberg, 2013, pp. 137–154.

[14] Y. Li, Y. Liu, W. Xu, W. Wang, B. Guo, All-hex meshing using singularity-restricted field, ACM Trans. Graph. 31 (2012) 177:1–177:11.

[15] G. Bunin, A continuum theory for unstructured mesh generation in two dimensions, Comput. Aided Geom. Des. 25 (2008) 14–40.

[16] J. Palacios, E. Zhang, Rotational symmetry field design on surfaces, ACM Trans. Graph. 26 (2007).

[17] F. Kälberer, M. Nieser, K. Polthier, Quadcover - surface parameterization using branched coverings, Computer Graphics Forum 26 (2007) 375–384.

[18] N. Ray, B. Vallet, W. C. Li, B. Lévy, N-symmetry direction field design, ACM Trans. Graph. 27 (2008) 10:1–10:13.

[19] Y. Hon, M. Li, Y. Melnikov, Inverse source identification by green's function, Engineering Analysis with Boundary Elements 34 (2010) 352 – 358.

[20] ITI Transcendata, CADfix website. `http://www.transcendata.com/products/cadfix` (Accessed: 02/2013).

[21] H. J. Fogg, Automatic Generation of Multiblock Decompositions for Meshing, Ph.D. thesis, Queen's University Belfast, School of Mechanical and Aerospace Engineering, 2014.

[22] H. J. Fogg, C. G. Armstrong, T. T. Robinson, Multi-block decomposition using cross-fields, in: J. P. Moitinho de Almeida, D. P, C. Tiago, N. Pares (Eds.), Proceedings of Adaptive Modelling and Simulation, Lisbon, 2013, pp. 254 – 267.

[23] D. Bommes, M. Campen, H.-C. Ebke, P. Alliez, L. Kobbelt, Integer-grid maps for reliable quad meshing, ACM Trans. Graph. 32 (2013) 98:1–98:12.

[24] D. Bommes, B. Lévy, N. Pietroni, E. Puppo, C. Silva, M. Tarini, D. Zorin, Quad-mesh generation and processing: A survey, Computer Graphics Forum 32 (2013) 51–76.