Jean-Philippe PERNOT, George MORARU, Philippe VERON - Repairing triangle meshes built from scanned point cloud - Journal of Engineering Design - Vol. 18, n°5, p.459-473 - 2007

Repairing triangle meshes built from scanned point cloud

J-P. PERNOT*, G. MORARU and P. VÉRON

LSIS–UMR CNRS 6168, CER ENSAM, 2 cours des Arts et Métiers, 13617 Aix-en-Provence cedex 1, France

The Reverse Engineering process consists of a succession of operations that aim at creating a digital representation of a physical model. The reconstructed geometric model is often a triangle mesh built from a point cloud acquired with a scanner. Depending on both the object complexity and the scanning process, some areas of the object outer surface may never be accessible, thus inducing some deficiencies in the point cloud and, as a consequence, some holes in the resulting mesh. This is simply not acceptable in an integrated design process where the geometric models are often shared between the various applications (*e.g.* design, simulation, manufacturing). In this paper, we propose a complete toolbox to fill in these undesirable holes. The hole contour is first cleaned to remove badly-shaped triangles that are due to the scanner noise. A topological grid is then inserted and deformed to satisfy blending conditions with the surrounding mesh. In our approach, the shape of the inserted mesh results from the minimization of a quadratic function based on a linear mechanical model that is used to approximate the curvature variation between the inner and surrounding meshes. Additional geometric constraints can also be specified to further shape the inserted mesh. The proposed approach is illustrated with some examples coming from our prototype software.

Keywords: Reverse Engineering; Integrated design; Geometric modelling; Holes in meshes; Triangle mesh deformation; Approximated curvature variation minimization; Shape manipulations

1. Introduction

Reverse Engineering (RE) is a powerful technique used to create a digital representation of an existing physical object. In mechanical engineering, the final aim is classically to be able to analyse and/or to upgrade products whose geometric models are not available. This lack of information is due to the fact that the concerned parts were created before CAD had become so mainstream. In other cases, even if the original parts have been created using a CAD software, the physical object may get modified by additional manufacturing and finishing processes (*e.g.* sanding and polishing). The only way to get the changes into the CAD model is to reverse engineer them. Basically, the RE process begins with the acquisition of a point cloud from the physical object outer surface. These dense unorganized data points are then used to compute either a triangle mesh (Hoppe *et al.*, 1992; Edelsbrunner and Mucke, 1994; Amenta *et al.*, 1998; Barnardini *et al.*, 1999) or a set of B-Spline/NURBS surfaces (Ma and Kruth, 1995; Forsey and Bartels, 1995; Eck and Hoppe, 1996) or a subdivision surface (Ma and Zhao, 2000; Lee *et al.*, 2000; Takeuchi *et al.*, 2000). Nevertheless, the triangulation of the

*Corresponding author. Email: jean-philippe.pernot@aix.ensam.fr

DOI: 10.1080/09544820701403797

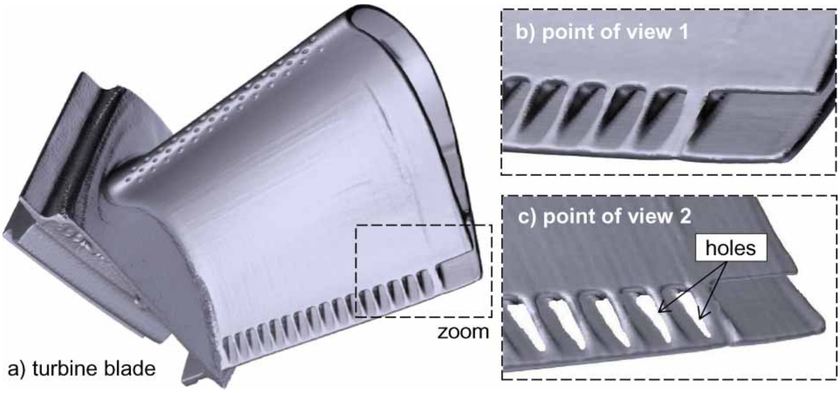

Figure 1. Example of holes (c) in the polyhedral model of a turbine blade (a, courtesy Georgia Tech) scanned from a point of view (b).

point cloud is often a preliminary step to the creation of more complex surfaces. Therefore, it is crucial to be able to produce a triangle mesh that best fits the object outer surface.

Even though the data acquisition is performed from various points of view, both the object complexity and the technology of the adopted acquisition system (*e.g.* Coordinate Measuring Machines or laser scanning) can be such that some areas of the object outer surface are inaccessible. This may induce some deficiencies in the point cloud and, as a result, a set of holes in the triangle mesh. Figure 1(a) shows the result of the acquisition of a turbine blade with a laser. Using the point of view (b), the scanner is unable to reach some portions of the object visible from another point of view (c). This is due to the optical occlusion phenomenon. The geometric model being often used as an intermediate object shared between the actors of the design process, the presence of undesired holes may induce unexpected results when doing rapid prototyping or Finite Element analysis for example. These drawbacks are simply unacceptable.

In this paper, we propose some new algorithms and models to fill in these undesired holes. The filling process works in four main steps, as illustrated in figure 2. First, the hole contour is identified while looking for boundary edges (figure 2(a)). The hole contour is then cleaned (figure 2(b)) to remove badly oriented and degenerated triangles (section 3). These irrelevant triangles are due to the scanner noise. The hole is then filled in figure 2(c) with one of our predefined topological grids (section 4) which is finally deformed (figure 2(d)) to satisfy a specific blending criterion with the surrounding mesh (section 5). In our approach, the resulting mesh is obtained while minimizing a quadratic function based on a linear mechanical model of bar networks. Among the various possibilities, we are notably able to minimize an approximation of the curvature variation across the hole contour. The user can also add some geometric constraints to shape further the interior of the inserted mesh and to recover some

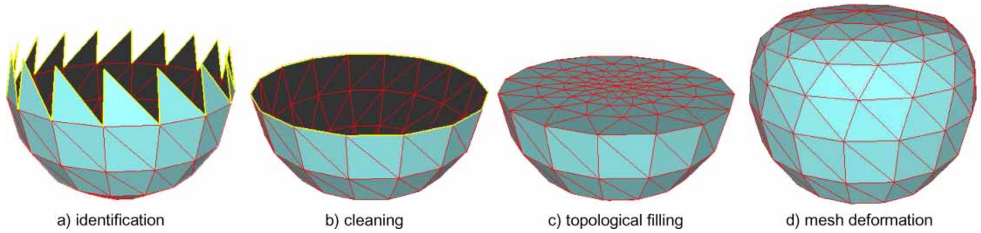

Figure 2. Overall filling process on a simple example: after the identification and cleaning of a hole contour, the missing area is filled with a topological grid whose final shape results from a deformation process.

more complex shapes. These are strong improvements with respect to the existing techniques found in the literature (section 2). The filling process is further exemplified with results coming from our prototype software and illustrating various application domains, such as the reverse engineering of a mechanical product or the reverse engineering of a statue (section 6). The limits and possible upgrades are finally discussed in section 7.

2. Related work

Various techniques have been proposed to fill in undesired holes in meshes. Two main categories can be distinguished: the geometric and non-geometric approaches.

Among the non-geometric approaches, Curless and Levoy (1996) use a volumetric representation to detect the areas that have to be filled in. Davies *et al.* (2002) apply a volumetric diffusion process to extend a signed distance function through this volumetric representation until its zero set bridges whatever holes may be present. This iterative approach is particularly well adapted for complex geometrical and topological holes. A similar approach has been developed by Nooruddin and Turk (2003) for the simplification and the repairing of polygonal meshes. Verdera *et al.* (2003) also represent the surface of interest in implicit form, and fill in the holes with a system of geometric partial differential equations derived from image inpainting algorithms. These equations are based on the geometric characteristics of the known mesh (*e.g.* the curvatures) and are applied only at the holes and a neighbourhood of them. Because these equations are anisotropic and based on geometry, they lead to a slightly slower algorithm than the one of Davies *et al.* (2002). Clarenz *et al.* (2004) use a finite element method to minimize the integral of the squared mean curvature (the so-called Willmore energy) of the filled hole. Their process is iterative and can only ensure a tangency continuity with the surrounding mesh.

Considering the geometric approaches, Liepa (2003) proposes a filling process relatively similar to the one of figure 2. The hole is first detected and filled with a minimum area triangulation of its 3D contour (Barequet and Sharir, 1995). The triangulation is refined, so that the triangle density agrees with the density of the triangles of the surrounding mesh (Pfeifle and Seidel, 1996). The filled hole is finally smoothed with a fairing technique based on an umbrella operator (Kobbelt *et al.*, 1998). The faired meshed is obtained while solving a system of linear equations. It satisfies tangency blending conditions with the initial mesh. Schneider and Kobbelt (2001) propose a fairing technique based on solving a non-linear fourth-order partial differential equation. G^1 boundary conditions are satisfied but the resolution is iterative. In addition, the implicit fairing of Desbrun *et al.* (1999) requires an iterative resolution process. Other approaches are based on the Moving Least Squares projection, which induces a non-linear minimization process (Wang and Oliveira, 2003; Tekumalla and Cohen, 2004).

The main drawbacks of most of these approaches concern the use of iterative resolution processes to solve non-linear equations, the non-satisfaction of curvature blending conditions between the inner and surrounding meshes, and the impossibility to prescribe additional geometric constraints to shape the inner mesh further. The second limit is illustrated in figure 3.

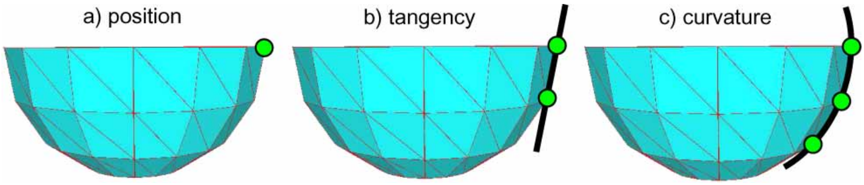

Figure 3. Depending on the number of vertex rows taken into account, either some position or tangency or curvature information can be extracted from the surrounding mesh.

It shows that to be able to extract curvature information from the surrounding mesh, at least three rows of vertices have to be taken into account. These are the three limits we try to overcome. We discuss how successful we are in section 7.

3. Identification and cleaning of the hole contour

In this work, all the meshes are supposed to be triangular, oriented, manifold and connected. In particular, two separate holes will have no vertices in common (otherwise those would be singular), and a given hole will not have islands (otherwise the mesh would not be connected). Given these hypotheses, the contour of the hole can be automatically identified while looking for a closed cycle of boundary edges, *i.e.* those edges that are not shared between two triangles.

Since the scanner noise may lead to the creation of degenerate triangles, the hole contour must be pre-processed to remove all the bad triangles before inserting the topological grid. The pre-treatments concern the thin triangles, *i.e.* those having one or two small angles (figures 4(a) and (b)), and the badly oriented ones, *i.e.* those having widely varying orientations with respect to the surrounding mesh (figure 4(c)).

3.1 *Deletion of badly oriented triangles*

Triangles defined by two boundary edges can rotate around their interior edge without affecting the other triangles (figure 4(c)). This degree of freedom, combined with the uncertainty on the position of the vertices connected to the boundary edges (scanner noise), may give to these triangles an orientation completely different from the one of their adjacent triangles. To avoid undesired undulations in the neighbourhood of the hole, all the triangles defined by two boundary edges–and, as a consequence, potentially badly oriented–are recursively deleted (black triangles on the example of figure 5(a).

Using this algorithm, some significant details of the contour may disappear. To overcome this limit, an additional condition is added: a triangle having two boundary edges can be deleted only if its adjacent triangle does not have any boundary edge (figure 5(b)).

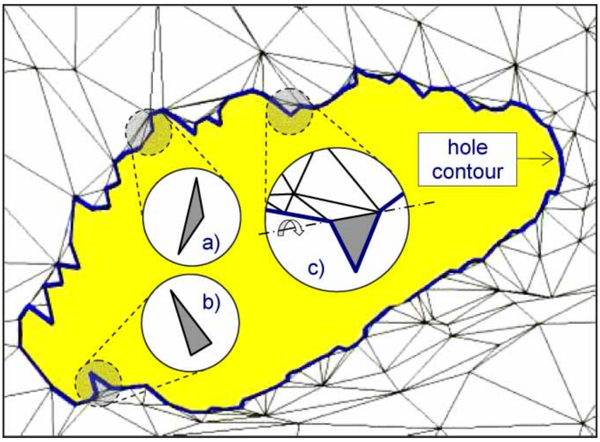

Figure 4. Hole contour identification and undesired triangles resulting from the scanner noise (a, b, c).

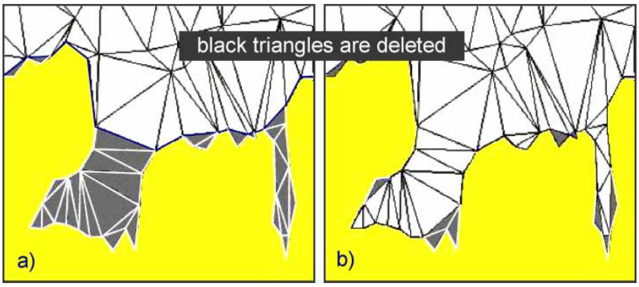

Figure 5. Deletion of potentially badly oriented triangles which preserves (b), or not (a), some significant details of the contour.

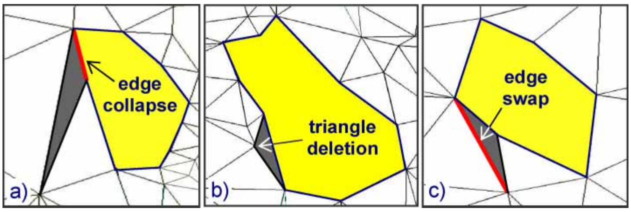

Figure 6. Three types of degenerated triangles.

3.2 *Treatment of degenerated triangles*

Degenerated triangles are defined by one or two small angles (figures 4(a) and (b)). To detect such triangles without computing their angles, we use an indicator introduced to check the quality of Finite Element (FE) meshes (Ciarlet, 1978) and defined as follows:

$$Q = \alpha \frac{\rho}{h} = \alpha \frac{S}{hp} \tag{1}$$

is the *aspect ratio* of a given triangle with $\alpha = 2\sqrt{3}$ a normalization coefficient, so that $Q = 1$ for an equilateral triangle, h is the longest edge length, ρ is the radius of the incircle, S is the area of the triangle and p its half-perimeter. This quality factor belongs to the interval $[0, 1]$ with the limit zero corresponding to flat triangles. In FE analysis, it is commonly accepted that a triangulation is a good one if the aspect ratio of the worst triangle is greater than 0.5. Here, we are less demanding than for FE analysis and more degenerated triangles are accepted. As a consequence, only the triangles whose ratio is smaller than 0.25 will be considered as degenerated. This threshold has been found empirically while analysing the variation of the ratio according to the triangle angles.

All the triangles considered as degenerated are then treated separately according to three possible configurations. First, if the triangle has one small edge with respect to the two others, the smallest edge is collapsed (figure 6(a)). Otherwise, if the longest edge is a boundary edge, the triangle is deleted (figure 6(b)), else the longest edge is swapped (figure 6(c)).

4. Topological filling

In the proposed approach, the hole is filled in with a topological grid obtained by meshing a two-dimensional ellipse having the same number of boundary vertices as the hole contour (figures 7(a) to 7(c)). The ellipse is defined by a unique user-specified parameter $h = b/a$

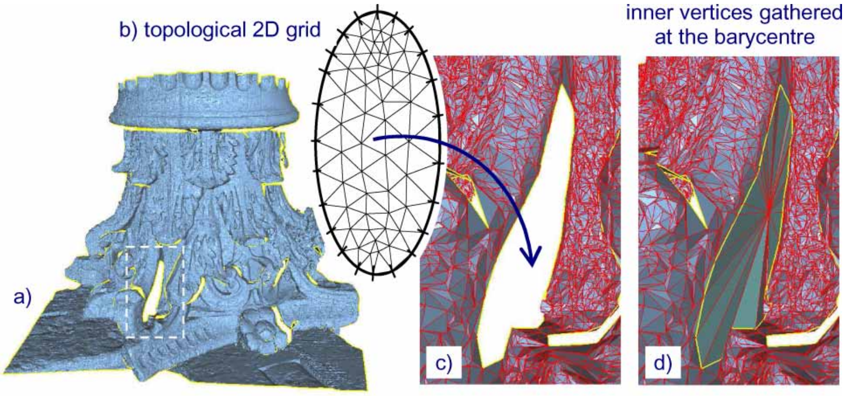

Figure 7. Topological filling with a meshed two-dimensional ellipse.

defining the ratio between the length a of its major axis and the length b of its minor axis. Since at this step the scale of the ellipse is not important, the length a of the major axis will always be equal to 1.0. The connections between the inner vertices, spread randomly on the ellipse, are defined using the Delaunay triangulation. The grid contour is then stretched and merged with the contour of the hole to be filled in. The new position of the inner vertices is unknown and will be computed during the inner mesh deformation step (section 5). They can be initially gathered together at the barycentre of the boundary vertices (figure 7(d)) since their initial position does not affect the result of the deformation.

The advantage of such a process lies in the decoupling between the topology and the geometry: the hole is first filled and the shape of the inner mesh is then determined. This algorithm gives good results when the hole to be filled looks like an ellipse. Unfortunately, it may lead to degenerated triangles when the hole has a much more complex shape, or when the boundary edges have heterogeneous sizes. In the future, it could be interesting to use the minimum area triangulation of Barequet and Sharir (1995) as a basis for determination of the topological grid.

5. Mesh deformation

The positions of the free vertices, *i.e.* the inner mesh vertices that are not on the hole contour, are obtained by deforming the inner mesh according to specific blending conditions specified between the inner and surrounding meshes. The deformation itself results from the minimization of a quadratic function based on a linear mechanical model of bar networks.

5.1 *Mechanical model of bar networks*

Basically, the positioning of the mesh vertices requires a manual displacement of each vertex (figure 8(a)). In order to save time and automate the deformation process, we use a mechanical model of bar network (Schek, 1974) coupled to the inner and surrounding meshes. The vertices and edges of the meshes match respectively the nodes and the bars of our bar network. Each bar can be seen as a spring with a null initial length and a stiffness q_i (more precisely a force density). To preserve the static equilibrium state of the structure, external forces f_i have to be applied to the nodes (figure 8(b)). If these external forces were not applied to the network, all the nodes would be gathered together at a single point. The linear relationships between

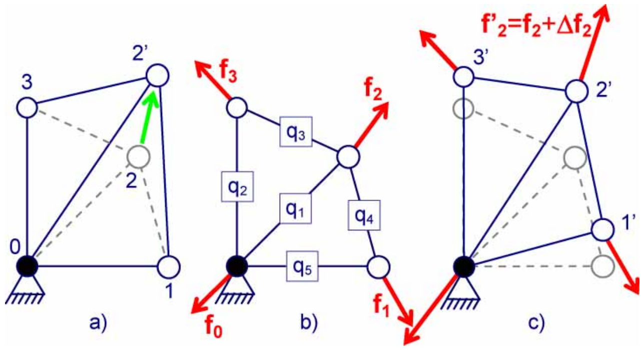

Figure 8. Deformation of a network with (c) and without (a) a coupling to our mechanical model (b).

the external forces and the position of the nodes enable intuitive shape modifications through the manipulation of a restricted set of external forces. In the example of figure 8(c), only the external force applied to the second node is modified to produce the displacement of all the free nodes (see Pernot, 2004) for a complete survey of free form deformation techniques).

Formalization Given $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$, the three vectors containing the components of the 3D coordinates of the N_n nodes of the bar network coupled to the control vertices of the geometric model, the $\boldsymbol{f}_x$, $\boldsymbol{f}_y$ and $\boldsymbol{f}_z$ components of the external forces applied to these nodes can be obtained by using the following $(3 \times N_n)$ equations expressing the bar network static equilibrium:

$$\begin{aligned} \boldsymbol{f}_x &= ({}^t\mathbf{C}.\mathbf{Q}.\mathbf{C}).\boldsymbol{x}, \\ \boldsymbol{f}_y &= ({}^t\mathbf{C}.\mathbf{Q}.\mathbf{C}).\boldsymbol{y}, \\ \boldsymbol{f}_z &= ({}^t\mathbf{C}.\mathbf{Q}.\mathbf{C}).\boldsymbol{z}, \end{aligned} \tag{2}$$

where $\mathbf{Q}$ is the force density matrix of size $(N_b \times N_b)$ being N_b the number of bars. $Q_{ij} = q_j.\delta_{ij}$ with δ_{ij} the Kroenecker's symbol and $q_j = f_j/\ell_j$ the force density into the jth bar of length ℓ_j. $\mathbf{C}$ is the branch-node matrix of size $(N_b \times N_n)$ expressing the connectivity of the network (Pernot, 2004).

Following Pernot *et al.* (2005a), a distinction between free and blocked nodes can be performed in the equations (2). It gives rise to the two following sets of equations:

$$\begin{aligned} \boldsymbol{f}_{fn_x} &= \mathbf{D}_f.\boldsymbol{x}_{fn} + \mathbf{D}_{bf}.\boldsymbol{x}_{bn}, \\ \boldsymbol{f}_{fn_y} &= \mathbf{D}_f.\boldsymbol{y}_{fn} + \mathbf{D}_{bf}.\boldsymbol{y}_{bn}, \\ \boldsymbol{f}_{fn_z} &= \mathbf{D}_f.\boldsymbol{z}_{fn} + \mathbf{D}_{bf}.\boldsymbol{z}_{bn}, \end{aligned} \tag{3}$$

for the external forces applied to free nodes and:

$$\begin{aligned} \boldsymbol{f}_{bn_x} &= {}^t\mathbf{D}_{bf}.\boldsymbol{x}_{fn} + \mathbf{D}_b.\boldsymbol{x}_{bn}, \\ \boldsymbol{f}_{bn_y} &= {}^t\mathbf{D}_{bf}.\boldsymbol{y}_{fn} + \mathbf{D}_b.\boldsymbol{y}_{bn}, \\ \boldsymbol{f}_{bn_z} &= {}^t\mathbf{D}_{bf}.\boldsymbol{z}_{fn} + \mathbf{D}_b.\boldsymbol{z}_{bn}, \end{aligned} \tag{4}$$

for the external forces applied at blocked nodes. The different $\mathbf{D}_i$ matrices are obtained through the decompositions of the ${}^t\mathbf{C}.\mathbf{Q}.\mathbf{C}$ matrix.

Conversely, being given a set of external forces applied to the nodes of the bar network, the position of the free nodes are given by:

$$\begin{aligned} \boldsymbol{x}_{fn} &= (\mathbf{D}_f)^{-1}.(\boldsymbol{f}_{fn_x} - \mathbf{D}_{bf}.\boldsymbol{x}_{bn}), \\ \boldsymbol{y}_{fn} &= (\mathbf{D}_f)^{-1}.(\boldsymbol{f}_{fn_y} - \mathbf{D}_{bf}.\boldsymbol{y}_{bn}), \\ \boldsymbol{z}_{fn} &= (\mathbf{D}_f)^{-1}.(\boldsymbol{f}_{fn_z} - \mathbf{D}_{bf}.\boldsymbol{z}_{bn}). \end{aligned} \tag{5}$$

These last equations show how it is possible to manipulate indirectly the node positions, and consequently the inner vertices, through the manipulation of external forces (see Pernot, 2004, for the treatment of configurations where the $\mathbf{D}_f$ matrix is singular). The unknowns of the deformation process are not the positions but the external forces themselves. Even if only the external forces applied at the free nodes are necessary to compute the new position of the free nodes, some external forces applied to the blocked nodes connected to at least one free bar, *i.e.* a bar that changes length during the process, may vary during the deformation and can therefore be taken into account for the definition of the objective function to be minimized (section 5.3). They are linearly dependent on the external forces applied at free nodes and the corresponding equations can be obtained using a combination of equations (4) and (5).

5.2 *Optimization problem formulation*

To avoid the direct manipulation of the external forces applied to the free nodes, *i.e.* the inner mesh vertices that are not on the hole contour, an optimization problem is written as follows:

$$\begin{cases} \boldsymbol{G}(\boldsymbol{F}) = \mathbf{0} \\ \min \phi(\boldsymbol{F}) \end{cases} \tag{6}$$

where the unknown vector $\boldsymbol{F}$ contains the components of all the external forces applied to free nodes. Such a formulation clearly decouples:

- the *geometric constraints* specified either between the inner and surrounding meshes (*e.g.* position, tangency or curvature blending conditions) or directly on the inner mesh to further constrain its shape. These constraints give rise to a set of possibly non-linear equations directly linking the position of the free vertices. The resulting constraint vector $\boldsymbol{G}$ can be expressed as a function of the external forces applied to the free nodes using equation (5).
- from the *function* ϕ to minimize. Various deformation behaviours can be defined through the combination of several geometric and/or mechanical quantities related to the bar network. It is the heart of our approach.

Returning to the deformation of the topological grid, no constraints are specified between the inner and surrounding meshes. The inner mesh is deformed while minimizing an objective function that takes into account some geometric information (*e.g.* normal or curvature evolution) available on the surrounding mesh (section 5.3). Thus, there are no non-linear equations or over-constrained configurations to be solved. Some geometric constraints may be specified to adjust the internal shape of the inner mesh (section 5.4). In any case, they are not used to specify the blending conditions between the two meshes.

5.3 *Curvature variation minimization*

As already stated, most of the time, the optimization problem is reduced to the minimization of the objective function $\phi(\boldsymbol{F})$. Theoretically, any combination of basic quantities expressed as a quadratic function of the unknown vector can be used to define this single objective function (Pernot *et al.*, 2005a). The minimization of these quantities gives rise to a system of linear equations that can be solved easily (no need of an iterative process).

Among the various combinations that can be used, the simplest one is the minimization of all the external forces:

$$\phi_F = \sum_i \boldsymbol{f}_i^2 = \sum_i f_{x_i}^2 + f_{y_i}^2 + f_{z_i}^2 \tag{7}$$

Since the external forces applied to the free nodes are independent from each other (figure 9(a)), the minimization of this sum on all the free nodes always produces a null unknown vector (no crossed terms in the quadratic function). This tends to minimize the curvature of the underlying geometry while producing areas as planar as possible (figure 10(a)). This minimization is interesting in the context of tensile textile structures (Véron *et al.*, 1998). Moreover, one can notice that an external force is only influenced by both the position of the

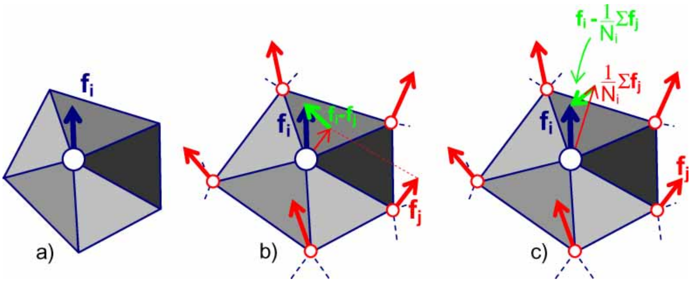

Figure 9. External forces taken into account when minimizing either the force applied to the *i*th node (a), or the variations of this force with respect to the surrounding ones (b), or its variation with respect to the average of the surrounding ones (c).

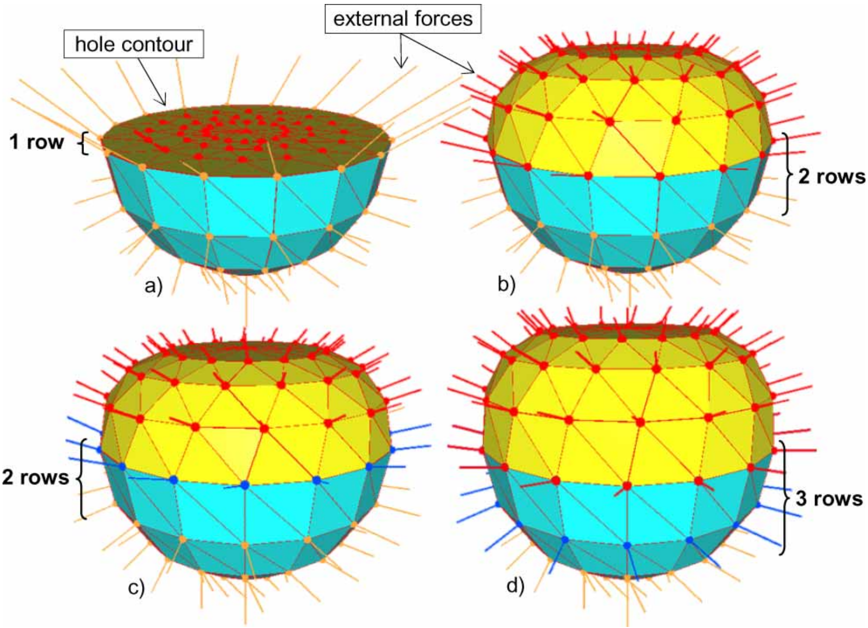

Figure 10. Minimization of ϕ_F applied to free nodes and to nodes of the hole contour (a, b). Minimization of $\phi_{\Delta^r F}$ applied to free nodes and to nodes of the hole contour (c, d).

node on which it is applied and the position of the vertices directly connected to it (figure 9(a)). Therefore, only the information relative to the position of the hole contour vertices is considered (the case of figure 3(a)). This does not answer our requirements since neither the tangency evolution nor the curvature evolution are preserved across the hole contour. To take into account more geometric information on the surrounding mesh, one can add in the sum of equation (7) the external forces applied to the nodes of the hole contour. Using a combination of the equations (4) and (5), one can obtain a quadratic objective function with squared and crossed terms. The influence area is larger since it uses geometric information relative to the hole contour and to the second row of the surrounding mesh, *i.e.* relative to all the vertices that are connected to the vertices of the hole contour (figure 10(b)). Unfortunately, such a minimization cannot pretend more than the preservation of the tangency with the surrounding mesh (figure 3(b)).

To reach the level of curvature blending condition, one should use geometric information relative to the three first rows of the surrounding mesh (the case of figure 3(c)). While noticing that the variation between two successive external forces represents in some sense the evolution of the curvature between the two nodes, the following minimization can be adopted:

$$\phi_{\Delta^r F} = \sum_i \sum_j \left[\boldsymbol{f}_i - \boldsymbol{f}_j \right]^2 \tag{8}$$

where the sum on i is performed on all the nodes and the sum on j is restricted to the nodes connected to node i. The use of such a function enables the minimization of the variations between couples of external forces applied to nodes directly connected together. The influence area of these basic quantities is greater than in the previous case (figure 10(c)). This is due to the fact that all the external forces applied to the nodes j, directly connected to a given node i, are also taken into account (figure 9(b)) and that each external force is influenced by both the position of the node on which it is applied and the position of the vertices directly connected to it. But our initial requirements are still not satisfied if the minimization of this sum is performed on all the free nodes (figure 10(c)). In this case, only the two first rows of the surrounding mesh are used (the case of figure 3(b)). To take into account the third row, the sum on i also has to take into account the nodes of the hole contour (figure 10(d)). In this case, the evolution of the curvature is preserved when crossing the hole contour. More exactly, the curvature variation minimization is approximated by the external forces relative variation minimization.

To restrict the influence of both the triangle mesh topology (*i.e.* connections between the vertices) and the shape of the triangles (*i.e.* size of the edges) on the final result, two major adaptations are proposed:

- to avoid an imbalance between quantities computed at nodes connected to only a few faces and quantities computed at nodes connected to many faces, the influence factor $1/N_i$ is introduced with N_i the number of edges connected to a given node i:

$$\phi_{\Delta^{pr} F} = \sum_i \sum_j \frac{1}{N_i} \left[\boldsymbol{f}_i - \boldsymbol{f}_j \right]^2 \tag{9}$$

- to avoid an imbalance between parts of the mesh with long edges and parts with short edges, the **Q** matrix (eq. 2) is initialised with densities inversely proportional to the length of the bars. Thus, the norms of the external forces applied to the nodes of each bar network are almost equal everywhere (figure 11).

Moreover, one can also notice in the formulation of equation (9) that the variations between the external forces are taken into account separately. In order to globalize the effect of

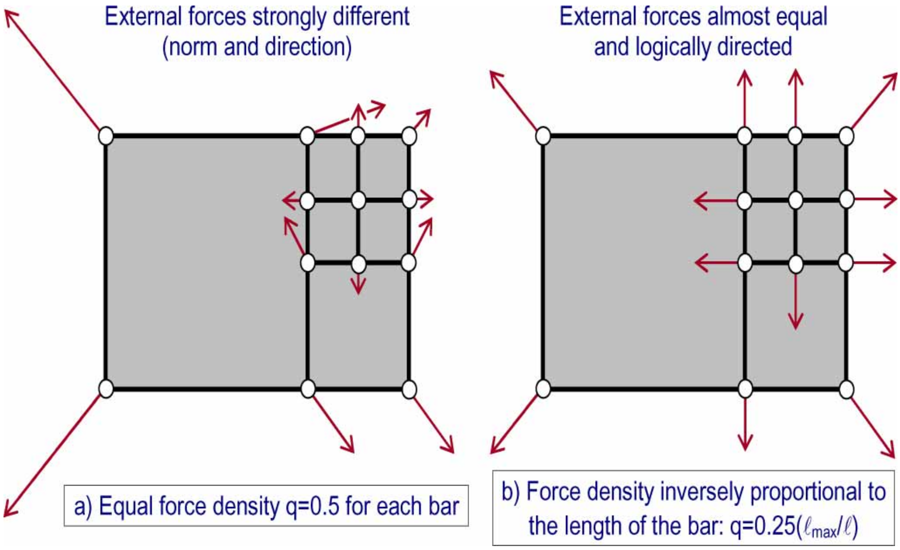
External forces strongly different (norm and direction)
External forces almost equal and logically directed
a) Equal force density q=0.5 for each bar
b) Force density inversely proportional to the length of the bar: q=0.25(ℓmax/ℓ)

Figure 11. Influence of the force densities initialisation over the norm and the direction of the external forces.

the external forces surrounding a given node i, the following new adaptation is proposed (figure 9(c)):

$$\phi_{\Delta^{gpr}F} = \sum_i \left[\boldsymbol{f}_i - \frac{1}{N_i} \sum_j \boldsymbol{f}_j \right]^2 \tag{10}$$

To finish, it is useless to try to use more rows of the surrounding mesh since the supplementary terms would be constant and would not affect the result of the minimization. Moreover, being able to minimize the curvature variation between the inner and surrounding meshes is enough for most applications.

5.4 *Shape adjustments*

As stated in section 5.2, it is possible to combine the use of an adapted minimization criterion with the specification of an optional constraint vector $\boldsymbol{G}(\boldsymbol{F})$. Here, geometric constraints can be used to further shape the interior of the inner mesh. This is very interesting when the user holds extra information concerning the shape of the missing part (*e.g.* photography of the physical object).

In practice, any constraint expressed as a function of the coordinates of the mesh vertices is acceptable. Using equation (5), it is always possible to go back to a set of equations that depends on the unknown vector $\boldsymbol{F}$. In case of non-linear constraints, a linearization is performed at the first order and the resolution using a Lagrangian becomes iterative. Among the various possibilities, Pernot (2004) has proposed a set of point constraints linking a parametric point lying on a mesh (not necessarily one of the vertices) with a geometric point in 3D space. Either position, tangency, or distance conditions may be specified between the two. An approach similar to the one of Cheutet *et al.* (2005) could also be used to insert a planar area inside the inner mesh. In the case of a closed mesh, one could also constrain the interior volume (Lien and Kajiya, 1984). An example that uses several constraint lines is proposed in section 6.

6. Results and discussion

The proposed example corresponds to a configuration where the initial model of an artificial climbing hold needs to be updated after a sanding operation (figure 12). To illustrate the power of the approach, a hole is created inside a complex area. After a cleaning operation, the hole is filled in with a topological grid defined with about 100 boundary vertices. See the shape of the hole, the user has instantiated manually a ratio $h = 0.5$. The shape of the inner mesh can be determined while using either the external forces or the approximated curvature variation minimizations. In both cases, no additional constraints are specified. Using the minimization of the external forces, nothing ensures a smooth connection between the inner and surrounding meshes and the repaired area is easily identifiable on the final mesh (figure 12(b_i)). On the contrary, the minimization of the external forces' relative variation smoothes the connection (figure 12(c_i)) but the inserted area is distinguishable. This example shows the limit of the proposed approach. Effectively, as it is, our approach is unable to retrieve complex missing shapes. To overcome this limit, some shape adjustments can be performed while specifying additional constraints (section 5.4). Here, the user has created

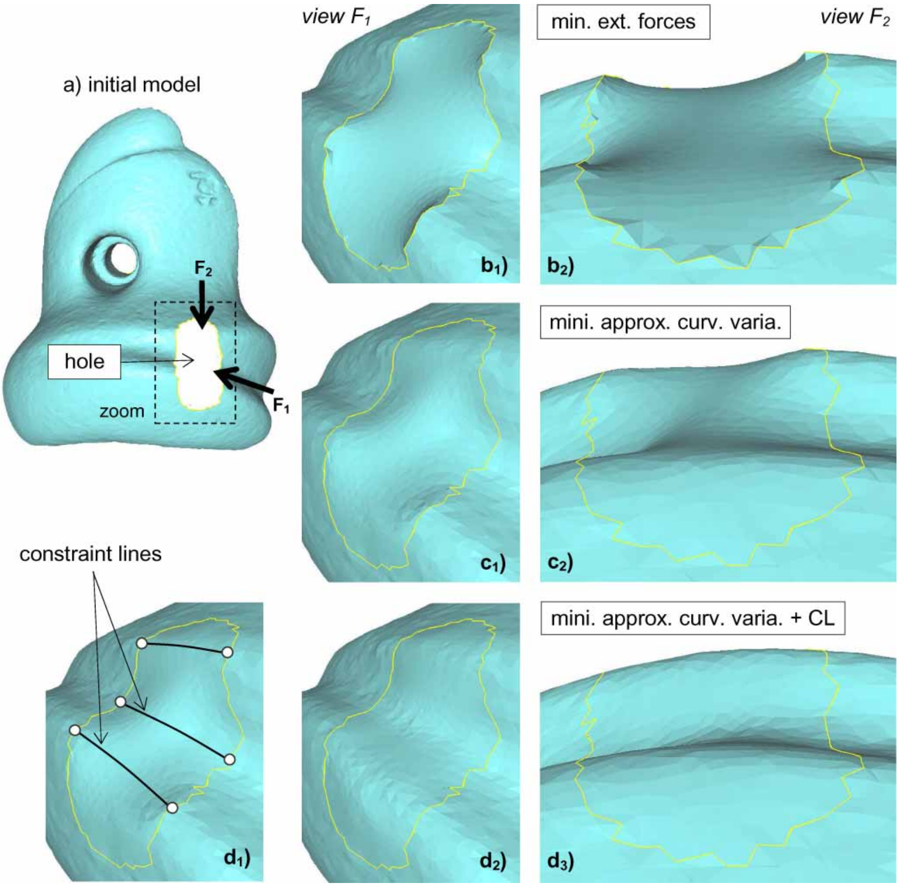

Figure 12. Filling a hole inside the polyhedral model of a digitized artificial climbing hold (courtesy Tomoadour). The minimization of the external forces does not produce a smooth connection with the surrounding mesh (b_i). This is improved while using the minimization of the approximated curvature variation (c_i). Details of the inner shape can be retrieved while specifying manually additional constraint lines the mesh has to satisfy (d_i).

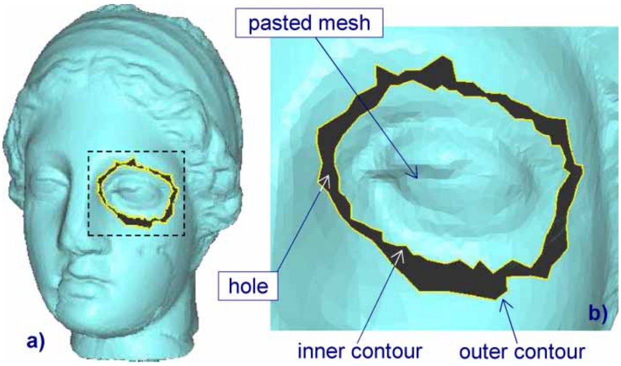

Figure 13. Example requiring a grid topological equivalent to a ring (courtesy University of Thessaloniki).

manually three constraint lines joining the two sides of the hole. The various points have been chosen according to the shape of the surrounding mesh, *i.e.* in such a way that the constraint lines prolong what could be the edges of the missing part (figure 12(d_1)). These lines are then discretized into a set of point constraints constraining parametric points lying on the mesh with geometric points lying on the curves. The points on the mesh are obtained by projection. These constraints plus the minimization of the approximated curvature variation form an optimization problem whose solution is close to the shape that was initially removed (figures 12(d_2) and (d_3)).

Other examples could also be proposed to illustrate the interest of such a hole-filling process. One such example in the cultural heritage preservation domain where the goal is to obtain a digital representation of historical monuments, statues and so on. Here again, the main constraint lies in the fact that the filled areas must not be easily identifiable. Unfortunately, the use of constraint lines will not be always sufficient to reveal the shape of complex physical objects and some details may never be recovered. Figure 13(a) shows a geometric model where the left eye of a statue has been voluntarily removed. In this example, it could be interesting to be able to copy and paste the right eye inside the area corresponding to the left eye (figure 13(b)). To be able to treat such a configuration and find the appropriate blending mesh, our filling algorithm must be upgraded to enable the insertion of another type of topological grid. Here, a grid topologically equivalent to a ring would be appropriate, the number of vertices being defined by the inner and outer contours.

More generally, the definition of grids enabling the filling of holes having several islands should be considered. It could ease the resolution of configurations where several shapes (possibly coming from a features library) are pasted inside a single hole, or simply when the digitized model presents holes having one or several islands. These improvements should affect solely the topological filling step.

Finally, one can notice that as soon as the hole does not look like an ellipse, the inner mesh is stretched anisotropically, thus producing degenerated triangles, which may ease the distinction between the inner and surrounding meshes. To overcome these limits, two alternatives can be imagined and will be further studied in the future. First, the shape of the topological grids could be adapted to the shape of the holes. It means that a grid topologically equivalent to a disk (*i.e.* one contour) can be obtained from an area bounded by either a circle or an ellipse or a rectangle or any other complex contour. Here, the difficulty is to find a 2D contour that looks like the shape of the hole. The second idea could be to refine the minimum area triangulation of Barequet and Sharir (1995).

7. Conclusion

In this paper, a complete toolbox to fill in holes in meshes during the reverse engineering of products has been proposed. The approach is modular, so that each module can be improved separately. The hole contour is first identified and cleaned to removed degenerated and badly oriented triangles that are due to the scanner noise. The hole is then filled in with a topological grid built from the boundary vertices and whose shape is defined by the ratio between the lengths of the minor and major axes of an ellipse. This grid is then deformed to satisfy specific blending conditions between the inner and surrounding meshes. New methods and tools have been imagined at each of these steps. The deformation itself results from the minimization of an objective function that uses a linear mechanical model to approximate the curvature variation between the two meshes. Several adaptations have been proposed to restrict the influence of both the triangle mesh topology and the shape of the triangles. The whole process has been validated on several examples coming from our prototype software. It gives very good results in configurations where the hole looks like an ellipse.

Additional work should be performed to be able to adapt both the topology (disk, ring and so on) and the shape (circle, rectangle and so on) of the grid to the shape of the hole. One idea could be to refine the minimum area triangulation of Barequet and Sharir (1995). This improvement should solely affect the topological filling step.

Our long-term objective is to structure the geometric models that are used all along the product lifecycle (*e.g.* sketches, polyhedral models, B-Rep models, FE models). The definition of the relationships between these structured models will give rise to a single multi-representation model, shared between the actors of the product lifecycle, and on which the semantic proper to each task will be attached (Pernot *et al.*, 2005b). This also corresponds to one of the long term objectives of the European Network of Excellence (Aim@Shape, 2004). The set of models, methods and tools proposed in this paper forms a complete toolbox used to clean and repair the polyhedral models in order to prepare their structuring.

References

Aim@Shape, Advanced and innovative models and tools for the development of semantic-based systems for handling, acquiring, and processing knowledge embedded in multidimensional digital objects, *European Network of Excellence, Key Action: 2.3.1.7 Semantic-based knowledge systems, VI Framework, URL: http://www.aimatshape.net*, 2004.

Amenta, N., Bern, M. and Kamvysselis, M., A new Voronoi-based surface reconstruction algorithm, *Proceedings of the 25th annual conference on computer graphics and interactive techniques*, 1998.

Barequet, G. and Sharir, M., Filling gaps in the boundary of a polyhedron, *Computer Aided Geometric Design*, 1995, **12**(2), 207–229.

Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C. and Taubin, G., The ball-pivoting algorithm for surface reconstruction, *IEEE Transactions on Visualization and Computer Graphics*, 1999, **5**(4), 349–359.

Cheutet, V., Pernot, J.-P., Léon, J.-C., Falcidieno, B. and Giannini, F., Insertion of planar areas into free-form surfaces in early product design, *Proceedings of ASME DAC'05: International Design Engineering Technical Conferences And Design Automation Conference*, 2005.

Ciarlet, P.G., *The Finite Element Method for Elliptic Problem* (North Holland), 1978.

Clarenz, U., Diewald, U., Dziuk, G., Rumpf, M. and Rusu, R., A finite element method for surface restoration with smooth boundary conditions, *Computer Aided Geometric Design*, 2004, **21**(5), 427–445.

Curless, B. and Levoy, M., A volumetric method for building complex models from range images, *Processing of SIGGRAPH'96*, 1996, pp. 303–312.

Davis, J., Marschner, S.R., Garr, M. and Levoy, M., Filling holes in complex surfaces using volumetric diffusion, *Processing of the first International Symposium on 3D Data, Visualization and Transmission*, 2002.

Desbrun, M. Meyer, M. Schrder, P. and Barr, A.H., Implicit fairing of irregular meshes using diffusion and curvature flow, *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, 1999, pp. 317–324.

Eck, M. and Hoppe, H., Automatic reconstruction of B-spline surfaces of arbitrary topological type, *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996.

Edelsbrunner, H. and Mücke, E.P., Three-dimensional alpha shapes, *ACM Transactions on Graphics*, 1994, **13**(1).

Forsey, D.R. and Bartels, R.H., Surface fitting with hierarchical splines, *ACM Transactions on Graphics*, 1995, **14**(2).

Hoppe, H., DeRose, T., Duchamp, T., McDonald, J. and Stuetzle, W., Surface reconstruction from unorganized points, *Proceedings of the 19th annual conference on computer graphics*, 1992, pp. 71–78.

Kobbelt, L., Campagna, S., Vorsatz, J. and Seidel, H.-P., Interactive multi-resolution modeling on arbitrary meshes, *Proceedings of SIGGRAPH'98*, 1998, pp. 105–114.

Lee, A., Moreton, H. and Hoppe, H., Displaced subdivision surfaces, *Proceedings SIGGRAPH'00*, 2000.

Lien, L. and Kajiya, J., A symbolic method for calculating the integral properties of arbitrary nonconvex polyhedra, *IEEE Comput. Graph. Appl.*, 1984, **4**(9), 35–41.

Liepa, P., Filling holes in meshes, *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry Processing* (*SGP'03*), 2003, pp. 200–205.

Ma, W. and Kruth, J.P., Parameterization of randomly measured points for least squares fitting of B-spline curves and surfaces, *Computer-Aided Design*, 1995, **27**(9), 663–675.

Ma, W. and Zhao, N., Catmull–Clark surface fitting for reverse engineering applications, *Proceedings of Geometric Modeling and Processing, Theory and Applications*,2000, pp. 274–283.

Nooruddin, F.S. and Turk, G., Simplification and Repair of Polygonal Models Using Volumetric Techniques, *IEEE Transactions on Visualization and Computer Graphics*, 2003, **9**(2), 191–205.

Pernot, J.-P., Fully free form deformation features for aesthetic and engineering designs, PhD thesis, INP-Grenoble (France) and University of Genoa (Italy), 2004 http://www.3s.hmg.inpg.fr/ci/doc/.

Pernot, J.-P., Guillet, S., Léon, J.-C., Falcidieno, B. and Giannini, F., Shape tuning in fully free form deformation features, *Journal of Computing and Information Science in Engineering* (*JCISE*), 2005a, **5**(1), 95–103.

Pernot, J.-P., Falcidieno, B., Giannini, F. and Léon, J.-C., Fully free form deformation features for aesthetic shapes design, *Journal of Engineering Design* (*JED*), 2005b, **16**(2), 115–133.

Pfeifle, R. and Seidel, H.-P., Triangular B-splines for blending and filling of polygonal holes, *Proceedings of Graphics Interface*, 1996, pp. 186–193.

Schek, H.J., The force density method for form finding and computation of general networks, *Computational Methods in Applied Mechanics and Engineering*, 1974, **3**(2), 115–134.

Schneider, R. and Kobbelt, L., Geometric fairing of irregular meshes for free-form surface design, *Computer-Aided Geometric Design*, 2001, **18**(4), 359–379.

Takeuchi, S., Kanai, T., Suzuki, H., Shimada, K. and Kimura, F., Subdivision surface fitting with QEM-based mesh simplification and reconstruction of approximated B-spline surfaces, *Proceedings of the 8th pacific conference on Computer Graphics and Applications*, 2000, pp. 202–212.

Tao Ju, Robust repair of polygonal models, *ACM Transaction on Graphics* (*TOG*), 2004, **23**(3), pp. 888–895.

Tekumalla, L.S. and Cohen, E., A hole-filling algorithm for triangular meshes, Technical report UUCS-04-019, School of Computing, University of Utah, USA, 2004, http://www.cs.utah.edu/techreports/.

Verdera, J., Caselles, V., Bertalmio, M. and Sapiro, G., Inpainting surface holes, *International Conference on Image Processing* (*ICIP*), 2003.

Véron, P., Trompette, P. and Léon, J.-C., Integrated design and collaborative engineering of fabric structures, *Engineering with Computers*, 1998, **14**, 23–35.

Wang, J. and Oliveira, M.M., A hole-filling strategy for reconstruction of smooth surfaces in range images, *Proceedings of SIBGRAPI'03*, 2003.