Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 225 (2007) 165–182
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Unstructured surface mesh adaptation using the Laplace–Beltrami target metric approach

Glen Hansen [a,*], Andrew Zardecki [b]

[a] *Multiphysics Methods Group, Idaho National Laboratory, Idaho Falls, ID 83415-3840, United States*
[b] *Los Alamos National Laboratory, United States*

Received 20 June 2006; received in revised form 15 November 2006; accepted 17 November 2006
Available online 26 January 2007

Abstract

This paper develops a set of adaptive surface mesh equations by using a harmonic morphism, which is a special case of a harmonic map. These equations are applicable both to structured and unstructured surface meshes, provided that the underlying surface is given in a parametric form. By representing the target metric of the mesh as a sum of a *coarse-grained* component and a component quadratic in surface gradients, an improved surface mesh may be obtained. The weak form of the grid equations is solved using the finite element approximation, which reduces the grid equations to a nonlinear, algebraic set. Examples of structured and unstructured meshes are used to illustrate the applicability of the proposed approach.

MSC: 65M50; 51P05; 65N30

Keywords: Finite elements; Galerkin methods; Surface mesh generation; Surface mesh adaptation

1. Introduction

Many effective algorithms exist for the generation of a mesh that discretizes a surface contained in three-dimensional space. A typical approach begins with the discretization of the boundary of the surface (e.g., its edges). In this method, bounding curves that define the surface edges are first discretized (e.g., the method advanced by Khamayseh and Kuprat [1]). Given a suitable boundary discretization, a surface mesh may then be formed using a surface grid generation algorithm such as an advancing front method or boundary parameter interpolation.

Several considerations influence the suitability of the resulting surface grid:

* Corresponding author. Tel.: +1 208 526 8494.
E-mail addresses: Glen.Hansen@inl.gov (G. Hansen), azardecki@bellsouth.net (A. Zardecki).

0021-9991/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2006.11.033

(1) the requirements of the volume mesh generation algorithm that often follows the surface discretization process,
(2) the requirements of the ultimate computational physics application that will use the surface or volume mesh, and
(3) the need to incorporate detailed geometric characteristics of the surface upon which the mesh lies into the surface mesh geometry itself.

This paper is focused on the last requirement. A "high quality" surface mesh on complex (e.g., highly-curved) surfaces is created by developing an elliptic surface mesh adaptation method based on the concept of harmonic morphisms, which is a special case of a harmonic map. This approach extends existing elliptic surface grid generation methodology as it lends itself to a convenient implementation for addressing unstructured and hybrid surface meshes. The Laplace–Beltrami grid equations that form the basis for the method have been shown to be effective for other grid smoothing tasks in two- and three-dimensional geometries [2,3]. These equations employ a target metric tensor that, together with the driving terms, controls the ultimate form and geometric characteristics of the final adapted grid. The method relies on modification of the target metric tensor presented previously to include information about the spatial gradients of the surface. As such, the proposed approach is conceptually similar to an approach advanced by Liseikin [4], who writes the Beltrami equations employing a metric tensor derived from a monitor surface. Unlike Liseikin, the *coarse-graining procedure*, in which the average node positions are used to estimate the target metric tensor [5], is used on unstructured surface meshes.

The framework of harmonic mappings, introduced by Eells and Sampson [6], has been used extensively for efficient grid generation as exemplified by the articles of Mastin and Thompson [7,8] and Ivanenko [9]. Khamayseh and Mastin [10] formulate an elliptic generation system for parametrically-defined surfaces. In theoretical physics, Misner applies harmonic mapping theory to a class of nonlinear field equations [11].

Techniques based on harmonic maps have been used by various authors to adapt the grid to physical requirements such as shock wave formation [12–14]. When the grid is defined on a two-dimensional surface contained in three-dimensional space, the adaptation process consists of moving the grid nodes in a manner that captures the curvature of the surface in contrast to the original, typically uniform, grid.

The increased significance of unstructured and hybrid surface grids in simulation applications [15,16] provides a motivation for this study of grid adaptation based on a uniform framework of elliptic smoothers. For surfaces defined parametrically or represented analytically by rational B-splines, the proposed approach generalizes the results presented by Khamayseh and Mastin [10], who restrict their investigation to structured surface grids.

The body of this paper is organized as follows. Section 2 summarizes the basic notion of a harmonic morphism, also termed a *semiconformal mapping*. This is a special case of a harmonic mapping. Under semiconformal maps the scalar products of vectors in tangent spaces are related by a location-dependent dilatation factor. In Section 3, the Euler–Lagrange equations of harmonic morphisms are used to formulate the two-dimensional surface grid equations. In this setting the applicability of the approach presented by Khamayseh and Mastin is extended to unstructured grids. This section also discusses the implementation and improvement of the proposed method. Section 4 presents examples of selected surface grid domains. Finally, in Section 5 conclusions and recommendations are formulated.

2. Semiconformal mapping

Given two Riemannian manifolds (M,g) and (N,h) of dimension m and n, equipped with metrics g and h, respectively, consider a regular map $u : M \to N$. In local coordinates $(\xi^1,\ldots,\xi^m)$ on M and $(u^1,\ldots,u^n)$ on N, the map u is written as

$$u(\xi) = (u^1(\xi^1,\ldots\xi^m),\ldots,u^n(\xi^1,\ldots\xi^m)).$$

Note that the symbol u is used to represent both the mapping and the coordinate label (i.e., (u^1,u^2) or (u,v)) to remain consistent with the literature [17,18]. The energy density of u is defined as

$$e(u)(\xi) = \frac{1}{2} g^{\alpha\beta}(\xi) \frac{\partial u^i}{\partial \xi^\alpha} \frac{\partial u^j}{\partial \xi^\beta} h_{ij}(u(\xi)). \tag{1}$$

In this development, index notation is employed, which implies summation over repeated indices. In Eq. (1), $g^{\alpha\beta}$ is the contravariant metric tensor given as an inverse matrix with respect to the covariant metric tensor $g_{\alpha\beta}$ of M,

$$g_{\alpha\beta} g^{\alpha\beta} = \delta^\beta_\alpha, \tag{2}$$

where δ^β_α is the Kronecker delta function.

The energy of the map u is

$$E[u] = \frac{1}{2} \int_M g^{\alpha\beta}(\xi) \frac{\partial u^i}{\partial \xi^\alpha} \frac{\partial u^j}{\partial \xi^\beta} h_{ij}(u(\xi)) \sqrt{g}\, \mathrm{d}\xi^1 \cdots \mathrm{d}\xi^n, \tag{3}$$

where $g = \det(g_{\alpha\beta})$.

The critical points of the energy functional satisfy Euler–Lagrange equations of the form

$$\tau(u) \equiv \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi^\alpha} \left(\sqrt{g} g^{\alpha\beta} \frac{\partial u^k}{\partial \xi^\beta} \right) + g^{\alpha\beta}(\xi) \Gamma^k_{ij}(u(\xi)) \frac{\partial u^i}{\partial \xi^\alpha} \frac{\partial u^j}{\partial \xi^\beta} = 0, \tag{4}$$

where Γ^k_{ij} are the Christoffel symbols of N, constructed with the aid of the metric h

$$\Gamma^k_{ij} = \frac{1}{2} h^{km} \left(\frac{\partial h_{im}}{\partial u^j} + \frac{\partial h_{jm}}{\partial u^i} - \frac{\partial h_{ij}}{\partial u^m} \right), \tag{5}$$

which are pulled back to M. Here h_{im} and h^{km} refer to covariant and contravariant components of the metric tensor of N. The left-hand side of Eq. (4) represents the tension field $\tau(u)$ of u.

In the theory of Riemann surfaces, one seeks a special coordinate system, in which the element of arc length is proportional to the Euclidean form. Such coordinates, termed isothermal, satisfy a pair of Laplace–Beltrami equations. A mapping between two Riemann surfaces is called conformal if it preserves the angle between two curves passing through each point of the surface. In particular, a mapping is conformal if it maps an isothermal system of coordinates into another isothermal system. A generalization of the idea of a conformal map to the case of Riemannian manifolds leads to the concept of a harmonic morphism. By definition, a continuous mapping $u : M \to N$ is a harmonic morphism if $f \circ u$ is a harmonic function, for every function f which is harmonic in an open set $V \subset N$. A harmonic function f in V is a function satisfying the Laplace equation

$$\frac{1}{\sqrt{h}} \frac{\partial}{\partial u^m} \left(\sqrt{h} h^{mn} \frac{\partial f}{\partial u^n} \right) = 0. \tag{6}$$

When $m \geqslant n$, a C^1-map $u : M \to N$ is called semiconformal with dilatation $\lambda(\xi)$ if the restriction of u to the set of points of M at which $\mathrm{d}u$ does not vanish is a conformal submersion. Here, $\mathrm{d}u$ is the differential of u. This definition means that at any point of M at which $\mathrm{d}u \neq 0$, the restriction of $\mathrm{d}u$ to the orthogonal complement $K^\perp_\xi$ of $K_\xi = ker\, \mathrm{d}u_\xi$

$$\mathrm{d}u_\xi|_{K^\perp_\xi} : K^\perp_\xi \to T_{u(\xi)} N \tag{7}$$

is a surjective and conformal map. In this expression, ker is the kernel of the differential map; the subset of the tangent space where the differential vanishes. In terms of local coordinates, Eq. (7) may be written as a condition that, restricted to the orthogonal complement of the kernel, $\mathrm{d}u$ simply multiplies the scalar products by $\lambda(\xi)$:

$$g_{\alpha\beta}(\xi) X^\alpha Y^\beta = \lambda^{-2}(\xi) h_{ij}(u(\xi)) \frac{\partial u^i}{\partial \xi^\alpha} \frac{\partial u^j}{\partial \xi^\beta} X^\alpha Y^\beta. \tag{8}$$

Since Eq. (8) holds for all tangent vectors X and Y, one has

$$g_{\alpha\beta}(\xi) = \lambda^{-2}(\xi) h_{ij}(u(\xi)) \frac{\partial u^i}{\partial \xi^\alpha} \frac{\partial u^j}{\partial \xi^\beta}. \tag{9}$$

For mesh generation, constant dilatation will be employed for simplicity. Much of this study will be based on $\lambda = 1$, but other values $\lambda < 1$ may be useful or perhaps mandatory for particular problems. As will be seen later in the paper, λ multiplies the Beltrami surface differential terms; in effect, it might be considered a parameter that can be selected to control the extent of the adaptation of the surface mesh to the character of the surface.

For a semiconformal map u with dilatation λ, the tension field of u becomes

$$\tau(u) = \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi^\alpha} \left(\sqrt{g} g^{\alpha\beta} \frac{\partial u^k}{\partial \xi^\beta} \right) - \frac{\lambda^2(\xi)}{\sqrt{h}} \left(\frac{\partial}{\partial u^i} \sqrt{h} h^{ik} \right)(\xi). \tag{10}$$

Eq. (10) is obtained by combining Eqs. (4) and (9) together with the identity

$$h^{ik} \Gamma^k_{ij} = -\frac{1}{\sqrt{h}} \left(\frac{\partial}{\partial u^i} \sqrt{h} h^{ik} \right), \tag{11}$$

which follows from a straightforward computation [19]. The Euler–Lagrange equations corresponding to the tension field given by Eq. (10) are

$$\tau(u) = 0. \tag{12}$$

The harmonic morphisms form a subset of harmonic maps: A mapping $u : M \to N$ is a harmonic morphism if and only if u is a semiconformal, harmonic mapping. In the case of manifolds M and N of equal dimension, one can also prove [20] that harmonic morphisms are precisely conformal mappings if $m = n = 2$, whereas for higher dimensions harmonic morphisms are conformal mappings with constant dilatation. In the special case of $\lambda(\xi) \equiv 1$, the harmonic morphisms are semiconformal maps if $n = m = 2$; for $m = n \geqslant 3$, they are homothetic (related by expansion or geometric contraction). In general, harmonic maps do not compose into harmonic maps, whereas harmonic morphisms do.

3. Surface grid equations

In the context of grid generation, it is customary to consider the physical domain as a map of the parametric domain with coordinates (u, v); see Fig. 1. When the grid is structured, a rectilinear grid in computational space (ξ, η) generates a grid in the parameter space, which maps to a curvilinear grid on the surface [21]. In general, as stressed in this paper, computational space is not required to carry a global grid; it is rather a Riemannian manifold characterized by local coordinates.

Within the framework of the harmonic morphism approach, the surface grid equations of Khamayseh and Mastin [10] are a direct consequence of the Euler–Lagrange Eq. (12), in which the tension field is given by Eq. (10) with $\lambda(\xi) = 1$. With the notation $(u, v) = (u^1, u^2)$, the grid equations in two dimensions read

$$\begin{aligned} \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi^\alpha} \left(\sqrt{g} g^{\alpha\beta} \frac{\partial u}{\partial \xi^\beta} \right) &= F(u(\xi, \eta), v(\xi, \eta)), \\ \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi^\alpha} \left(\sqrt{g} g^{\alpha\beta} \frac{\partial v}{\partial \xi^\beta} \right) &= G(u(\xi, \eta), v(\xi, \eta)), \end{aligned} \tag{13}$$

where the driving terms on the right-hand side of Eq. (13) are the Beltrami differentials

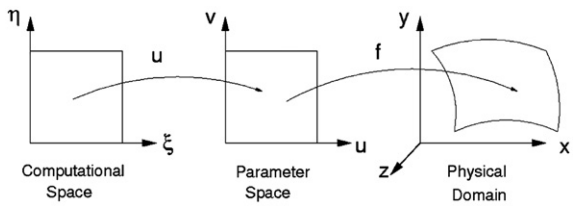

Fig. 1. Physical domain viewed as a composite mapping $f \circ u$ from parameter space and computational space.

$$F(u,v) = \frac{1}{\sqrt{h(u,v)}}\left[\frac{\partial}{\partial u}\left(\frac{h_{vv}}{\sqrt{h(u,v)}}\right) - \frac{\partial}{\partial v}\left(\frac{h_{uv}}{\sqrt{h(u,v)}}\right)\right],$$
$$G(u,v) = \frac{1}{\sqrt{h(u,v)}}\left[\frac{\partial}{\partial v}\left(\frac{h_{uu}}{\sqrt{h(u,v)}}\right) - \frac{\partial}{\partial u}\left(\frac{h_{uv}}{\sqrt{h(u,v)}}\right)\right]. \tag{14}$$

In Eq. (14), h_{uu}, h_{uv}, and h_{vv} denote the independent components of the metric tensor of N, and $h(u,v) = h_{uu}h_{vv} - h_{uv}^2$.

These adaptive grid equations express the fact that the mapping from computational space to parameter space is a semiconformal mapping with constant dilatation. In principle, the dilatation factor $\lambda(\xi)$ can be used to amplify the significance of driving terms that account for adaptation. The grid equations are adaptive since the initial mesh in the parameter space is adapted to the surface features in the physical space. Although Eq. (13) are formally identical to the equations derived by Khamayseh and Mastin, their significance is different. In fact, the underlying grid in parameter space need not be structured. It may consist of, for example, quadrilaterals and triangles of arbitrary connectivity. Equation (13), therefore, extend the validity of the Khamayseh and Mastin approach to a larger class of grids.

Adaptive grid generators often employ the concept of "monitor functions," which control the mesh movement towards an optimal configuration. A well-known example of a monitor function is provided by selecting a surface in three-dimensional Euclidean space, defined explicitly in terms of a function $f(u,v)$

$$x = u,$$
$$y = v,$$
$$z = cf(u,v), \tag{15}$$

where c is a scaling constant. The covariant metric tensor of this surface is

$$h_{ij} = h_{ij}^{\text{euclid}} + h_{ij}^{\text{quad}}, \tag{16}$$

where the Euclidean term is the unit matrix

$$[h_{ij}^{\text{euclid}}] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{17}$$

and the quadratic term is

$$[h_{ij}^{\text{quad}}] = c\begin{bmatrix} \left(\frac{\partial f}{\partial u}\right)^2 & \frac{\partial f}{\partial u}\frac{\partial f}{\partial v} \\ \frac{\partial f}{\partial u}\frac{\partial f}{\partial v} & \left(\frac{\partial f}{\partial v}\right)^2 \end{bmatrix}, \tag{18}$$

where $i,j = 1,2$. Mesh adaptation based on the monitor function induced by this metric tensor has been considered by Huang and Sloan [22] and MacKenzie [23]. A more general monitor function was introduced by Ceniceros and Hou [24].

In the finite element approach, the region Ω in the parameter space is partitioned into elements $\Omega = \{\Omega_e\}$. The finite element cells in the parametric domain are viewed as maps of the same canonical master element. For the quadrilateral cells, using the bilinear map as an example, the mapping is implemented through the basis functions

$$\varphi_1(\xi,\eta) = \frac{1}{4}(1-\xi)(1-\eta),$$
$$\varphi_2(\xi,\eta) = \frac{1}{4}(1+\xi)(1-\eta),$$
$$\varphi_3(\xi,\eta) = \frac{1}{4}(1+\xi)(1+\eta),$$
$$\varphi_4(\xi,\eta) = \frac{1}{4}(1-\xi)(1+\eta). \tag{19}$$

Given a quadrilateral Ω_e in the parameter plane with vertices (u_m^e, v_m^e), $m = 1, \ldots 4$, the transformation

$$\begin{aligned} u^e(\xi,\eta) &= \sum_{m=1}^{4} u_k^e \varphi_m(\xi,\eta), \\ v^e(\xi,\eta) &= \sum_{m=1}^{4} u_k^e \varphi_m(\xi,\eta), \end{aligned} \tag{20}$$

maps the square master element $\Omega_{\xi\eta} = \{-1 \leqslant \xi \leqslant 1, -1 \leqslant \eta \leqslant 1\}$ into Ω_e. Similar equations hold for triangular elements [2]. When the mesh cells are triangular, the summation in the discretized equations runs over three, not four, cell vertices. Equation (20) signify that each element Ω_e in parameter space is an image of the same master element $\Omega_{\xi\eta}$ in the computational space, with variables ξ and η playing the role of local coordinates.

Assume now that the metric in the parameter space is Euclidean, given by Eq. (17). Setting $\lambda = 1$ in Eq. (9), by virtue of Eqs. (20) one obtains

$$g_{\alpha\beta}^e(\xi) = \sum_{m=1}^{4}\sum_{n=1}^{4}(u_m^e u_n^e + v_m^e v_n^e)\frac{\partial\varphi_m}{\partial\xi^\alpha}\frac{\partial\varphi_n}{\partial\xi^\beta} \tag{21}$$

for the components of metric tensor associated with element Ω_e.

Equation (21), discussed by Lautersztajn and Samuelsson [25], defines the metric consistent with the given grid. As was demonstrated in an earlier work [2], the direct use of this metric does not lead to any grid motion when employed within the Laplace–Beltrami scheme.

To account for the spatial variation of the surface, the next logical step is to supplement the unit tensor in the parameter space by terms containing the gradients of f, where f describes the form of the surface, cf. Eq. (15). This replaces the metric tensor by a target metric tensor given as a sum of the unit tensor and a correction term. The correction term is expressed as a quadratic form in the components of the gradient of f. Such a procedure is reminiscent of the approach of D'Azevedo [26], who interpreted the Hessian matrix of the data function as metric tensor measuring the local approximation error. In the computational space (ξ, η), the metric tensor is obtained from Eq. (16) through the transformation given by Eq. (9), in which $\lambda = 1$.

As will be demonstrated in the following section, it is possible to improve the adaptation of the method if the Euclidean component of the metric in Eq. (17) is replaced by the coarse-grained metric, introduced in Ref. [3]. The coarse-graining procedure is an algorithm devised to compute an element's *target* metric tensor for use with the Laplace–Beltrami (LB) smoothing method. This development uses on the prescriptive specification of an ideal form and size of the element of interest, based on the characteristics of neighbor elements and solution features near the element in question. The LB method was derived to accept this input data as a tensor, the target metric tensor, that describes the geometry of this idealized element. The mechanics of specifying a target metric tensor is attractive from two perspectives; a target tensor can be readily computed automatically, without user input or free parameters, and it is intuitive. Indeed, from a user perspective, one specifies some local method that will improve a general element and the Laplace–Beltrami solution process acts to globalize that prescription across the entire mesh.

The term "coarse-graining" describes one possible local prescription algorithm. Consider the shaded triangle element ABC shown in Fig. 2. When compared with its neighbor elements (quadrilateral elements DAI, $ACHI$, EBD, $EFCB$, GCF, and triangles DBA and CGH), triangle ABC is somewhat smaller in area but is nicely shaped, geometrically. The coarse-graining procedure is an algorithm where an "idealized triangle" $\overline{ABC}$ is formed by calculating new locations of the triangle vertices, with the goal of specifying a target element size that is larger with respect to its neighbors. In this case, the new vertex location is calculated from an unweighted average of its edge-connected neighbors. For example, vertex $\bar{C}$'s coordinates are calculated by averaging the coordinates of nodes A, B, F, G, and H. The remaining coarse grained vertices $\bar{A}$ and $\bar{B}$ are calculated, similarly, to form a target triangle $\overline{ABC}$ on which the metric tensor is computed for input to the Laplace–Beltrami method. Clearly, one can develop other prescriptions and/or include solution adaptive behavior, if desired. This simple coarse-graining procedure is effective on the problems considered here, as will be demonstrated in Section 4 of this paper, and it is readily implementable for three-dimensional hybrid mesh topologies within parallel applications. The coarse-grained metric is formally expressed by Eq. (21), but the

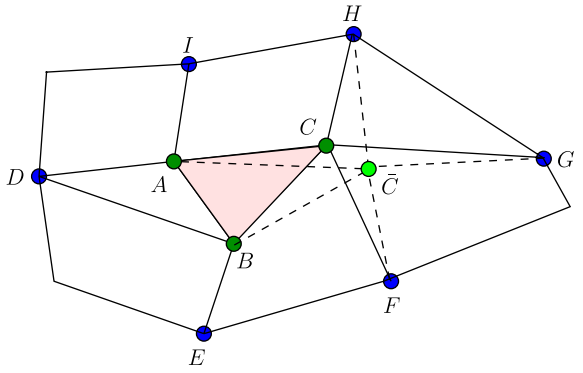

Fig. 2. Coarse graining: The target metric is based on averaged vertex positions, such as $\bar{C}$, which is the average of vertices A, B, F, G, and H.

locations of the vertices are calculated as an average of locations of vertices connected to a given vertex by an edge. The coarse-grained metric thus becomes

$$g_{\alpha\beta}^{\text{coarse}}(\xi) = \sum_{m=1}^{4}\sum_{n=1}^{4}(\langle u_m^e\rangle\langle u_n^e\rangle + \langle v_m^e\rangle\langle v_n^e\rangle)\frac{\partial\varphi_m}{\partial\xi^\alpha}\frac{\partial\varphi_n}{\partial\xi^\beta}, \tag{22}$$

in which

$$\begin{aligned}\langle u_m^e\rangle &= \frac{1}{N}\sum_{n=1}^{N} u_{mn}^e,\\ \langle v_m^e\rangle &= \frac{1}{N}\sum_{n=1}^{N} v_{mn}^e.\end{aligned} \tag{23}$$

The summation in Eqs. (23) extends over all nodes $1,\ldots,N$ connected to node u_m^e by an edge. The final form of the target metric tensor thus becomes

$$g_{\alpha\beta}(\xi) = g_{\alpha\beta}^{\text{coarse}}(\xi) + \frac{\partial u^i}{\partial\xi_\alpha}\frac{\partial u^j}{\partial\xi_\beta}h_{ij}^{\text{quad}}, \tag{24}$$

where h_{ij}^{quad} is given by Eq. (18).

This approach bears a relationship to the theory advanced by Liseikin [4], who introduces a monitor metric tensor defined as

$$h_{ij}^s = p(u,v)h_{ij} + q(u,v)\frac{\partial f}{\partial u^i}\frac{\partial f}{\partial u^j},$$

where $p(u,v)$ and $q(u,v)$ are smooth functions, and where h_{ij} is the metric tensor resulting from the surface geometry. When the coarse-grained metric is used, the effect of the metric is to smooth out the mesh non-homogeneities. The second part of the metric tensor, together with the driving terms, accounts for adaptivity.

As in Ref. [3], the system of partial differential grid equations, (13), is solved numerically using the finite element technique. This process converts the differential equations to a system of nonlinear algebraic equations of the form

$$\sum_{n=1}^{N} K_{mn}(a)a_n^i = -f_m^i, \quad i = 1,2, \tag{25}$$

where the stiffness matrix K_{mn} is expressed as

$$K_{mn} = \int_\Omega \frac{\partial\varphi_m}{\partial\xi^\alpha}\sqrt{g}g^{\alpha\beta}\frac{\partial\varphi_n}{\partial\xi^\beta}\,\mathrm{d}^2\xi. \tag{26}$$

The driving term in Eq. (25) is

$$f_m^i = \int_\Omega \varphi_m F^i \sqrt{g}\, \mathrm{d}^2\xi, \quad i = 1, 2. \tag{27}$$

In Eq. (25), the unknown coefficients a_n^i are the expansion coefficients of the coordinates u^i in terms of the basis functions $\varphi_{n(u)}$. The numerical details of the implementation are discussed in Ref. [3].

It is important to note that Eqs. (15) describe only a subset of surfaces that may be of interest to an application. Consequently, the metric tensor defined by Eq. (16) is not a general form of the metric tensor of an arbitrary surface. Indeed, the corkscrew surface considered in the next section is described by a more general form of the metric tensor than Eq. (16). Fortunately, the notion of the target metric allows one to ignore inconvenient complexity of the actual metric tensor of the surface; one again employs the chosen target metric estimation algorithm. For example, consider a parametric surface such as the corkscrew generated by Eq. (31). In this case, the actual surface metric is more complicated than Eq. (16), but the coarse-graining algorithm, Eqs. (18) and (22)–(24), is still employed to obtain the components of the metric used to drive the grid smoothing process.

Of additional note, the Beltrami differential driving terms control adaptation. The results shown in this paper are representative of surfaces described parametrically or by Eq. (15); the Beltrami differentials are computed by analytic differentiation of the surface description equation. This limitation is for convenience as it is conceptually straightforward to calculate these differentials on any surface description including those defined by sets of triangles or other polygonal data, using numerical means. The implementation of such a method is sufficiently involved that its presentation will be postponed for a future paper.

4. Numerical examples

The proposed surface grid generation method is concerned with two general qualitative goals: (1) capture surface details within the final surface mesh, and (2) provide effective results for general unstructured surface meshes.

The first example deals with a monkey saddle surface shown in Fig. 3. The monkey saddle has a cubic polynomial representation, $z = x^3 - 3x^2y$. The left image is constructed from a transfinite interpolation, in which the elements on the surface appear to be of similar size and uniformly distributed across the surface. The right image shows the adapted mesh in the physical space. Note that the elements of the adapted mesh have increased in size near the center of the surface. Additionally, away from the center, the element edges tend to intersect each other at angles closer to 90° than the original result.

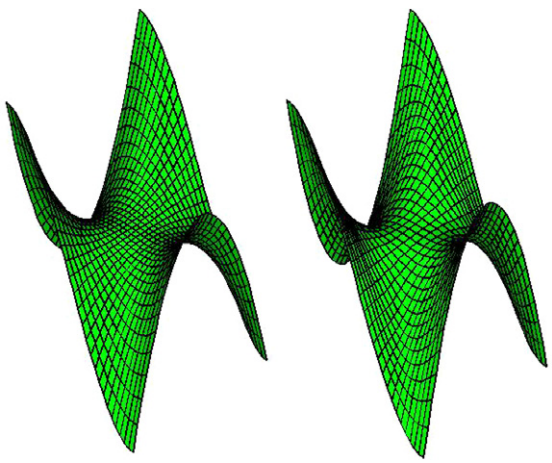

Fig. 3. The monkey saddle surface shown in physical space, hosting a Cartesian grid. To the left is the initial, uniform, grid; on the right is the adapted result.

If one now considers the Gaussian curvature of the surface, as shown in Fig. 4, it is possible to gain further insight into the adaptation process. Comparing Figs. 4 and 3, it is evident that the elements of the surface characterized by large gradients in Gaussian curvature are predominantly affected by the adaptation approach. In contrast to planar mesh smoothing, where the driving terms in Eq. (13) disappear; surface mesh adaptation is controlled by the Beltrami driving terms. These terms contain only the first derivatives of the metric tensor components, whereas the Gaussian curvature requires the second order derivatives. The similarity between these two entities may be seen by considering Brioschi's formula for the Gaussian curvature [27]. Under the assumption that cross-term metric component h_{uv} is negligible, the Gaussian curvature is given as

$$K_{\mathrm{G}} = -\frac{1}{2\sqrt{h}}\left[\frac{\partial}{\partial u}\left(\frac{1}{\sqrt{h}}\frac{\partial h_{vv}}{\partial u}\right) + \frac{\partial}{\partial v}\left(\frac{1}{\sqrt{h}}\frac{\partial h_{uu}}{\partial u}\right)\right], \tag{28}$$

where $h = h_{uu}h_{vv}$. Equation (28) shows that the Gaussian curvature is related to the derivative of the arithmetic average of the two Beltrami differentials.

The effect of curvature is often easier to visualize if the surface mesh is viewed in parameter space, as illustrated in Fig. 5. Apparent in the figure is a qualitative relationship between the Gaussian curvature of the surface and the concentration of mesh node points.

The second goal of this paper is to demonstrate effective surface mesh quality improvement for unstructured meshes. Fig. 6 illustrates a triangle mesh generated on the monkey saddle surface. In this case, the initial mesh was generated by scattering a random distribution of nodes across the surface, then using Delaunay triangulation to create the mesh. Given the low quality of the initial mesh on this surface and the large number of triangles present, it is not possible to see effects of the adaptation algorithm on this triangle mesh. More interesting, however, is the improvement of the visual quality of the triangle elements when the method was applied. Indeed, an effective elliptic surface mesh adaptation approach should smooth any variation of element size on the surface.

The process of concentrating smaller elements towards regions of surface curvature often has a side effect; it may decrease the overall geometric quality of the mesh elements by stretching or deforming them. A useful mesh generation algorithm must typically achieve some compromise between capturing surface details and the quality of elements, especially if it is not possible to change the surface element connectivity or mesh topology as the adaptation process evolves. Indeed, the proposed approach appears to improve element "quality," as it adapts the mesh. The definition of surface element quality, however, is a strong

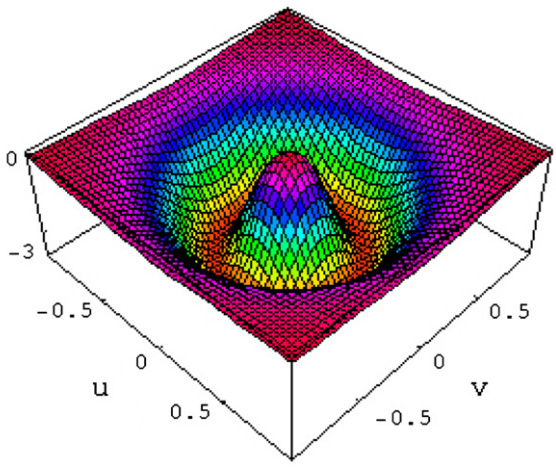

Fig. 4. Gaussian curvature of the monkey saddle surface.

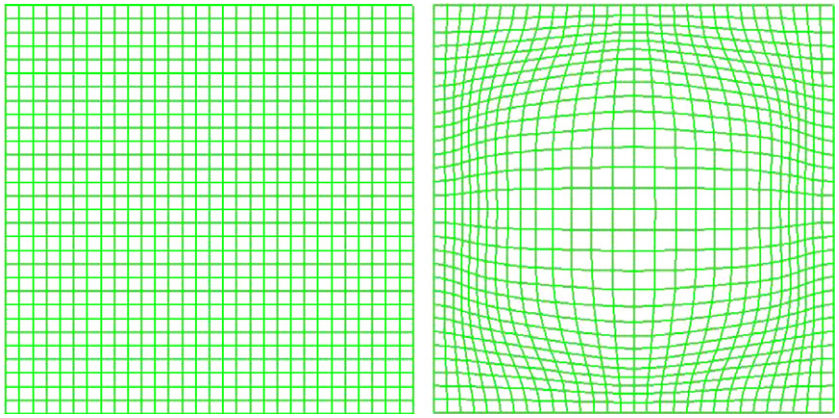

Fig. 5. The monkey saddle surface shown in parameter space. To the left is the initial, uniform, grid; on the right is the adapted result.

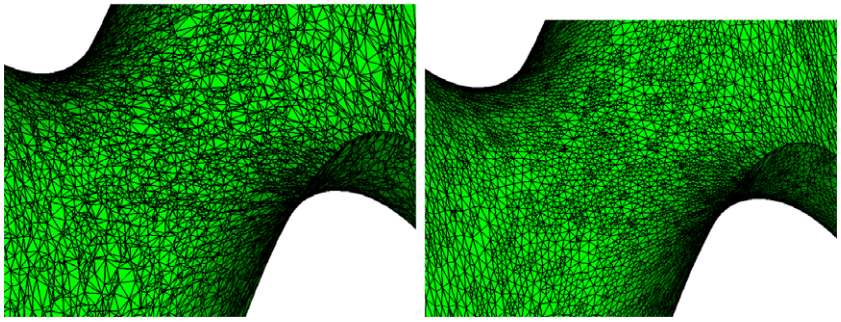

Fig. 6. A fragment of monkey saddle surface shown in physical space, corresponding to a triangular unstructured grid in parameter space. To the left is the initial, uniform, grid; on the right is the adapted result.

function of the requirements of the ultimate application where the mesh will be used, and thus difficult to describe in general terms.

Fig. 7 shows the parametric domain of a table surface, described by the equation

$$z = a \tanh 80\left[\frac{1}{16} - \left(x - \frac{1}{2}\right)^2 - \left(y - \frac{1}{2}\right)^2\right]. \tag{29}$$

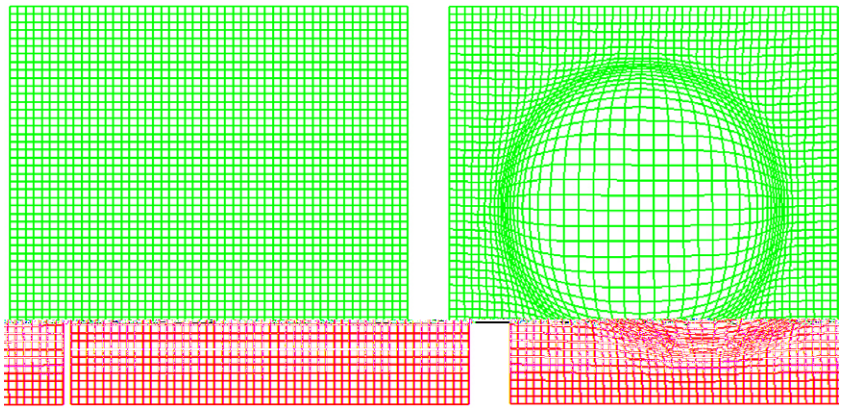

Fig. 7. The table surface shown in parameter space. To the left is the initial, uniform, grid; on the right is the adapted result.

The physical domain of the surface, with $a = 1/2$, is depicted in Fig. 8. This result reinforces the previous observations of the ability of the method to concentrate the mesh towards areas where the surface curvature changes abruptly, cf., Fig. 9. The steep sides of the table function require a finer mesh to capture this detail; the mesh generator indeed concentrates the mesh towards this region as desired. Additionally, on the sides of the table, the method visually improves both the aspect ratios of the surface elements and the skew angles of these elements. Figs. 7 and 8 were generated using only the coarse-graining metric estimation algorithm (Eq. (24), where $h = 0$).

Figs. 10 and 11 show the effect of the target metric adjustment. In Figs. 10 and 11, the target metric contains a contribution from the Euclidean term and a component quadratic in surface gradients. For the table surface, the scaling factor in Eq. (18) is $c = -0.001$. Fig. 10 was generated using a metric prescription calculated from Eq. (16), and Fig. 11 was generated using Eq. (24).

Fig. 12 displays a corrugated surface, such as one might use as a first-approximation to a threaded region of a bolt. The corrugated surface is a surface of revolution, defined by the parametric equations

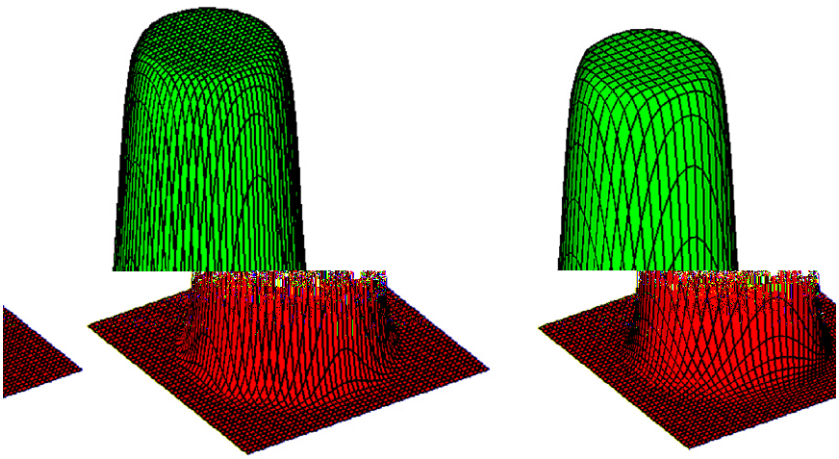

Fig. 8. The table surface shown in physical space, corresponding to the Cartesian grid in parameter space. To the left is the initial, uniform, grid; on the right is the adapted result.

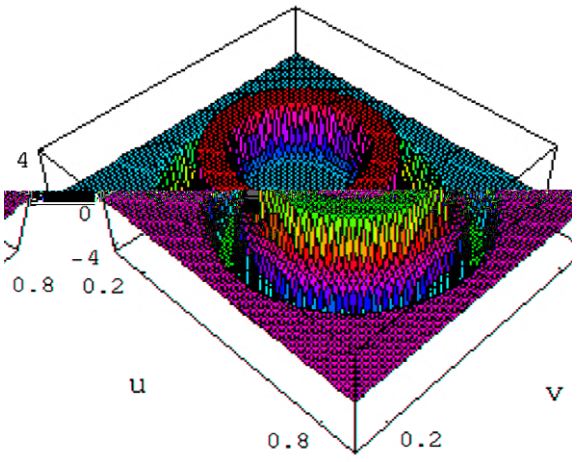

Fig. 9. Gaussian curvature of the table surface.

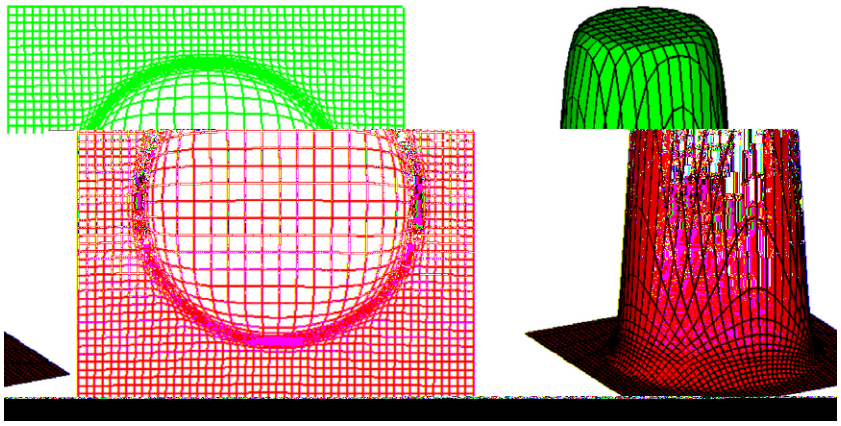

Fig. 10. The table surface adaptive grid shown in parameter space (left) and physical space (right). The target metric consists of the sum of a Euclidean term and a contribution quadratic in surface gradients.

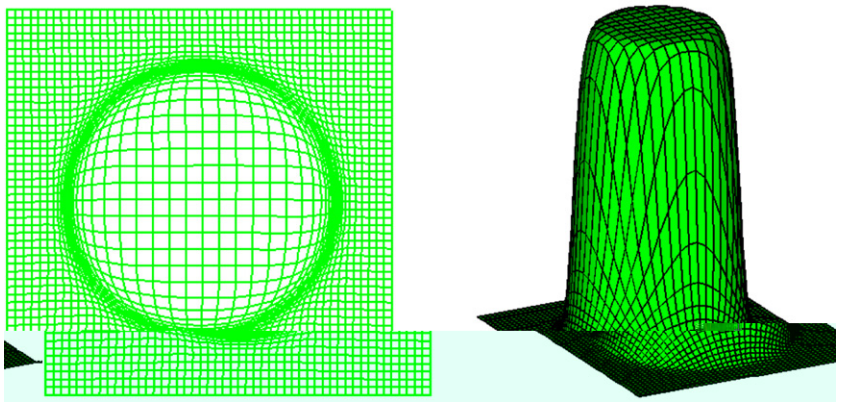

Fig. 11. The table surface adaptive grid shown in parameter space (left) and physical space (right). The target metric consists of the sum of a coarse-grained contribution and a component quadratic in surface gradients.

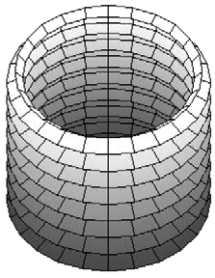

Fig. 12. Corrugated surface.

$$\begin{aligned} x &= (4+\frac{1}{2}\sin 3v)\cos u, \\ y &= (4+\frac{1}{2}\sin 3v)\sin u, \\ z &= v. \end{aligned} \tag{30}$$

Fig. 13 shows a small section of the triangle surface mesh in parameter space, both initially and after adaptation. Lastly, Fig. 14 illustrates the result in physical space. Again, it is difficult to see evidence of adaptation of the initial triangle mesh, but the general improvement in mesh element quality is immediately apparent.

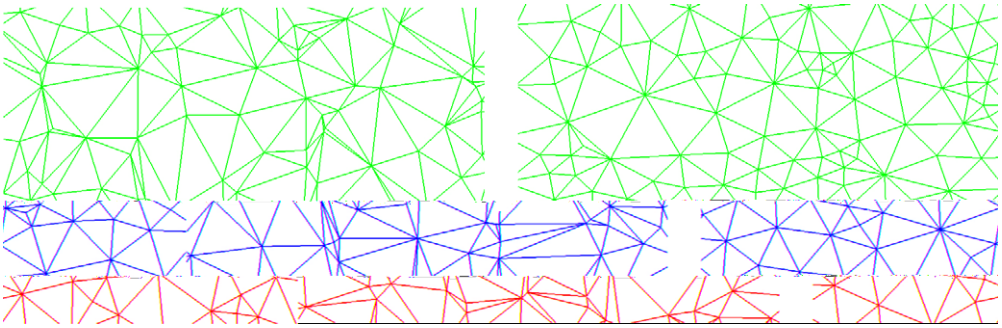

Fig. 13. The corrugated surface shown in parameter space. To the left is the initial, uniform, grid; on the right is the adapted result.

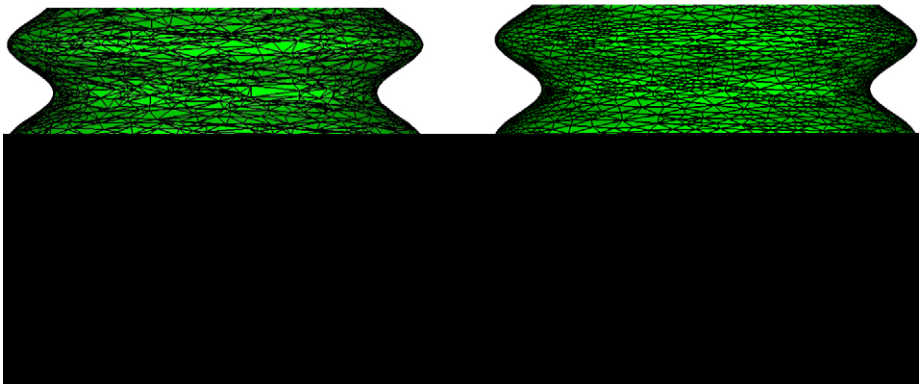

Fig. 14. The corrugated surface shown in physical space, corresponding to the triangular grid. To the left is the initial, uniform, grid; on the right is the adapted result.

As a closing example, consider the corkscrew domain shown in Fig. 15. The parametric equations of the corkscrew surface are

$$\begin{aligned} x &= \cos v \cos u, \\ y &= \cos v \sin u, \\ z &= \sin v + u. \end{aligned} \tag{31}$$

This surface might result by holding the top and bottom points of a sphere, then rotating the top point one full revolution in a counter-clockwise direction with respect to the bottom point. This surface exhibits a helix spiral and cusps formed near the root of the helix.

On applying the surface adaptation method to the corkscrew surface discretized with a structured rectangular mesh, one immediately encounters the problem of severe surface element distortion that originates at the cusps near the root of the helix. This distortion results in actual folding of grid lines near the root as the solution method proceeds toward convergence. The results shown thus far have employed a semiconformal map using a constant dilatation parameter λ (ξ) of unity (see Eq. (10)). The complex corkscrew surface requires a scaling of the dilatation parameter to provide acceptable element quality. For non-unity constant dilatation, the grid equations in two dimensions read

$$\begin{aligned} \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi^\alpha} \left(\sqrt{g} g^{\alpha\beta} \frac{\partial u}{\partial \xi^\beta} \right) &= \lambda^2 F(u(\xi,\eta), v(\xi,\eta)) \\ \frac{1}{\sqrt{g}} \frac{\partial}{\partial \xi^\alpha} \left(\sqrt{g} g^{\alpha\beta} \frac{\partial v}{\partial \xi^\beta} \right) &= \lambda^2 G(u(\xi,\eta), v(\xi,\eta)). \end{aligned} \tag{32}$$

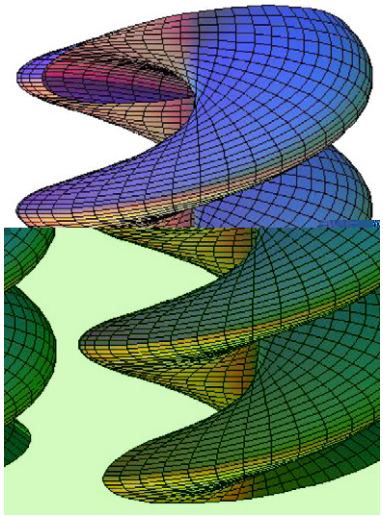

Fig. 15. Corkscrew surface (the twisted sphere).

Table 1
The quality factor (see text) of the surface elements as a function of the semiconformal map dilatation parameter λ

Deviation from orthogonality	Initial mesh	Value of λ^2					
		0.0002	0.0005	0.001	0.002	0.004	0.005
Mean	30.8	28.7	25.3	19.1	16.2	27.4	31.5
Std. dev.	13.4	12.8	12.1	12.4	11.6	15.4	17.3

Table 1 presents data that show the "quality" of the structured surface mesh as a function of dilatation parameter λ, for selected values of λ^2 ranging from 0.0002 to 0.005. The final value of λ should be selected to provide the most acceptable mesh to the application requesting it. This paper adopts a simple quality metric based on the deviation of the intra-element angles from 90°. Mastin [28] discusses a qualitative bound on this orthogonality metric, "Reasonable departure from orthogonality ($\phi \leqslant 45°$) is therefore of little concern when the rate-of-change of grid spacing is reasonable." This study will assume that the optimal mesh exhibits the lowest possible *departure from orthogonality* of the angles internal to the elements.

For a surface mesh, the orthogonality metric extends to the calculation of intra-element angle deviation with respect to the surface itself. For each element in the mesh, the angles between the element edges are calculated using the dot product of the edge vectors. Table 1 presents both the mean deviation of the element angles from orthogonality, and the standard deviation of this departure from orthogonality, for various values of λ^2. Given this metric, the value of $\lambda^2 = 0.002$ results in the best mesh; the mesh exhibiting the least aggregate departure from orthogonality with respect to the surface.

Fig. 16 shows the initial mesh and the adapted structured rectangular mesh in parameter space for this object. Fig. 17 shows this result in physical space. This illustration is also colored based on the orthogonality quality metric; elements in red exhibit the greatest departure from orthogonality (containing at least one angle of roughly 45°, with respect to the surface), where blue elements are nearly orthogonal. The effectiveness of the surface adaptation algorithm is readily apparent, considering both the mean quality factor shown in Table 1 and this figure. Indeed, the figure shows the adapted mesh contains fewer red elements and considerably more blue. Further, note the significant improvement in element quality near the waist of the corkscrew (far from the root).

Fig. 18 displays the initial mesh and an adapted unstructured triangular mesh in parameter space for the corkscrew object. Fig. 19 shows this result in physical space. There is again a visual improvement in the

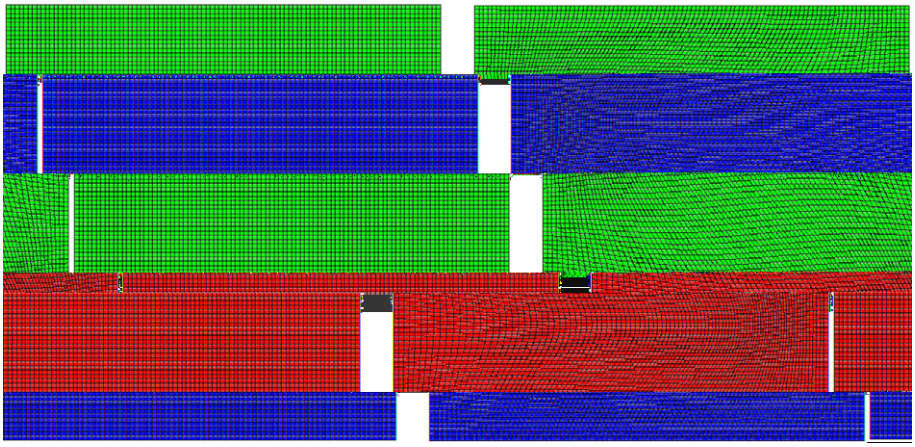

Fig. 16. The corkscrew surface in parameter space showing the Cartesian grid surface discretization and using a dilatation parameter $\lambda^2 = 0.002$. To the left is the initial, uniform, grid; on the right is the adapted result.

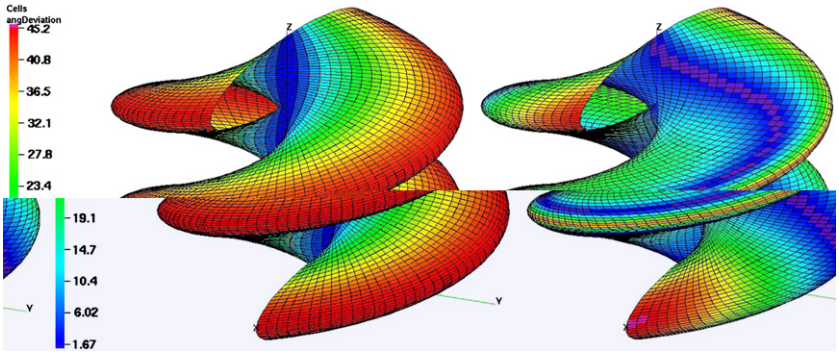

Fig. 17. The corkscrew surface shown in physical space corresponding to the Cartesian grid and a dilatation parameter $\lambda^2 = 0.002$. To the left is the initial, uniform, grid; on the right is the adapted result.

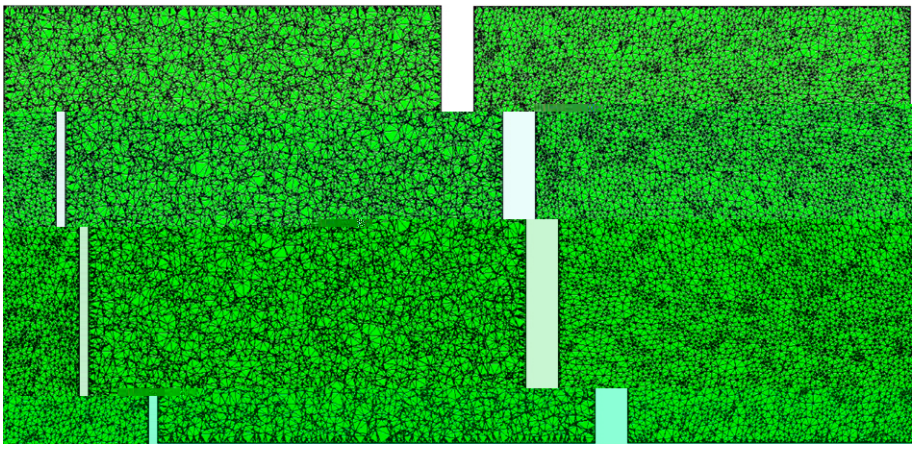

Fig. 18. The corkscrew surface shown in parameter space illustrating the triangle grid. To the left is the initial grid; on the right is the adapted version.

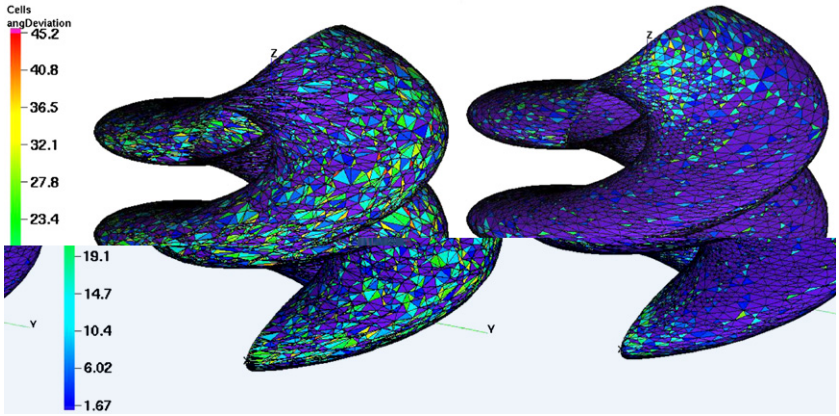

Fig. 19. The corkscrew surface shown in physical space corresponding to the triangle grid. To the left is the initial grid; on the right is the adapted version using a $\lambda^2 = 0.001$. Coloration again indicates surface element quality factor.

Table 2
The quality factor (see text) of the surface triangles for a semiconformal map dilatation parameter $\lambda^2 = 0.001$

Deviation from orthogonality	Initial mesh	Value of λ^2
		0.001
Mean	11.2	3.95
Std. dev.	12.4	8.07

element quality as the mesh is adapted, and the coloration reflects significant improvement in the element quality factor. This result employs a dilatation parameter λ^2 of 0.001.

Table 2 shows quality factor statistics for the triangle surface mesh. In this case, the "optimal" triangle would be an equilateral triangle with respect to the surface. Thus, the mean angle deviation shown in the table is in units of angular variation away from 60° between any two edges of the triangle. The statistics show that the adaptation algorithm provides significant improvement in both element quality factor and consistency for triangle surface meshes, at least for this simple element quality metric.

5. Conclusions

This paper proposes a method for the smoothing and adaptation of surface meshes based on the theory of harmonic morphisms. An elliptic system of partial differential equations is presented that is valid for both structured and unstructured meshes. One of the principal goals for this work was to develop a surface mesh motion algorithm based on an elliptic method; to leverage the robustness, adaptation properties, and effectiveness of this class of methods. This approach is closely related to the method advanced by Khamayseh and Mastin [10]. Indeed, it may be viewed as an extension of the work of Khamayseh and Mastin to a wider class of applications involving general unstructured surface meshes.

To study the effectiveness of the approach, several test problems were used where the behavior of the method could be assessed qualitatively. These problems included the monkey saddle, the table surface, the corrugated surface, and a corkscrew (twisted sphere) surface. Both structured and unstructured meshes were studied on these surfaces. The proposed method behaves similarly to other elliptic methods; it provides adaptation to surface features, and it is effective on unstructured meshes.

Future work in this area should include a quantitative assessment of this and other methods of surface mesh generation. This assessment will first require the formulation of a quantitative metric that includes the relationship between two competing interests, the degree of adaptation desired and the geometric integrity of the elements in the resulting mesh. Such a metric should be a function of the requirements of the ultimate

simulation application and the assumptions within the computational model. In general, there is a compromise between adaptation and element geometric quality; on a given domain, regions of the mesh may exist where additional adaptation decreases the global quality of the resulting mesh. The proposed approach allows the magnitude of adaptation to be controlled by the use of a constant dilatation parameter λ. The problem of how to treat surface singular points is also not addressed here.

This study was limited to surface descriptions similar to Eqs. (15) and (31) (i.e., surfaces given in analytic or parametric form). While this describes a useful class of problems, this implementation should be extended to general, discrete surface descriptions such as triangulated surfaces and to add surface mesh p-refinement. While this is conceptually straightforward, the implementation will require the accurate numerical approximation of the surface Beltrami differentials and the ability to effectively address surface quality issues.

Finally, the Laplace–Beltrami target metric approach employing the coarse-grained metric estimation method is best described as a nonlinear diffusion equation system. The resultant finite-element problem (26) is a nonlinear algebraic system that is challenging to solve [2,3]. While the solution of this system is not competitive on a single processor execution time basis with basic methods like as Laplace smoothing, it should be possible to develop a parallel, scalable multilevel approach to solve this system [29].

Acknowledgements

The submitted manuscript has been authored by a contractor of the U.S. Government under Contract No. W-7405-ENG-36 (LA-UR-04-5239) and No. DE-AC07-05ID14517. Accordingly, the US Government retains a non-exclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes.

References

[1] A. Khamayseh, A. Kuprat, Hybrid curve point distribution algorithms, SIAM J. Sci. Comput. 23 (2002) 1464–1484.
[2] G. Hansen, A. Zardecki, D. Greening, R. Bos, A finite element method for unstructured grid smoothing, J. Comput. Phys. 194 (2004) 611–631.
[3] G. Hansen, A. Zardecki, D. Greening, R. Bos, A finite element method for three-dimensional unstructured grid smoothing, J. Comput. Phys. 202 (2005) 281–297.
[4] V.D. Liseikin, A Computational Differential Geometry Approach to Grid Generation, Springer, Berlin, Heidelberg, New York, 2004.
[5] G.A. Hansen, R.W. Douglass, A. Zardecki, Mesh Enhancement: Selected Elliptic Methods, Foundations and Applications, Imperial College Press, London, 2005.
[6] J. James Eells, J.H. Sampson, Harmonic mappings of Riemannian manifolds, Am. J. Math. 86 (1964) 109–160.
[7] C.W. Mastin, J.F. Thompson, Discrete quasiconformal mappings, Z. Angew. Math. Phys. 29 (1978) 1–11.
[8] C.W. Mastin, J.F. Thompson, Quasiconformal mappings and grid generation, SIAM. J. Sci. Stat. Comput. 5 (2) (1984) 305–310.
[9] S.A. Ivanenko, Harmonic mappings, in: J.F. Thompson, B.K. Soni, N.P. Weatherill (Eds.), Handbook of Grid Generation, CRC Press, Boca Raton, FL, 1999, pp. 8.1–8.43, Chapter 8.
[10] A. Khamayseh, C.W. Mastin, Computational conformal mapping for surface grid generation, J. Comput. Phys. 123 (1996) 394–401.
[11] C.W. Misner, Harmonic mappings as models for physical theories, Phys. Rev. D 18 (1978) 4510–4524.
[12] A.S. Dvinsky, Adaptive grid generation from harmonic maps on Riemannian manifolds, J. Comput. Phys. 95 (1991) 450–476.
[13] J.U. Brackbill, J.S. Saltzman, Adaptive zoning for singular problems in two dimensions, J. Comput. Phys. 46 (1982) 342–368.
[14] J.U. Brackbill, An adaptive grid with directional control, J. Comput. Phys. 108 (1993) 38–50.
[15] D.L. Marcum, A. Gaither, Unstructured surface grid generation using global mapping and physical space approximation, in: Eighth International Meshing Roundtable, Sandia National Laboratory, 1999, pp. 397–406.
[16] U. Tremel, F. Deister, O. Hassan, N.P. Weatherill, Parallel generation of unstructured surface grids, in: 12th International Meshing Roundtable, Sandia National Laboratory, 2003, pp. 43–53.
[17] J. Eells, B. Fuglede, Harmonic Maps Between Riemannian Polyhedra, Cambridge University Press, Cambridge, 2001.
[18] S. Nishikawa, Variational Problems in Geometry, American Mathematical Society, Providence, RI, 2002.
[19] D.M. Deturck, J.L. Kazdan, Some regularity theorems in Riemannian geometry, Ann. Scient. Éc. Norm. Sup. 14 (1981) 249–260.
[20] B. Fuglede, Harmonic morphisms between Riemannian manifolds, Ann. Inst. Fourier (Grenoble) 28 (1978) 107–144.
[21] S.P. Spekreijse, Elliptic grid generation based on Laplace equations and algebraic transformations, J. Comput. Phys. 118 (1995) 38–61.
[22] W. Huang, D.M. Sloan, A simple adaptive grid method in two dimensions, SIAM J. Sci. Comput. 15 (4) (1994) 776–797.
[23] J.A. MacKenzie, The efficient generation of simple two-dimensional adaptive grids, SIAM J. Sci. Comput. 19 (1998) 1340–1366.
[24] H.D. Ceniceros, T.Y. Hou, An efficient dynamically adaptive mesh for potentially singular solutions, J. Comput. Phys. 172 (2001) 609–639.

[25] N. Lautersztajn-S, A. Samuelsson, On application of differential geometry to computational mechanics, Comput. Methods Appl. Eng. 150 (1–4) (1997) 25–38.
[26] E.F. D'Azevedo, On adaptive mesh generation in two dimensions, in: Eighth International Meshing Roundtable, 1999, pp. 109–117.
[27] A. Gray, Modern Differential Geometry of Curves and Surfaces with Mathematica, 2nd ed., CRC Press, Boca Raton, FL, 1998.
[28] C.W. Mastin, Truncation error on structured grids, in: J.F. Thompson, B.K. Soni, N.P. Weatherill (Eds.), Handbook of Grid Generation, CRC Press, Boca Raton, FL, 1999, pp. 32.1–32.10, Ch.32.
[29] M. Berndt, G. Hansen, J.D. Moulton, Efficient nonlinear solvers for Laplace-Beltrami smoothing of three-dimensional unstructured grids, Tech. Rep. LAUR-05-6189, Los Alamos National Laboratory (2005).