Engineering with Computers (2010) 26:281–288
DOI 10.1007/s00366-009-0158-x

ORIGINAL ARTICLE

On discrete boundaries and solution accuracy in anisotropic adaptive meshing

Konstantin Lipnikov · Yuri Vassilevski

Received: 23 November 2005 / Accepted: 11 May 2007 / Published online: 19 December 2009

Abstract In the adaptive mesh generation, the space mesh should be adequate to the surface mesh. When the analytical surface representation is not known, additional surface information may be extracted from triangular surface meshes. We describe a new surface reconstruction method which uses approximate Hessian of a piecewise linear function representing the discrete surface. Efficiency of the proposed method is illustrated with two CFD applications.

Keywords Surface reconstruction · Hessian recovery · Adaptive meshes

1 Introduction

Multi-scale simulations of physical systems are most effective when they are combined with adaptive methods. The adaptive methods reduce greatly the demand for larger number of unknowns and improve accuracy of the simulations via grid adaptation near fine-scale features of a solution. In this paper, we consider a metric based mesh adaptation [1–6] where the metric is induced by the Hessian (matrix of second derivatives) of the mesh solution. The focus of this paper is on the treatment of curved (non-planar) internal and boundary surfaces.

In many applications, exact parametrization of curved surfaces may be unknown and the surfaces are described by triangular meshes, for instance, meshes coming from CAD systems. This reduces performance of adaptive methods due to the limited surface resolution. In order to resolve fine-scale features of the solution, the boundary faces of the adaptive mesh should be close to the unknown analytical surfaces. If analytical surfaces are sufficiently smooth (or piecewise smooth), the triangular meshes carry additional information about these surfaces. In this paper, we present numerical and theoretical analysis of a new surface reconstruction method.

There are many methods for higher order reconstruction of piecewise linear surfaces (see [7–10] and references therein). In [8, 9] the surface is parametrized and the desired surface characteristics are computed from the derivatives of functions specifying the parametrization. In [7, 8], the discrete surface is approximated by a piecewise quadratic surface using the best fit algorithm. The isogeometric analysis in [11] uses non-uniform rational B-splines (NURBS) to represent the input CAD geometry. The new method uses technique of the discrete differential geometry to compute an approximate Hessian of a piecewise quadratic function representing the reconstructed surface. The Hessian is computed in a weak sense by analogy with the finite element methods. The new method is local and can be easily parallelized.

The paper outline is as follows. In Sect. 2, we describe briefly the Hessian based adaptation methodology. In Sect. 3, we describe a new surface reconstruction method. In Sect. 4, we demonstrate efficiency of the new reconstruction method for two problems simulating flows around an obstacle.

K. Lipnikov (✉)
Los Alamos National Laboratory, Los Alamos, USA
e-mail: lipnikov@lanl.gov

Y. Vassilevski
Institute of Numerical Mathematics, Moscow, Russia
e-mail: vasilevs@dodo.inm.ras.ru

2 Hessian-based mesh adaptation

2.1 Quasi-optimal meshes

Let Ω_h be a tetrahedral mesh with $N(\Omega_h)$ elements and u_h be a continuous piecewise linear solution computed at mesh nodes with some numerical method which we denote by $\mathcal{P}_{\Omega_h}$. We shall simply write that $u_h = \mathcal{P}_{\Omega_h} u$ where u is an unknown exact solution. The ideal goal would be to find a mesh which minimizes the maximal norm of the discretization error $\|u - \mathcal{P}_{\Omega_h} u\|_\infty$. Note that this mesh may be anisotropic.

Our analysis is applicable to a more simple problem of minimizing the interpolation error, $\|u - \mathcal{I}_{\Omega_h} u\|_\infty$, where $\mathcal{I}_{\Omega_h}$ is the linear interpolation operator on mesh Ω_h. Our practical experience shows that the mesh minimizing the interpolation error results in a small discretization error for many classes of problems with non-smooth solutions. The optimal mesh is defined as follows:

$$\Omega_h^{\mathrm{opt}} = \arg \min_{N(\Omega_h) < N_{\max}} \|u - \mathcal{I}_{\Omega_h} u\|_\infty \qquad (1)$$

where $N_{\max}$ is the maximal number of mesh elements defined by the user. This problem was analyzed both theoretically and numerically in [1, 6, 12]. In fact, problem (1) was replaced by a simpler problem which provides a constructive way for finding an approximate solution of (1) which we refer to as a *quasi-optimal* mesh. The quasi-optimal meshes provide the same asymptotic error reduction (see formula (4)) as the optimal meshes. It is proved in [1] that the mesh which is quasi-uniform in the metric $|H^h|$ derived from the Hessian H^h of u_h is also quasi-optimal. Recall that a mesh is called quasi-uniform if it is shape-regular and all simplexes have roughly the same size.

The metric $|H^h|$ is designed to be continuous and linear over each tetrahedron. It is built from the spectral decomposition of Hessian H^h (see [1] for more details):

$$H^h = W_h \Lambda_h W_h^T, \quad |H^h| = W_h |\Lambda_h| W_h^T,$$

where $\Lambda_h = \mathrm{diag}\{\lambda_1, \lambda_2, \lambda_3\}$ is the diagonal matrix, and

$$|\Lambda_h| = \mathrm{diag}\{\max\{|\lambda_1|; \varepsilon\}, \max\{|\lambda_2|; \varepsilon\}, \max\{|\lambda_3|; \varepsilon\}\}$$

is the diagonal matrix with $\varepsilon > 0$ being a user defined tolerance. The generation of meshes quasi-uniform in metric $|H^h|$ is based on the notion of a *mesh quality* which is introduced below.

Let G be a metric generated by a symmetric positive definite 3×3 matrix whose entries depend on point $\mathbf{x} \in \Omega$. For an element e in Ω_h, we denote by $|e|_G$ its volume in metric G and by $|\partial e|_G$ the total length of its edges (also in metric G). We define the mesh quality as

$$Q(\Omega_h) = \min_{e \in \Omega_h} Q(e) \qquad (2)$$

where $Q(e)$ is the quality of a single element e,

$$Q(e) = 6\sqrt[4]{2} \frac{|e|_G}{|\partial e|_G^3} F\left(\frac{|\partial e|_G}{6h^*}\right), \quad 0 < Q(e) \le 1. \qquad (3)$$

Here h^* is the mesh size in the G-uniform mesh with $N_{\max}$ elements,

$$h^* = \left(\frac{12}{\sqrt{2}} \frac{|\Omega|_G}{N_{\max}}\right)^{1/3},$$

and $F(t)$ is a continuous smooth function, $0 \le F(t) \le 1$, with the only maximum at point 1, $F(1) = 1$, and such that $F(0) = F(+\infty) = 0$. The last factor in (3) controls the size of the element, whereas the remaining factors control its shape. Note $Q(e) = 1$ if and only if $|\partial e|_G = 6\ h^*$ and $6\sqrt[4]{2}|e|_G = |\partial e|_G^3$. These conditions hold only when element e is the equilateral simplex of size h^* in metric G.

The generation of a G-quasi-uniform mesh is now reduced to generation of the mesh Ω_h with quality $Q(\Omega_h) \sim 1$. Since the mesh quality is as good as the quality of the worst element, its quality can be increased with a *local* mesh modification around this element. The list of such modifications includes alternation of topology with node deletion/insertion, edge/face swapping and node movement (see Fig. 1 for 2D analogs of local mesh modifications and [1] for more details).

The local mesh modifications such as node deletion/insertion and edge/face swapping are well described in the literature while implementation of the node movement requires additional comments. It is driven by minimization of the smooth functional $\mathcal{F} : \mathbb{R}^3 \to \mathbb{R}$, of the node position $\mathbf{x}$, defined as a reciprocal of the mesh quality (2), i.e. $1 \le \mathcal{F}(\mathbf{x}) < \infty$. Some restrictions have to be imposed on mesh modifications to keep the mesh unfolded and to preserve internal and boundary surfaces.

2.2 Adaptive iterative algorithm

We use the following algorithm to build a quasi-optimal mesh:

- Generate any initial tetrahedrization $\Omega_h^{(1)}$ of the computational domain.
- For $k = 1, 2,...,$ repeat

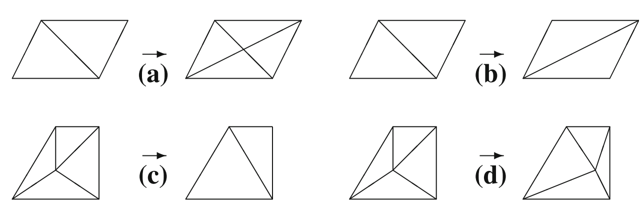

Fig. 1 Local topological operations for triangular meshes: **a** node insertion, **b** edge swapping, **c** node deletion, and **d** node movement

- Compute on $\Omega_h^{(k)}$ the discrete solution u_h and generate the discrete Hessian-based metric $|H^h|$.
- Terminate the adaptive loop if the mesh quality in metric $|H^h|$ is bigger than Q_0 which is the user defined number (e.g., $Q_0 = 0.4$).
- Generate the mesh $\Omega_h^{(k+1)}$ which is quasi-uniform in the metric $|H^h|$ and is such that $Q(\Omega_h^{(k+1)}) > Q_0$. To do this, we use local mesh modifications such as node deletion/insertion, edge/face swapping and node movement (see Fig. 1).

It is proved in [1, 12] that quasi-optimal meshes in polyhedral domains result in the asymptotically optimal estimate:

$$\|u - \mathcal{I}_{\Omega_h} u\|_\infty \sim N(\Omega_h)^{-2/3}. \tag{4}$$

In Sect. 3 we demonstrate numerically that (4) holds in a more general case of curved boundaries. We also show that the optimal estimate is violated when these boundaries are defined by triangular meshes.

The described algorithm consists of three independent steps with different complexity. Complexity of the discrete solution depends greatly on the discretization scheme and the numerical method. The construction of the discrete Hessian is proportional to the number of elements. Complexity of generation of $|H^h|$-quasi-uniform mesh depends on many factors such as quality of the initial mesh and solution anisotropy. In general, it is reduced with each adaptive iteration. Convergence of the adaptive iterations has been studied in [1].

2.3 Treatment of surface constraints

The distinctive geometrical features of any model are internal and boundary surfaces and their intersections (model edges). Let us consider a particular model surface $\Gamma \subset \mathbb{R}^3$ and a model edge $\Theta \subset \mathbb{R}^3$. If the analytical formulas for Γ and Θ were available, they could be used in the adaptive mesh generation method described above. In this article we consider the case when these formulas are not available and Γ and Θ are modeled with faces and edges of the original mesh $\Omega_h^{(1)}$.

Let Γ_h be the triangulated surface of the original mesh $\Omega_h^{(1)}$ approximating Γ with triangular faces Γ_t,

$$\Gamma_h = \bigcup_t \Gamma_t.$$

Similarly, let Θ_h be a poly-line formed by the edges of $\Omega_h^{(1)}$ approximating Θ. In this paper, we fix (freeze) nodes on Γ_h and Θ_h. This imposes simple constraints on the local mesh modifications and leaves enough freedom for realization of mesh modifications with surrounding tetrahedra. Still, the fixed-nodes constraints may result in an unnecessarily fine mesh in regions where solution u_h is smooth. The more adequate treatment of discrete boundaries is described in [13].

In Sect. 4, we shall demonstrate that accuracy of boundary representation makes significant impact on accuracy of the discrete solution. The accuracy may be improved if we assume that the underlying surfaces are sufficiently smooth or piecewise smooth. Then the discrete surface Γ_h carries additional information about Γ. The new surface reconstruction method is described in the next section.

3 Piecewise quadratic extrapolation of piecewise linear surfaces

In this section, we consider again the model surface Γ. To simplify the presentation, we assume that Θ is its boundary. We assume also that nodes of Γ_h and Θ_h belong to Γ and Θ, respectively, although this assumption is not necessary in practice. The piecewise quadratic extrapolation $\tilde{\Gamma}_h$ of Γ_h is defined as the continuous surface being the union of non-overlapping pieces $\tilde{\Gamma}_t$ of local quadratic extrapolations over faces Γ_t,

$$\tilde{\Gamma}_h = \bigcup_t \tilde{\Gamma}_t.$$

The local extrapolation $\tilde{\Gamma}_t$ is described by a quadratic function $\varphi_{2,t}$. Hereafter, we shall omit the superscript t whenever it does not result in confusion. For our purposes, it will be convenient to describe the function φ_2 in a local coordinate system (ξ_1, ξ_2) associated with the plane of Γ_t. In this coordinate system, the 2D multi-point Taylor formula [14] for the quadratic function φ_2 with the Hessian $H^{\varphi_2} = \{H_{pq}^{\varphi_2}\}_{p,q=1}^2$ reads

$$\varphi_2(\xi) = -\frac{1}{2}\sum_{i=1}^{3}(H^{\varphi_2}(\xi - \mathbf{a}_i), (\xi - \mathbf{a}_i))p_i(\xi) \tag{5}$$

where $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ are vertices of the triangle Γ_t and $p_i(\xi)$ is a continuous piecewise linear function such that $p_i(\mathbf{a}_j) = \delta_{ij}$ (the P_1 finite element basis function for point $\mathbf{a}_i$).

In order to recover the Hessian H^{φ_2}, we use another representation for it. Let us assume that numbers $\alpha_i = (H^{\varphi_2}\ell_i, \ell_i)$, $i = 1, 2, 3$, representing the projection of this Hessian on edges ℓ_i of Γ_t, are given. Hereafter, we use ℓ_i for both the mesh edge and the corresponding vector. In the local coordinate system, vectors ℓ_i are described by two coordinates, $\ell_i = (l_1^i, l_2^i)$. We assume that the vector ℓ_i begins at vertex $\mathbf{a}_i$ and ends at vertex $\mathbf{a}_{i+1}$ with $\mathbf{a}_4 = \mathbf{a}_1$. Then, the definition of α_i gives

$$\left(\begin{pmatrix} H_{11}^{\varphi_2} & H_{12}^{\varphi_2} \\ H_{12}^{\varphi_2} & H_{22}^{\varphi_2} \end{pmatrix}\begin{pmatrix} l_1^i \\ l_2^i \end{pmatrix}, \begin{pmatrix} l_1^i \\ l_2^i \end{pmatrix}\right) = \alpha_i$$

which in turn results in the system of three linear equations for the unknown entries of the matrix H^{φ_2} :

$$l_1^i l_1^i H_{11}^{\varphi_2} + l_2^i l_2^i H_{22}^{\varphi_2} + 2 l_1^i l_2^i H_{12}^{\varphi_2} = \alpha_i, \quad i = 1,2,3. \tag{6}$$

Lemma 1 *The matrix of the system (6) is non-singular.*

Proof Let us denote the coefficient matrix of system (6) by B. Note that $\ell_1 + \ell_2 + \ell_3 = 0$. Using this fact in direct calculations of the determinant of matrix B, we get

$$|\det B| = 2\left|l_1^1 l_2^2 - l_1^2 l_2^1\right|^3 = 16|\Gamma_t|^3 > 0 \tag{7}$$

where $|\Gamma_t|$ is the area of the triangle Γ_t. This proves the assertion of the lemma. □

Now, we use results of [1] where the algorithm for computing the discrete Hessian $H^h(\mathbf{a}_i)$ of a continuous piecewise linear function is presented and analyzed. We define α_i as the average of two nodal approximations,

$$\alpha_i = \left((H^h(\mathbf{a}_i)\ell_i, \ell_i) + (H^h(\mathbf{a}_{i+1})\ell_i, \ell_i)\right)/2, \tag{8}$$

associated with the edge ℓ_i. There are two exceptions from this rule. If $\mathbf{a}_i \in \Theta_h$ and $\mathbf{a}_{i+1} \notin \Theta_h$, then α_i is equal to $(H^h(\mathbf{a}_{i+1})\ell_i, \ell_i)$. If $\mathbf{a}_i \in \Theta_h$ and $\mathbf{a}_{i+1} \in \Theta_h$, then $\alpha_i = 0$. This implies that the nodal approximation of the Hessian is not recovered at model edges and therefore the traces of Γ_h and $\tilde{\Gamma}_h$ on Θ_h coincide.

It remains to describe how we recover $H^h(\mathbf{a}_i)$ for every interior node $\mathbf{a}_i$ of Γ_h. We begin by introducing a few additional notations. For each $\mathbf{a}_i$, we define the superelement σ_i as a union of all triangles of Γ_h sharing $\mathbf{a}_i$. Then, we define a plane approximating in the least square sense the nodes of this superelement and associate this plane with a local coordinate system (ξ_1, ξ_2). Let $\hat{\sigma}_i$ be the projection of σ_i onto the $\xi_1\xi_2$-plane. Further, let $\varphi(\xi_1, \xi_2)$ be the continuous function representing locally Γ, and $\varphi_h^i(\xi_1, \xi_2)$ be the continuous piecewise linear function representing σ_i. We assume that both functions are single-valued over $\hat{\sigma}_i$. Finally, we denote the Hessian of φ by H^{φ}.

The components H_{pq}^h, $p, q = 1, 2$, of the discrete Hessian H^h are defined in a weak sense by

$$\int_{\hat{\sigma}_i} H_{pq}^h(\mathbf{a}_i)\psi_h \mathrm{d}S = -\int_{\hat{\sigma}_i} \frac{\partial \varphi_h^i}{\partial \xi_p}\frac{\partial \psi_h}{\partial \xi_q}\mathrm{d}S, \tag{9}$$

where ψ_h is the P_1 finite element basis function for point $\mathbf{a}_i$. Note that the discrete Hessian $H^h(\mathbf{a}_i)$ is a geometric characteristic of the surface Γ at point $\mathbf{a}_i$ (related to its curvature) and therefore is invariant of the position of the projection plane associated with the superelement σ_i.

The proposed reconstruction method is local and therefore it can be easily parallelized. Its computational cost is proportional to the number of surface triangles. It is pertinent to note that this cost is negligent compared to the cost of anisotropic mesh adaptation.

The approximation properties of our extrapolation method were analyzed in [13]. Here, we present the main theoretical result and shift the focus on numerical analysis of the new method. For every triangle Γ_t, we define a superelement σ^t as union of superelements σ_i corresponding to vertices $\mathbf{a}_i$ of Γ_t. Again, we use the local coordinate system (ξ_1, ξ_2) associated with the triangle Γ_t. Let $\hat{\sigma}^t$ (resp., $\hat{\Gamma}_t$) be the projection of σ^t (resp., Γ_t) onto the $\xi_1\xi_2$-plane. We define the constant tensor $H_{\sigma^t}^{\varphi}$ for the superelement $\hat{\sigma}^t$ as

$$\mathbf{x}_{\sigma^t} = \arg\max_{\xi \in \hat{\sigma}^t} |\det H^{\varphi}(\boldsymbol{\xi})|), \quad H_{\sigma^t}^{\varphi} = H^{\varphi}(\mathbf{x}_{\sigma^t}). \tag{10}$$

Theorem 1 [13] *Let edges of a triangle Γ_t be interior edges of Γ_h and $\hat{\sigma}_t$ be a quasi-uniform triangulation with size h. Let $\varphi(\xi_1, \xi_2)$ be a $C^2(\hat{\sigma}^t)$ function representing locally Γ and $\varphi_h = \mathcal{I}_{\hat{\sigma}^t}\varphi$ be a continuous piecewise linear function representing σ^t. Moreover, let H^{φ} and H^h be the differential and discrete Hessians of φ and φ_h, respectively, such that*

$$\left\|H_{pq}^{\varphi} - H_{\sigma^t,pq}^{\varphi}\right\|_{L_\infty(\hat{\sigma}^t)} < \delta, \tag{11}$$

$$\left\|\vec{\nabla}(\varphi - \mathcal{I}_{\hat{\sigma}^t}\varphi)\right\|_{L_2(\hat{\sigma}^t)} < \epsilon. \tag{12}$$

Then, the quadratic function φ_2 describing $\tilde{\Gamma}_t$ and defined by (5), (6), (8) and *(9) satisfies*

$$\|\varphi - \varphi_2\|_{L_\infty(\hat{\Gamma}_t)} \le C\left(\epsilon + \delta h^2\right) \tag{13}$$

where constant C is independent of δ, ε, h and φ.

Generally speaking, the values of ε and δ depend on the smoothness of φ. For sufficiently smooth functions, the right-hand side of (13) is expected to be smaller than $\|\varphi - \varphi_h\|_{L_\infty(\hat{\Gamma}_t)} \sim Ch^2$. Our numerical experiments show that the extrapolation results in more accurate solution of PDEs.

4 Numerical experiments

4.1 Convection–diffusion problem

As the first model problem, we consider the convection–diffusion equation

$$\begin{aligned}
-0.01\Delta u + \mathbf{b}\cdot\nabla u &= 0 \quad \text{in } \Omega \\
u &= g \quad \text{on } \Gamma_{\text{in}} \\
\frac{\partial u}{\partial n} &= 0 \quad \text{on } \Gamma_{\text{out}} \\
u &= 0 \quad \text{on } \partial\Omega \setminus (\Gamma_{\text{in}} \cup \Gamma_{\text{out}}).
\end{aligned} \tag{14}$$

Here $\mathbf{b} = (1,0,0)^T$ is the velocity field, $\Omega = (0,1)^3 \setminus B_{0.5}(0.18)$ is the computational domain with $B_{0.5}(r) = \{\mathbf{x} : \sum_{i=1}^{3}(x_i - 0.5)^2 \le r^2\}$, $\Gamma_{\text{in}} = \{\mathbf{x} \in \partial\Omega : x_1 = 0\}$, $\Gamma_{\text{out}} =$

$\{\mathbf{x} \in \partial\Omega : x_1 = 1\}$, and $g(x_2, x_3) = 16x_2(1-x_2)x_3(1-x_3)$ is the standard Poiseille profile of the entering flow.

The solution u to (14) possesses a boundary layer along the upwind side of the spherical obstacle $B_{0.5}(0.18)$ and is very smooth in the shadow region of this obstacle. Since the exact solution is not known, we replace it with the piecewise linear finite element solution u_* computed on a very fine adaptive (quasi-optimal) mesh containing more than 1.28 million tetrahedra (see Fig. 2, left). To build the adaptive mesh, we used the analytical representation of $\partial\Omega$. The trace of the adapted mesh on the surface of the obstacle shows coarsening in the shadow region and refinement in the upwind region (see Fig. 2, right).

In the first set of experiments (the left picture in Fig. 3), we demonstrate the asymptotic result (4) with u_* instead of u. The L_∞-norm of error fits the analytical curve $60N(\Omega_H)^{-2/3}$.

In the second set of experiments (the right picture in Fig. 3), the boundary $\Gamma = \partial B_{0.5}$ (0.18) is approximated with a quasi-uniform mesh Γ_{h_0}. We measure the L_∞-norm of error as a function of $N(\Omega_h)$ for three different values of h_0. The top and the bottom curves in the right picture in Fig. 3 corresponds to $h_0 = 0.025$ and $h_0 = 0.0125$, respectively. The error is saturated quickly due to the limited boundary resolution. We observe that the saturated error, ε_{h_0}, is almost reciprocal to h_0^2: $\varepsilon_{0.025} = 0.067$ and $\varepsilon_{0.0125} = 0.021$. This is probably related to the second order approximation of the smooth boundary Γ by the piecewise linear surface Γ_{h_0}.

In these experiments, we fixed the nodes on Γ_{h_0}. The fixed-node constraints result in the unnecessarily fine mesh only in the shadow region of the obstacle (see Fig. 4, right picture). Therefore, the mesh is too stretched there in contrast to the case of an analytical representation of the obstacle boundary (see Fig. 4 (left) and Fig. 2 for the mesh trace). This results in mesh elements with a lower quality in the shadow region. However, the excessive refinement and the low quality of these elements do not affect the value of the saturated error. The number of extra elements is small compared to $N(\Omega_h)$ and the solution is very smooth in these elements.

In the third set of experiments, we study the effect of the piecewise quadratic extrapolation $\tilde{\Gamma}_{h_0}$ of Γ_{h_0} on accuracy of the discrete solution. We compare saturation

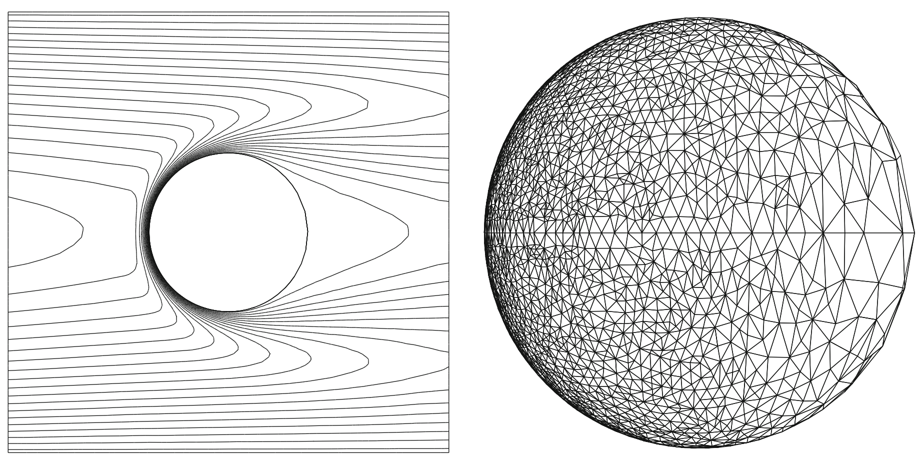

Fig. 2 Isolines of solution u_* in the x_1x_2 plane passing through the center of the obstacle (*left*) and the trace of a typical quasi-optimal mesh at the obstacle with analytical boundary (*right*)

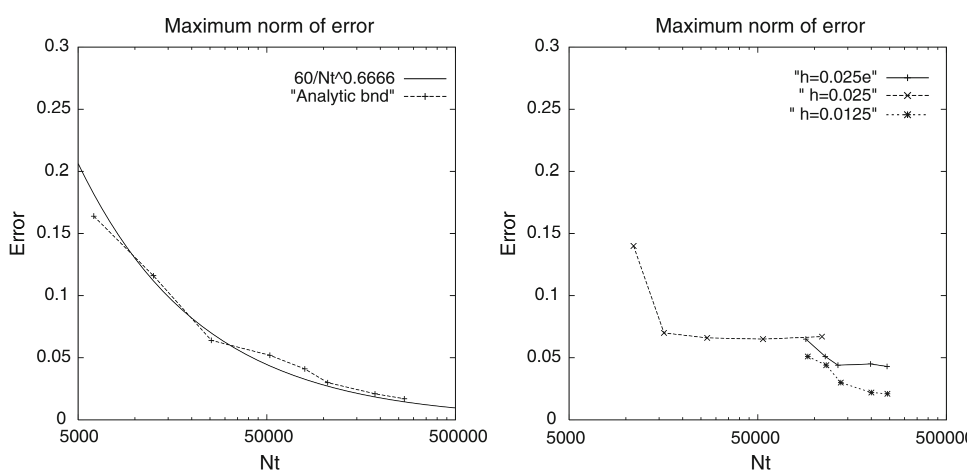

Fig. 3 Convergence analysis: using analytical representation of the obstacle boundary (*left*), using three discrete models $\Gamma_{0.025}, \Gamma_{0.0125}$ and $\Gamma^*_{0.025}$ for $\partial B_{0.5}(0.18)$ (*right*)

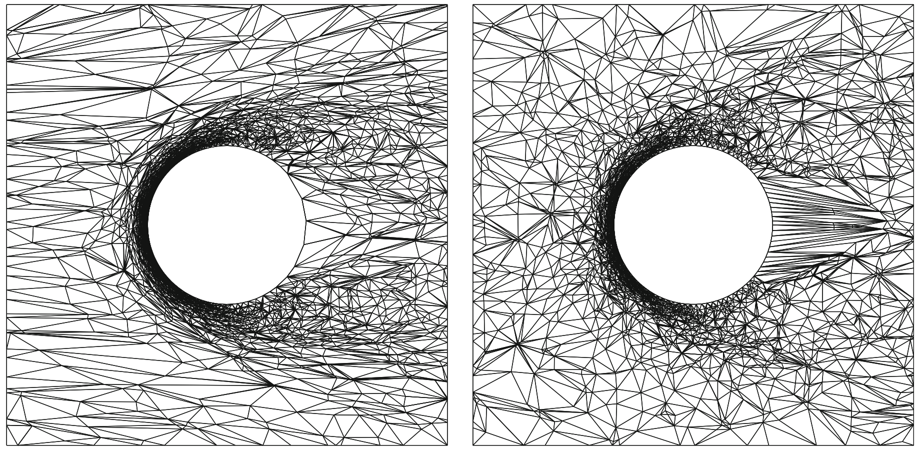

Fig. 4 The mesh cuts in the x_1x_2 plane passing through the center of the obstacle. The *left* picture corresponds to the case of an analytical boundary representation. The *right* picture corresponds to the case of fixed-node constraints. Both meshes have approximately the same number of elements ($\sim 200k$)

errors for three surface meshes: $\Gamma_{0.025}$, $\Gamma_{0.0125}$ and $\Gamma^*_{0.0125}$. The third mesh is obtained from $\Gamma_{0.0125}$ by projecting its mesh nodes onto $\tilde{\Gamma}_{0.025}$. This mesh must provide the saturation error $\varepsilon^*_{h_0}$ which is between the saturation errors on the other two meshes. This is illustrated in the right picture of Fig. 3 where $\varepsilon_{0.0125} = 0.021$, $\varepsilon_{0.025} = 0.067$, and $\varepsilon^*_{0.0125} = 0.043$.

Another approach to building a piecewise linear surface $\Gamma^*_{0.0125}$ is based on the uniform refinement of $\Gamma_{0.025}$ with subsequent projection of new mesh nodes onto $\tilde{\Gamma}_{0.025}$. We use the first approach because it gives the most rigorous comparison of saturation errors on meshes $\Gamma_{0.0125}$ and $\Gamma^*_{0.0125}$.

4.2 Potential transonic flow

As the second model problem, we consider the stationary irrotational adiabatic flow of the ideal gas over an obstacle. The obstacle is the spherical cap which is defined as the intersection of the ball with radius 0.625 centered at point (1, 0.75, −0.375) and half-space $x_3 \geq 0$. The computational domain is the parallelepiped $[0,\ 2] \times [0,\ 1.5] \times [0,\ 1]$ without the spherical cap.

The velocity potential φ satisfies

$$\begin{aligned} \operatorname{div}\left[\left(1 - \frac{|\nabla\varphi|^2}{c}\right)^{\alpha} \nabla\varphi\right] &= 0 \quad \text{in } \Omega, \\ \partial\varphi/\partial n &= \mathbf{u}_\infty \cdot \mathbf{n} \quad \text{on } \Gamma_\infty, \\ \partial\varphi/\partial n &= 0 \quad \text{on } \Gamma_{\text{bot}}. \end{aligned} \tag{15}$$

Here $\mathbf{u}_\infty$ is the flow speed at infinity which is transformed into the Neumann boundary condition for the velocity potential on Γ_∞. The boundary Γ_∞ consists of five planar pieces of $\partial\Omega$ which are entire faces of the parallelepiped, and $\Gamma_{\text{bot}} = \partial\Omega \setminus \bar{\Gamma}_\infty$. Further, $c = \frac{2c_0^2}{\gamma-1}$, c_0 is the speed of sound in the motionless gas, $\alpha = 1/(\gamma - 1)$, and $\gamma = 1.4$.

The inflow is parallel to x_1 axis and its speed is 0.66 Mach. The details of the finite element solution of this problem can be found in [15].

The flow exhibits the transonic regime: a shock is formed on the obstacle surface behind the supersonic zone. Figure 5 shows the obstacle surface, the trace of the adapted mesh on it, the velocity isosurface corresponding to 1.1 Mach, and the velocity isolines at x_1 x_3 symmetry plane of the computational domain. The computations were performed with $N_{\max} =$ 20,000.

The model mesh is quasi-uniform and has 3,814 tetrahedra. Its trace on the obstacle surface consists of 66 triangles. Before adaptation, the model mesh is uniformly refined so that the initial mesh $\Omega_h^{(1)}$ consists of approximately 30 thousands tetrahedra. We have intentionally kept the surface mesh coarse enough to show visual effects of different solution methods.

We run three adaptive simulations. In the first simulation, exact parametrization of the obstacle surface is known so that all boundary nodes belong to the analytical surfaces. In two other simulations, we assume that the parametrization is not known and that the only available surface data are those 66 coarse triangles. In the second simulation, the triangulated surface is frozen and the new boundary nodes are put on this discrete surface. In the third simulation, after the uniform refinement of the model mesh, the new boundary nodes are projected onto the computed piecewise quadratic surface. The refined triangulation is considered to be the new surface mesh and the simulation proceeds as the second one.

The velocity isolines shown in the Fig. 5 indicate that the shock is smeared for the second simulation (middle picture). Moreover, the velocity isosurface is not even simply connected and covers smaller domain than in the other simulations. The solution computed in the third simulation (bottom picture) is closer to that from the first simulation with the analytical surface (top picture).

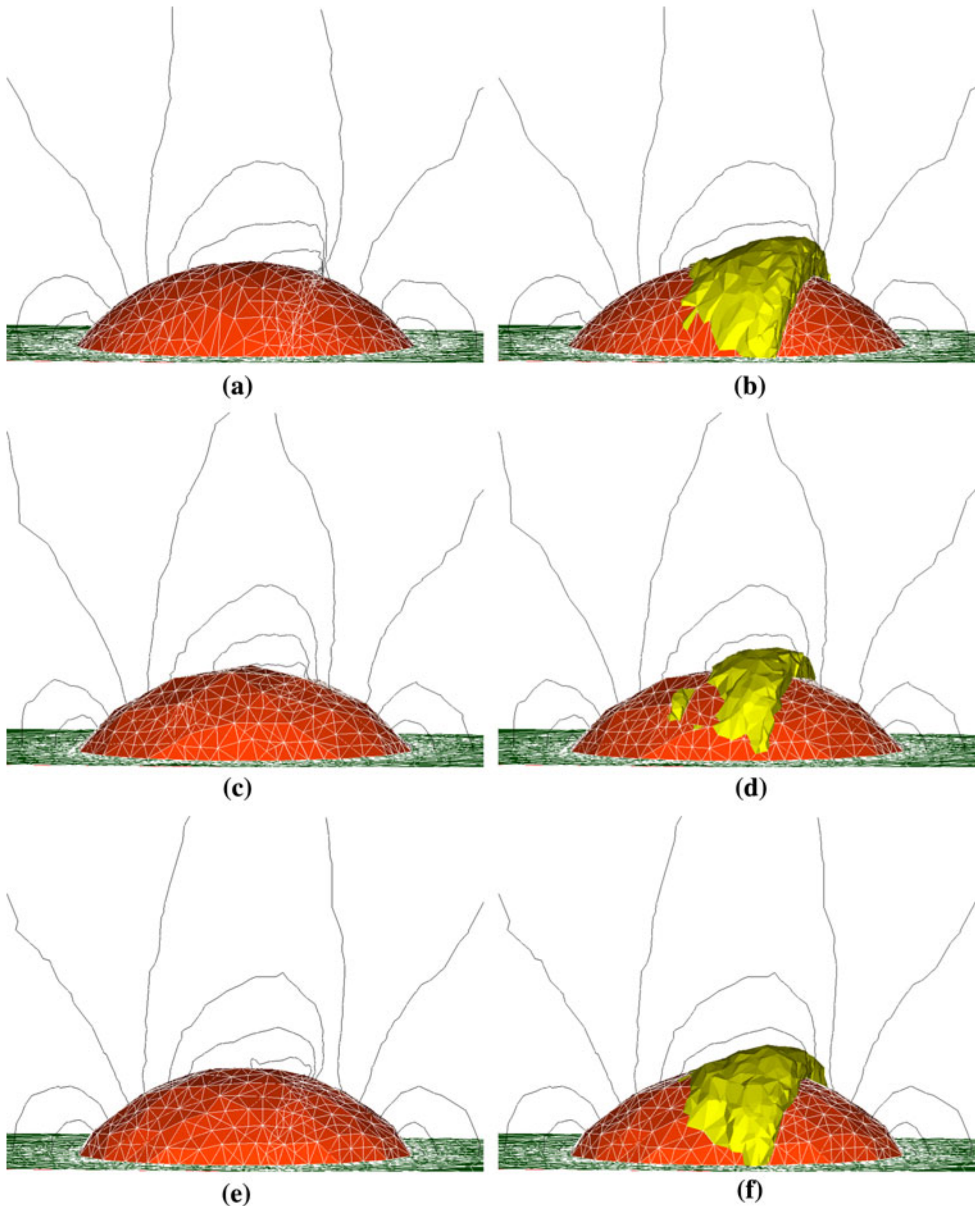

Fig. 5 The obstacle, trace of the adapted mesh, velocity isosurface of 1.1 Mach, and velocity isolines at x_1x_3 symmetry plane of the domain for computations with the analytical obstacle surface (*top*), piecewise linear obstacle surface (*middle*), and reconstructed obstacle surface (*bottom*)

5 Conclusion

We have shown numerically that representation of curved surfaces with triangular meshes restricts the use of adaptive methods. For a particular convection–diffusion problem, the saturated discretization error is proportional to h^2 where h is the size of the quasi-uniform mesh approximating the curved surface. We have proposed and analyzed a new surface reconstruction technique which improves performance of adaptive methods. The new technique has been applied to two CFD applications.

Acknowledgments The work of the second author has been supported by the RFBR grant 04-07-90336 and the academic program "Computational and informatic issues of the solution of large problems". The authors thank Dr. D.Svaytskiy for useful comments regarding the Hessian recovery methods.

References

1. Agouzal A, Lipnikov K, Vassilevski Y (1999) Adaptive generation of quasi-optimal tetrahedral meshes. East-West J Numer Math 7:223–244
2. Buscaglia GC, Dari EA (1997) Anisotropic mesh optimization and its application in adaptivity. Int J Numer Methods Eng 40:4119–4136
3. Dompierre J, Vallet M-G, Fortin M, Habashi WG, Ait-Ali-Yahia D, Tam A (1995) Edge-based mesh adaptation for CFD. CERCA Report R95-73.
4. Borouchaki H, Hecht F, Frey PJ (1998) Mesh gradation control. Int J Numer Methods Eng 43:1143–1165

5. Lipnikov K, Vassilevski Y (2003) Parallel adaptive solution of 3D boundary value problems by Hessian recovery. Comput Methods Appl Mech Eng 192:1495–1513
6. Vassilevski Y, Lipnikov K (2003) Optimal triangulations: existence, approximation and double differentiation of P_1 finite element functions. Comput Math Math Phys 43:827–835
7. Garimella RV, Swartz BK (2003) Curvature estimation for unstructured triangulations of surfaces. Report LA-UR-03-8240, Los Alamos National Laboratory.
8. McIvor AM, Valkenburg RJ (1997) A comparison of local surface geometry estimation methods. Mach Vis Appl 10:17–26
9. Mokhtarian F, Khalili N, Yuen P (2001) Curvature computation of free-form 3-D meshes at multiple scales. Comput Vis Image Understand 83:118–139
10. Petitjean S (2002) A survey of methods for recovering quadrics in triangle meshes. ACM Comput Surv 32:211–262
11. Hughes TJR, Cottrell JA, Bazilevs Y (2005) Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. Comput Methods Appl Mech Eng 194:4135–4185
12. Agouzal A, Vassilevski Y (2005) An unified asymptotical analysis of interpolation errors for optimal meshes. Doklady Academii Nauk 405:1–4
13. Vassilevski Y, Dyadechko V, Lipnikov K (2005) Hessian based anisotropic mesh adaptation in domains with discrete boundaries. Russ J Numer Anal Math Model 20:391–402
14. Ciarlet PG, Wagschal C (1971) Multipoint Taylor formulas and applications to the finite element method. Numer Math 17:84–100
15. Berger H, Warnecke G, Wendland W (1990) Finite elements for transonic potential flows. Numer Methods PDEs 6:17–42