ELSEVIER

Computer Aided Geometric Design 17 (2000) 545–577

COMPUTER AIDED GEOMETRIC DESIGN

www.elsevier.com/locate/comaid

Optimal development of doubly curved surfaces

Guoxin Yu, Nicholas M. Patrikalakis, Takashi Maekawa *

Massachusetts Institute of Technology, Department of Ocean Engineering, Design Laboratory, Cambridge, MA 02139-4307, USA

Received June 1999; revised February 2000

Abstract

This paper presents algorithms for optimal development (flattening) of a smooth continuous curved surface embedded in three-dimensional space into a planar shape. The development process is modeled by in-plane strain (stretching) from the curved surface to its planar development. The distribution of the appropriate minimum strain field is obtained by solving a constrained nonlinear programming problem. Based on the strain distribution and the coefficients of the first fundamental form of the curved surface, another unconstrained nonlinear programming problem is solved to obtain the optimal developed planar shape. The convergence and complexity properties of our algorithms are analyzed theoretically and numerically. Examples show the effectiveness of the algorithms. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Optimal development; Strain field; Nonlinear programming; Line heating; Doubly curved surfaces

1. Introduction

In engineering applications, there exist two kinds of surfaces, developable surfaces and non-developable surfaces, which are also called singly and doubly curved surfaces, respectively. A developable surface has zero Gaussian curvature at all points, while a non-developable surface has non-zero Gaussian curvature at least in some region. A developable surface is highly favorable in metal forming since it can be formed only by bending without tearing or stretching. For this reason, developable surfaces are widely used in manufacturing parts whose materials are not easily amenable to stretching. However, surfaces of many engineering structures are commonly fabricated as doubly curved shapes to fulfill functional requirements such as hydrodynamic, aesthetic, or structural. For

* Corresponding author. E-mail: tmaekawa@deslab.mit.edu.

PII: S0167-8396(00)00017-0

example, a large portion of the shell plates of ship hulls or airplane fuselages are doubly curved surfaces.

Given a three-dimensional design surface, which represents a face of a curved plate or shell, the first step of the fabrication process is flattening or planar development of this surface into a planar shape so that the manufacturer can not only determine the initial shape of the flat plate but also estimate the strain distribution required to form the shape. Then the planar shape is formed into an approximation of the design surface by various approaches such as forming by matching dies, by continuous hammering, or by line heating using an oxyacetylene torch, laser, or heat by induction. This planar shape is usually not unique since, theoretically, a large variety of initial planar shape can be deformed into the curved surface if adequate stretching or shrinkage is allowed. However, in real practice, a planar development corresponding to minimum stretching or shrinkage is highly desirable for the following reasons:

(1) it saves material;
(2) it reduces the work needed to form the planar shape to the doubly curved design surface.

Early surface development procedures were implemented in shipyards based on geodesic development during the last three decades, mainly for ship hull plates whose Gaussian curvature is very small. More recently, Letcher (1993) presents a basic geometric theory for flattening and fabrication of doubly curved plates. The mapping from the curved surface to its planar development is modeled by adding in-plane strains to the curved surface. The strain field is obtained by solving a generalized Poisson’s equation with the source term equal to the Gaussian curvature. However, since the problem is formulated as a boundary value problem, a good solution relies on a well specified boundary condition which is hard to know beforehand. Also, the differential equation is formulated in an orthogonal coordinate system and it is not trivial to formulate in a non-orthogonal coordinate system. Ueda et al. (1994) investigate the relation between the final shape of a plate and the inherent strain. They compute the strain caused by deformation from the initial configuration to the final one using large deformation elastic FEM analysis. Since the initial configuration is usually the projection of the doubly curved surface onto the x–y plane, their approach can only be applied to the cases when the 3D surface is relatively flat, i.e., the curvature is small.

Manning (1980) developed a procedure for surface development based on an isometric tree. A tree of lines with a spine and branches is first drawn on the curved surface. Then the spine and the branch curves are developed isometrically onto planar curves, using the geodesic curvature of the spine and branches on the surface as the curvature of the planar curves. The envelope of the developed pattern forms the planar developed shape. Obviously, the shape of the planar development depends on the choice of the spine and branch curves, since in this development scheme, the stretching along both the spine and branch curves is zero. This procedure is applied in the shoemaking industry and may not be applicable in metal forming. Another disadvantage of this procedure is that it does not provide the field of strain (deformation). Hinds et al. (1991) develop doubly curved surfaces by first approximating them by quadrilateral facets, then flattening these platelets allowing some gaps in the developed patterns. This method is applied in the clothing industry. The disadvantage of this method is that the developed shape depends on the starting edge chosen and again if used in metal forming, it is not guaranteed that the forming process is realizable from the planar shape to the curved surface. Azariadis and Aspragathos (1997) extend the work by Hinds et al. (1991) to reduce the gaps by minimizing the Euclidean distances of pairs of corresponding points between two successive strips. The quality of the development approaches in (Hinds et al., 1991; Azariadis and Aspragathos, 1997) largely depends on the choice of guide-strip or starting edge.

Cho et al. (1998) present an algorithm to approximately develop a doubly curved surface by minimizing the mapping error function for locally isometric mapping between a given and developed surface net. The method has been applied to construct an auxiliary planar domain of triangulation for tessellating trimmed parametric surface patches, which sufficiently preserves the shape of triangles when mapped into three-dimensional space. Again, the applicability of Cho's method (Cho et al., 1998) for metal forming is unclear. The disadvantage of the available literature is that there is no general algorithm for optimal development of general curved surfaces for metal forming process.

In this paper, we develop algorithms for optimal development of a general doubly curved surface in the sense that the strain from the surface to its planar development is minimized. A tensile strain (stretching) from the curved surface to its planar development is assumed which corresponds to forming from the planar shape to curved surface by the line heating approach (Scully, 1987; Yu et al., 1999). When a plate is being formed by line heating, plastic deformation is produced by the thermal stresses generated during the heating and subsequent cooling of the plate. During the heating process, temperature along heating lines increases rapidly, causing the metal at the heated region to expand. In the mean time, the expanded metal is constrained by the surrounding cooler metal, and compressive stresses result. When the heat is removed, the plate cools and the metal contracts, resulting in residual compressive strains inside the plate. The temperature gradient across the thickness of the plate causes the strain gradient across the thickness, which results in the curvature of the plate. The paper is structured as follows: Section 2 reviews differential geometry of surfaces, as well as derives some important theorems on the derivative of the first fundamental form coefficients of the offset surface with respect to the offset distance. Section 3 presents the algorithms for surface development based on the strains along isoparametric lines. Section 4 present the algorithms for surface development based on the strains along principal curvature directions. Section 5 analyzes the complexity and the convergence of the algorithms with respect to the number of grid points. Section 6 illustrates the performance of the surface development algorithms by means of several examples and concluding remarks are provided in Section 7. A more detailed treatment of the results of this paper can be found in a recent thesis by the first author (Yu, 1999).

2. Surface theory

2.1. Background on differential geometry of surfaces

A parametric surface in 3D Euclidean space is defined by (Hoschek and Lasser, 1993)

$$\mathbf{r} = \mathbf{r}(u, v), \tag{1}$$

where the parameters u and v are restricted to some intervals (i.e., $u_1 \leqslant u \leqslant u_2$, $v_1 \leqslant v \leqslant v_2$) leading to parametric surface patches. This rectangular domain D of u, v is called *parametric space* and it is frequently the unit square. If derivatives of the surface are continuous up to the rth order, the surface is said to be of class r, denoted C^r. We assume the surface is smooth enough so that all the (partial) derivatives given in the paper are meaningful.

2.1.1. First and second fundamental forms

Consider a curve C on a surface S defined by $\mathbf{r} = \mathbf{r}(u(t), v(t))$. The arc length $\mathrm{d}s$ of the curve on the surface is given by (do Carmo, 1976)

$$(\mathrm{d}s)^2 = I = \mathrm{d}\mathbf{r}\cdot\mathrm{d}\mathbf{r} = E\,\mathrm{d}u^2 + 2F\,\mathrm{d}u\,\mathrm{d}v + G\,\mathrm{d}v^2, \tag{2}$$

where

$$E = \mathbf{r}_u\cdot\mathbf{r}_u, \qquad F = \mathbf{r}_u\cdot\mathbf{r}_v, \qquad G = \mathbf{r}_v\cdot\mathbf{r}_v \tag{3}$$

and subscripts u, v denote partial derivatives. I is called the first fundamental form, and E, F, G are called coefficients of the first fundamental form.

In order to quantify the curvatures of a surface S, we consider a curve C on S which passes through a point P as shown in Fig. 1. $\mathbf{t}$ is the unit tangent vector; $\mathbf{n}$ is the unit normal vector and $\mathbf{k}$ is the curvature vector of the curve C at point P.

$$\mathbf{k} = \frac{\mathrm{d}\mathbf{t}}{\mathrm{d}s} = \kappa\mathbf{n} = \mathbf{k}_n + \mathbf{k}_g. \tag{4}$$

The component along the unit surface normal $\mathbf{N} = \frac{\mathbf{r}_u\times\mathbf{r}_v}{|\mathbf{r}_u\times\mathbf{r}_v|}$ is the normal curvature vector $\mathbf{k}_n$ expressed as

$$\mathbf{k}_n = \kappa_n\mathbf{N}, \tag{5}$$

where κ_n is called the normal curvature of the surface at P in the direction $\mathbf{t}$.

The second fundamental form is given by

$$II = -\mathrm{d}\mathbf{r}\cdot\mathrm{d}\mathbf{N} = L\,\mathrm{d}u^2 + 2M\,\mathrm{d}u\,\mathrm{d}v + N\,\mathrm{d}v^2, \tag{6}$$

where

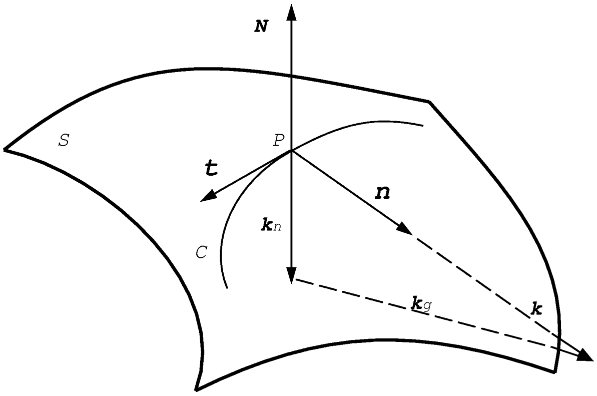


Fig. 1. Definition of normal curvature.

$$L = \mathbf{N} \cdot \mathbf{r}_{uu} = -\mathbf{r}_u \cdot \mathbf{N}_u, \qquad M = \mathbf{N} \cdot \mathbf{r}_{uv} = -\mathbf{r}_u \cdot \mathbf{N}_v = -\mathbf{r}_v \cdot \mathbf{N}_u,$$
$$N = \mathbf{N} \cdot \mathbf{r}_{vv} = -\mathbf{r}_v \cdot \mathbf{N}_v \tag{7}$$

are coefficients of the second fundamental form. The normal curvature can be expressed by

$$\kappa_n = \frac{II}{I} = \frac{L + 2M\lambda + N\lambda^2}{E + 2F\lambda + G\lambda^2}, \tag{8}$$

where $\lambda = \mathrm{d}v/\mathrm{d}u$.

2.1.2. Gauss curvature

The extreme values of κ_n can be obtained by evaluating $\mathrm{d}\kappa_n/\mathrm{d}\lambda = 0$ of Eq. (8), which gives after several algebraic manipulations:

$$\kappa_n^2 - 2H\kappa_n + K = 0. \tag{9}$$

The values K and H are called Gauss (Gaussian) and mean curvature respectively. They are functions of the coefficients of the first and second fundamental forms as follows:

$$K = \frac{LN - M^2}{EG - F^2}, \tag{10}$$
$$H = \frac{EN + GL - 2FM}{2(EG - F^2)}. \tag{11}$$

Alternatively, the Gaussian curvature K can be expressed as a function of E, F, G and their derivatives (Struik, 1950).

$$\begin{aligned} 4\bigl(EG - F^2\bigr)^2 K = {} & E\bigl(E_v G_v - 2F_u G_v + G_u^2\bigr) \\ & + F(E_u G_v - E_v G_u - 2E_v F_v + 4E_u F_v - 2F_u G_u) \\ & + G\bigl(E_u G_u - 2E_u F_v + E_v^2\bigr) \\ & - 2\bigl(EG - F^2\bigr)(E_{vv} - 2F_{uv} + G_{uu}). \end{aligned} \tag{12}$$

2.2. Theorems on the gradients of the first fundamental form coefficients

In this section, some theorems are presented on the gradients of the first fundamental form coefficients of the offset surface along the offset distance direction, which correspond to the gradients of those coefficients across the thickness for a curved shell plate. These theorems show that the gradients of the first fundamental form coefficients of the offset surface provide the mechanism of surface curvature. In metal forming, this means that the non-uniformity of the tensile or compressive strains across the thickness generates the gradients of the first fundamental form coefficients of the offset surfaces across plate thickness, which in turn generates curvature of the formed plate. To our knowledge, these results are new in the CAGD area.

For a curved shell plate with thickness h, we consider $\mathbf{r}(u, v)$ as the mid-surface if its offset surfaces with signed distances $h/2$ and $-h/2$ are the upper and lower surfaces.

Theorem 2.1. *The coefficients of the second fundamental form of a parametric surface can be expressed by the derivatives of the coefficients of the first fundamental form of its offset surface with respect to the offset distance d, evaluated at $d = 0$.*

Proof. Let a progenitor parametric surface (called mid-surface) be defined by (1) and the coefficients of its first fundamental form be given in Eq. (3). Then the offset surface with signed distance d along the normal from the mid-surface is:

$$\hat{\mathbf{r}}(u,v) = \mathbf{r}(u,v) + d\mathbf{N}(u,v), \tag{13}$$

where $\mathbf{N}$ is the unit normal vector of the surface $\mathbf{r}(u,v)$ at (u,v). The first fundamental form coefficients of the offset surface are functions of u, v, d:

$$\hat{E} = \hat{\mathbf{r}}_u \cdot \hat{\mathbf{r}}_u, \qquad \hat{F} = \hat{\mathbf{r}}_u \cdot \hat{\mathbf{r}}_v, \qquad \hat{G} = \hat{\mathbf{r}}_v \cdot \hat{\mathbf{r}}_v. \tag{14}$$

Their derivatives with respect to d, evaluated at mid-surface, i.e., for $d = 0$ are:

$$\left.\frac{\partial \hat{E}}{\partial d}\right|_{d=0} = 2\mathbf{r}_u \cdot \mathbf{N}_u = -2L, \tag{15}$$

$$\left.\frac{\partial \hat{F}}{\partial d}\right|_{d=0} = \mathbf{r}_u \cdot \mathbf{N}_v + \mathbf{r}_v \cdot \mathbf{N}_u = -2M, \tag{16}$$

$$\left.\frac{\partial \hat{G}}{\partial d}\right|_{d=0} = 2\mathbf{r}_v \cdot \mathbf{N}_v = -2N. \tag{17}$$

Therefore, the second fundamental form coefficients L, M, N of the mid-surface can be expressed by the derivatives of $\hat{E}$, $\hat{F}$, $\hat{G}$, evaluated at the mid-surface. □

A similar result can be derived for the metrics along principal curvature directions.

Corollary 2.1. *Let the parameters along maximum principal curvature and minimum principal curvature directions be s and t. Then*

$$\left.\frac{\partial(\hat{\mathbf{r}}_s \cdot \hat{\mathbf{r}}_s)}{\partial d}\right|_{d=0} = -2k_{\max}\big(\hat{\mathbf{r}}_s \cdot \hat{\mathbf{r}}_s\big)|_{d=0}, \tag{18}$$

$$\left.\frac{\partial(\hat{\mathbf{r}}_t \cdot \hat{\mathbf{r}}_t)}{\partial d}\right|_{d=0} = -2k_{\min}\big(\hat{\mathbf{r}}_t \cdot \hat{\mathbf{r}}_t\big)|_{d=0}. \tag{19}$$

Proof. For the offset at distance d along the normal from the mid-surface, as defined in Eq. (13),

$$\begin{aligned}\hat{\mathbf{r}}_s \cdot \hat{\mathbf{r}}_s &= \big(\hat{\mathbf{r}}_u u_s + \hat{\mathbf{r}}_v v_s\big) \cdot \big(\hat{\mathbf{r}}_u u_s + \hat{\mathbf{r}}_v v_s\big) \\ &= \hat{E}u_s^2 + 2\hat{F}u_s v_s + \hat{G}v_s^2.\end{aligned} \tag{20}$$

Therefore, after taking partial derivatives of Eq. (20) with respect to d, and using Eqs. (15)–(17), we obtain

$$\left.\frac{\partial(\hat{\mathbf{r}}_s \cdot \hat{\mathbf{r}}_s)}{\partial d}\right|_{d=0} = -2\big(Lu_s^2 + 2Mu_s v_s + Nv_s^2\big). \tag{21}$$

From Eq. (8),

$$\kappa_{\max} = \frac{L + 2M\lambda + N\lambda^2}{E + 2F\lambda + G\lambda^2} = \frac{Lu_s^2 + 2Mu_s v_s + Nv_s^2}{Eu_s^2 + 2Fu_s v_s + Gv_s^2}, \tag{22}$$

which results in

$$Lu_s^2 + 2Mu_s v_s + Nv_s^2 = k_{\max}\left(Eu_s^2 + 2Fu_s v_s + Gv_s^2\right) = k_{\max}\left(\hat{\mathbf{r}}_s \cdot \hat{\mathbf{r}}_s\right)\big|_{d=0}. \tag{23}$$

Thus by substituting Eq. (23) into Eq. (21), we obtain Eq. (18). Similarly, along the minimum principal curvature direction, where the parameter is t, we obtain Eq. (19). □

Eqs. (15)–(17), (18) and (19) play an important role in surface development algorithms, since in engineering applications, curved plates have finite thickness, no matter how thin they are.

3. Surface development along isoparametric directions

In this section, algorithms for surface development along isoparametric directions are presented. The process of surface development is expressed by tensile strains along u, v isoparametric directions. This corresponds to forming a plate into a curved surface only by shrinkage which can be realized by line heating process (Scully, 1987; Yu et al., 1999).

3.1. Determination of strain field

3.1.1. Formulation

We assume that the surface is defined by a parametric vector equation of the form (1). The surface and its planar development are shown in Fig. 2. The coefficients of the first fundamental form of the curved surface are given by Eq. (3).

Assume that during metal forming by line heating, the normal strain along u line is $\epsilon^u(u,v) \leqslant 0$, and the normal strain along v line is $\epsilon^v(u,v) \leqslant 0$. On the contrary, the strains

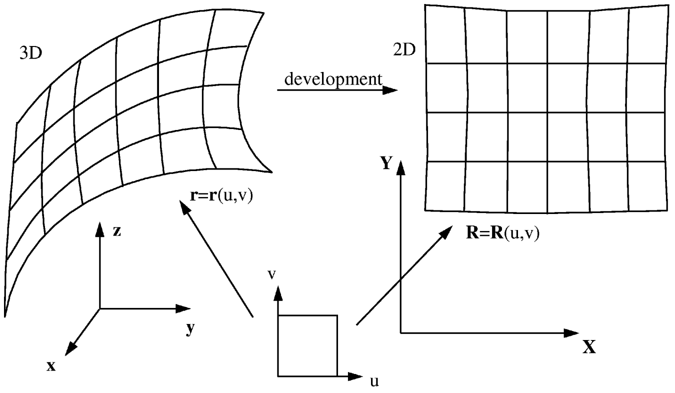


Fig. 2. Curved surface and its planar development.

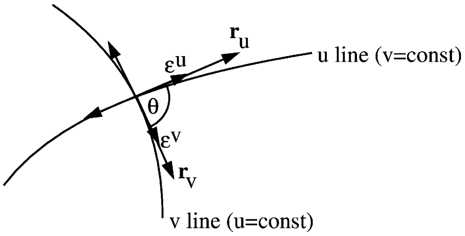


Fig. 3. Strain distribution produced during surface development.

due to development from curved surface to its planar development are $\varepsilon^u(u,v) \geqslant 0$ and $\varepsilon^v(u,v) \geqslant 0$, as shown in Fig. 3. Normal strains are a non-dimensional quantity defined by the ratio of extension or shrinkage of a fiber and its original length. After surface development an infinitesimal length $|\mathbf{r}_u\,\mathrm{d}u|$ changes to $(1+\varepsilon^u)|\mathbf{r}_u\,\mathrm{d}u|$, and an infinitesimal length $|\mathbf{r}_v\,\mathrm{d}v|$ changes to $(1+\varepsilon^v)|\mathbf{r}_v\,\mathrm{d}v|$, according to the definition of strain. Thus we have

$$|\mathbf{R}_u| = \bigl(1+\varepsilon^u\bigr)|\mathbf{r}_u|, \qquad |\mathbf{R}_v| = \bigl(1+\varepsilon^v\bigr)|\mathbf{r}_v|, \tag{24}$$

where $\mathbf{R}(u,v)$ is the planar development (see Fig. 2). The first fundamental form coefficients of the developed surface $\mathbf{R}(u,v)$ are given by

$$e = \mathbf{R}_u \cdot \mathbf{R}_u, \qquad f = \mathbf{R}_u \cdot \mathbf{R}_v, \qquad g = \mathbf{R}_v \cdot \mathbf{R}_v. \tag{25}$$

After substituting Eq. (24) and the relations

$$\mathbf{R}_u \cdot \mathbf{R}_u = |\mathbf{R}_u|^2, \qquad \mathbf{R}_v \cdot \mathbf{R}_v = |\mathbf{R}_v|^2$$

into Eq. (25), the coefficients of the first fundamental form of the planar developed surface are

$$e = \bigl(1+\varepsilon^u\bigr)^2 E, \qquad f = \bigl(1+\varepsilon^u\bigr)\bigl(1+\varepsilon^v\bigr)F, \qquad g = \bigl(1+\varepsilon^v\bigr)^2 G. \tag{26}$$

Here, in computing f, we assume the angle between $\mathbf{r}_u$ and $\mathbf{r}_v$ does not change after surface development. This is equivalent to ignoring the effect of shear strain.

We then minimize the strains $\varepsilon^u(u,v)$ and $\varepsilon^v(u,v)$ which satisfy the condition that after adding these strains to the doubly curved surface, it maps to a planar shape on which Gaussian curvature is zero. This minimization is done in an integral sense using the squares of the strains. Using Eq. (12), this results into

$$\begin{aligned} &\min \iint_D \bigl\{\bigl(\varepsilon^u\bigr)^2 + \bigl(\varepsilon^v\bigr)^2\bigr\}|\mathbf{r}_u \times \mathbf{r}_v|\,\mathrm{d}u\,\mathrm{d}v \\ &\quad = \min \iint_D \bigl\{\bigl(\varepsilon^u\bigr)^2 + \bigl(\varepsilon^v\bigr)^2\bigr\}\sqrt{EG-F^2}\,\mathrm{d}u\,\mathrm{d}v, \end{aligned} \tag{27}$$

such that

$$\begin{aligned} 0 = \bigl\{&e\bigl(e_v g_v - 2f_u g_v + g_u^2\bigr) + f(e_u g_v - e_v g_u - 2e_v f_v + 4e_u f_v - 2f_u g_u) \\ &+ g\bigl(e_u g_u - 2e_u f_v + e_v^2\bigr) \end{aligned}$$

$$-2\bigl(eg-f^2\bigr)(e_{vv}-2f_{uv}+g_{uu})\bigr\}/4\bigl(eg-f^2\bigr)^2, \qquad (28)$$
$$\varepsilon^u(u,v)\geqslant 0;\ \varepsilon^v(u,v)\geqslant 0;\ (u,v)\in D,$$

where D denotes the parametric domain. Since the problem is solved numerically with a solution satisfying the constraint at a given tolerance, we keep the denominator in Eq. (28) (i.e., $4(eg-f^2)^2$) so that the tolerance of constraint has explicit meaning of Gaussian curvature. It can be shown that minimizing the strains $\varepsilon^u(u,v)$ and $\varepsilon^v(u,v)$ is equivalent to minimizing the magnitude of the strains $\epsilon^u(u,v)$ and $\epsilon^v(u,v)$. We choose to work with $\varepsilon^u(u,v)$ and $\varepsilon^v(u,v)$ since we are starting from the curved design surface.

Alternatively, we can also use $(\varepsilon^u+\varepsilon^v)$ instead of $\{(\varepsilon^u)^2+(\varepsilon^v)^2\}$ in the above integral objective function. In this case, the objective function represents the area difference between the doubly curved surface and the planar development to the first order. We use the quadratic objective function $\{(\varepsilon^u)^2+(\varepsilon^v)^2\}$ here instead of the linear one to make the solution easier. After substituting Eqs. (26) into the above formulae, we obtain an optimization problem with respect to $\varepsilon^u(u,v)$ and $\varepsilon^v(u,v)$.

This constrained minimization problem is discretized by using the finite difference method and trapezoidal rule of integration. A grid of $N_g^u\times N_g^v$ points in the parametric domain are used in the discretization. Therefore, the total number of variables is $2N_g^uN_g^v$. To guarantee the independence of each constraint, constraints are imposed at the internal points of the grid, so there are $(N_g^u-2)\times(N_g^v-2)$ constraints.

After discretization, the objective function becomes

$$\sum_{i=1}^{N_g^u}\sum_{j=1}^{N_g^v}\alpha_{ij}\bigl(\bigl(\varepsilon_{ij}^u\bigr)^2+\bigl(\varepsilon_{ij}^v\bigr)^2\bigr)\sqrt{E_{ij}G_{i,j}-F_{ij}^2}\,\Delta u\,\Delta v, \qquad (29)$$

where following the trapezoid rule of integration (Dahlquist and Björck, 1974)

$$\begin{cases}\alpha_{ij}=1 & \text{when } 1<i<N_g^u;\ 1<j<N_g^v,\\ \alpha_{ij}=0.5 & \text{when } 1<i<N_g^u;\ j=1 \text{ or } j=N_g^v,\\ \alpha_{ij}=0.5 & \text{when } i=1 \text{ or } i=N_g^u;\ 1<j<N_g^v,\\ \alpha_{ij}=0.25 & \text{when } i=j=0 \text{ or } i=N_g^u,\ j=N_g^v,\\ \alpha_{ij}=0.25 & \text{when } i=N_g^u,\ j=0 \text{ or } i=0,\ j=N_g^v.\end{cases} \qquad (30)$$

As $\Delta u,\Delta v\to 0$, the error between the objective functions in (27) and (29) due to numerical integration is $\mathrm{O}((\Delta u)^2, (\Delta v)^2)$ as is well known (Dahlquist and Björck, 1974).

We use second order central difference method to approximate all the derivatives in Eq. (28) at internal points of the grid. As $\Delta u,\Delta v\to 0$, the resulting errors are of the order $(\Delta u)^2$ or $(\Delta v)^2$, or $\Delta u\cdot\Delta v$ (Dahlquist and Björck, 1974).

After discretization, we obtain a nonlinear optimization problem with a convex cost function and nonlinear polynomial constraints. This nonlinear programming problem is solved by using the Fortran NAG routine E04VDF (1990), which is designed to solve the nonlinear programming problem—the minimization of a smooth nonlinear function subject to a set of constraints on the variables. Although theoretically, it may be possible to have multiple solutions for the optimization problem (27)–(28), it is very rare in practice to have multiple solutions given the physical nature of the problem.

3.1.2. Solution method

The minimization problem in Section 3.1.1 is a special case of the following nonlinear programming problem (the symbols in this section follow the general practice in the optimization literature, especially, vectors are expressed as a normal letter instead of a bold one):

$$\begin{aligned} &\text{minimize} \quad \psi(q) \\ &\text{subject to} \quad l \leqslant \begin{bmatrix} q \\ A_L q \\ c(q) \end{bmatrix} \leqslant u, \end{aligned} \tag{31}$$

where $\psi(q)$ is a smooth nonlinear function, A_L is a constant matrix, and $c(q)$ is a vector of smooth nonlinear constraint functions. This form allows full generality in specifying other types of constraints. In particular, the i(th) constraint may be defined as an equality by setting $l_i = u_i$. If certain bounds are not present, the associated elements of l or u can be set to special values that will be treated as $-\infty$ or $+\infty$. In our problem, q is the vector of strains $q = (\varepsilon_{00}^u, \varepsilon_{00}^v, \varepsilon_{01}^u, \varepsilon_{01}^v, \ldots, \varepsilon_{10}^u, \varepsilon_{10}^v, \ldots)^{\mathrm{T}}$; $\psi(q)$ is the objective function after discretization and is expressed in (29); the matrix A_L is empty, and $c(q)$ is a vector of $(N_g^u - 2)(N_g^v - 2)$ nonlinear constraints coming from discretization of Eq. (28) at $(N_g^u - 2) \times (N_g^v - 2)$ internal grid points. The vector l is a zero vector, and vector $u = (+\infty, +\infty, \ldots, +\infty, 0, 0, \ldots, 0)^{\mathrm{T}}$, with $2N_g^u N_g^v$ infinities and $(N_g^u - 2)(N_g^v - 2)$ zeros.

E04VDF is an implementation of a sequential quadratic programming (SQP) method (Gill et al., 1981; Bertsekas, 1995). Let q_0 denote the initial estimate of the solution. During the k(th) “major iteration” of E04VDF ($k = 0, 1, \ldots$), a new estimate is defined by

$$q_{k+1} = q_k + \alpha_k p_k,$$

where the vector p_k is the solution of a QP subproblem, to be described below. The positive scalar α_k is chosen to produce a sufficient decrease in an augmented Lagrange function; the procedure that determines α_k is the line search method.

The QP subproblem that defines p_k is of the form

$$\begin{aligned} &\text{minimize} \quad g^{\mathrm{T}} p + \frac{1}{2} p^{\mathrm{T}} H p \\ &\text{subject to} \quad \bar{l} \leqslant \begin{bmatrix} p \\ A_Q p \end{bmatrix} \leqslant \bar{u}, \end{aligned} \tag{32}$$

where the vector g is the gradient of ψ at q_k; the matrix H is a positive definite quasi-Newton approximation to the Hessian of an augmented Lagrangian function.

Let m_L denote the number of linear constraints (the number of rows in A_L), and let m_N denote the number of nonlinear constraints (the dimension of $c(q)$). The matrix A_Q in (32) has $m_L + m_N$ rows, and is defined as

$$A_Q = \begin{bmatrix} A_L \\ A_N \end{bmatrix}$$

where A_N is the Jacobian matrix of $c(q)$ evaluated at q_k. Let l in (31) be partitioned into three sections: the first n component (denoted by l_B), corresponding to the bound constraints; the next m_L components (denoted by l_L), corresponding to the linear constraints; and the last m_N components (denoted by l_N), corresponding to the nonlinear constraints. The vector $\bar{l}$ in (32) is partitioned in the same way, and is defined as

$$\bar{l}_B = l_B - q_k, \quad \bar{l}_L = l_L - A_L q_k \quad \text{and} \quad \bar{l}_N = l_N - c_k,$$

where c_k is $c(q)$ evaluated at q_k. The vector $\bar{u}$ is defined in an analogous fashion.

In general, solving the QP subproblem for p_k is itself an iterative procedure. Hence a “minor iteration” of E04VDF corresponds to an iteration within the QP algorithm. In our implementation, the starting point of the minimization is that all the strains are chosen to be zero.

3.1.3. Strain gradients

After solving for the strain distribution at the mid-surface, we can determine the ideal gradient of the strains along the normal of the mid-surface. As mentioned in Sections 1 and 2, in metal forming, the non-uniformity of the tensile or compressive strains across the thickness generates the gradients of the first fundamental form coefficients of the offset surfaces across plate thickness, which in turn generates curvature of the formed plate. Based on Eq. (26), and

$$F = \mathbf{r}_u \cdot \mathbf{r}_v = |\mathbf{r}_u||\mathbf{r}_v| \cos\theta = \sqrt{E}\sqrt{G}\cos\theta, \tag{33}$$

where θ is the angle between $\mathbf{r}_u$ and $\mathbf{r}_v$, the coefficients of the first fundamental form of the planar developed shape of the offset surface at distance d from the mid-surface are:

$$\begin{aligned} &\hat{e} = \left(1+\hat{\varepsilon}^u\right)^2 \hat{E}, \qquad \hat{f} = \left(1+\hat{\varepsilon}^u\right)\left(1+\hat{\varepsilon}^v\right)\sqrt{\hat{E}\hat{G}}\cos\left(\hat{\theta}+\Delta\theta\right), \\ &\hat{g} = \left(1+\hat{\varepsilon}^v\right)^2 \hat{G}, \end{aligned} \tag{34}$$

where $\hat{E}$, $\hat{F}$, $\hat{G}$ are the coefficients of the first fundamental form of the offset surface; $\hat{e}$, $\hat{f}$, $\hat{g}$ are the coefficients of the first fundamental form of the planar developed shape of the offset surface; $\hat{\theta}$ is the angle between isoparametric lines $u = \text{const}$ and $v = \text{const}$ on the offset surface; $\Delta\theta$ is the change of this angle after development. According to the assumption in this section, $\Delta\theta = 0$ at $d = 0$. Ideally, after development, the 2D developed shape of the offset surface is the same for any given offset distance d, thus we have

$$\frac{\partial \hat{e}}{\partial d} = 0, \tag{35}$$

$$\frac{\partial \hat{f}}{\partial d} = 0, \tag{36}$$

$$\frac{\partial \hat{g}}{\partial d} = 0. \tag{37}$$

After substituting $\hat{e}$ in Eq. (34) into Eq. (35), and after some manipulation by using Eq. (15), we have

$$\left.\frac{\partial[\ln(1+\varepsilon^u)]}{\partial d}\right|_{d=0} = \frac{L}{E}. \tag{38}$$

Similarly, Eq. (37) leads to

$$\left.\frac{\partial[\ln(1+\varepsilon^v)]}{\partial d}\right|_{d=0}=\frac{N}{G}. \tag{39}$$

After substituting $\hat{f}$ in Eq. (34) into Eq. (36), and a detail derivation, we obtain

$$\left.\frac{\partial[\ln(1+\varepsilon^u)+\ln(1+\varepsilon^v)]}{\partial d}\right|_{d=0}=\frac{2M}{F}+\tan\theta\left.\frac{\partial(\Delta\theta)}{\partial d}\right|_{d=0}. \tag{40}$$

During the derivation of Eq. (40), Eq. (16) is used, as well as the relation

$$\Delta\theta|_{d=0}=0.$$

After substituting Eqs. (38), (39) into Eq. (40), we obtain

$$\left.\frac{\partial(\Delta\theta)}{\partial d}\right|_{d=0}=\cot\theta\left(\frac{L}{E}+\frac{N}{G}-\frac{2M}{F}\right). \tag{41}$$

The system of Eqs. (38), (39) and (41) give out ideal strain gradients, and the gradient of change of angle between isoparametric directions. In other words, if the strain gradients of $\ln(1+\varepsilon^u)$ and $\ln(1+\varepsilon^v)$ at $d=0$ are equal to the ratios of the corresponding second and first fundamental form coefficients at the mid-surface before development, and the gradient of $\Delta\theta$ at $d=0$ satisfy the relation (41), the 2D developed shape of the offset surface with small offset distance will be the same.

3.2. Determination of planar developed shape

After solving the nonlinear minimization problem, we obtain the strains ε^u and ε^v at all grid points. We now determine the planar coordinates (X_{ij}, Y_{ij}) of the grid points at the corresponding planar development. Ideally, these coordinates (X_{ij}, Y_{ij}) should satisfy the following equations at all grid points:

$$\mathbf{R}_u\cdot\mathbf{R}_u=e,\qquad \mathbf{R}_u\cdot\mathbf{R}_v=f,\qquad \mathbf{R}_v\cdot\mathbf{R}_v=g, \tag{42}$$

where $\mathbf{R}=(X,Y)$, and e, f, g are obtained from Eqs. (26) as functions of u and v. After discretization of the above Eqs. (42) using finite difference method (central difference for internal points, and forward or backward difference for boundary points), we obtain a system of over-determined nonlinear polynomial equations. Instead of solving the system directly, we solve the following least squares error unconstrained minimization problem

$$\min\sum_{i=1}^{N_g^u}\sum_{j=1}^{N_g^v}(\mathbf{R}_u\cdot\mathbf{R}_u|_{ij}-e_{ij})^2+(\mathbf{R}_u\cdot\mathbf{R}_v|_{ij}-f_{ij})^2+(\mathbf{R}_v\cdot\mathbf{R}_v|_{ij}-g_{ij})^2. \tag{43}$$

This optimization problem can be solved by using the quasi-Newton method (Bertsekas, 1995) for finding an unconstrained minimization of a sum-of-squares of M_1 nonlinear functions in M_2 variables ($M_1 \geqslant M_2$). This can be done by using the NAG Fortran library routine E04GBF (1990). In the implementation, rigid body motion of the developed planar shape is prohibited by forcing the coordinates of the grid points:

$$(X_{ij},Y_{ij})|_{(i=0,j=0)}=(0,0)\quad\text{and}\quad X_{ij}|_{(i=0,j=1)}=0.$$

The starting points of the minimization are given by

$$(X_{ij}, Y_{ij}) = \left(\frac{i}{N_g^u}, \frac{j}{N_g^v}\right)\beta \quad (i = 1, \ldots, N_g^u;\ j = 1, \ldots, N_g^v),$$

where β is a scalar factor.

For the special case when the given surface is a developable surface, the first optimization problem (27)–(28) is solved with the solution $\varepsilon^u(u,v) = \varepsilon^v(u,v) = 0$, and only the second optimization needs to be solved, which is equivalent to development of a developable surface.

3.3. Inverse problem

In order to estimate the accuracy of this surface development algorithm, we solve an inverse problem. After determining the planar developed shape, we can compute e_{ij}, f_{ij}, g_{ij} at all grid points based on the planar coordinates (X_{ij}, Y_{ij}) using the discrete version of Eqs. (42). We then evaluate E_{ij}, F_{ij}, G_{ij} based on the e_{ij}, f_{ij}, g_{ij} and the strains ε_{ij}^u, ε_{ij}^v by using Eq. (26). Under the condition of ideal strain gradients (38) and (39) and the change of angles between isoparametric directions (41), the second fundamental form coefficients at grid points L_{ij}, M_{ij}, N_{ij} are recovered. In other words, we store $\frac{\partial[\ln(1+\varepsilon^u)]}{\partial d}|_{d=0}$, $\frac{\partial[\ln(1+\varepsilon^v)]}{\partial d}|_{d=0}$ and $\frac{\partial(\Delta\theta)}{\partial d}|_{d=0}$ at the developed grid points (X_{ij}, Y_{ij}), and recover the second fundamental form coefficients L_{ij}, M_{ij}, N_{ij} by solving for L, M, N in Eqs. (38), (39) and (41). In Eq. (41), we use the θ computed from the 3D surface to avoid the extra error due to approximating it from the 2D shape. Alternatively, we can compute θ in Eq. (41) from the 2D developed shape. This involves first fitting by splines the isoparametric lines in 2D developed shape based on the coordinates (X_{ij}, Y_{ij}), then computing θ. Since the inverse problem is an auxiliary problem, we use the θ computed from the 3D surface.

After obtaining the coefficients of the first and second fundamental forms, we solve a reverse problem to obtain $\mathbf{r}_{ij} = (x_{ij}, y_{ij}, z_{ij})$ using the least squares error minimization as follows:

$$\begin{aligned}\min \sum_{i=1}^{N_u}\sum_{j=1}^{N_v} &(\mathbf{r}_u \cdot \mathbf{r}_u|_{ij} - E_{ij})^2 + (\mathbf{r}_u \cdot \mathbf{r}_v|_{ij} - F_{ij})^2 + (\mathbf{r}_v \cdot \mathbf{r}_v|_{ij} - G_{ij})^2 \\ &+ \left(\mathbf{r}_{uu} \cdot (\mathbf{r}_u \times \mathbf{r}_v)|_{ij} - L_{ij}\sqrt{E_{ij}G_{ij} - F_{ij}^2}\right)^2 \\ &+ \left(\mathbf{r}_{uv} \cdot (\mathbf{r}_u \times \mathbf{r}_v)|_{ij} - M_{ij}\sqrt{E_{ij}G_{ij} - F_{ij}^2}\right)^2 \\ &+ \left(\mathbf{r}_{vv} \cdot (\mathbf{r}_u \times \mathbf{r}_v)|_{ij} - N_{ij}\sqrt{E_{ij}G_{ij} - F_{ij}^2}\right)^2. \end{aligned} \tag{44}$$

Suitable constraints can be imposed to get rid of rigid body motion. The solution method of problem (44) is the same as that of problem (43), except there are more variables here. The error of this surface development process can be measured by the distance between the grid points on the reconstructed surface and those on the original surface.

4. Surface development along principal curvature directions

In Section 3, surface development is expressed by tensile strains along the isoparametric lines. The assumption made is that the angle between isoparametric directions remains unchanged after a doubly curved surface is developed into a two-dimensional shape. This assumption is reasonable when the angle between isoparametric directions is large and the strains are small. In the case when the angle between $\mathbf{r}_u$ and $\mathbf{r}_v$ is small at some area of the surface, this assumption may cause errors which can not be ignored.

In this section, algorithms for surface development based on strains along principal curvature directions are presented. Since the principal curvature directions are independent of the parametrization of surfaces and are unique except at umbilic points, this surface development is more general. Also, since the angle between two principal curvature directions is a right angle, the assumption that this angle does not change significantly after development is more reasonable.

4.1. Determination of strain field

4.1.1. Formulation

We assume that the surface is defined by the parametric vector equation (1). The surface and its planar development are shown in Fig. 2. The coefficients of the first fundamental form of the curved surface are given by Eq. (3). We further assume that during the surface development process, the strains due to development from curved surface to its planar development are $\varepsilon^s(u,v) \geqslant 0$ and $\varepsilon^t(u,v) \geqslant 0$, along the maximum and minimum principal curvature directions, respectively. Therefore an infinitesimal length $|\mathbf{r}_s\,\mathrm{d}s|$ changes to $(1+\varepsilon^s)|\mathbf{r}_s\,\mathrm{d}s|$, and an infinitesimal length $|\mathbf{r}_t\,\mathrm{d}t|$ changes to $(1+\varepsilon^t)|\mathbf{r}_t\,\mathrm{d}t|$, according to the definition of strain. Thus we have

$$|\mathbf{R}_s| = \left(1+\varepsilon^s\right)|\mathbf{r}_s|, \qquad |\mathbf{R}_t| = \left(1+\varepsilon^t\right)|\mathbf{r}_t|, \tag{45}$$

where $\mathbf{R}(u,v)$ is the planar development. $\mathbf{R}(u,v)$ can also be considered as a parametric surface with its first fundamental form coefficients defined by Eq. (25). Since

$$\mathbf{R}_s \cdot \mathbf{R}_s = (\mathbf{R}_u u_s + \mathbf{R}_v v_s) \cdot (\mathbf{R}_u u_s + \mathbf{R}_v v_s) = e u_s^2 + 2 f u_s v_s + g v_s^2 \tag{46}$$

and

$$\mathbf{r}_s \cdot \mathbf{r}_s = (\mathbf{r}_u u_s + \mathbf{r}_v v_s) \cdot (\mathbf{r}_u u_s + \mathbf{r}_v v_s) = E u_s^2 + 2F u_s v_s + G v_s^2, \tag{47}$$

using the relations in Eqs. (45), (46) and (47), we obtain

$$e u_s^2 + 2 f u_s v_s + g v_s^2 = \left(1+\varepsilon^s\right)^2 \left(E u_s^2 + 2F u_s v_s + G v_s^2\right). \tag{48}$$

Similarly, along minimum principal curvature direction, we have

$$e u_t^2 + 2 f u_t v_t + g v_t^2 = \left(1+\varepsilon^t\right)^2 \left(E u_t^2 + 2F u_t v_t + G v_t^2\right). \tag{49}$$

We also assume that after development, the principal curvature directions remain orthogonal, which gives

$$\mathbf{R}_s \cdot \mathbf{R}_t = (\mathbf{R}_u u_s + \mathbf{R}_v v_s) \cdot (\mathbf{R}_u u_t + \mathbf{R}_v v_t) = 0. \tag{50}$$

Simplifying the above equation gives

$$e u_s u_t + f(u_s v_t + u_t v_s) + g v_s v_t = 0. \tag{51}$$

Then we have a system of three linear equations (48), (49) and (51) in e, f, g whose solution is given by

$$\begin{aligned} e = {} & \frac{v_t^2[Eu_s^2 + 2Fu_s v_s + Gv_s^2](1+\varepsilon^s)^2}{(v_s u_t - u_s v_t)^2} \\ & + \frac{v_s^2[Eu_t^2 + 2Fu_t v_t + Gv_t^2](1+\varepsilon^t)^2}{(v_s u_t - u_s v_t)^2}, \end{aligned} \tag{52}$$

$$\begin{aligned} f = {} & -\frac{u_t v_t[Eu_s^2 + 2Fu_s v_s + Gv_s^2](1+\varepsilon^s)^2}{(v_s u_t - u_s v_t)^2} \\ & - \frac{u_s v_s[Eu_t^2 + 2Fu_t v_t + Gv_t^2](1+\varepsilon^t)^2}{(v_s u_t - u_s v_t)^2}, \end{aligned} \tag{53}$$

$$\begin{aligned} g = {} & \frac{u_t^2[Eu_s^2 + 2Fu_s v_s + Gv_s^2](1+\varepsilon^s)^2}{(v_s u_t - u_s v_t)^2} \\ & + \frac{u_s^2[Eu_t^2 + 2Fu_t v_t + Gv_t^2](1+\varepsilon^t)^2}{(v_s u_t - u_s v_t)^2}. \end{aligned} \tag{54}$$

We minimize the strains $\varepsilon^s(u,v)$ and $\varepsilon^t(u,v)$ which satisfy the condition that after adding these strains to the doubly curved surface along principal curvature directions, the surface maps to a planar shape on which Gaussian curvature is zero. Using Eq. (12), this results into a minimization problem same as problem (27)–(28) except that ε^u, ε^v are replaced by ε^s, ε^t respectively. As shown in Section 3, this constrained minimization problem is discretized by using the finite difference method and trapezoidal rule of integration. The final formulation is similar to that in Section 3 except that $\varepsilon^u, \varepsilon^v$ are replaced by $\varepsilon^s, \varepsilon^t$. Again, the nonlinear constrained minimization problem is solved by using the Fortran NAG routine E04VDF (1990).

4.1.2. Strain gradients

After solving for the strain distribution at the mid-surface, we then can determine the ideal gradient of the strains along the normal of the mid-surface. As mentioned in Section 2, the strain gradients provide the mechanism of surface curvature in metal forming process. Based on Eqs. (48), along s direction, the relation of the first fundamental form coefficients of the offset surface of distance d along the normal from the mid-surface is:

$$\hat{e}u_s^2 + 2\hat{f}u_s v_s + \hat{g}v_s^2 = \left(1+\hat{\varepsilon}^s\right)^2\left(\hat{E}u_s^2 + 2\hat{F}u_s v_s + \hat{G}v_s^2\right). \tag{55}$$

Since after development, the 2D shape is the same across the thickness, we have

$$\frac{\partial(\hat{e}u_s^2 + 2\hat{f}u_s v_s + \hat{g}v_s^2)}{\partial d} = 0. \tag{56}$$

After substituting Eq. (55) into Eq. (56), and using the expression (20), we have

$$\frac{\partial}{\partial d}\left(\left(1+\hat{\varepsilon}^s\right)^2\left(\hat{\mathbf{r}}_s \cdot \hat{\mathbf{r}}_s\right)\right) = 0. \tag{57}$$

Expanding the above equation, we obtain at $d = 0$,

$$\left.\frac{\partial[\ln(1+\varepsilon^s)]}{\partial d}\right|_{d=0} = -\frac{1}{2\hat{\mathbf{r}}_s\cdot\hat{\mathbf{r}}_s}\left.\frac{\partial(\hat{\mathbf{r}}_s\cdot\hat{\mathbf{r}}_s)}{\partial d}\right|_{d=0} = \kappa_{\max}. \tag{58}$$

The last equality comes from Eq. (18). Similarly, along t direction,

$$\left.\frac{\partial[\ln(1+\varepsilon^t)]}{\partial d}\right|_{d=0} = -\frac{1}{2\hat{\mathbf{r}}_t\cdot\hat{\mathbf{r}}_t}\left.\frac{\partial(\hat{\mathbf{r}}_t\cdot\hat{\mathbf{r}}_t)}{\partial d}\right|_{d=0} = \kappa_{\min}. \tag{59}$$

4.2. *Determination of planar developed shape*

After solving the nonlinear minimization problem, we obtain the strains ε^s and ε^t at all grid points. The first fundamental form coefficients e, f, g of the planar developed shape is then obtained from Eqs. (52)–(54). The method in Section 3.2 can be used to determine the planar developed shape.

For the special case when the given surface is a developable surface, the first optimization problem is solved with the solution $\varepsilon^s(u,v) = \varepsilon^t(u,v) = 0$, and only the second optimization needs to be solved, which is equivalent to development of a developable surface.

4.3. *Inverse problem*

In order to estimate the accuracy of this surface development algorithm, we can compute e_{ij}, f_{ij}, g_{ij} at all grid points based on the planar coordinates (X_{ij}, Y_{ij}) using the discrete version of Eqs. (42). We then evaluate E_{ij}, F_{ij}, G_{ij} based on the e_{ij}, f_{ij}, g_{ij} and the strains ε^s, ε^t by using Eqs. (48) and (49), along with the following condition that the principal curvature directions are orthogonal.

$$Eu_su_t + F(v_su_t + u_sv_t) + Gu_su_t = 0. \tag{60}$$

Under the condition of ideal strain gradients along the principal curvature directions, the second fundamental form coefficients at grid points L_{ij}, M_{ij}, N_{ij} are recovered. Then we solve the problem (44) to reconstruct the 3D surface. Suitable constraints can be imposed to get rid of rigid body motion. The error of this surface development process can be measured by the distance between the grid points on the reconstructed surface and those on the original surface.

5. Analysis of the algorithms

5.1. *Convergence analysis*

In this section, we discuss the convergence of the surface development algorithms, i.e., the convergence of the discrete solution to the continuous solution $\varepsilon^u(u,v)$, $\varepsilon^v(u,v)$ to the optimization problem (27)–(28).

Theorem 5.1. *The error on the right side of Eq.* (28) *due to discretization is* $\mathrm{O}((\Delta u)^2, (\Delta v)^2, \Delta u\Delta v)$.

Proof. Let $h(\varepsilon^u(u,v), \varepsilon^v(u,v))$ be the right side of Eq. (28), then discretization introduces errors $\mathrm{O}((\Delta u)^2)$, $\mathrm{O}((\Delta v)^2)$, or $\mathrm{O}(\Delta u \Delta v)$ into all the derivatives in $h(\varepsilon^u(u,v), \varepsilon^v(u,v))$. By substituting $\partial e/\partial u$ with $(\partial e/\partial u) + \mathrm{O}((\Delta u)^2)$, $\partial e/\partial v$ with $(\partial e/\partial v) + \mathrm{O}((\Delta v)^2)$, etc. and expand the right side of Eq. (28), we obtain

$$\bar{h}\big(\varepsilon^u(u,v), \varepsilon^v(u,v)\big) = h\big(\varepsilon^u(u,v), \varepsilon^v(u,v)\big) + \mathrm{O}\big((\Delta u)^2, (\Delta v)^2, \Delta u \Delta v\big), \tag{61}$$

where $\bar{h}$ is the constraint after discretization. □

Since the error on the right side of Eq. (28) due to discretization is $\mathrm{O}((\Delta u)^2, (\Delta v)^2, \Delta u \Delta v)$, and the error on the objective function in (27) due to discretization is $\mathrm{O}((\Delta u)^2, (\Delta v)^2)$, as mentioned in Section 3.1.1, as $\Delta u \to 0$, $\Delta v \to 0$, both the objective function and constraints of the discretized optimization problem converge to those of the continuous problem. Thus the discretized optimization problem for strain determination converges to the continuous optimization problem (27)–(28).

In general, we expect that the discrete solution $\varepsilon^u_{ij}, \varepsilon^v_{ij}$ converges to the continuous solution $\varepsilon^u(u_i, v_j), \varepsilon^v(u_i, v_j)$ as $\Delta u \to 0$, $\Delta v \to 0$, and the error of variables ε^u_{ij}, ε^v_{ij} due to discretization error is of the order $\mathrm{O}((\Delta u)^2, (\Delta v)^2, \Delta u \Delta v)$. Since the error of variables ε^u_{ij}, ε^v_{ij} due to discretization error is of the second order, the error of the objective function is also of the second order. According to the relations (26), the error of e, f, g in (42) is also of the second order, which means the error of the final X, Y coordinates of the developed shape is of the second order with respect to $\Delta u = \Delta v$. Similar results exist for surface development along the principal curvature directions. See (Yu, 1999) for more details.

5.2. Complexity analysis

5.2.1. The algorithm for strain determination

As shown in Section 3.1.2, solution of the first constrained optimization involves solution of a sequential quadratic programming problems. In a major iteration, operations mainly include the formulation of the H matrix, vector g and the determination of the step α_k by line search. The dominant operations are in formulation of the H matrix by quasi-Newton method, which requires $\mathrm{O}(n^3)$ of operations, where n is the dimension of q. In our problem, $n = 2(N^u_g - 2)(N^v_g - 2)$, so the total number of operations is $\mathrm{O}((N^u_g N^v_g)^3)$. Solving the QP subproblem (32) has two distinct phases. In the first (the LP phase), an iterative procedure is carried out to determine a feasible point. The second phase (the QP phase) generates a sequence of feasible iterates in order to minimize the quadratic objective function. In both phases, a subset of the constraints—called the working set—is used to define the search direction at each iteration; typically, the working set includes constraints that are satisfied within the corresponding tolerance. The LP phase for determining a feasible point need only be carried out once in each major iteration, and possibly less than once, when the solution from the previous major iteration is feasible. The cost of linear programming depends on the method used; so let us assume the simplex method is used. Though the worst case performance of the method is exponential (Bertsimas and Tsitsiklis, 1997), this method operates on a $m \times n$ matrix to generate a solution usually in $\mathrm{O}(m^2 n)$ time (Gass, 1985). The constraint matrix in problem (32) is of the size $[2N^u_g N^v_g + (N^u_g - 2)(N^v_g - 2)] \times 2N^u_g N^v_g$, therefore, the LP takes $\mathrm{O}((N^u_g N^v_g)^3)$ time. The

QP phase involves 4 steps and takes $\mathrm{O}(n^3) = \mathrm{O}((N_g^u N_g^v)^3)$ time. See (Yu, 1999) for detail. In summary, each major iteration takes $\mathrm{O}((N_g^u N_g^v)^3)$ time.

5.2.2. The algorithm for planar developed shape determination

The unconstrained minimization problem (43) is a least square minimization of $m = 3N_g^u N_g^v$ functions with $n = 2N_g^u N_g^v$ variables. The general algorithm for nonlinear least-square problems

$$\min F(x) = \sum_{i=1}^{m} \left[f_i(x)\right]^2, \quad x \in E^n,\ m \leqslant n,$$

is given in (Gill, and Murray, 1978). A detailed analysis of the algorithm (Yu, 1999) shows that in each iteration of the non-constrained minimization, the required number of operations is $\mathrm{O}((N_g^u N_g^v)^3)$.

6. Examples

In this section, we demonstrate how the algorithms work for surface development based on strains along isoparametric lines and along lines of curvature. The surfaces in the examples include surfaces with all elliptical points (positive Gaussian curvature), surfaces with all hyperbolic points (negative Gaussian curvature), and more complex surfaces that have both positive and negative Gaussian curvature regions. Compared to the surfaces in shipbuilding industry, the surfaces used in the examples have much larger absolute value of Gaussian curvature and hence they are more difficult to develop. All examples were executed on a graphics workstation running at 200 MHz.

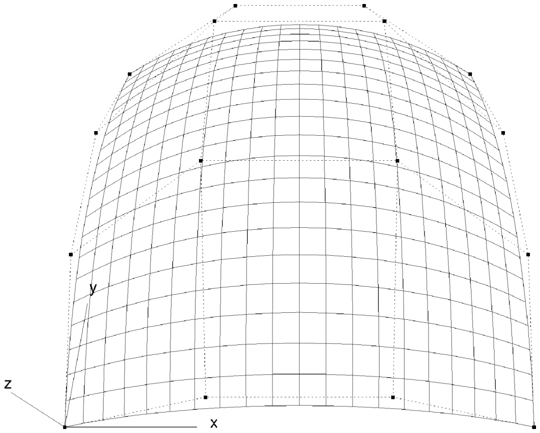


Fig. 4. The bi-cubic Bézier surface in Example 1.

6.1. Example 1

An elliptical bicubic Bézier surface $\mathbf{r}(u, v) = \sum_{i=0}^{3}\sum_{j=0}^{3}\mathbf{r}_{ij}B_{i,3}(u)B_{j,3}(v)$ with the following control points:

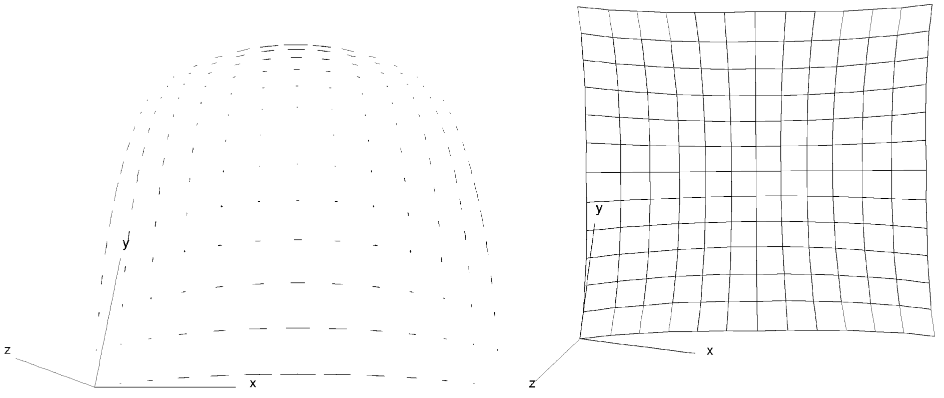
$(0, 0, 0)$	$(0, 1/3, 0.15)$	$(0, 2/3, 0.15)$	$(0, 1, 0)$
$(1/3, 0, 0.25)$	$(1/3, 1/3, 0.5)$	$(1/3, 2/3, 0.5)$	$(1/3, 1, 0.25)$
$(2/3, 0, 0.25)$	$(2/3, 1/3, 0.5)$	$(2/3, 2/3, 0.5)$	$(2/3, 1, 0.25)$
$(1, 0, 0)$	$(1, 1/3, 0.15)$	$(1, 2/3, 0.15)$	$(1, 1, 0)$

The surface along with its control polygon is shown in Fig. 4. The constrained minimization problem (27)–(28) or its counterpart for strains along principal curvature directions is discretized at 13×13 grid points which are equally distributed in u, v domain.

6.1.1. Results from surface development along isoparametric lines

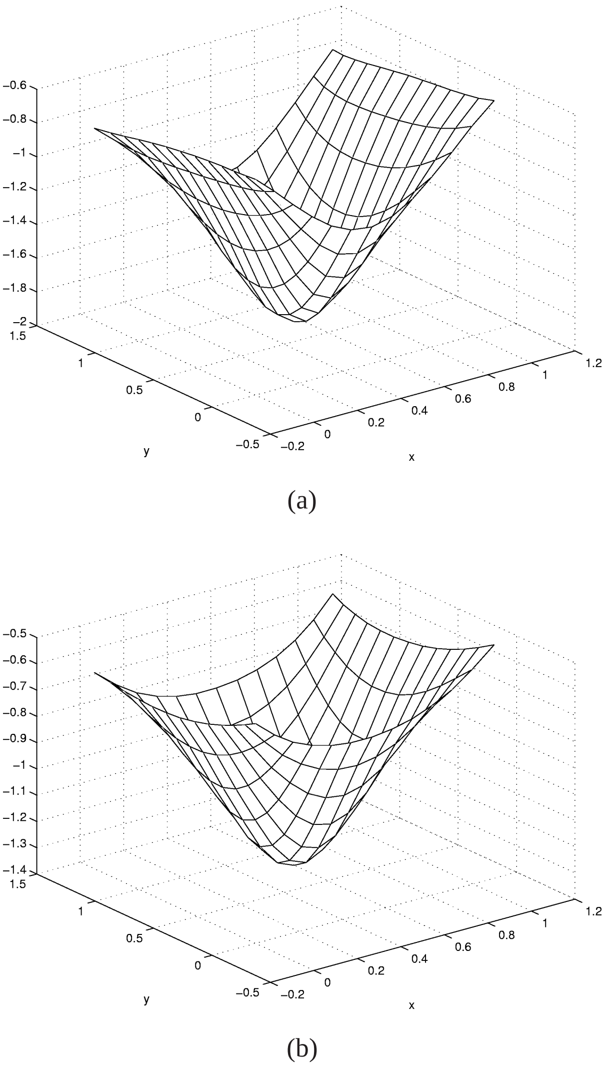
Fig. 5(a) shows the strain distribution after the constrained minimization problem was solved using tolerances of 10^{-5} for the constraints and 10^{-4} for the objective function. The strains are scaled to fit into the figure. The extreme values of the strain field are located at $(u, v) = (0, 0.5)$ or $(u, v) = (1.0, 0.5)$ with $(\varepsilon^u, \varepsilon^v) = (0.0012101, 0.203391)$, and at $(u, v) = (0.5, 0)$ or $(u, v) = (0.5, 1.0)$ with $(\varepsilon^u, \varepsilon^v) = (0.242961, 0.00136569)$. The objective function converges to the value of 6.3857×10^{-3} at the solution, and all the constraints are within the tolerance of 1.0×10^{-5}.

After development, the planar shape is shown in Fig. 5(b). The four corner points have coordinates of (0,0), (−0.14799, 1.10615), (1.18199, 0.15838), (1.03399, 1.26452) respectively. The final value of the formula (43) is less than 10^{-4} times $\sum_{i=1}^{N_g^u}\sum_{j=1}^{N_g^v}(e_{ij}^2 + f_{ij}^2 + g_{ij}^2)$, the sum of the squares of the right side of system (42) at all grid points.



(a) (b)

Fig. 5. (a) The strain distribution and (b) the corresponding 2D shape of the surface in Example 1, developed along isoparametric lines.



(a)

(b)

Fig. 6. Logarithmic strain gradients along (a) u-isoparametric line and (b) v-isoparametric line.

Figs. 6(a) and (b) show the ideal strain gradients $\frac{\partial[\ln(1+\varepsilon^u)]}{\partial d}|_{d=0}$ and $\frac{\partial[\ln(1+\varepsilon^v)]}{\partial d}|_{d=0}$ evaluated at grid points such that the first fundamental form coefficients of the 2D developed shape are constant at any offset distance d close to 0.

In order to estimate the accuracy of this surface development algorithm, we reconstructed the 3D surface by using the method in Section 3.3. We fixed 6 variables to avoid the rigid body motion in the surface reconstruction process. We set $(x, y, z) = (0, 0, 0)$ at $(u, v) = (0, 0)$, $(x, z) = (0, 0)$ at $(u, v) = (0, 1)$, and $z = 0$ at $(u, v) = (1, 0)$. After solving the problem (44), the obtained reconstructed surface is shown in Fig. 7 (solid line) along

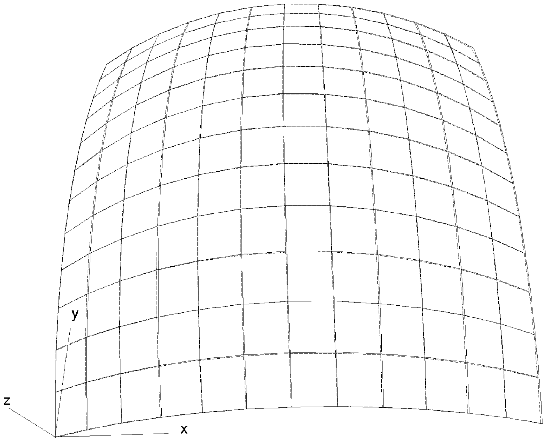


Fig. 7. The reconstructed (solid line) and the original surfaces (dashed line) in Example 1, when developed along isoparametric lines.

Table 1
CPU time for each optimization at various number of grid points (Example 1, development along isoparametric lines)

N_g	Niter1	obj1 (10^{-3})	CPU1 (s)	CPU1/Niter1 (s)	Niter2	CPU2 (s)	CPU2/Niter2 (s)
7	2	6.658	0.71	0.355	11	5.86	0.533
9	8	6.492	8.53	1.066	6	12.67	2.112
11	7	6.428	34.72	4.960	5	37.00	7.400
13	18	6.386	177.49	9.861	5	110.66	22.132
15	13	6.359	401.26	30.866	6	333.23	55.538
17	15	6.341	1097.28	73.152	5	658.71	131.742
19	33	6.327	3869.18	117.25	5	1390.80	278.16
21	37	6.317	8168.48	220.77	5	2761.26	555.25

with the original surface (dashed line). We see an excellent match of the reconstructed surface with the original surface. The maximum error (distance) between the grid points of reconstructed surface and that of the original surface is 0.00459319.

Table 1 shows the CPU time spent on each optimization for various numbers of grid points, objective functions, etc., where N_g is the number of grid points in both u and v directions; Niter1 is the number of iterations in the first optimization; obj1 is the converged value of the objective function in the first optimization; CPU1 is the CPU time spent on the first optimization; CPU1/Niter1 is the CPU time per iteration in the first optimization; Niter2 is the number of iterations in the second optimization; CPU2 is the CPU time spent on the second optimization; and CPU2/Niter2 is the CPU time per iteration in the second

Table 2
Accuracy of the surface development process (Example 1, development along isoparametric lines)

Number of grids	7	9	11	13	15	17	19	21
$\Delta u = \Delta v$	1/6	1/8	1/10	1/12	1/14	1/16	1/18	1/20
Error ($\times 10^{-3}$)	17.763	9.935	6.580	4.549	3.428	2.612	2.113	1.697

optimization. For various numbers of grid points, the tolerance for constraints is 10^{-5}, and the tolerance for objective function is 10^{-4}.

After data fitting, the CPU time per iteration for the first optimization is approximately $2.5829 \times 10^{-6}(N_g)^6$, while that for the second optimization is approximately $1.8853 \times 10^{-6}(N_g)^{6.4}$. We see that the CPU time per iteration in the first optimization agrees well with the theoretical results in Section 5; i.e., when $N_g^u = N_g^v = N_g$, the CPU time per iteration in the first optimization is $\mathrm{O}(N_g^6)$. However, the performance observed for the second optimization is slightly worse than the theoretical results. Instead of $\mathrm{O}(N_g^6)$ CPU time per iteration, we observed $\mathrm{O}(N_g^{6.4})$ time per iteration. It is not clear why this happened. One possibility may be because the NAG routine uses iterative methods inside each iteration, so that as N_g increases, the problems are increasingly ill-conditioned, thus requiring more mini-iterations. A thorough track of the running time for different phases of the algorithm may be used to resolve this problem.

Table 2 shows the maximum error due to the development and reconstruction process for various numbers of grid points. After a data fitting process was carried out which fitted the data in Table 2 with the function

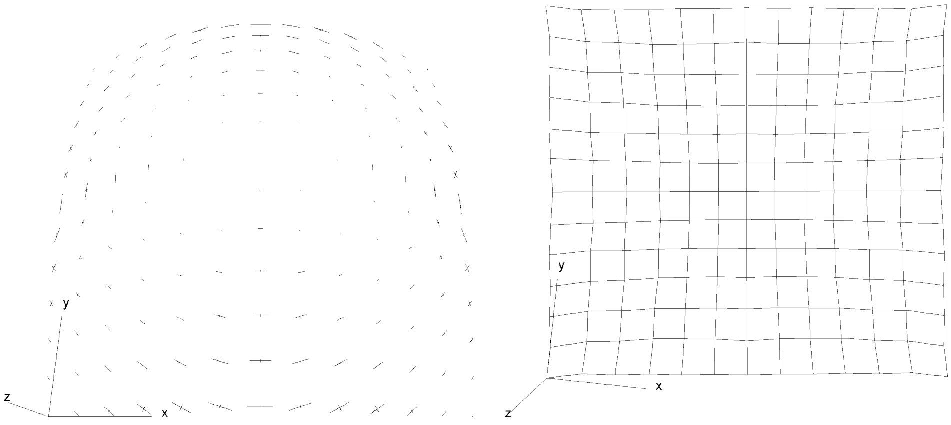
$$E = c(\Delta u)^a. \tag{62}$$

We obtain $c = 0.5689$, $a = 1.9397$. Here we see the error function due to the surface development and reconstruction process is of the order $a < 2$. This is partly because the assumption, that the angle between isoparametric lines does not change after development, introduces extra errors.

6.1.2. Results from surface development along principal curvature directions

Fig. 8(a) shows the strain distribution after the constrained minimization problem was solved using tolerances 10^{-5} for the constraints and 10^{-4} for the objective function. The strains are scaled to fit into the figure. The extreme values of the strain field are located at $(u, v) = (0, 0.5)$ or $(u, v) = (1.0, 0.5)$ with $(\varepsilon^s, \varepsilon^t) = (0.1808969, 0.006270948)$, and at $(u, v) = (0.5, 0)$ or $(u, v) = (0.5, 1.0)$ with $(\varepsilon^s, \varepsilon^t) = (0.001567142, 0.1790917)$. The objective function converges to 6.78804×10^{-3} at the solution, and all the constraints are within the tolerance of 1.0×10^{-5}.

After development, the planar shape is shown in Fig. 8(b). The four corner points have coordinates of (0,0), $(-0.11802, 1.09373)$, $(1.17949, 0.12786)$, $(1.06147, 1.22159)$ respectively. The final value of the formula (43) is less than 2×10^{-4} times $\sum_{i=1}^{N_g^u} \sum_{j=1}^{N_g^v} (e_{ij}^2 + f_{ij}^2 + g_{ij}^2)$, the sum of the squares of the right side of system (42) at all grid points.



(a) (b)

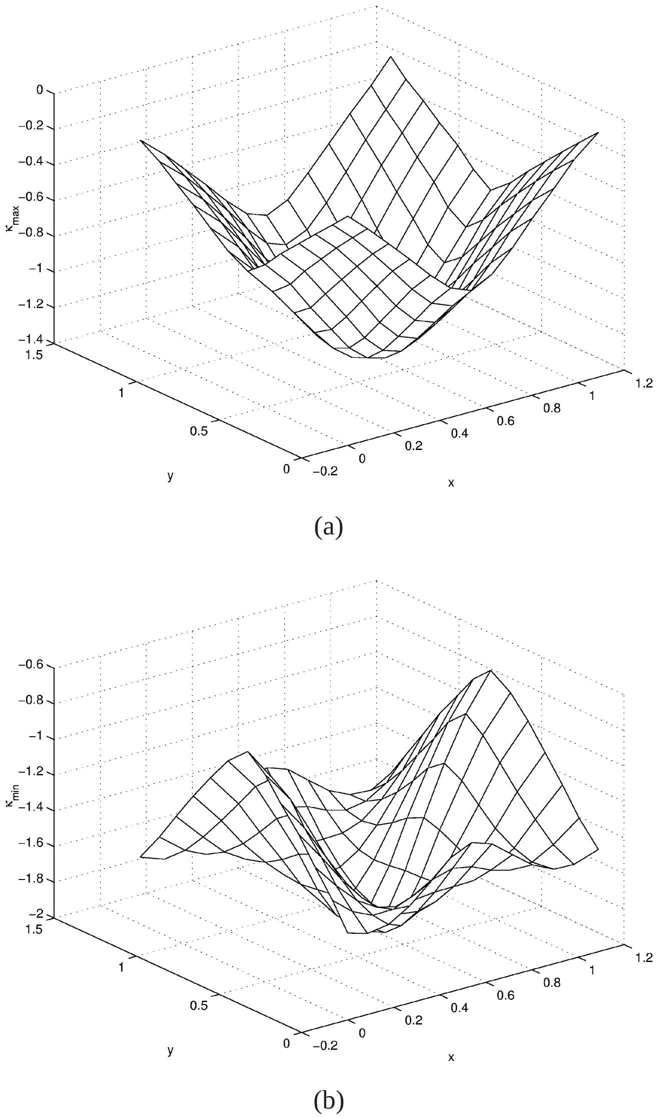
Fig. 8. (a) The strain distribution and (b) the corresponding 2D shape of the surface in Example 1, developed along the principal curvature directions.

Table 3
CPU time for each optimization at various numbers of grid points (Example 1, development along principal curvature directions)

N_g	Niter1	obj1 (10^{-3})	CPU1(s)	CPU1/Niter1(s)	Niter2	CPU2(s)	CPU2/Niter2(s)
7	6	6.389	1.45	0.242	11	5.66	0.515
9	6	6.610	6.52	1.087	6	11.84	1.973
11	9	6.703	32.16	3.573	6	41.75	6.958
13	13	6.788	124.99	9.615	5	106.97	21.394
15	16	6.826	413.91	25.870	5	298.96	59.792
17	15	6.842	942.79	62.853	5	682.85	136.57
19	18	6.846	2341.75	130.10	5	1419.11	283.82
21	28	6.850	5770.87	206.10	5	2761.5	552.30

Figs. 9(a) and (b) show the ideal strain gradient $\frac{\partial[\ln(1+\varepsilon^s)]}{\partial d}|_{d=0}$ and $\frac{\partial[\ln(1+\varepsilon^t)]}{\partial d}|_{d=0}$ evaluated at grid points such that the first fundamental form coefficients of the 2D developed shape are constant at any offset distance d close to 0.

In order to estimate the accuracy of this surface development algorithm, we reconstructed the 3D surface by using the method in Section 4.3. After getting rid of the rigid body motion, the obtained reconstructed surface is shown in Fig. 10 (solid line) along with the original surface (dashed line). We see an excellent match of the reconstructed surface with the original surface. The maximum error (distance) between the grid points of reconstructed surface and that of the original surface is 0.00509951.



(a)

(b)

Fig. 9. Logarithmic strain gradients along (a) maximum curvature direction (b) minimum curvature direction for the surface in Example 1, developed along the principal curvature directions.

Table 3 shows the CPU time spent on each optimization for various number of grid points, objective functions, etc. After data fitting, the CPU time per iteration for the first optimization is approximately $2.4895 \times 10^{-6}(N_g)^6$, while that for the second optimization is approximately $1.4834 \times 10^{-6}(N_g)^{6.4623}$. We see that the CPU time per iteration in the first optimization agrees well with the theoretical results in Section 5; i.e., when $N_g^u = N_g^v = N_g$, the CPU time per iteration in the first optimization is $\mathrm{O}(N_g^6)$. However,

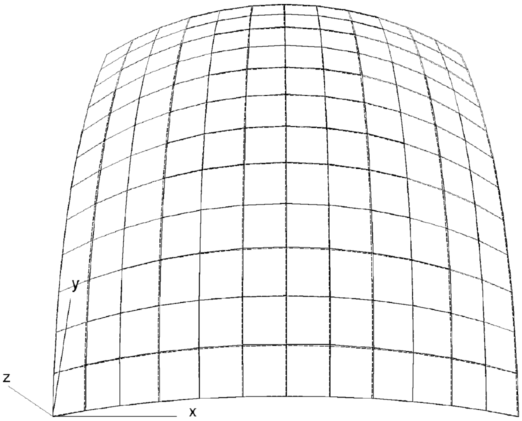


Fig. 10. The reconstructed (solid line) and the original surfaces (dashed line) for the surface in Example 1, developed along the principal curvature directions.

Table 4
Accuracy of the surface development process (Example 1, development along principal curvature directions)

Grid number	7	9	11	13	15	17	19	21
$\Delta u = \Delta v$	$1/6$	$1/8$	$1/10$	$1/12$	$1/14$	$1/16$	$1/18$	$1/20$
Error ($\times 10^{-3}$)	18.1004	9.9129	7.1817	5.0995	3.9349	3.0282	2.4936	1.9664

the performance observed for the second optimization is slightly worse than the theoretical results. Instead of $\mathrm{O}(N_g^6)$ CPU time per iteration, we observed $\mathrm{O}(N_g^{6.46})$ time per iteration.

Table 4 shows the maximum error due to the development and reconstruction process for various number of grid points. When a data fitting process was carried out to fit the data in Table 4 with Eq. (62), we obtain $c = 0.4433$, $a = 1.7987$. Here we also see the error function due to the surface development and reconstruction process is of the order $a < 2$. This is partly because the assumption, that the angle between curvature directions does not change after development, introduces extra errors.

6.2. Example 2

We use this example to show how the algorithms work and the convergence of the objective function during the process.

The second surface is a bicubic Bézier surface

$$\mathbf{r}(u, v) = \sum_{i=0}^{3} \sum_{j=0}^{3} \mathbf{r}_{ij} B_{i,3}(u) B_{j,3}(v),$$

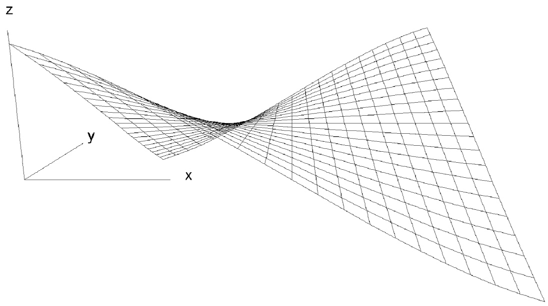


Fig. 11. The bicubic Bézier surface in Example 2.

where all the points on the surface are hyperbolic. The control points of the saddle-shaped surface are given by:

$(0,0,0.25)$	$(0,1/3,0.1)$	$(0,2/3,-0.1)$	$(0,1,-0.25)$
$(1/3,0,0.1)$	$(1/3,1/3,0.05)$	$(1/3,2/3,-0.05)$	$(1/3,1,-0.1)$
$(2/3,0,-0.1)$	$(2/3,1/3,-0.05)$	$(2/3,2/3,0.05)$	$(2/3,1,0.1)$
$(1,0,-0.25)$	$(1,1/3,-0.1)$	$(1,2/3,0.1)$	$(1,1,0.25)$

The surface is shown in Fig. 11. Again, 13 × 13 grid points are used in discretization.

6.2.1. *Results from surface development along isoparametric lines*

Fig. 12(a) shows the strain distribution after the constrained minimization problem was solved using tolerances of 10^{-5} for the constraints and 10^{-4} for the objective function. The strains are scaled to fit into the figure. The extreme values of the strain field are located at $(u,v)=(0.5,0.5)$ with $(\varepsilon^u,\varepsilon^v)=(0.0867214,0.0886954)$. The objective function converges to the value of 4.005672×10^{-3} at the solution, and all the constraints are within the tolerance of 10^{-5}.

After development, the planar shape is shown in Fig. 12(b). The four corner points have coordinates of (0,0), (−0.09223, 1.11186), (1.11534, 0.09258), (1.02311, 1.20444), respectively. The final value of the formula (43) is less than 10^{-5} times $\sum_{i=1}^{N_g^u}\sum_{j=1}^{N_g^v}(e_{ij}^2+f_{ij}^2+g_{ij}^2)$, the sum of the squares of the right side of system (42) at all grid points. Here we see a planar development similar to that in Example 1.

Table 5 shows the variation of the objective function in the first optimization with respect to $\Delta u(=\Delta v)$ and number of grid points. The objective function can be fitted with the curve $\mathrm{obj1}=0.003909+0.013492(\Delta u)^2$. As pointed out in Section 5, a quadratic convergence is observed in the objective function of the first optimization. If we allow extrapolation, we can estimate the objective function approaches 0.003909 as $\Delta u=\Delta v\to0$.

6.2.2. *Results from surface development along principal curvature directions*

Fig. 13(a) shows the strain distribution after the constrained minimization problem was solved using tolerances 10^{-5} for the constraints and 10^{-4} for the objective function. The

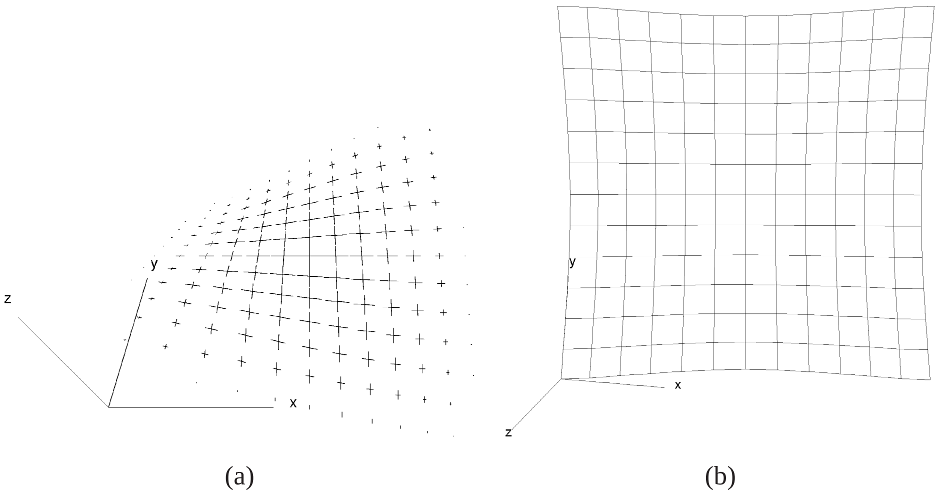


Fig. 12. (a) The strain distribution and (b) the corresponding 2D shape of the surface in Example 2, developed along isoparametric lines.

Table 5
The objective function of the 1st optimization (Example 2, development along isoparametric lines)

Grid number	7	9	11	13	15	17	19	21
$\Delta u = \Delta v$	$1/6$	$1/8$	$1/10$	$1/12$	$1/14$	$1/16$	$1/18$	$1/20$
obj1 $(\times 10^{-3})$	4.2831	4.125	4.038	4.006	3.970	3.964	3.953	3.945

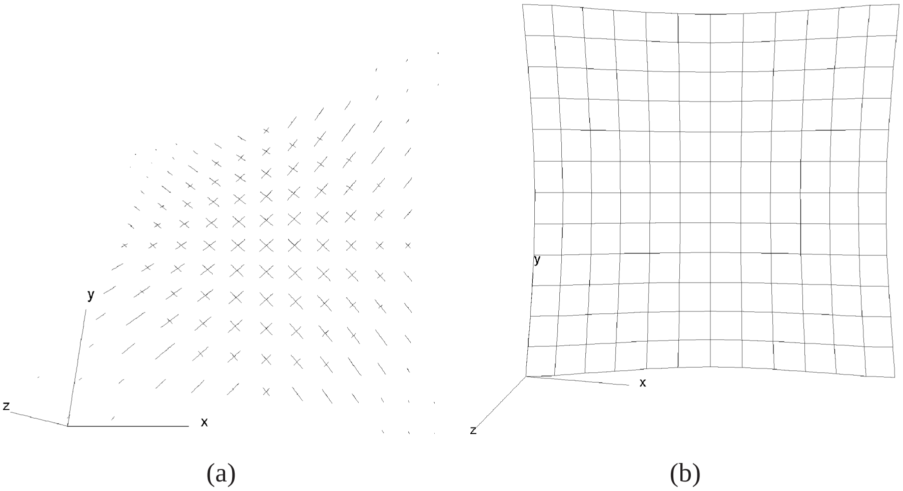


Fig. 13. (a) The strain distribution and (b) the corresponding 2D shape of the surface in Example 2, developed along the principal curvature directions.

Table 6
The objective function of the 1st optimization (Example 2, development along principal curvature directions)

Grid number	7	9	11	13	15	17	19	21
$\Delta u = \Delta v$	$1/6$	$1/8$	$1/10$	$1/12$	$1/14$	$1/16$	$1/18$	$1/20$
obj1 $(\times 10^{-3})$	3.1611	2.93382	2.8422	2.7909	2.7661	2.7486	2.7388	2.7331

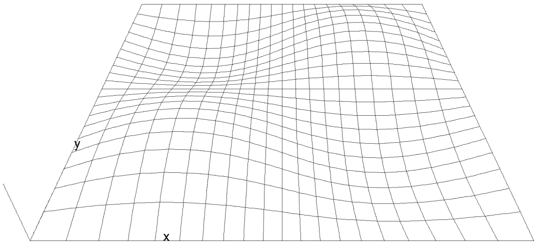


Fig. 14. A wave-like B-spline surface in Example 3.

strains are scaled to fit into the figure. As a comparison with the results in Section 6.2.1, the strains at $(u, v) = (0.5, 0.5)$ are $(\varepsilon^s, \varepsilon^t) = (0.05627, 0.05581)$. The objective function is 2.791×10^{-3} at the solution, and all the constraints are within the tolerance of 1.0×10^{-5}.

After development, the planar shape is shown in Fig. 13(b). The four corner points have coordinates of (0,0), $(-0.09909, 1.11083)$, $(1.11505, 0.09953)$, $(1.01596, 1.21035)$ respectively. The final value of the formula (43) is less than 3×10^{-5} times $\sum_{i=1}^{N_g^u} \sum_{j=1}^{N_g^v} (e_{ij}^2 + f_{ij}^2 + g_{ij}^2)$, the sum of the squares of the right side of system (42) at all grid points.

Table 6 shows the variation of the objective function in the first optimization with respect to $\Delta u (= \Delta v)$ and number of grid points. The objective function can be fitted with the curve $\text{obj1} = 0.002689 + 0.015442(\Delta u)^2$. As pointed out in Section 5, a quadratic convergence is observed of the objective function of the first optimization. If we allow extrapolation, we can estimate the objective function approaches 0.002689 as $\Delta u = \Delta v \to 0$.

6.3. Example 3

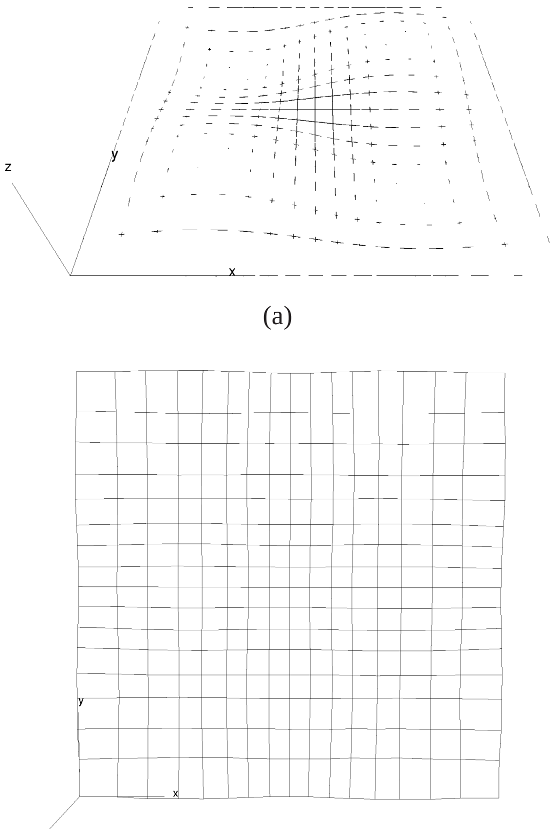
This example shows the performance of the algorithms on a general B-spline surface. A wave-like bicubic integral B-spline surface

$$\mathbf{r}(u, v) = \sum_{i=0}^{4} \sum_{j=0}^{4} \mathbf{r}_{ij} N_{i,4}(u) N_{j,4}(v)$$

on a uniform knot vector $[0,0,0,0,0.5,1,1,1,1]^2$ with the following control points (x,y,z)

$(0,0,0)$	$(0,0.25,0)$	$(0,0.5,0)$	$(0,0.75,0)$	$(0,1,0)$
$(0.25,0,0)$	$(0.25,0.25,0.2)$	$(0.25,0.5,0)$	$(0.25,0.75,-0.2)$	$(0.25,1,0)$
$(0.5,0,0)$	$(0.5,0.25,0)$	$(0.5,0.5,0)$	$(0.5,0.75,0)$	$(0.5,1,0)$
$(0.75,0,0)$	$(0.75,0.25,-0.2)$	$(0.75,0.5,0)$	$(0.75,0.75,0.2)$	$(0.75,1,0)$
$(1,0,0)$	$(1,0.25,0)$	$(1,0.5,0)$	$(1,0.75,0)$	$(1,1,0)$

is shown in Fig. 14. 17 × 17 grid points are used in discretization.



(a)

(b)

Fig. 15. (a) The strain distribution and (b) the corresponding 2D shape of the surface in Example 3, developed along isoparametric lines.

6.3.1. Results from surface development along isoparametric lines

Fig. 15(a) shows the strain distribution after the constrained minimization problem was solved using tolerances of 10^{-7} for the constraints and 10^{-4} for the objective function. The strains are scaled to fit into the figure. The strains at the center of the surface $(u, v) = (0.5, 0.5)$ are $(\varepsilon^u, \varepsilon^v) = (0.0555, 0.0550)$. The objective function converges to the value of 9.84842×10^{-4} at the solution, and all the constraints are within the tolerance of 10^{-7}.

After development, the planar shape is shown in Fig. 15(b). The four corner points have coordinates of $(0, 0)$, $(0.00315, 1.03617)$, $(1.03618, -0.00312)$, $(1.03933, 1.03306)$, respectively. The final value of the formula (43) is about 10^{-4} times $\sum_{i=1}^{N_g^u}\sum_{j=1}^{N_g^v}(e_{ij}^2 + f_{ij}^2 + g_{ij}^2)$, the sum of the squares of the right side of system (42) at all grid points.

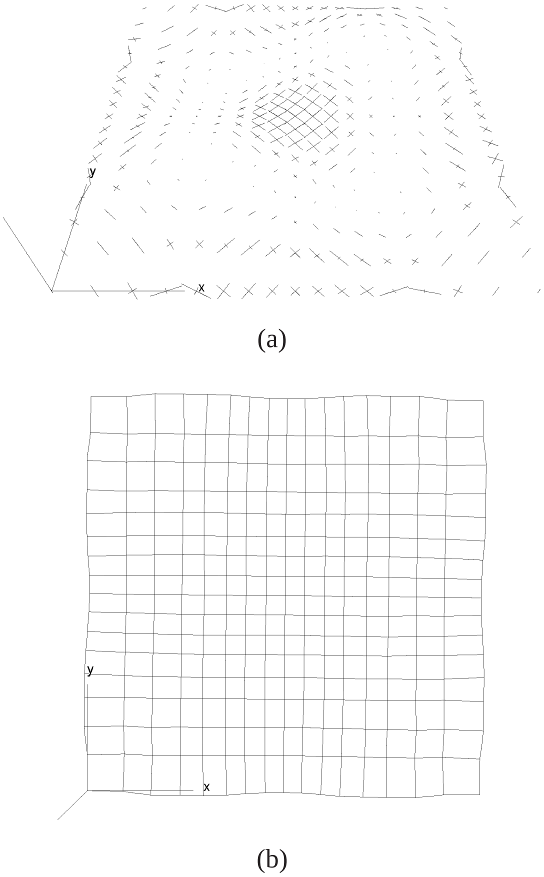


Fig. 16. (a) The strain distribution and (b) the corresponding 2D shape of the surface in Example 3, developed along the principal curvature directions.

6.3.2. Results from surface development along principal curvature directions

Fig. 16(a) shows the strain distribution after the constrained minimization problem was solved using the tolerance of 10^{-8} for the constraints and 10^{-4} for the objective function. The strains are scaled to fit into the figure. As a comparison with the results in Section 6.3.1, the strains at $(u, v) = (0.5, 0.5)$ are $(\varepsilon^s, \varepsilon^t) = (0.08258, 0.08267)$. The objective function is 1.6018×10^{-3} at the solution, and all the constraints are within the tolerance of 1.0×10^{-8}.

After development, the planar shape is shown in Fig. 16(b). The four corner points have coordinates of $(0, 0)$, $(0.01073, 1.03467)$, $(1.03467, -0.01074)$, $(1.04540, 1.02393)$, respectively. The final value of the formula (43) is about 1.4×10^{-4} times $\sum_{i=1}^{N_g^u} \sum_{j=1}^{N_g^v} (e_{ij}^2 + f_{ij}^2 + g_{ij}^2)$, the sum of the squares of the right side of system (42) at all grid points.

6.4. Discussion

The examples in this section show that the algorithms for surface development along isoparametric lines and principal curvature directions work well geometrically. The strains obtained and the CPU time spent on both methods are at the same magnitude. The 2D developed shapes are similar. Physically, however, development along principal curvature directions is more realizable. This can be seen from Eq. (41) that the gradient of the angle change $\frac{\partial(\Delta\theta)}{\partial d}|_{d=0}$ must satisfy. This gradient of the angle change $\frac{\partial(\Delta\theta)}{\partial d}|_{d=0}$ is hard to control during metal forming process, since $\frac{\partial(\Delta\theta)}{\partial d}|_{d=0}$ is not directly related to the strain gradients. $\Delta\theta$ is due to shear strain, which is of the second order compared with the normal strains, and is not simply related to the temperature distribution during metal forming process by line heating. The gradients of the ideal principal strains in Eqs. (58) and (59), however, are easier to control by controlling the temperature gradient throughout the plate thickness.

7. Concluding remarks

Algorithms based on nonlinear optimization for development of a doubly curved surface have been presented in this paper. Examples of development of a surface with all elliptical points, a surface with all hyperbolic points, and a general B-spline surface show the effectiveness of the algorithms. Compared with the available algorithms for surface development, the algorithms proposed here always find a solution that only stretching is required from curved surface to its planar development, or only shrinkage is required from planar development to the curved surface. This corresponds to forming of the surface by using (laser or torch) line heating. For other manufacturing process, the formulation of the minimization problem only needs to be slightly modified to take account of dilations from a planar shape to a curved surface.

Comparison of the two surface development methods along isoparametric lines and principal curvature directions shows no significant difference between their performance, although development along principal curvature directions gives out strain gradients which are more realizable.

The examples show that the algorithms are time-consuming when the number of grid points is large. An improvement may be possible if we explore the banded properties of the Jacobian matrix after discretization of the constraints in the first optimization and the least squares functions of the second optimization. Because of the finite difference method in approximating all derivatives, after discretization, the constraint in the first optimization or the least squares function in the second optimization only involve the variables at the neighboring points. We may also subdivide a surface into a number of subpatches and optimally develop each of them sequentially. This way, the total CPU time would be cut significantly. Of course the continuity between neighboring subpatches needs to be enforced, and the final solution may not be a global optimal solution. These are tasks for future research. In addition, application of the methods developed in manufacturing simulation is also a subject of future research.

Acknowledgements

Funding for this research was obtained in part from the New Industry Research Organization (NIRO), and from the MIT Sea Grant and Department of Ocean Engineering. The authors thank Professor J.N. Tsitsiklis and the referees for their valuable comments.

References

Azariadis, P. and Aspragathos, N. (1997), Design of plane developments of doubly curved surfaces, Computer-Aided Design 29 (10), 675–685.

Bertsekas, D.P. (1995), Nonlinear Programming, Athena Scientific, Belmont, MA.

Bertsimas, D. and Tsitsiklis, J.N. (1997), Introduction to Linear Optimization, Athena Scientific, Belmont, MA.

Cho, W., Patrikalakis, N.M. and Peraire, J. (1998), Approximate development of trimmed patches for surface tessellation, Computer-Aided Design 30 (14), 1077–1087.

Dahlquist, G. and Björck, A. (1974), Numerical Methods, Prentice-Hall, Englewood Cliffs, NJ.

do Carmo, P.M. (1976), Differential Geometry of Curves and Surfaces, Prentice-Hall, Englewood Cliffs, NJ.

Gass, S.I. (1985), Linear Programming: Methods and Applications, McGraw-Hill, New York.

Gill, P.E. and Murray, W. (1978), Algorithms for the solution of the nonlinear least-squares problem, SIAM J. Numer. Anal. 15, 977–992.

Gill, P.E., Murray, W. and Wright, A. (1981), Practical Optimization, Academic Press, New York.

Hinds, B.K., McCartney, J. and Woods, G. (1991), Pattern development for 3D surfaces, Computer-Aided Design 23 (8), 583–592.

Hoschek, J. and Lasser, D. (1993), Fundamentals of Computer Aided Geometric Design, A.K. Peters, Wellesley, MA (Translated by L.L. Schumaker).

Letcher, J.S. Jr. (1993), Lofting and fabrication of compound-curved plates, J. Ship Research 37 (2), 166–175.

Manning, J.R. (1980), Computerized pattern cutting: Methods based on an isometric tree, Computer-Aided Design 12 (1), 43–47.

Numerical Algorithms Group (1990), NAG Fortran Library Manual, Vols. 1–8, 14th edn., Mark, Oxford, England.

Scully, K. (1987), Laser line heating, J. Ship Production 3 (4), (1987), 237–246.

Struik, D.J. (1950), Lectures on Classical Differential Geometry, Addison-Wesley, Cambridge, MA.

Ueda, K., Murakawa, H., Rashwan, A.M., Okumoto, Y. and Kamichika, R. (1994), Development of computer-aided process planning system for plate bending by line heating (report 1)—relation between final form of plate and inherent strain, J. Ship Production 10 (1), 59–67.

Yu, G. (1999), Optimal development of doubly curved surfaces, Master's Thesis, Massachusetts Institute of Technology, Cambridge, MA.

Yu, G., Masubuchi, K., Maekawa, T. and Patrikalakis, N.M. (1999), A Finite Element Model for metal forming by laser line heating, in: Chryssostomidis, C. and Johansson, K. (Eds.), Proceedings of the 10th International Conference on Computer Applications in Shipbuilding, ICCAS-99, Vol. 2, MIT, Cambridge, MA, 409–418.