The Flattening of Arbitrary Surfaces by Approximation with Developable Stripes

Simon Kolmanič & Nikola Guid
University of Maribor, Slovenia

Abstract: The problem of surface flattening is an old and widespread problem. It can be found in cartography and in various industry branches. In cartography, this problem is already solved by various projections. In the industry, however, it is still present. In the article, some existent methods and the problems connected with them are given. We also present the new method based on generation of developable stripes, which can be unrolled without any distortion into the plane. In this way, the problem of overlapping of some parts of resulting flat pattern is eliminated. The overlapping problem is especially hard on the doubly curved surfaces. We show that this problem is successfully solved.

Key words: surface flattening, ruled surfaces, developable surfaces, Gaussian curvature, flat pattern

1. INTRODUCTION

The problem of deriving patterns from the 3D surface is an old and well-known problem. It is present in car, aeroplane, shipbuilding, and shoe industry. In all cases, the plane material is used for composition of some parts, like the aeroplane wings, parts of car bodies, ship hulls, and shoe uppers. It is important to have flat patterns with as little distortion as possible. Distortions like overlapping and tearing are completely eliminated only if surface is developable (Faux & Pratt, 1981). In that case, the plane material covered with flat pattern can be cut out and bent back into a 3D surface. If the surface is not developable, a flattening approximation has to be calculated to eliminate the distortions. The problem of surface flattening is present since the discovery that the Earth is round. In cartography, this problem is solved by various projections that minimize the distortion of the flattened surface (eg. Richardus & Adler, 1972). Those projections are specialized to flatten the sphere into the plane, but the flattening of the arbitrary surfaces is far more complicated.

On general surfaces, we can define three kinds of curvatures: elliptic, hyperbolic, and parabolic. For the surface-flattening problem, there are important two of them: elliptic and hyperbolic. If we can find both curvatures on the same surface, then this surface is doubly curved (Farin, 1995). Doubly curved surfaces represent the hardest problem in the surface flattening, because the distortions that arise during the flattening process are large.

Distortions generated by the flattening process can be divided into two groups: gaps and overlaps. The overlaps in 2D plane result the gaps in 3D space, which have to be stuffed with special gussets. On the other hand, the gaps in 2D plane do not represent any problems in 3D, they have to be just stitched together.

There exist many methods that tray to solve the problem of flattening of the general surface. The one of the first solutions for automated surface flattening can be found in early seventies in shoe industry (Darvill *et al.*, 1974). The method is based on *isometric trees*, whose correct mathematical definition gave Manning in (Manning, 1980). He showed that the correct solution could not always be found. From then on, many methods have been presented mostly specialized on the special surface representations or surface types. We have divided those methods into two groups according to the way of surface flattening. If the method unrolls the whole surface in one piece, the method is the member of a group of *methods for the flattening of the surface in one piece*. This group of methods is especially efficient if the surface is developable. Other methods for surface flattening try to divide the surface in the smaller patches, which can be unrolled in the plane without or only with a few distortions. Those methods are members of the group for *per partes surface flattening*.

Many of methods in the first group are working with developable surfaces (Gurunathan *et al.* 1987, Aumann 1991, Lang *et al.* 1992, Pottman *et al.* 1995). If the surface is developable, its development is in fact rotation of the points into the tangent plane around its linear generators. Only in that case, the surface can be unrolled into the plane without any distortions. If the surface is not developable, the distortions like the tearing and overlapping occur. The distortion can be reduced with the reorientation of cuts and overlaps, after the flattening process is finished (eg. Parida & Mudur, 1993). This technique can be combined with the most general method of this group, which works with triangulated surfaces and is based on the Gaussian curvature on triangulated surfaces (Calladine, 1984). The definition of

Gaussian curvature on triangulated surfaces enables quick flattening of the surface but in case of doubly curved surfaces the serious problem of gaps and overlaps is present (Hinds *et al.*, 1991).

The result is improved with the combination of flattening technique based on Gaussian curvature with reorientation technique shown in (Azariadis & Asparagathos, 1997). Another consideration on reducing of distortion is given in (McCartney *at al.*, 1999). In the group for development of surfaces in one piece, we can also find methods that flatten surfaces with the help of finite elements (Shimada & Tada, 1991).

The main idea for the methods for per partes surface flattening was presented in (Bennis *et al.*, 1991), where a new method for non-distorted texture mapping was described. The new approach of that texture mapping technique was to flatten the surface in the plane of the texture. To accomplish that on 3D surface, a starting isoparametric curve is selected. The surface is flattened in the plane as long as the distortion is in the allowed limits, so the distances between the points or distances and the geodesic curvature are preserved. If the distortion becomes too large, the flattening is stopped and a new starting curve has to be selected. Flattening results of this method are not particularly good, although the problem of overlapping is eliminated. The results are improved by the method presented in (Elber, 1995). He approximates a B-spline surface with the ruled stripes. To reduce the number of stripes, he uses the local maximums of normal curvature in the v direction. Because of that and of the use of degree raising and knot refinement of ruled stripes in order to calculate the difference between original surface and ruled stripes, the method is slow. If the surface, which has to be flattened, is a surface of revolution, then probably the fastest method for surface flattening, presented in (Hoschek, 1998), can be used. The main idea of that method is that the isoparametric curves in the u direction are circles. These circles can be fast approximated with k-gons. In this way, the stripes in the v direction are obtained, which can be flattened easily and also very fast.

2. FLATTENING OF THE ARBITRARY SURFACES USING DIVIDE-AND-CONQUER STRATEGY

If a surface is developable, it can be unrolled in the surface very easily and without any distortions. Since this is not valid if the surface is not developable in mathematically sense, our idea is to approximate the arbitrary surface with a set of developable stripes. To implement this idea, the two problems have to be solved: the generation of developable stripes and the fast procedure of surface approximation has to be found. Solutions for those problems can be found in subsections 2.1 and 2.2.

2.1 Developable Surfaces

If the Gaussian curvature K of surface $\mathbf{s}(u, v)$ is zero everywhere, the surface is developable (Faux & Pratt, 1981). Although this is characteristic property of developable surfaces, it cannot be used for construction of such a surface. To find the method for construction of the developable surface, we have to concentrate on superset of it, namely on the class of ruled surfaces. The ruled surface is defined with one or two curves (directrices), and the set of linear segments – linear generators (Gurunathan *et al.*, 1987). In this article, we concentrate on the ruled surface, defined with two directrices (see Figure 1), and show the condition when such a surface is developable.

Let $\mathbf{c}_1=\mathbf{c}_1(u)$ and $\mathbf{c}_2=\mathbf{c}_2(u')$ be two directrices (see Figure 1), where u and u' are parameters of both curves. A point P_1 on $\mathbf{c}_1$ is represented by $P_1=\mathbf{c}_1(u_1)$, $u_a < u_1 < u_b$, and point P_2 on $\mathbf{c}_2$ is represented by $P_2=\mathbf{c}_2(u_1')$, $u_a' \leq u_1' \leq u_b'$. A surface in Figure 1 is then defined by the next equation (Gurunathan *et al.*, 1987):

$$\mathbf{s}(u,u',v)=(1-v)\mathbf{c}_1(u)+v\mathbf{c}_2(u') \tag{1}$$

The ruled surface is developable if the tangent planes at all points along P_1P_2 coincide. At P_1, the tangent plane passes through P_1P_2 and the tangent to $\mathbf{c}_1$ at P_1.

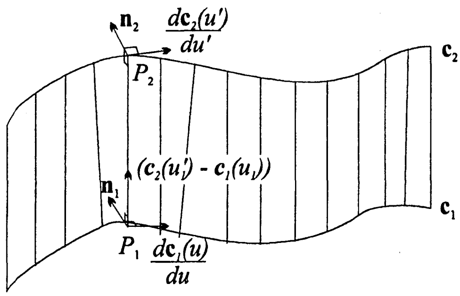


Figure 1: A ruled surface generated by two directrix curves

The normals to this tangent plane in points P_1 and P_2 are given by:

$$\mathbf{n}_1=\frac{d\mathbf{c}_1(u)}{du}\times\left(\mathbf{c}_2(u')-\mathbf{c}_1(u)\right) \quad \text{and} \tag{2}$$

$$\mathbf{n}_2 = \frac{d\mathbf{c}_2(u')}{du'} \times (\mathbf{c}_2(u') - \mathbf{c}_1(u))$$

The condition for developability is:

$$\mathbf{n}_1 \times \mathbf{n}_2 = 0 . \tag{3}$$

Eq. 3 is used for developable surface construction. The construction is started with two directrices, where one of them is selected as the primary directrix. The first point of the linear generator on the primary directrix is given by the step for moving along the primary directrix. The step for moving along the secondary directrix is much smaller. If, for example, the curve $\mathbf{c}_1$ is the primary directrix and the point P_1 is given, the proper point P_2 on the $\mathbf{c}_2$ has to be found yet. The point, that best satisfies the condition (Eq. 3) is selected as P_2. The search is faster if we consider that the generators are not allowed to intersect. The described algorithm for generation of the developable surface has been successfully tested on the various types of directrix curves. Although all testing results were good, it still could happen, that Eq. 3 could not be satisfied on some parts of the secondary directrix, if directrix curves were too heterogeneous. In this case a gap is generated in the developable surface. We can handle this situation by generating only a ruled surface or by generating a triangular patch to fill the problematic area. First method is faster and according to Elber (Elber, 1995), it represents a good solution, therefore we use it in our procedure.

2.2 Approximation of surfaces with developable stripes

Consider a surface $\mathbf{s}(u, v)$, where the parameters u and v are values from interval [0, 1]. The surface is approximated with developable stripes in the v direction, where the isoparametric curves in the u direction, $\mathbf{s}(u, v_b)$ and $\mathbf{s}(u, v_e)$, are taken as directrices. The procedure is started with $v_b = 0$ and $v_e = 1$. The developable stripe is constructed between these two isoparametric curves. The approximation is valid if the difference between the points on the original surface and on developable stripe is in the allowed limits. The distance is calculated by the following equation:

$$e = \frac{e_x + e_y + e_z}{3}, \text{ where} \tag{4}$$

$$\mathbf{e} = \frac{1}{mn}\sum_{i=0}^{n}\sum_{j=0}^{m}\left|\mathbf{s}_{ij} - \bar{\mathbf{s}}_{ij}\right|,$$

where $\mathbf{s}_{ij}$ are the points of the original surface, $\bar{\mathbf{s}}_{ij}$ are points of a developable stripe, m is number of points in the u direction, and n is the number of points in the v direction on the area defined by values v_b and v_e. If the difference between the original surface and developable stripe is larger than the given tolerance limit ε, the stripe has to be exchanged with two narrower stripes. Narrower stripes are defined with the next equation:

$$v_m = \frac{v_b + v_e}{2}, \tag{5}$$

where the first stripe is (v_b, v_m) and the second one is (v_m, v_e). The described procedure is repeated until all stripes are within the allowed limits. In Figure 2, the process of approximation of the sphere can be seen.

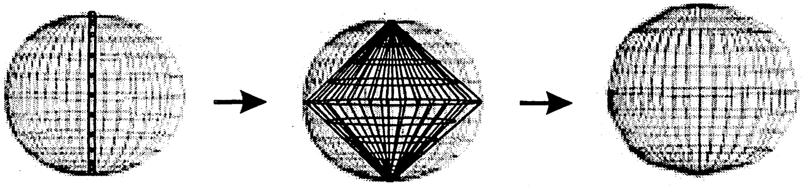


Figure 2: Approximation of the sphere with developable stripes

It is obvious that the problem of approximation is divided into two smaller identical problems that can be solved easily. This is typical for the divide-and-conquer strategy. Because of that, we will title our method as *divide-and-conquer method.* The described method can be used on all surfaces, where the nonintersecting directrices can be found. By the NURBS surfaces, we simply use isoparametric curves in the u direction. By scanned surfaces, the directrix curves are cross-section curves.

3. UNROLLING THE SURFACE INTO THE PLANE

Unrolling of the developable surface is in fact mapping of the 3D quadrangles into the plane, where the distances and the angles are preserved. The mapping process must be fast and accurate. Therefore we construct rather the quadrangles in the plane than to rotate the original 3D quadrangles into the plane.

The quadrangles with the predefined length can be constructed in many different ways, which depend of the quadrangle shape. By the Hoschek method, all quadrangles have the shape of isosceles trapezium, which can be constructed very easy and also very fast. By the flattening of arbitrary surfaces, the quadrangles are not symmetrical and therefore its construction is not so simple.

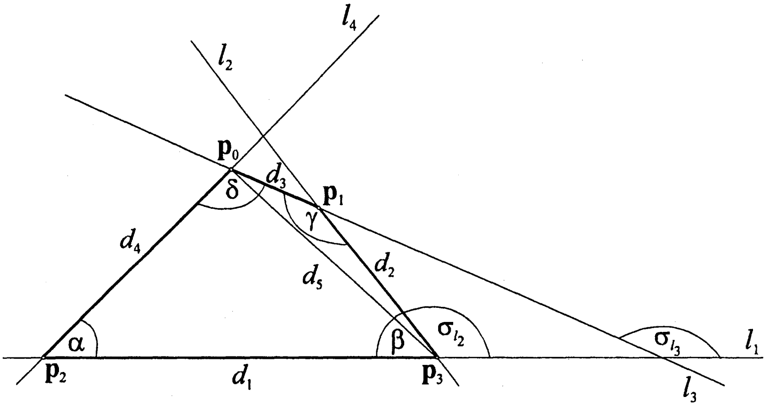


Figure 3: Construction of arbitrary quadrangle

An arbitrary quadrangle is defined by four distances d_1, d_2, d_3, and d_4, but if we want to construct such a quadrangle in the plane we also need the values of three inner angles: α, β, and γ (Figure 3). These values are calculated by the following equations:

$$\alpha = \arctan\left(\frac{r_1}{s_1 - d_5}\right),$$

$$\beta = 2\left(\arctan\left(\frac{r_1}{s_1 - d_4}\right) + \arctan\left(\frac{r_2}{s_2 - d_3}\right)\right), \tag{6}$$

$$\gamma = 2\arctan\left(\frac{r_2}{s_2 - d_5}\right),$$

where

$$s_1 = \frac{d_4 + d_1 + d_5}{2}, \tag{7}$$

$$r_1 = \sqrt{\frac{(s_1 - d_4)(s_1 - d_1)(s_1 - d_5)}{s_1}},$$

$$s_2 = \frac{d_5 + d_2 + d_3}{2} \text{ and}$$

$$r_2 = \sqrt{\frac{(s_2 - d_5)(s_2 - d_2)(s_2 - d_3)}{s_2}}.$$

After the distances and angle values are known, the construction can be started. We start to construct the quadrangle by defining the position for one of the edge points, for example the point $\mathbf{p}_2$. Beside the position of the first edge point, we can also choose the slope of the first edge, which lies on the line l_1. To reduce computational complexity, we define that the line l_1 is horizontal. In this way, the position of the point $\mathbf{p}_3$ can be calculated as $(x_2 + d_1, y_2)$. The calculation of the position of the point $\mathbf{p}_1$ is not so easy since the slope β have to be preserved. In Figure 3, it can be seen that the points $\mathbf{p}_3$ and $\mathbf{p}_1$ lie on the same line (l_2). The line l_2 is determinate by the position of the point $\mathbf{p}_3$ and the angle σ_{l_2}, which is defined by the next equation:

$$\sigma_{l_2} = \pi - \beta. \tag{8}$$

The position of the point $\mathbf{p}_1$ can be calculated by the following equation:

$$x_{1,2} = \frac{-(2kN - 2x - 2yk) \pm \sqrt{(2kN - 2x - 2yk)^2 - 4(1 + k^2)(x^2 + y^2 + N^2 - d^2 - 2y\,N)}}{2 + 2k^2}, \tag{9}$$

where $k = \tan(\sigma_{l_2})$ and $N = y_3 - kx_3$ are parameters of equation $y = kx + N$ that define the line l_2. The values x and y are the coordinate values of the point $\mathbf{p}_3 = (x_3, y_3)$. Finally, the coordinate y of the point $\mathbf{p}_1$ is calculated by the equation for the line l_2. With the use of Eq. 9, two possible solutions are generated. We take the point whose distance to point $(x_3, y_3 - d_2)$ is shorter.

The construction of the quadrangle is finished by the calculation of the position of the point $\mathbf{p}_0$. The point $\mathbf{p}_0$ lies on the line l_3 that is determinate by the position of the point $\mathbf{p}_1$ and the angle value σ_{l_3} that is calculated by the next equation:

$$\sigma_{l_3} = 2\pi - \beta - \gamma \tag{10}$$

The position of the point $\mathbf{p}_0$ can be calculated by the Eq. 9, where $k = \tan(\sigma_{l_3})$, $N = y_1 - kx_1$, and the values x and y are the coordinate values of the point $\mathbf{p}_1 = (x_1, y_1)$. By the use of Eq. 9, we again get two possible solutions. The correct one is the point whose distance to point $(x_1 - d_3, y_1)$ is smaller.

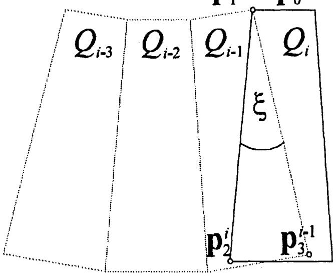


Figure 4: Preserving of the neighbouring relation by quadrangle mapping

The flattening of a quadrangle is finished after the neighbouring relation to previous flattened quadrangle is established. This can be done in two steps. First we have to translate the generated quadrangle i so that the points $\mathbf{p}_1^{i-1}$ and $\mathbf{p}_0^i$ coincide. Second the generated quadrangle has to be rotated by the angle value ξ, for edges $\mathbf{p}_3^{i-1}\mathbf{p}_1^{i-1}$ and $\mathbf{p}_2^i\mathbf{p}_0^i$ to coincide (Figure 4).

The angle value ξ is calculated by the same equation than the angle value α, where d_1 is a distance between points $\mathbf{p}_0^i$ and $\mathbf{p}_2^i$, d_5 is a distance between the points $\mathbf{p}_2^i$ and $\mathbf{p}_3^{i-1}$, d_4 is a distance between the points $\mathbf{p}_1^{i-1}$ and $\mathbf{p}_3^{i-1}$. The described algorithm for flattening of arbitrary quadrangle is used by the Elber and the divide-and-conquer method. This algorithm consumes far more time than the flattening of the quadrangles in the Hoschek method, but it is necessary if we want to flatten arbitrary surfaces.

4. EXAMPLES OF SURFACE FLATTENING

In this section, we compare the flattening results of the divide-and-conquer method, with the results of three already existing methods: the method based on Gaussian curvature (Calladine, 1984), the Elber method (Elber, 1995), and the Hoschek method (Hoschek, 1998). The first method is a member of the group for surface flattening in one piece. Other two methods are in the group for per partes surface flattening. We compare these methods on two surfaces: sphere and torus. The sphere is a typical elliptical surface. On the torus, there can be found all three curvatures: elliptical, hyperbolical, and parabolic (Farin, 1996).

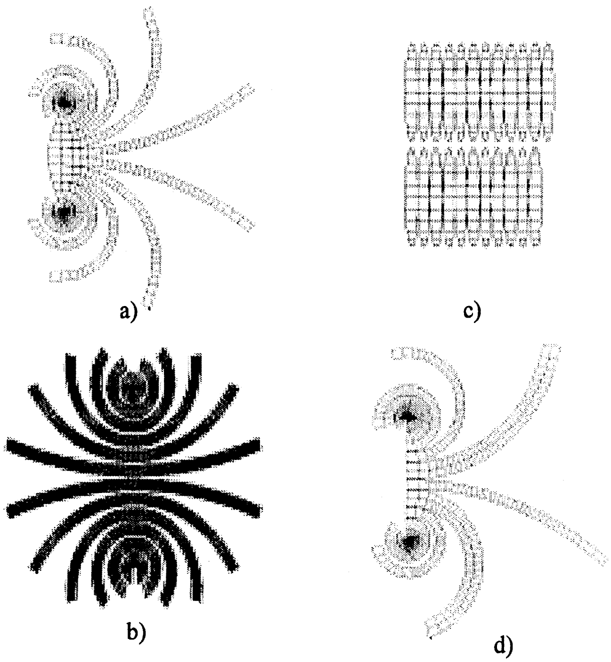


Figure 5: Flattenig of the sphere: a) with the Gaussian method, b) with the Elber method, c) with the Hoschek method, d) with the divide-and-conquer method

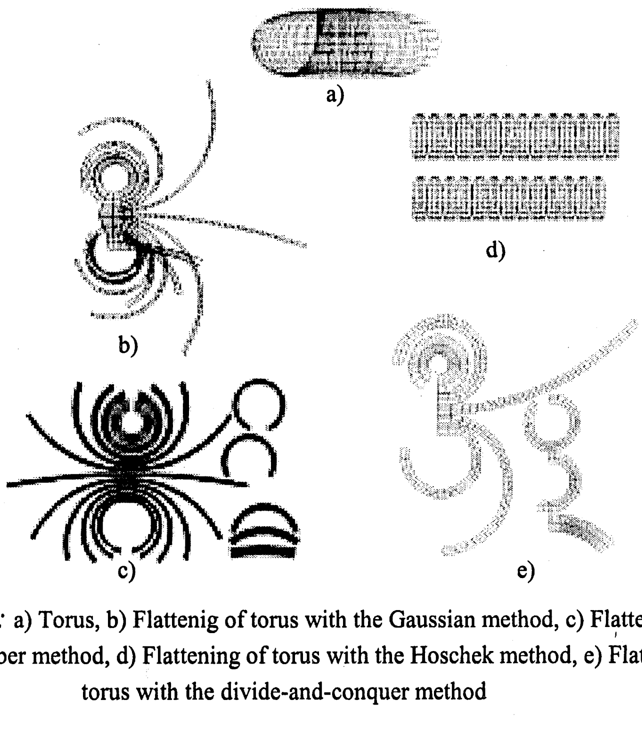
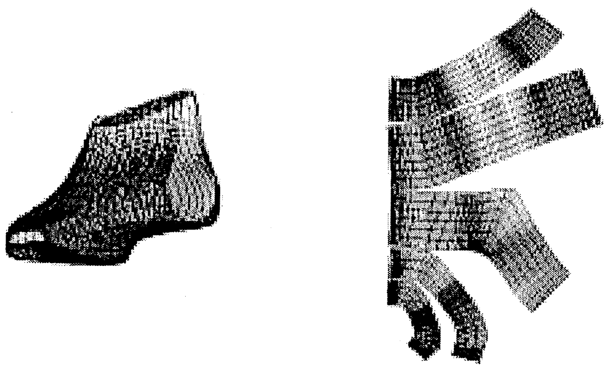


Figure 6: a) Torus, b) Flattenig of torus with the Gaussian method, c) Flattening of torus with the Elber method, d) Flattening of torus with the Hoschek method, e) Flattening of torus with the divide-and-conquer method

Because of that fact, the torus is the best testing surface that can be flattened by all four methods. The results of flattened sphere can be seen in Figure 5. The flat patterns generated by the Elber (Figure 5b) method, the Hoschek (Figure 5c) and the divide-and-conquer one (Figure 5d) consist of a set of stripes, whose number is controlled by the tolerance limit. By the method based on Gaussian curvature, shortly Gaussian method, the resulting flat pattern (Figure 5a) consists only of one piece. All flat patterns can be used to compose the original sphere. This is not the case by the flattening of the torus. In this case, the flat pattern gained by the Gaussian method (Figure 6b) is almost useless, although it can be improved by the methods presented in (Azariadis & Asparagathos, 1997). On the other hand, the flat patterns generated by the Elber (Figure 6c), the Hoschek (Figure 6d), and the divide-and-conquer method (Figure 6e) are still good.



a) b)

Figure 7: a) Shoe last obtained by 3D scanner, b) Flattening of the shoe last

The last example is surface flattening of the digitized shoe last by the divide-and-conquer method (Figure 7b). Similar results cannot be achieved by other three methods.

5. CONCLUSION

In this article, we have presented the new method for flattening of arbitrary surfaces, which is based on the divide-and-conquer strategy by the approximation of the surface with the developable stripes. The method is a member of the group of methods for per partes surface flattening. Methods of that group are very efficient if the surface is doubly curved. We have compared our method with a few existing methods. The method is not the fastest, but it is more general.

Our future work will be concentrated on elastic plane materials and the best positioning of particular stripes in order to minimize the waste of plane material. We will also consider the use of this method on scanned surfaces like the shoe lasts with style-lines to obtain useful patterns of shoe uppers.

REFERENCES

Farin, G. (1996). Curves and Surfaces for Computer - Aided Geometric Design a Practical Guide: fourth edition, Academic Press

Faux, I. D. and Pratt, M. J. (1981). Computational geometry for design and manufacture: Chichester: Ellis Harwood

Richardus, P. and Adler, R. K. (1972). Map projections for geodesists, cartographers and geographers: North-Holland, Amsterdam

Aumann, G. (1991). Interpolation with Developable Bézier Paches: Computer Aided Geometric Design, Vol. 8, 409-420

Azariadis, P. and Asparagathos, N. (1997). Design of Plane Developments of Doubly Curved Surfaces: Computer - Aided Design, Vol. 29, No. 10, 675 – 685

Bennis, C., Vezien, J. M. and Inglesias, G. (1991). Piecewise Surface Flattening for Non-Distorted Texture Mapping: Computer Graphics (Proc. Siggraph '91), Vol 25. No. 4, 237 – 246

Darvill, A .N., Williams, D. E. and Taylor, B. E. (1974). Method and Apparatus for Deriving the Net Form for Shoe Uppers: BP 1375 822 Patent Office, London

Elber, G. (1995). Model Fabrication Using Surface Layout Projection: Computer - Aided Design, Vol. 27, No. 4, 283 - 291

Gurunathan, B. and Dhande, S. G. (1987). Algorithms for Development of Certain Classes of Ruled Surfaces: Computers & Graphics, Vol. 11, No. 2, pp. 105-112

Hinds, B. K., McCarthney, J. and Woods, G. (1991). Pattern Development for 3D Surfaces: Computer - Aided Design, Vol. 23, No. 8, pp. 583 – 592

Hoschek, J. (1998). Approximation of Surface of Revolution by Developable Surfaces: Computer - Aided Design, Vol. 30, No. 10, 757-763

Lang, J. and Röschel, O. (1992). Developable (1, n)-Bézier surfaces: Computer Aided Geometric Design, Vol. 9, 291 – 298

Manning, J. R. (1980), Computerized Pattern Cutting Methods Based an Isometric Tree, Computer - Aided Design, Vol. 12, No. 1, pp. 43 - 47,

McCarthney, J., Hinds, B. K. and Seow, B. L. (1999). The flattening of triangulated surfaces incorporating darts and sussets: Computer - Aided Design, Vol. 31, No. 4, 249 – 260

Parida, L., and Mudur (1993), S. P., Constraint - satisfying Planar Development of Complex Surfaces, Computer - Aided Design, Vol. 25, No. 4

Pottman, H. and Farin, G. (1995). Developable Rational Bézier and B - Spline Surfaces: Computer Aided Geometric Design, Vol. 12, 513 – 531

Shimada, T. and Tada, Y. (1991). Approximate Transformation of an Arbitrary Curved Surface into a Plane Using Dynamic Programming: Computer - Aided Design, Vol. 23, No. 2, 153 - 159

Calladine, C. R. (1984). Gausssian Curvature and Shell Structure: Proceedings of the conference Mathematics of Surfaces, 179 - 196