Strip Approximation Using Developable Bézier Patches: A Local Optimization Approach

Chih-Hsing Chu[1*], Charlie C. L. Wang[2] and Chi-Rung Tsai[1]

[1]National Tsing Hua University, chchu@ie.nthu.edu.tw
[2]The Chinese University of Hong Kong, cwang@acae.cuhk.edu.hk

ABSTRACT

This paper proposes a geometric design method for approximation of a strip defined by two space curves with freeform developable surfaces. Given a ruling defined by two sample points taken from each curve, it calculates the feasible patches starting with the ruling. A local optimal solution is selected among them in terms of surface assessment criteria based on a heuristic algorithm. An aggregate of Bézier patches in the conical form, either triangular or quadrilateral, is thus created. Each patch is then degree elevated to gain extra degrees of freedom. They produce G^1 across the patch boundaries by modifying the control points while preserving the surface developability. Test examples with different design parameters are illustrated to validate the feasibility of the proposed method. In comparison with previous studies using the BBT (Bridge Boundary Triangulations) method, this work allows surface design with freeform developable patches, generates better results in the surface assessment, and provides more flexible control on the approximation shape. It serves as a simple but effective approach for geometric design of developable surfaces.

Keywords: Bézier patches, developable surfaces, triangular patches, quadrilateral patches, computer-aided geometric design.

1. INTRODUCTION

Developable surfaces are a subset of ruled surfaces which can be unfolded (or developed) into a plane without tearing or stretching during the process. This property, referred to as the developability, greatly eases manufacture of 3D objects. Therefore, developable shapes are widely used in many occasions such as sheet-metal forming [1], ship building [2-3], windshield design, and fabrication of apparels including shoes and clothing [4-5]. Parts are first modeled with developable surfaces in 3D space. They are then flattened into a planner pattern. The manufacturing process starts with cutting a material according to the pattern. Unrolling the cut material simply resumes its original 3D shape. A final step is sometimes applied to assemble different pieces by welding or sewing in order to form the final product.

Two approaches have been proposed for CAGD of freeform developable surfaces. First, a surface can be represented as a tensor product of degree (1, n) with non-linear constraints imposed by the developability. Designers are only allowed to specify the shape in a limited manner, e.g. some but not all of the control points. The remaining parameters must be solved from the constrained system. Previous studies employed different techniques to simplify the solution process [6-10]. However, such an approach may lack practicality in modeling of complex shapes due to the restricted degrees of freedom in the surface design [11]. Alternatively, one can treat a developable surface as an envelope of one parameter set of tangent planes. The surface thus becomes a curve in dual projective space [12]. Design methods were proposed for Bézier and B-spline surfaces based on the duality theory [13-14].

Many engineering products consist of double-curved surfaces. They are certainly not perfectly developable. In practice, it is tolerable to allow certain deviations in the developability. CAGD methods that approximate a 3D shape using developable surfaces have been introduced to address this need. Several studies [15-19] concerned with interpolation and approximation algorithms based on the dual approach. Leopoldseder and Pottmann [16] modeled a given developable surface by surfaces of revolution. Each pair of consecutive rulings and tangent planes that approximates the given surface is interpolated by smoothly linked circular cones. This work was extended to allow the input as a point cloud for applications in reverse engineering [17-18]. On the other hand, several literatures [20-21, 22-24] were focused on increasing or optimizing the developability of a surface based on its tessellation representation. Wang et al. [20] presented a method based on function optimization for increasing the developability of a trimmed NURBS surface by adjusting the positions and weights of the surface control points. The optimization process reduces the deformation

of the original patch while preserves G^0 continuity across the boundaries of the given trimmed surface patch. Tang and Wang [22] proposed a modeling algorithm that interpolates two given space curves (i.e. a strip) with an aggregate of triangles. The interpolation task was formulated as a variant of boundary triangulations [23] and thus transformed into the shortest-path problem. Their later work [24] optimized the result based on a variety of objective functions originating from different CAD/CAM applications. Shape approximation using developable surfaces also possesses a wide range of industrial applications. In addition to fabrication of sheet materials in a traditional manner, it has been recently applied to generation of tool path in five-axis flank milling [25] and simulation of robot motions [26].

This paper introduces a geometric design method for approximation of a strip with freeform developable surfaces. Given two boundary curves in space, this method generates an aggregate of Bézier patches in the conical form, consisting of triangular and quadrilateral patches. Starting with a ruling determined by two sample points from each curve, four candidate patches of degree two are calculated. Simple heuristics are applied to select one solution among them corresponding to a local optimum in terms of a surface evaluation criterion. The resultant patches connect with only positional continuity. The next step performs degree elevation on each patch to gain extra degrees of freedom. G^1 continuity is produced across the degree-elevated patches by adjusting their control polygons, whereas maintaining the surface developability. Test examples consisting of highly convoluted curves demonstrate the feasibility of our method. In comparison with the Bridge Boundary Triangulations (BBT) method [22], this work offers better surface developability in the result, simpler solution for quick implementation, and more design parameters for the shape control as well as further optimizations of the surface.

2. PRELIMINARIES

2.1 Developable Bézier Patch

Given two curves $\mathbf{P}(u)$ and $\mathbf{Q}(u)$ in 3D space, a ruled surface is constructed by linking each pair of corresponding curve points (with equal u) with a line segment $\mathbf{PQ}$, referred to as a ruling. The surface $\mathbf{R}$ is described as:

$$\mathbf{R}(t, u) = (1-t)\,\mathbf{P}(u) + t\,\mathbf{Q}(u),\ (t, u) \in [0, 1] \times [0, 1] \tag{1}$$

where t is the parameter along the ruling. Generally the tangent lines to the curves $\mathbf{P}(u)$ and $\mathbf{Q}(u)$ at any given point do not lie in the same plane. If these tangent lines and the corresponding ruling remain coplanar, then the surface becomes developable. This condition can be represented in terms of the triple scalar product of the two tangent vectors and the ruling vector $\mathbf{P}(u) - \mathbf{Q}(u)$ [9]:

$$\dot{\mathbf{P}}(u) \times \dot{\mathbf{Q}}(u) \bullet [\mathbf{P}(u) - \mathbf{Q}(u)] = 0 \tag{2}$$

Substituting the Bézier representation of both curves into Eq. (2) leads to a system of equations that must be imposed on the Bézier control points to ensure the surface developability. Previous work [9] derived the developability constraints from the de Casteljau algorithm for simplification of the solution process. For a quadratic Bézier patch shown in Fig. 1, they can be expressed as:

$$\mathbf{a_1} \bullet \mathbf{c_0} \times \mathbf{c_1} = 0 \tag{3}$$

$$\mathbf{a_2} \bullet \mathbf{c_1} \times \mathbf{c_2} = 0 \tag{4}$$

$$\mathbf{a_1} \bullet \mathbf{c_0} \times \mathbf{c_2} + \mathbf{a_2} \bullet \mathbf{c_0} \times \mathbf{c_1} = 0 \tag{5}$$

$$\mathbf{a_1} \bullet \mathbf{c_1} \times \mathbf{c_2} + \mathbf{a_2} \bullet \mathbf{c_0} \times \mathbf{c_2} = 0 \tag{6}$$

where $\mathbf{a_i} = \mathbf{P_i} - \mathbf{P_{i-1}}$ for i = 1, 2 and $\mathbf{c_j} = \mathbf{Q_j} - \mathbf{P_j}$ for j = 0, 1, 2.

The above equations reveal important geometric characteristics of the Bézier control polyhedron. Eq. (3) indicates that the first two pairs of control points must lie in the same plane. The last two pairs of control points are also coplanar because of Eq. (4). These two constraints are referred to as the co-planarity condition.

2.2 Conical Form of Developable Bézier Patch

Imposing proper geometric restrictions on a patch can sometimes simplify the developability constraints and the solution process of the constrained control points. Several previous studies [6-7, 10] applied this technique to make the surface design solvable. The resultant patches are special cases of the most general developable patch. Certainly these pre-defined limitations consume some degrees of freedom in the patch design, and thus reduce the modeling capability of the surface. When the extensions of all the trapezoids in the Bézier control polyhedron intersect at a point $\mathbf{O}$ (see Fig. 2), the first two co-planarity conditions Eqs. (3-4) will be automatically satisfied. This special case is referred to as the generalized conical form [9]. Vectors $\mathbf{c_0}$, $\mathbf{c_1}$, and $\mathbf{c_2}$ in the patch must satisfy [9]:

$$\frac{\mathbf{c}_0}{\mathbf{P}_0\mathbf{O}} = \frac{\mathbf{c}_1}{\mathbf{P}_1\mathbf{O}} = \frac{\mathbf{c}_2}{\mathbf{P}_2\mathbf{O}} = f \tag{7}$$

where f and $\mathbf{O}$ are referred to as the scaling factor and the projection point, respectively. The above equation indicates that one boundary curve is simply a scaled copy of the other curve. This conclusion also reveals one important characteristic for the control polyhedron of a Bézier patch in the conical form, i.e. any two control point pairs must remain coplanar. Any Bézier ruled patch with one boundary reduced into a single point is a triangular developable patch [9]. That is, any surface constructed by linking from a projection point to a Bézier curve becomes developable. Despite its limited modeling capability, the conical model provides simple but useful design methods in many applications of developable patches [27].

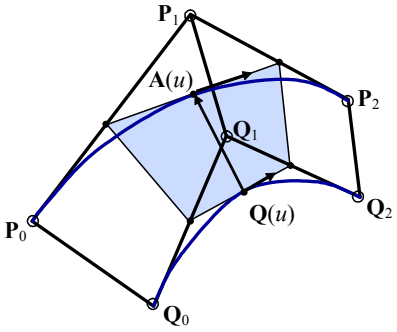


Fig. 1: Developability condition in a quadratic Bézier surface patch.

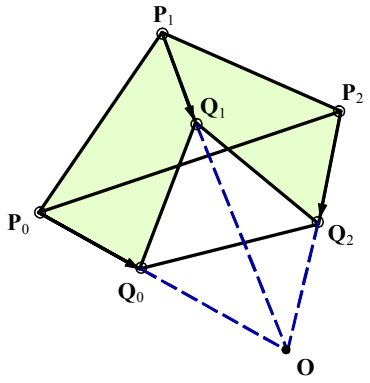


Fig. 2: A quadratic Bézier patch in generalized conical form.

3. MODELING WITH DEVELOPABLE BÉZIER PATCH

Given two boundary curves $\mathbf{P}(u)$ and $\mathbf{Q}(u)$, each will be sampled into a set of points $P=\{\mathbf{p_1}, \mathbf{p_2}, \ldots,\mathbf{p_n}\}$ and $Q=\{\mathbf{q_1}, \mathbf{q_2}, \ldots,\mathbf{q_m}\}$. To interpolate P and Q with maximal developability is theoretically a variation optimization problem [24]. This paper proposes a different approach which does not need to solve such a complex problem. Our method still takes P and Q as input data, similar to the BBT method [22]. Thus the results of both methods can be compared with each other. At any ruling $\mathbf{p_iq_i}$, there are two different elements we can choose to start with: a triangular or a quadrilateral developable Bézier patch. The following algorithms describe how to calculate the control points in each case.

3.1 Generation of a Triangular Patch

As described above, a triangular developable Bézier patch is defined with a projection point and a Bézier boundary curve. We choose the projection point and the end control points of the curve from the input point sets P and Q, i.e. they are located on the given boundaries. A curve constructed in this manner would be close proximity to the boundaries. As shown in Fig. 3, suppose $\mathbf{p_i}$ and $\mathbf{p_{i+1}}$ are two consecutive points in the set P, with the tangent vectors to the original curve denoted as $\mathbf{t_i}$ and $\mathbf{t_{i+1}}$, respectively. They are chosen to be the end control points of a quadratic Bézier curve. The second control point will be generated in two ways. The first possibility is the intersection between the tangent lines defined by $\mathbf{t_i}$ and $\mathbf{t_{i+1}}$. If they do not intersect, which is usually the case, then it is computed in a more complex manner. The middle point over the curve segment $\mathbf{p_ip_{i+1}}$ (denoted as $\mathbf{cp_i}$), $\mathbf{p_i}$, and $\mathbf{p_{i+1}}$ determine a plane $\mathbf{S}$. Let the tangent lines project into the plane, forming $\mathbf{t'_i}$ and $\mathbf{t'_{i+1}}$. Their intersection becomes the second control point. The projection point is selected from the other boundary.

3.2 Generation of a Quadrilateral Patch

Given four points on the given boundaries of the strip, as shown in Fig. 4, their corresponding boundary curve segments are denoted as $\mathbf{p_iq_i}$ and $\mathbf{p_{i+1}q_{i+1}}$. Since these four points are constructed by Step 2 in the algorithm described above, they are coplanar – thus $\mathbf{p_iq_i}$ and $\mathbf{p_{i+1}q_{i+1}}$ intersect at the projection point $\mathbf{O}$. Note that the control polygon of one boundary curve must be a scaled copy of the other in the conical form. This relation indicates that only three control points among $\mathbf{p_i}$, $\mathbf{q_i}$, $\mathbf{p_{i+1}}$, and $\mathbf{q_{i+1}}$ can be freely specified once the projection point has been chosen. Thus the end ruling may intersect the boundary at a different point $\mathbf{q_{i+1}'}$ from $\mathbf{q_{i+1}}$, given that $\mathbf{p_i}$, $\mathbf{q_i}$, and $\mathbf{p_{i+1}}$ as the chosen control points. The second control point $\mathbf{mp_i}$ of the longer curve segment is calculated from $\mathbf{p_i}$ and $\mathbf{p_{i+1}}$ with the same heuristic adopted in the generation of a triangular patch. Once it has been obtained, the remaining control point $\mathbf{mq_i}$ of the shorter segment is then determined by Eq (3).

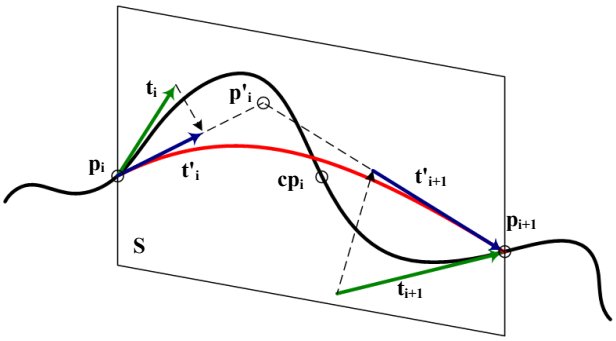


Fig. 3: Generation of the control polyhedron for a triangular developable Bézier patch.

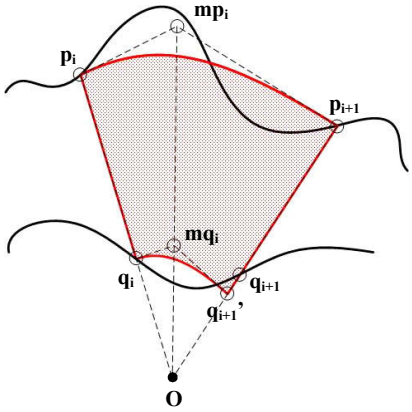


Fig. 4: Generation of the control polyhedron for a quadrilateral developable Bézier patch.

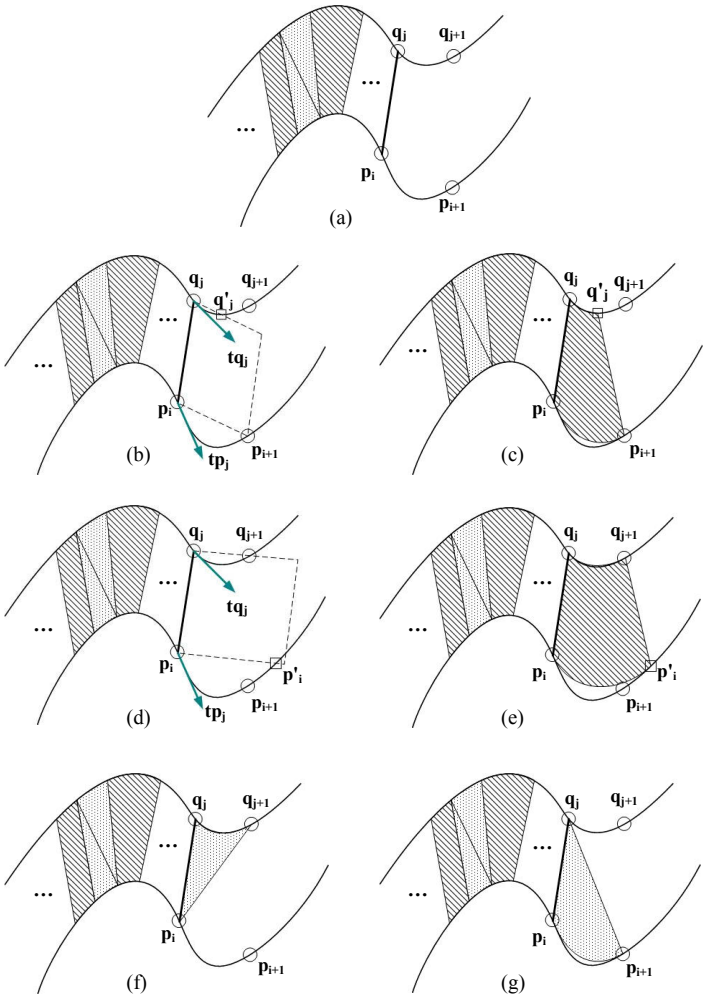


Fig. 5: Generation of a developable patch at a ruling.

3.3 Algorithm for Generation of Developable Patches in the Conical Form

Step 1: Generate point sets from the boundary curves

This step is to divide the given boundary curves into two point sets with equal arc length. The number of points in *P* and *Q* does not have to be equal.

Step 2: Test feasibility of constructing a quadrilateral patch

The proposed method adopts a heuristic in the approximation, i.e. we prefer use of a quadrilateral patch, Following the same idea, this step checks whether a quadrilateral patch can be constructed in the first place, even though a triangular one always exists. Fig. 5(a) shows that $\mathbf{p_iq_j}$ is a ruling to start with. The tangent vectors to the curves at these points are $\mathbf{tp_i}$ and $\mathbf{tq_j}$. If they lie on the same side of the triangle $\mathbf{p_ip_{i+1}q_j}$, then go to **Step 3** for testing other conditions that must be satisfied in construction of a developable patch. The similar test is carried out when both vectors remain on one side of the triangle $\mathbf{p_iq_jq_{j+1}}$. A triangular patch will be considered only when those vectors are located on different sides of the two triangles.

Step 3: Test the construction feasibility of a developable patch

Given a ruling $\mathbf{p_iq_j}$, there are four different ways in constructing the next patch. First, we can choose a triangular or a quadrilateral patch. Moreover, the projection point of each patch is likely to be located on either curve. The potential

solutions are referred to as candidate patches in this paper. Their corresponding construction procedures are described as follows:

(1) Quadrilateral patch formed by $\mathbf{p_i p_{i+1} q_j}$: we calculate the intersection between the plane determined by $\mathbf{p_i}$, $\mathbf{p_{i+1}}$, and $\mathbf{q_j}$ and the boundary curve $\mathbf{Q}$. The one nearest to $\mathbf{q_j}$ (denoted as $\mathbf{q'_j}$) is chosen when there are multiple solutions, as shown in Fig. 5(b). If the distance between $\mathbf{q'_j}$ and $\mathbf{q_j}$ is smaller than the Hausdorff distance specified by the user, then $\mathbf{p_i}$, $\mathbf{p_{i+1}}$, and $\mathbf{q_j}$ allow the construction of a quadrilateral developable patch. These points cannot form a quadrilateral developable patch when there is no intersection or the limitation of the distance is not satisfied.

(2) Quadrilateral patch formed by $\mathbf{p_i q_j q_{i+1}}$: we calculate the intersection between the plane determined by $\mathbf{p_i}$, $\mathbf{q_i}$, and $\mathbf{q_{j+1}}$ and the boundary curve $\mathbf{P}$. The one nearest to $\mathbf{p_j}$ (denoted as $\mathbf{p'_j}$) is chosen when there are multiple solutions, as shown in Fig. 5(d). If the distance between $\mathbf{p'_j}$ and $\mathbf{p_j}$ is smaller than the Hausdorff distance specified by the user, then $\mathbf{p_i}$, $\mathbf{q_i}$, and $\mathbf{q_{j+1}}$ allow the construction of a quadrilateral developable patch.

(3) Triangular patch formed by $\mathbf{p_i q_j q_{i+1}}$: the next point $\mathbf{q_{i+1}}$ on the curve $\mathbf{Q}$ is chosen to form a triangular developable patch along with the ruling $\mathbf{p_i q_j}$, as shown in Fig. 5(f), with pi as the projection point. It is necessary to check the distance between $\mathbf{p_i}$ and $\mathbf{q_{i+1}}$ with respect to the Hausdorff distance.

(4) Triangular patch formed by $\mathbf{p_i p_{i+1} q_j}$: the next point $\mathbf{p_{i+1}}$ on the curve $\mathbf{P}$ is chosen to form a triangular developable patch along with the ruling $\mathbf{p_i q_j}$, as shown in Fig. 5(g), with $\mathbf{q_i}$ as the projection point.

Step 4: Choose an optimal solution from the candidate patches

Step 3 may generate more than one feasible patch starting with $\mathbf{p_i q_j}$. This step helps select an optimal solution among them. Since a quadrilateral patch has higher preference, it surpasses both triangular patches. If there are two feasible quadrilateral patches (with the projection point on the different sides), they are evaluated by a given criterion. The one corresponding to a smaller value will be selected. The similar evaluation process is applied to choose between two triangular patches.

Step 5: Repeat Steps 2 to 4 until the end ruling reaches $\mathbf{p_m q_n}$.

4. CONTINUITY ADJUSTMENT WITH DEGREE ELEVATION

Any two consecutive patches generated using the above algorithms only guarantee positional continuity across the patch boundary. This section will introduce a technique to improve the continuity of the approximation result. The main idea is to gain additional degrees of freedom for finer adjustment of a Bézier patch through degree elevation [11]. These extra free parameters not only allow the continuity adjustment, but they can be also used to maintain the surface developability simultaneously. The principle of degree elevation for a Bézier patch will be discussed as follows.

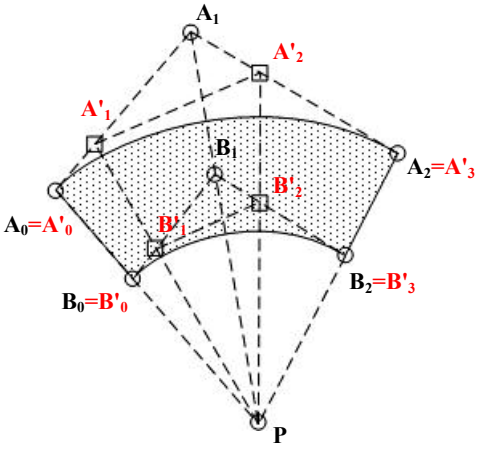


Fig. 6: Degree elevation for a quadratic Bézier ruled surface.

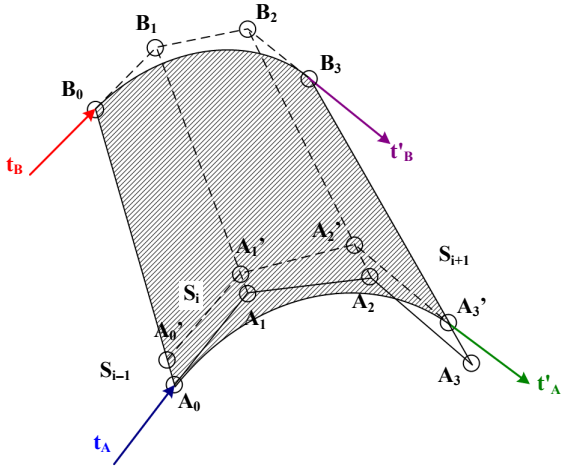


Fig. 7: Continuity adjustment for a quadrilateral patch.

4.1 Degree Elevation

Degree elevation increases the degree of a curve without changing the shape of the curve. This technique is often used for combining two curves with different degrees, which helps simplify the complexity of geometric processing. Given a

Bézier curve of degree n with control points $\mathbf{p}_0, \mathbf{p}_1, ..., \mathbf{p}_n$. The control points $\mathbf{p'}_i\ (i = 0, ..., n+1)$ for the Bézier curve of degree (n+1) can be written as [28]:

$$\mathbf{p}_0' = \mathbf{p}_0,\ \ \mathbf{p}_{n+1}' = \mathbf{p}_n$$
$$\mathbf{p}_i' = \frac{i}{n+1}\mathbf{p}_{i-1} + \left(1 - \frac{i}{n+1}\right)\mathbf{p}_i,\quad i = 1,\ 2,\ ....,n \tag{8}$$

For a quadratic Bézier ruled surface with $\mathbf{A_0}$-$\mathbf{A_1}$-$\mathbf{A_2}$ and $\mathbf{B_0}$-$\mathbf{B_1}$-$\mathbf{B_2}$ as the control points of its boundary curves (see Fig. 6), the new control points for the same surface of degree three become:

$$\mathbf{A'_0} = \mathbf{A_0},\ \mathbf{A'_4} = \mathbf{A_3},\ \mathbf{B'_0} = \mathbf{B_0},\ \mathbf{B'_4} = \mathbf{B_3}$$
$$\mathbf{A}_1' = \frac{1}{3}\mathbf{A}_0 + \frac{2}{3}\mathbf{A}_1,\ \ \mathbf{A}_2' = \frac{2}{3}\mathbf{A}_1 + \frac{1}{3}\mathbf{A}_2,\ \ \mathbf{B}_1' = \frac{1}{3}\mathbf{B}_0 + \frac{2}{3}\mathbf{B}_1,\ \ \mathbf{B}_2' = \frac{2}{3}\mathbf{B}_1 + \frac{1}{3}\mathbf{B}_2 \tag{9}$$

A cubic patch degree-elevated from a quadratic developable patch still preserves the developability of the surface, since the elevation process only changes the parameterization of the surface, not its shape. For a patch in the conical form, the projection point remains the same and each control point pair satisfies the scaled relationship. However, there are more design handles in the new representation of the surface that can be utilized to modify the control points with finer shape control. The next section introduces a set of algorithms that generate G^1 continuity across the patch boundaries based on this idea.

4.2 Algorithms for Continuity Adjustment with Degree Elevation

Continuity Adjustment for Quadrilateral Patch

Fig. 7 shows the control points of a quadrilateral patch involved in the adjustment process. Initially, the patch $\mathbf{S_i}$ connects to $\mathbf{S_{i-1}}$ and $\mathbf{S_{i+1}}$ along the rulings $\mathbf{A_0B_0}$ and $\mathbf{A_3B_3}$, respectively, both with positional continuity. For $\mathbf{S_{i-1}}$, the tangent vectors to the boundary curves at $\mathbf{A_0}$ and $\mathbf{B_0}$ are $\mathbf{t_A}$ and $\mathbf{t_B}$, which has been determined when processing the patch $\mathbf{S_{i-1}}$ (i.e., they are fixed and coplanar – when $\mathbf{S_i}$ is a starting patch, we simply assign the average of $\mathbf{t_A}$ and $\mathbf{t_B}$ as the strip tangent). To achieve G^1 across $\mathbf{A_0B_0}$, we must let (1) $\mathbf{A_1}$ lies in the direction of $\mathbf{t_A}$, and (2) $\mathbf{B_1}$ lies in the direction of $\mathbf{t_B}$.

To achieve G^1 across $\mathbf{A_3B_3}$ is slightly tricky. Instead of the original control polygon, a scaled copy (denoted as $\mathbf{A_0'}$-$\mathbf{A_1'}$-$\mathbf{A_2'}$-$\mathbf{A_3'}$) contains the last control point $\mathbf{A_3'}$ located on the boundary. We have $\mathbf{A_2A_3}$//$\mathbf{A_2'A_3'}$ in the conical form. Thus, G^1 requires (1) $\mathbf{A_2}$ (or $\mathbf{A_2'}$) lies in the direction of $\mathbf{t'_A}$, and (2) $\mathbf{B_2}$ lies in the direction of $\mathbf{t'_B}$. To determine the control points on $\mathbf{S_{i+1}}$, $\mathbf{t'_A}$ and $\mathbf{t'_B}$ are chosen as the average of the original tangent vectors on the boundary curves at $\mathbf{A_3}$ (or $\mathbf{A_3'}$) and $\mathbf{B_3}$. Then, we have $\mathbf{A_1} = \mathbf{A_0} + w_1\mathbf{t_A}$ and $\mathbf{A_2} = \mathbf{A_3} - w_2\mathbf{t'_A}$. They also specify the positions of $\mathbf{B_1}$ and $\mathbf{B_2}$. After preserving the G^1 continuity and the developability, we still have 2-DOF for further adjustment of the shape of $\mathbf{S_i}$, i.e. w_1 and w_2. $\mathbf{t'_A}$ and $\mathbf{t'_B}$ act as the starting tangent vectors for the next patch $\mathbf{S_{i+1}}$. Note that all tangent vectors are normalized.

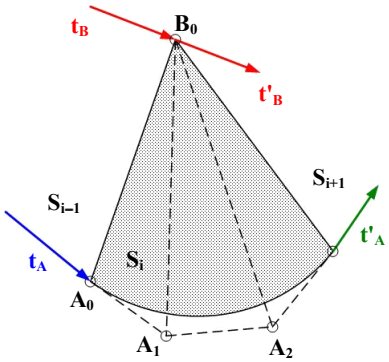


Fig. 8: Continuity adjustment for a triangular patch.

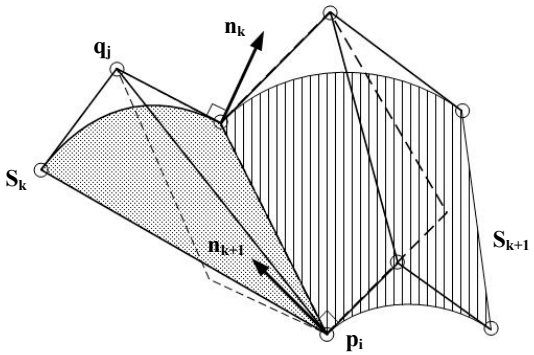


Fig. 9: Calculation of normal variation across two consecutive patches.

Continuity Adjustment for Triangular Patch

When processing a triangular patch $\mathbf{S_i}$, the situation becomes more complex than the quadrilateral patch – we need to analyze the possible configurations on the next patch $\mathbf{S_{i+1}}$. As shown in Fig. 8, if only considering about the

developability on the triangular patch $\mathbf{S_i}$, the tangent vectors $\mathbf{t_A}$, $\mathbf{t_B}$, $\mathbf{t'_A}$ and $\mathbf{t'_B}$ do not have to be coplanar. However, we need to take into account the developability constraints imposed by the next patch $\mathbf{S_{i+1}}$ at the same time when choosing them. In detail,

- When $\mathbf{S_{i+1}}$ is a quadrilateral patch, we must have $\mathbf{t'_A}//\mathbf{t'_B}$ so that $\mathbf{S_{i+1}}$ becomes developable. Meanwhile, we need to have $\mathbf{t_B}//\mathbf{t'_B}$ so that the G^1 continuity is preserved at $\mathbf{B_0}$. Therefore, in this configuration, the tangent vectors on the end ruling must satisfy $\mathbf{t'_B}//\mathbf{t_B}$ and $\mathbf{t'_A}//\mathbf{t'_B}$.
- When $\mathbf{S_{i+1}}$ is a triangular patch with the projection point at $\mathbf{B_0}$ (or $\mathbf{A_3}$), we give $\mathbf{t_B}//\mathbf{t'_B}$ so that the G^1 continuity is given at $\mathbf{B_0}$. Then, $\mathbf{t'_A}$ is assigned to follow the boundary curve tangent of the strip at $\mathbf{A_3}$.

After determining the directions of the tangent vector on the start and end rulings of $\mathbf{S_i}$, the control points $\mathbf{A_1}$ and $\mathbf{A_2}$ can be computed by $\mathbf{A_1} = \mathbf{A_0} + w_1\mathbf{t_A}$ and $\mathbf{A_2} = \mathbf{A_3} - w_2\mathbf{t'_A}$. In this case, we still have 2-DOF to adjust the shape of $\mathbf{S_i}$ after preserving G^1 and developability.

5. IMPLEMENTATION RESULTS

5.1 Surface Evaluation Criteria

Given a pair of spatial curves, different approximation results can be generated and all interpolate the same curves. They should be accessed quantitatively by some criteria depending on specific applications. These criteria also work as optimization objectives in the design of the approximation algorithms. For example, one optimization objective could be minimal area, i.e. the resultant patch set has the minimal surface area among all the solutions, the co-called Plateau's problem [29]. Other optimization objectives include maximal developability, minimal bending energy, and minimal normal variation [24]. This paper will adopt normal variation and bending energy as the major objectives to be minimized in the surface construction process.

Fig. 9 illustrates two consecutive patches connecting along the ruling $\mathbf{p_iq_j}$. Since the both patches are in the conical form, any two pairs of the control points must be co-planar. The corresponding normal vectors of $\mathbf{S_k}$ and $\mathbf{S_{k+1}}$ at the ruling are $\mathbf{n_k}$ and $\mathbf{n_{k+1}}$, respectively. The normal variation (or normal twist) across $\mathbf{p_iq_j}$ can be expressed as:

$$N_v(S_k, S_{k+1}) = 1 - n_k \cdot n_{k+1} \tag{10}$$

It becomes null when the two vectors are in the same directions. The total normal variation of the approximation result consisting of M patches becomes:

$$N_T(M) = \sum_{k=0}^{M-1} N_v(S_k, S_{k+1}) \tag{11}$$

Note that the calculation of the normal variation is regardless of the patch type (triangular or quadrilateral).

Strain energy was considered a good objective for functional optimization of surface fairness [30]. It gives an integral measure of the surface curvature. This study employs bending energy as one objective that is locally minimized in the construction process of the surface. It can be simplified into the following form along a ruling shared by two successive patches with a small bending angle:

$$U_k = K\frac{A\sin^2\theta}{L^2} \tag{12}$$

where K is a coefficient determined by the thickness of surface and the Young's modulus [22]. As shown in Fig. 10, L is the moment arm of the patch $\mathbf{S_k}$ with respect to the rotation axis $\mathbf{p_iq_j}$. It is computed as the maximal perpendicular distance from the patch to the axis. θ is the angle extended by the tangent planes of $\mathbf{S_k}$ and $\mathbf{S_{k+1}}$ at the ruling. A is the surface area of $\mathbf{S_{k+1}}$. K is set to one for simplification purpose. The total bending energy containing in an aggregate of M developable patches can be written as:

$$U_B(M) = \sum_{k=0}^{M-1} U_k \tag{13}$$

5.2 Test Results

This section presents a number of examples to validate the feasibility of the proposed method. Different input parameters are examined to characterize their individual effects on the result. The same boundary curves employed by the previous work [24] will be used for comparison purpose. The input parameters include the optimization objective (O_L), the numbers of sample point on the boundaries (N_p/N_q), the Hausdorff distance for triangular (H_3) and quadrilateral patches (H_4), and the maximal length ratio between the boundaries of quadrilateral patch (r_a). The output properties include the surface evaluation value, the number of triangular patches n_3, and the number of quadrilateral patches n_4. NT and BE denote "normal variation" and "bending energy" respectively in the results.

Fig. 11 illustrates the test result based on the BBT method with the same input parameters as Fig. 12 except r_a, which the method does not offer. Fig. 12(a) demonstrates the patch set generated by the current method, which consists of

33 triangular and 39 quadrilateral patches. The planar pattern unfolded from the patches is shown in Fig. 12(b). Our method outperforms the BBT method in the total normal variation. The second strip to be interpolated consists of two highly convoluted curves. Fig. 13 is the test result generated by the BBT method. Fig. 13 shows the approximation result of the current method based on optimization of bending energy. It contains only four quadrilateral patches, as the curvature varies radically along the curves. Note that the distribution of the triangular patches is different from the one shown in Fig. 14. The current method still produces a slightly better result in terms of bending variation.

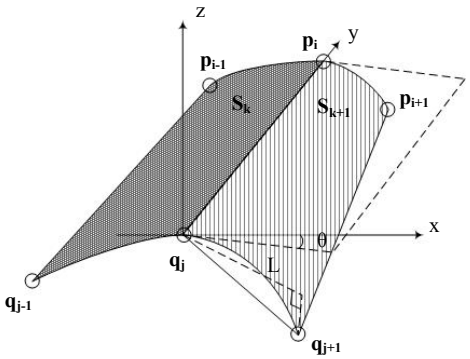


Fig. 10: Calculation of bending energy across two consecutive patches.

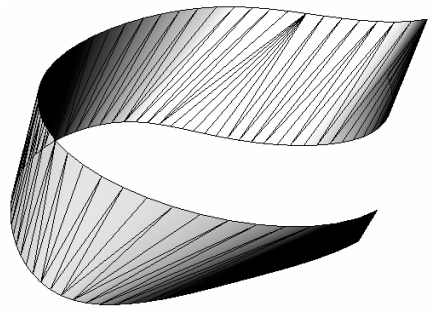
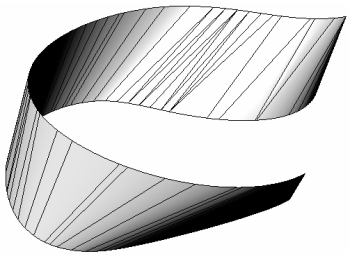
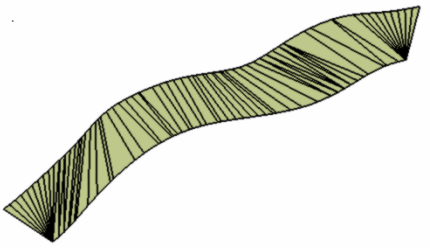


Fig. 11: The first test result of the BBT method [24].



(a)

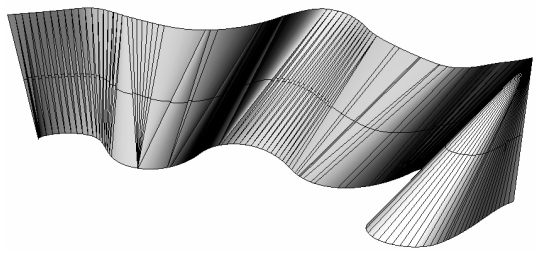


(b)

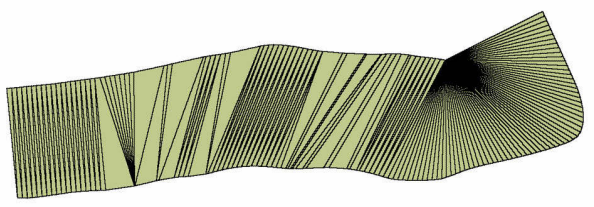
Input Parameter					
O_L	N_p	N_q	H_3	H_4	r_a
NT	55	65	4	4	1.2

Current Method			BBT Method
n_3	n_4	N_T	N_T
33	39	0.128747	5.3

Fig. 12: The first test result based on optimization of normal variation.



(a)



(b)

Input Parameter					
O_L	N_p	N_q	H_3	H_4	r_a
BE	144	91	4	10	1.5

Current Method			BBT Method
n_3	n_4	U_B	U_B*
200	4	80.5874	77.0

Fig. 13: The second test example based on optimization of bending energy.

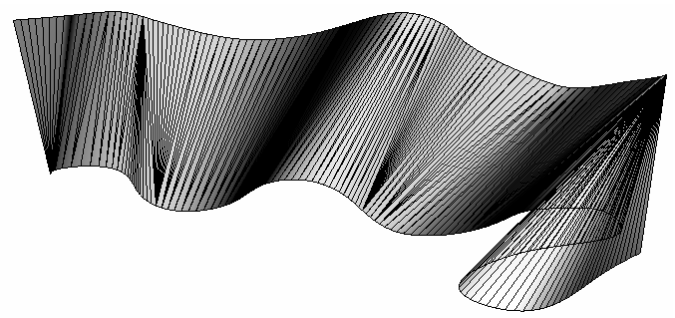


Fig. 14: The second test result of the BBT method [24].

6. CONCLUSIONS AND FUTURE WORK

This paper presents a new method that interpolates a strip specified by two space curves with developable patches in the conical form. The computation procedure contains two steps. The first step calculates the feasible patches that connect to a given ruling defined by two points on each curve. A heuristic is proposed to select a local optimal solution among them in terms of a surface assessment criterion. The result consists of consecutive developable Bézier patches, triangular or quadrilateral, with positional continuity across the patch boundaries. Degree elevation is next conducted on each patch to produce extra degrees of freedom. Geometric algorithms are provided for adjusting the control points of the patches with these design parameters. They allow G^1 continuity across the patches, preserve the developability of the surfaces, and maintain their proximity to the boundary curves at the same time. Numerous examples are generated using different input parameters and objectives. They demonstrate that our method outperforms the BBT (Boundary Bridge Triangulations) method previously developed.

This work provides a simple but effective method for interpolating two boundary curves using developable surfaces. In comparison with previous studies, it allows approximation of a strip with freeform developable patches, better evaluation values of surface, and more flexible shape control of the result. One major advantage is the ability of using triangular and quadrilateral patches simultaneously in the surface design. The former fits well in the highly convoluted areas whereas the latter approximates smooth regions with fewer patches. Another benefit is being able to perform local shape adjustment via degree elevation. A complete (or nearly) developable shape can be thus obtained.

One major limitation of this work is that the result is not a global optimum (if it exists). A possible solution is to couple with some global search scheme (e.g. Genetics Algorithm) that computes the approximation patches using the proposed method in one iteration. Another important issue is to fully utilize the extra degrees of freedom produced by degree elevation. They allow multiple objectives or more complex objectives in the surface design. For instance, several strips can be interpolated simultaneously with developable patches for modeling intricate shapes. Such an approach can significantly enhance the practicality of developable surfaces in product realization. Our future research is focused on this.

7. REFERENCES

[1] Mancewicz, M. J.; Frey, W. H.: Developable surfaces: properties, representations and methods of design, GM Research Publication GMR-7637, 1992.

[2] Lamb, T.: Shell development computer aided lofting – Is there a problem or not? Journal of Ship Production, 11(1), 1995, 34-46.

[3] Norlan, T. J.: Computer aided design of developable surfaces, Marine Technology, 8, 1971, 233-242.

[4] Hinds, B. K.; McCartney, J.; Woods G.: Pattern development for 3D surfaces, Computer-Aided Design, 23(8), 1991, 583-592.

[5] Wang, C. C. L.; Wang, Y.; Yuen, M. M. F.: Design automation for customized apparel products, Computer-Aided Design, 37(7), 2005, 675-691.

[6] Aumann, G.: Interpolation with developable Bézier patches, Computer Aided Geometric Design, 8, 1991, 409-420.

[7] Lang, J.; Röschel, O.: Developable (1, n)-Bézier surfaces, Computer Aided Geometric Design, 9, 1992, 291-298.

[8] Maekawa, T.; Chalfant, J. S.: Design and tessellation of B-spline developable surfaces, ASME Transaction Journal of Mechanical Design, 120, 1998, 453-461.

[9] Chu, C. H.; Séquin, C. H.: Developable Bézier patches: properties and design, Computer-Aided Design, 34(7), 2002, 511-527.
[10] Chu, C. H.; Chen, J. T.: Geometric design of developable composite Bézier surfaces, Computer Aided Design and Applications, 1(3), 2004, 531-540.
[11] Aumann, G.: Degree elevation and developable Bézier surfaces, Computer Aided Geometric Design, 20, 2004, 661-670.
[12] Pottmann, H.; Wallner, J.: Computational Line Geometry, Springer-Verlag, 2001.
[13] Bodduluri, R. M. C.; Ravani, B.: Geometric design and fabrication of developable Bézier and B-spline surfaces, ASME Transactions Journal of Mechanical Design, 116, 1994, 1024-1048.
[14] Bodduluri, R. M. C.; Ravani, B.: Design of developable surfaces using duality between plane and point geometry, Computer-Aided Design, 25, 1995, 621-632.
[15] Randrup, T.: Approximation of surfaces by cylinders, Computer-Aided Design, 30, 1998, 807–812.
[16] Leopoldseder, S.; Pottmann, H.: Approximation of developable surfaces with cone spline surfaces, Computer-Aided Design 30, 1998, 571–582.
[17] Pottmann, H.; Randrup, T.: Rotational and helical surface approximation for reverse engineering, Computing, 60, 1998, 307–323.
[18] Chen, H. Y.; Lee, I. K.; Leopoldseder, S.; Pottmann, H.; Randrup, T.; Wallner, J.: On surface approximation using developable surfaces, Graphical Models and Image Processing, 61, 1999, 110-124.
[19] Pottmann, H.; Wallner, J.: Approximation algorithms for developable surfaces, Computer Aided Geometric Design, 16, 1999, 539-556.
[20] Wang, C. C. L.; Wang, Y.; Yuan, Y. F. M.: On increasing the developability of a trimmed NURBS surface, Engineering with Computers, 20, 2004, 54-64.
[21] Wang, C. C. L. and Tang, K., Achieving developability of a polygonal surface by minimum deformation: a study of global and local optimization approaches, The Visual Computer, 20, 521-539, 2004.
[22] Tang, K.; Wang, C. C. L.: Modeling developable folds on a strip, Journal of Computing and Information Science in Engineering, 5, 2005, 35-47.
[23] Frey, W. H.: Boundary triangulations approximating developable surfaces that interpolate a closed space curve, 13(2), 2002, 285–302.
[24] Tang, K.; Wang, C. C. L.: Optimal boundary triangulations of an interpolating ruled surface, Journal of Computing and Information Science in Engineering, 5, 2005, 291-301.
[25] Chu, C. H.; Chen, J. T.: Automatic tool path generation for 5-axis flank milling based on developable surface approximation, International Journal of Advanced Manufacturing Technology, 29(7-8), 2006, 707-713.
[26] Lee, J. H.: Modeling generalized cylinders using direction map representation, Computer-aided Design and Applications, 1(3), 2004, 541-550.
[27] Chu, C. H.; Chen, J. T.: Characterizing design degrees of freedom for composite developable Bézier surfaces and their applications in design and manufacturing, Robotics & CIM, 23(1) , 2007, 116-125.
[28] Farin, G.: Curves and Surfaces for Computer Aided Geometric Design, Academic Press, 1997.
[29] Monterde, J.: Bézier surfaces of minimal area: the Dirichlet approach, Computer Aided Geometric Design, 21, 2004, 117–136.
[30] Moreton, H. P.; Séquin, C. H.: Functional optimization for fair surface design, SIGGRAPH, 1992, 167-176.