

T-Spline Polygonal Complexes

Abdulwahed M. Abbas

The University of Balamand, abbas@balamand.edu.lb

ABSTRACT

The notion of polygonal complexes was originally conceived as a means for exact interpolation of uniform B-spline curves by Doo-Sabin (and later on by Catmull-Clark) subdivision surfaces. Starting from the theoretical origin of these complexes, this paper provides a general formulation of this notion that covers all quad-based (uniform/non-uniform) B-spline as well as NURBS surfaces. This formulation is generalized even further to cope with the extra-requirements brought about in the context of T-spline surfaces while, at the same time, maintaining previous formulations as particular instances of that.

Keywords: B-splines, polygonal complexes, subdivision, NURBS, T-spline surfaces.

1. INTRODUCTION

Interpolation (of a point by a curve or a surface, as well as of a curve by a surface) and, equivalently, the notion of *skinning*, have proved to be of considerable benefit in geometric modeling, and has never ceased to find many sensitive and interesting applications in science, engineering and technology. This made interpolation a subject of focus of research for a few decades now (see [16, 17] for a survey and other references therein).

Interpolation is traditionally achieved through the solution of the typical system of linear equations with many variables [6, 19]. However, the drive for efficiency led researchers to bordering techniques such as fitting [7, 26] and approximation [20]. By comparison, the focus this paper will be on exact interpolation.

Thus, this notion has been visited by so many researchers in so many different modeling contexts to the degree that there is no further need to emphasize its importance to modeling. This argument gains even more force when raised in the context of exact interpolation as opposed to approximate interpolation (or fitting).

Against this background, the earliest notion of polygonal (strip) complexes was proposed by Nasri [18] as a means of interpolating uniform quadratic B-spline curves by Doo-Sabin subdivision surfaces [5]. Later on, Nasri [15] redeployed this notion to also support the interpolation of uniform cubic B-spline curves by Catmull-Clark subdivision surfaces [4].

As suggested above, the initial motivation behind a polygonal complex is that, *under the corresponding subdivision scheme, it admits a B-spline limit curve of the same degree* (see Fig. 1 and Fig. 2). Thus, when a complex is embedded within a polygonal mesh, its limit curve is *automatically* interpolated by the surface limit of subdivision of the polygonal mesh, without the need for any additional overheads.

More interestingly, as suggested in the previous paragraph, nothing prevents the extension of this notion to any other subdivision scheme [9].

This notion was employed later, under well-specified constraints [3], for the interpolation of any arbitrarily intersecting network of curves by a subdivision surface (see Fig. 3). This works for Doo-Sabin [13] and Catmull-Clark [3] as well as for Loop [8] subdivision schemes.

Furthermore, polygonal complexes can also serve as local neighborhoods for holding information useful for exercising further control over the limit surface; e.g., normal direction and local curvature constraints (see [12] for an illustration of that).

Before delving into the main body of this paper, it is important to emphasize that the *necessity* of this research may be further justified by simply comparing the immediacy and ease with which the interpolation effects are obtained with the help of polygonal complexes against previously employed methods.





Fig. 1: Doo-Sabin Polygonal strip Complex and its limit curve: (a) Initial Complex, (b) One Subdivision Step and (c) Limit Curve.



Fig. 2: Catmull-Clark Polygonal strip Complex and its limit curve: (a) Initial Complex, (b) One Subdivision Step and (c) Limit Curve.



Fig. 3: Interpolating an intersecting network of curves by a subdivision surface: (a) Doo-Sabin (b), Catmull-Clark and (c) Loop.

This is in addition to other benefits (mentioned in the previous paragraph) that are gained from the ability to embed such complexes inside the control mesh of the surface being modeled.

The rest of this paper is structured as follows. Section 2 introduces preliminary definitions intended to make this paper more self-contained. This also lays the grounds for the derivations conducted in later sections of this paper. Section 3 provides a specification of the interpolation task which forms the basis of the formulation of polygonal complexes presented in section 4.

Section 5 shows that the original notion of polygonal complexes for subdivision surfaces is just an instance of this formulation, which is extended to also cover the NURBS domain. Section 6 generalizes this formulation even further to suit the requirements of the T-spline domain, and section 7 concludes the paper and provides some suggestions for further work.

2. PRELIMINARIES

The historical perspective drawn in the paper getting back to B-spline surfaces (especially the factorization hint drawn in section 2.3) provides a clue as to why a polygonal complex represents a curve interpolated by the respective surface. This hint applies equally well to NURBS and serves as an indispensable framework that guides the development of this idea in the T-spline domain.

2.1. B-Spline Curves

Given a sequence of control points $[p_0, p_1, ..., p_m]$ and a non-decreasing sequence of knots $[t_0, t_1, ..., t_n]$, a Bspline curve [21] of degree *p* is defined as follows:

$$C(t) = \sum_{i=0}^{m} N_i^p(t) p_i, \text{ where } t \in [t_p..t_{m+1}]$$
(2.1a)

Here, m = n - p - 1 and N_i^p is the degree *p* B-spline basis function recursively defined by:

$$\begin{split} N_i^0(t) &= 1 \text{ if } t_i \le t < t_{i+1} \text{ or } 0 \text{ otherwise and} \\ N_i^p(u) &= \frac{t - t_i}{t_{p+i} - t_i} N_i^{p-1}(t) \\ &+ \frac{t_{p+i+1} - t}{t_{p+i+1} - t_{i+1}} N_{i+1}^{p-1}(t) \text{ when } p > 0 \quad (2.1b) \end{split}$$

Differing versions of the function are adopted by many authors, where differences are to do with the use being made of the subscripts versus the end conditions of the recursion.

Since $N_i^p(t) = 0$ for all t such that $t < t_i$ or $t > t_i$ t_{i+p+1} , a control point p_i influences the curve only in the range $t_i < t \le t_{i+p+1}$. Furthermore, for any tsuch that $t_i \leq t < t_{i+1}$, at most p+1 degree p basis functions are non-zero:

$$N_{i-p}^{p}(t), N_{i-p+1}^{p}(t), \dots, N_{i-2}^{p}(t), N_{i-1}^{p}(t), N_{i}^{p}(t)$$

The number of those functions reduces to *p* when

case in the uniform situation. For example, in the cubic case, when a knot t_i is in the span[$t_p..t_{m+1}$], the expression:

$$N_{i-3}^{3}(t_{i})p_{i-3} + N_{i-2}^{3}(t_{i})p_{i-2} + N_{i-1}^{3}(t_{i})p_{i-1}$$

evaluates to a point interpolated by the curve C(t) of Eqn. (2.1a). Furthermore, if p_{i-2} is replaced by the point:

$$\frac{1}{N_{i-2}^3(t_i)} \left[-N_{i-3}^3(t_i)p_{i-3} + p_{i-2} - N_{i-1}^3(t_i)p_{i-1} \right] \quad (2.1c)$$

in the associated control polygon, the resulting curve will interpolate p_{i-2} .

2.2. B-Spline Surfaces

Given the following items (see Fig. 4):

- a set of control points p_{ij} , where $0 \le i \le m_1$ and $0 \le i \le m_2$;
- a knot vector in the *u* direction, $[u_0, u_1, ..., u_{n_1}]$;
- a knot vector in the v direction, $[v_0, v_1, ..., v_{n_2}]$;
- a degree p_1 in the *u* direction (such that $m_1 = n_1 p_1 1$); and
- a degree p_2 in the v direction (such that $m_2 = n_2 p_2 1$);
- and given a knot $u \in [u_{p_1}..u_{m_1+1}]$ and a knot $v \in [v_{p_2}..v_{m_2+1}]$,



Fig. 4: Control points grid of a B-spline surface.

a B-spline surface [21] is defined by the following expression:

$$S(u,v) \equiv \sum_{i=0}^{m_1} \sum_{j=0}^{m_2} N_i^{p_1}(u) N_j^{p_2}(v) p_{ij}$$
(2.2a)

2.3. Factorization

For any particular parameter v such that $v_j \le v < v_{j+1}$, $N_i^{p_1}(u)$ may be factored out of the inner summation of

Eqn. (2.2), since it is constant along the u direction. This reduces the expression to:

$$\sum_{i=0}^{m_1} N_i^{p_1}(u) \sum_{k=j-p_2}^j N_k^{p_2}(v) p_{ik}$$
(2.2b)

3. INTERPOLATION OF A CURVE BY A B-SPLINE SURFACE

The research conducted in this paper is a natural outgrowth of conventional polygonal complexes developed in previously-visited modeling domains (essentially Doo-Sabin, Catmull-Clark and Loop subdivision surfaces) in the same way that T-splines are a natural outgrowth from NURBS surfaces. So, in order to eliminate the element of surprise, it is rather necessary to explain the context against which this idea is further developed in the T-spline domain.

3.1. Interpolation: fundamental specification

Generalizing the notion of polygonal complexes should start from a specification of the interpolation process itself. In fact, regardless of how it is achieved, the interpolation of a B-spline curve C(t)by a B-spline surface S(u, v) may be specified by the following formula:

$$\forall t \in [t_p..t_{m+1}] \exists u \in [u_{p_1}..u_{m_1+1}] \exists v \in [v_{p_2}..v_{m_2}+1]C(t)$$

= S(u, v) (3.1)

Accordingly, with reference to Eqn. (2.2b), the surface S(u, v) would be interpolating the curve $C'(u) \equiv \sum_{i=0}^{m_1} N_i^{p_1}(u) p'_i$, where $p'_i = \sum_{k=j-p_2}^{j} N_k^{p_2}(v) p_{ik}$, for every $i = 0..m_1$.

This is a mere generalization to surfaces of the observation made for curves at the end of section 2.1. Moreover, as noted before, the expression of p'_i reduces to $\sum_{k=j-p_2}^{j-1} N_k^{p_2}(v) p_{ik}$ when v is one of the knots (i.e. v_j) in the span $[v_{p_2}..v_{m_2+1}]$, which will be the case for the rest of the paper.

4. POLYGONAL COMPLEXES FOR B-SPLINE SURFACES OF ANY DEGREE

Since $N_j^{P_2}(v)$ is also constant, for any particular parameter v along the columns of the grid, the note made at the end of section 3.1 also suggests that the control points of the curve C'(u) can also be obtained through the following matrix multiplication:

$$R \times M$$
 (4.1)

Where *R* is a single row matrix consisting of the sequence $\left(N_k^{p_2}(v)\right)_{k=j-p_2..j-1}$, and *M* consists of p_2 consecutive rows of points $(p_{ik})_{i=0..m_1}$, starting from

row number $j - p_2$. The matrix M so described represents a polygonal complex of degree p_2 , and the curve C'(u) is the associated interpolated curve obtained as such (see Fig. 5).



Fig. 5: Cubic polygonal complex marked on a B-spline surface grid.

As it is currently formulated, the curve is always iso-parametric, along one or the other of the parameters of the surface; i.e., row-wise as well as columnwise.

Beside its immediacy, the above derivation embodies as particular cases those reported by Abbas [2], Nasri [15], Maekawa [10] and Yamaguchi [28] for the special case of uniform bi-cubic B-spline surfaces.

Since the subject of the discussion is no longer restricted to subdivision surfaces, it might be worth it to update the definition of a polygonal complex so as to take into account the non-uniform situation. Accordingly, it should now become: *the minimal neighborhood of the interpolating surface that holds the necessary information for the interpolated curve to be specified and constructed*.

5. APPLICATIONS

This section briefly surveys some of the applications that have already made use of polygonal complexes, illustrating the simplicity and ease with which interpolation is achieved. It focus mainly on interpolating B-spline curves by subdivision surfaces, without missing the fact that other applications of interest have already made use of these complexes, as mentioned in the introduction of this paper.

5.1. Curve Interpolation by Doo-Sabin Subdivision Surfaces

As far as this discussion is concerned, the only feature required from Doo-Sabin subdivision surfaces is the uniform bi-quadratic B-spline feature. As such, a polygonal complex will be a $2 \times N$ matrix of points of the control mesh, while the matrix *R* will be

$$\frac{1}{2} \times \begin{bmatrix} 1 & 1 \end{bmatrix}$$

Thus, when *M* is embodied within a control mesh, the matrix $R \times M$ specifies a control polygon of a quadratic B-spline curve interpolated by the limit Doo-Sabin subdivision surface of the embodying control mesh.

In the opposite direction, when a polygon P is replaced (in the embodying control mesh of a surface) by a two-row matrix M such that these two rows are symmetric with respect to P, the resulting subdivision surface will interpolate the quadratic B-spline curve limit of P itself.

5.2. Curve Interpolation by Catmull-Clark Subdivision Surfaces

As far as this discussion is concerned, the only feature required from Catmull-Clark subdivision surfaces is the uniform bi-cubic B-spline feature. As such, the polygonal complex will be a $3 \times N$ matrix of points of the control mesh, while the matrix *R* will be:

$$\frac{1}{6} \times \begin{bmatrix} 1 & 4 & 1 \end{bmatrix}$$

Thus, when *M* is embodied within a control mesh, the matrix $R \times M$ specifies a control polygon of a cubic B-spline curve interpolated by the limit Catmull-Clark subdivision surface of the embodying control mesh.

In the opposite direction, when the middle polygon m of M is replaced (in M) by the polygon:

$$\frac{1}{4} \times \begin{bmatrix} -1 & 6 & -1 \end{bmatrix} \times M$$

the embodying subdivision surface will interpolate the cubic B-spline curve limit of the middle polygon *m* itself. This particular application of the technique is at the basis of a process often called *lofting* [24] or *skinning* [27, 14] (see Fig. 6).



Fig. 6: A Catmull-Clark skinning subdivision surface.

5.3. Polygonal Complexes for NURBS Surfaces

This formulation of polygonal complexes and their associated curve is directly extendible to the *NURBS* [19] domain. The only extra feature that is required

to deal with here is the weight associated with each control point. However, the weight may simply be hidden as a multiplicative scalar inside the matrix M representing the polygonal complex with the corresponding point itself (more details on that may be found in the following section).

6. POLYGONAL COMPLEXES FOR T-SPLINE SURFACES

The following technical results should be considered as a culmination of the conceptual idea that could not have been justified without the inclusion of the discussion of the earlier part of the paper. Moreover, this gradual introduction of this idea represents a methodology that could be of value for anyone aiming to introduce polygonal complexes to any other modeling domain.

In this sense, this section shows that the specification of section 3.1 may also be exploited to derive further generalizations of polygonal complexes to cater for the requirements of other modeling domains, which is T-splines in this particular case.

6.1. T-Spline Surfaces : basics

T-splines surfaces [22, 23] were conceived as a generalization of *NURBS* [19]. The main distinguishing features of these surfaces are (see Fig. 7):



- a) The control mesh (called T-mesh) is no longer required to be a rectangular control grid, in the sense that constituting (rows and columns) control polygons do not necessarily stretch from one side of the control mesh to the other. This gives rise to so-called T-junctions, together with meaningful reduction in the number of control points required to specify the resulting surface.
- b) A knot inference mechanism that permits the automatic generation of the local knot vectors \hat{u}_i and \hat{v}_j associated with any particular point p_{ij} of the T-mesh, depending on the configuration of its immediate neighborhood on the T-mesh.

c) The ability to perform local refinement; a desirable but hardly obtainable feature in other existing modeling domains in general.

By comparison to the regularity of the B-splines setting, it is natural to anticipate that the extra features arising in the T-spline domain will imply that the formulation of a polygonal complex here will not flow as easily. Moreover, the expression representing the associated curve will be less straightforward to derive without further manipulation. This extra manipulation is quite apparent, for example, in the formulation of polygonal complexes in the Loop subdivision setting (see [8] and [9]).

6.2. T-Spline Surfaces : reformulation

Following the formulation of B-Spline polygonal complexes, a parallel treatment here will start from the basic expression of a T-spline surface:

$$T(u,v) \equiv \frac{\sum_{i \in I} \sum_{j \in I_i} w_{ij} B[\hat{u}_i, \hat{v}_j](u,v) p_{ij}}{\sum_{i \in I} \sum_{j \in I_i} w_{ij} B[\hat{u}_i, \hat{v}_j](u,v)}$$
(6.2a)

where *I* denotes the set of indices of the pointpopulated columns of the grid embodying the Tmesh, and *I_i* denotes the set of point-populated indices of column *i* of the same grid. Moreover, $\langle \hat{u}_i, \hat{v}_j \rangle$ is the pair of knot vectors associated with the control point p_{ij} , along the *i* and the *j* directions respectively.

Additionally, without sacrificing too much generality, the following discussion will be restricted to bi-cubic T-spline surfaces. Accordingly, the blending function *B* is as follows:

$$B[\hat{u}, \hat{v}](u, v) = N_0^3[\hat{u}](u)N_0^3[\hat{v}](v)$$
(6.2b)

Here, $N_0^3[\hat{u}](u)$ and $N_0^3[\hat{v}](v)$ are the same cubic B-spline basis function defined above but, this time, calculated with respect to the local knot vector \hat{u} and \hat{v} , respectively.

For the sake of simplicity, the subscript 0 and the superscript 3 of the B-spline basis function N will be suppressed from now on. Furthermore, the weights w_{ij} associated with individual points p_{ij} of the T-mesh will also be ignored, without any loss of generality, since they can always be treated in the same way suggested for *NURBS*. Finally, the denominator of Eqn. (6.2a) will also be ignored, since it plays no more role than a multiplicative constant.

Thus, under the above assumptions, the summation in Eqn. (6.2a) reduces to:

$$T(u,v) \equiv \sum_{i \in I} \sum_{j \in I_i} N[\hat{u}_i](u) N[\hat{v}_j](v) p_{ij}$$
(6.2c)

6.3. Further Manipulations

In the cubic case, and following our discussion in section 2, the influential elements of the set I_i over



Fig. 8: The control points of the T-mesh: (a) The colored points of a polygonal complex and (b) The associated knot vectors.

the inner summation of Eqn. (6.2c) are no more than three, which reduces this summation to:

$$\sum_{i \in I} (N[\hat{u}_{i<}](u)N[\hat{v}_{j<}](v)p_{ij<} + N[\hat{u}_{i}](u)N[\hat{v}_{j}](v)p_{ij} + N[\hat{u}_{i>}](u)N[\hat{v}_{j>}](v)p_{ij>})$$
(6.3)

Here, p_{ii} (if it exists) is the control point on the crossing of row *j* with column *i* of the T-mesh, where *j* refers to the index of the iso-parametric curve corresponding to parameter v of the T-mesh.

Additionally, $p_{ii<}$ (if it exists) is the control point on the first populated row below *j* crossing with column i of the T-mesh, but still within the area of influence of $\langle \hat{u}_i, \hat{v}_j \rangle$ of p_{ij} . Likewise, p_{ij} (if it exists) is the control point having similar role to that of p_{ii} but this time above row *j* on column *i* of the T-mesh. The notation of the associated knot vectors should be accorded the natural interpretation matching that. Fig. 8 indicates the colored nodes on the T-mesh that should, in principle, belong to the polygonal complex.

6.4. The Associated Curve

The cubic B-spline curve associated with the T-spline polygonal complex should look like the following:

$$\sum_{i \in J_j} N_{ij}^3 [\hat{w}_{ij}](w) q_{ij}$$
(6.4a)

Here, J denotes the set of indices of the pointpopulated rows of the grid embodying the T-mesh, and J_j denotes the set of point-populated indices of row *j* of the same grid. Accordingly, in order to simplify matching between various terms of the curve expression in Eqn. (6.4a) against the corresponding ones in the surface expression of Eqn. (6.2c), two complications need to be resolved:

• As opposed to the regularity of B-splines, the sets I and J_i in the T-spline context, are not identical. In fact, in the case of T-splines, $J_i \subseteq I$ in general, because not every populated column necessarily implies a control point q_{ii} for the corresponding curve (see Eqn. (6.2a)).

The nature of the T-splines knot inference mechanism implies that the derived knot vectors may not necessarily be identical along any given row of the T-mesh. Consequently, the associated basis functions $N_i^{p_1}(u)$ may not be conducive to factorization as in the case of expression of Eqn. (2.2b).

The first complication is addressed by insisting that *I* and J_i should be made identical; i.e., $J_i \equiv I$. This may be achieved through the application of Tspline local refinement, which introduces a point on the main row of the complex for every populated column of the T-mesh.

This process always yield a result (see Fig. 9), due to the fact that local refinement does not add any extra rows or columns to the rectangular grid underlying the T-mesh, except maybe during the insertion of the first knot (which is not the situation here).



The shaded area of Fig. 10(a) illustrates an instance of a T-spline non-regular polygonal complex obtained



Fig. 11: The interpolated curves: (a) Interpolated curve 1 and (b) Interpolated Curve 2.

as a result of this refinement mechanism. Note that, in the particular instance depicted in this figure, a node on a particular column of the T-mesh does not have to have counterparts on either of the other two rows. However, the middle row has to always be populated. The resulting polygonal complex should be taken as a natural growth since, as one might expect, a regular polygonal complex is just a particular case of that.

Since the nodes of the middle row of the complex are all associated with the same parameter v, such a polygonal complex may be traced directly along this parameter so as to produce a curve interpolated by the T-spline surface corresponding to the T-mesh of Fig. 9 (see Fig. 11(a) and Fig. 11(b)). Note here that, although Fig. 11(b) is a slight deformation of Fig. 11(a), interpolation of the same curve is still maintained there, nevertheless.

However, this gives rise to another problem; that of showing that the obtained interpolated curve is a cubic B-spline curve in the classical sense. In fact, with reference to Eqn. (6.4a), the identity $(J_j \equiv I)$ does not by itself guarantee that a sequence of points $(q_i)_{i \in K}$ may be generated to satisfy:

$$N[\hat{u}_{i}](u)q_{ij} = N[\hat{u}_{i<}](u)N[\hat{v}_{j<}](v)p_{ij<} + N[\hat{u}_{i}](u)N[\hat{v}_{j}](v)p_{ij} + N[\hat{u}_{i>}](u)N[\hat{v}_{j>}](v)p_{ij>}$$
(6.4b)

Thus, the second complication referred to above remains unresolved, in the sense that the required factorization over all terms of summation of Eq. (6.4b) remains inapplicable, wherever $N[\hat{u}_i](u)$ is different from $N[\hat{u}_{i>}](u)$ and/or from $N[\hat{u}_{i>}](u)$.

For this reason, a further round of refinement operations is still required to refine $\hat{u}_{i<}$ and $\hat{u}_{i>}$, thus splitting the defying terms $N[\hat{u}_{i<}](u)$ and $N[\hat{u}_{i>}](u)$

into parts that are equal to $N[\hat{u}_i](u)$, and consequently admissible for factorization.

6.5. The Formulation

The shaded area of Fig. 10(b) illustrates an instance of a T-spline polygonal complex after this second round of local refinement operations. Again, this should be taken as a natural growth since, as one would perhaps expect, a regular polygonal complex is just a particular case of that.

In the worst case scenario, too many refinement operations may turn the T-mesh into a *NURBS* like control mesh (actually, this is the method pursued in [25] to achieve interpolation), but the technique would still be working because, as explained above, a *NURBS* polygonal complex is just particular a case of those of T-splines.

At the end, the above sequence of manipulations yields a sequence $(q_{ij})_{i \in J_i}$ defined by:

$$q_{ij} \equiv N[\hat{v}_{j<}](v)p_{ij<} + N[\hat{v}_{j}](v)p_{ij} + N[\hat{v}_{j>}](v)p_{ij>} \quad (6.5a)$$

However, for any given parameter v, the terms $N_{j<}^{p_2}(v)$, $N_j^{p_2}(v)$ and $N_{j>}^{p_2}(v)$ may not necessarily be similar along any column. Consequently, the formulation of the control points of the curve associated with a polygonal complex given in Eqn. (4) will have to be generalized in order to cope with the T-spline additional requirements. In fact, these control points are now restricted to the first diagonal of the matrix multiplication in Eqn. (4).

In the cubic case, as illustrated in Eqn. (6.5b), M is now a $3 \times m_1$ matrix of points with the i^{th} column consisting of the three points $p_{ij<}$, p_{ij} and $p_{ij>}$, for all i ranging from 0 to $m_1 - 1$. Furthermore, R is now an $m_1 \times 3$ matrix of scalars, with the i^{th} row consisting of the three terms $N_{j<}^{p_2}(v)$, $N_{j>}^{p_2}(v)$ and $N_{j>}^{p_2}(v)$, for all i ranging from 0 to $m_1 - 1$.

In this sense, Eqn. (4) turns out to be a particular case of Eqn. (6.5b), were all rows of the matrix R are



Fig. 12: Further illustrations of the interpolation task via T-spline Polygonal Complexes.

identical.

$$\begin{bmatrix} N[\hat{v}_{0<}](v) & \dots & N[\hat{v}_{(n-1)<}](v) & N[\hat{v}_{n<}](v) \\ N[\hat{v}_{0}](v) & \dots & N[\hat{v}_{(n-1)}](v) & N[\hat{v}_{n}](v) \\ N[\hat{v}_{0>}](v) & \dots & N[\hat{v}_{(n-1)}>](v) & N[\hat{v}_{n>}](v) \end{bmatrix}^{T} \\ \times \begin{bmatrix} p_{0<} & \dots & p_{(n-1)<} & p_{n<} \\ p_{0>} & \dots & p_{(n-1)>} & p_{n>} \\ p_{0>} & \dots & p_{(n-1)>} & p_{n>} \end{bmatrix}$$
(6.5b)

6.6. Further Applications: T-Spline Surface Skinning

A direct application of Eqn. (6.5a) may be the generation of a T-spline surface interpolating a sequence of iso-parametric curves (the same processed referred to above as *lofting* or *skinning*). In fact, in such a situation, the inverse of Eqn. (6.5a) may be used. That is:

$$p_{ij} = \frac{1}{N[\hat{v}_j](v)} (-N[\hat{v}_{j<}](v)p_{ij<} + q_{ij} - N[\hat{v}_{j>}](v)p_{ij>})$$
(6.6)

When focusing on the points of the control mesh existing along a fixed parameter line as the control points of a B-spline curve (C), lofting (or skinning) concerns the techniques that should be used for modifying the rest of the control point of the mesh so that the surface resulting from these modifications will interpolate the initially given curve (C).

In this sense, the benefits of polygonal complex resides in the fact that a control point on the middle row of a polygonal needs to be repositioned with respect to only the other two points existing on the wings of this complex.

Again, this is a major simplification of the skinning task, when compared to previous techniques.

This should be compared to the result presented in [11], where the skinning process, in the T-spline domain, is attempted without making use of polygonal complexes. Actually, the research described in [11] constitutes the main motivation behind the work reported in the present paper, since it paved the way for a formulation of T-spline polygonal complexes, which makes the present paper a natural outgrowth of [1] directed toward T-splines surfaces.

6.7. Further Remarks

Fig. 12 also presents another set of figures illustrating the effectiveness of T-spline polygonal complexes with regard to the curve interpolation task.

One might complain from the uniformity of these figures. However, this uniformity mainly arises from the fact that each figure is interpolating a single curve along a single (horizontal or vertical) parameter line.

This may be helped by interpolating more than one curve (on the same parameter direction), by making sure that the section occupied by each curve is isolated from all the others. This method is employed in [14] in the context of subdivision surfaces (also see fig. 6).

This may also be extended to interpolate two interecting curves, on two orthogonal parameter directions, but with a special arrangement around the point where these two curves are interecting (see [3]).

However, when it comes to interpolating more than two curves with the same intersection point, this takes the problem to a higher level of difficulty, since one of these curves will necessarily be non-isoparameteric with respect to the interpolating surface, which represents a problem that has so far been defying exact solutions for a few decades now.

7. CONCLUSIONS AND FURTHER WORK

This paper introduces polygonal complexes to the T-spline domain, starting from the theoretical foundations of this notion. Considering the generality of the approach, this work should also be expandable to cover more liberal domains.

A major effort (expanded throughout this paper) went toward showing that those complexes developed in the respective domains are basically emanating from the same origin and are developed within the same unifying framework and serves the same purpose in those respective domains.

Continuity of the interpolating surface is never affected, since nothing of the essentials relating to the construction of the surface is touched during the development of the polygonal complex.

The usefulness of polygonal complexes, beside many other things, resides in their ability to achieve interpolation at very little computational cost, with relatively simple operations. However, the research conducted in this paper should be seen as just a first step. The natural research direction that should follow should perhaps go toward the applications of these complexes in the T-spline domain, which have been briefly mentioned in the introduction and also in main body of the paper.

ACKNOWLEDGEMENT

The research work reported in this paper is supported by a grant from the Lebanese Council for Scientific Research for the academic year 2012–2013.

REFERENCES

- [1] Abbas, A.: Generalizing Polygonal Complexes across Modeling Domains, Computer Graphics International, 12–15 June, Bournemouth, U.K., 2012.
- [2] Abbas, A.; Nasri, A.; Maekawa, T.: Generating B-spline curves with points, normals and curvature: a constructive approach, The Visual Computer, 26, 2010.
- [3] Abbas, A.; Nasri, A.: Interpolating Multiply Intersecting Curves Using Catmull-Clark Subdivision Surfaces, Computer-Aided Design & Applications, 1(1-4), 2004, 25-32.
- [4] Catmull, E.; Clark, J.: Recursively generated B-spline surfaces on arbitrary topological meshes, Computer Aided Design, 10(6), 1978, 350–355. http://dx.doi.org/10.1016/00 10-4485
- [5] Doo, D.; Sabin, M.: Behavior of recursive division surfaces near extraordinary points, Computer Aided Design, 10(6), 1978, 356-360. http://dx.doi.org/10.1016/0010-4485
- [6] Halstead, M.; Kass, M.; DeRose, T.: Efficient, fair interpolation using Catmull-Clark surfaces, Computer Graphics (Proceedings of ACM SIG-GRAPH' 93), 1993, 35–44.
- [7] Hoschek, J.; Jüttler, B.: Techniques for fair and shape preserving surface fitting with tensorproduct B-splines, Shape Preserving Representations in Computer-Aided Geometric Design, 1999, 163–185. Nova Science, New York.
- [8] Ma, W.; Wang, H.: Interpolating an arbitrary number of joint B-spline curves by Loop surfaces, Computers and Graphics, 36(5), 2012.
- [9] Ma, W.; Wang, H.: Loop subdivision surfaces interpolating B-spline curves, Computer-Aided Design, 41(11), 2009, 801–811. http://dx.doi. org/10.1016/j.cad.2009.03.011
- [10] Maekawa, T.; Matsumoto, Y.; Namiki, K.: Interpolation by geometric algorithm, Computer-Aided Design, 39(4), 2007, 313–323. http://dx. doi.org/10.1016/j.cad.2006.12.008

- [11] Nasri, A.-H.; Sinno, K.; Zheng, J.: Local T-spline surface skinning, The Visual Computer, 28(6-8), 2012, 787-797. http://dx.doi.org/10.1007/ s00371-012-0692-1
- [12] Nasri A.; Sabin, M.; Abu Zaki, R.; Nassiri, N.; Santina, R.: Feature curves with cross curvature control on Catmull-Clark subdivision surfaces, Computer Graphics International, 2006, 761–768.
- [13] Nasri, A.: Interpolating an Unlimited Number of Curves Meeting at Extraordinary Points on Subdivision Surfaces, Computer Graphics Forum, 22(1), 2003, 87–97. http://dx.doi.org/10.1111/ 1467-8659.t01-1-00648
- [14] Nasri, A.; Abbas, A.; Hasbini, I.: Skinning Catmull-Clark Subdivision Surfaces with Incompatible Cross-Section Curves, Pacific Graphics, 2003, 102–111.
- [15] Nasri, A.; Abbas, A., Designing Catmull-Clark subdivision surfaces with curve interpolation constraints, Computer and Graphics, 26(3), 2002, 393-400. http://dx.doi.org/ 10.1016/S0097-8493
- [16] Nasri, A.; Sabin, M.: Taxonomy of Interpolation Conditions in Recursive Subdivision Curves, The Visual Computer Journal, 18(4), 2002, 259– 272.
- [17] Nasri, A.; Sabin, M.: Taxonomy of Interpolation Conditions in Recursive Subdivision Surfaces, The Visual Computer Journal, 18(6), 2002, 382– 403.
- [18] Nasri, A.: Recursive Subdivision of Polygonal Complexes and their applications in CAGD, Computer Aided Geometric Design Journal, 17, 2000, 595-615. http://dx.doi.org/10.1016/ S0167-8396(00)00015-7
- [19] Piegl, L.; Tiller, W.: The NURBS Book, Springer-Verlag, New York, NY, 1997. http://dx.doi.org/ 10.1007/978-3-642-59223-2
- [20] Pottmann, H.; Leopoldseder, S.; Hofer, M.: Approximation with Active B-Spline Curves and Surfaces, Pacific Conference on Computer Graphics and Applications, 2002, 8–25.
- [21] Prautzsch, H.; Boehm, W.; Paluszny, W.: Bézier and B-Spline techniques, Springer, 2002. http://dx.doi.org/10.1007/978-3-662-04919-8
- [22] Sederberg, T.; Cardon, W.; Finnigan, D.; North, G.; Zheng, N.; Lyche, T.: T-spline Simplification and Local Refinement, ACM Transactions on Graphics, 23(3), 2004. http://dx.doi.org/ 10.1145/1015706.1015715
- [23] Sederberg, T.; Zheng, J.; Bakenov, A.; Nasri, A.: T-splines and T-NURCCS, ACM Transactions on Graphics, 22(3), 2003, 477-484. http://dx.doi.org/10.1145/882262.882295
- [24] Schaefer, S.; Warren, J.; Zorin, D.: Lofting curve networks using subdivision surfaces, Proc. Eurographics Symposium on Graphics, 2004, 105–116.

- [25] Sinno, K.: T-Spline Skinning Surfaces, Master Thesis, Department of Computer Science, American University of Beirut, Lebanon, 2008.
- [26] Weiss, V.; Andor, L.; Renner, G.; and Várady, T.: Advanced surface fitting techniques, Computer Aided Geometric Design, 19(1), 2002, 19–42. http://dx.doi.org/10.1016/S0167-8396
- [27] Woodward, C.: Skinning techniques for interactive B-spline surface interpolation, Computer-Aided Design, 20(8), 1988, 441–451. http://dx. doi.org/10.1016/0010-4485(88)90002-4
- [28] Yamaguchi, F.: Curves and Surfaces in Computer Aided Geometric Design, Springer-Verlag, NY, 1988. http://dx.doi.org/10.1007/978-3-642-48952-5