

On the interpolation of non-iso-parametric curves

Abdulwahed M. Abbas¹  and Ahmad H. Nasri² 

¹The University of Balamand, Lebanon; ²American University of Beirut, Lebanon

ABSTRACT

Exact curve interpolation has frequently been addressed in a variety of modeling domains. The suggested solutions uniformly assume the interpolated curve to be iso-parametric along one or the other of the parameter lines controlling the interpolating surface. However, once this assumption is abandoned, the difficulty of this problem rises to a degree that no known method is currently in existence which is able to solve this problem efficiently enough for practical purposes. In this context, this paper presents an algorithm capable of embedding a cubic B-spline curve within a *regular* quad-based Catmull-Clark subdivision surface along any arbitrary direction not necessarily *along* any of the parameter lines controlling the surface.

KEYWORDS

Non-iso-parametric curves; curve interpolation; Catmull-Clark subdivision surfaces; polygonal complexes; X-configurations

1. Introduction

Curve interpolation (also known as lofting [10] or skinning [8]) has been addressed in numerous modeling domains: B-spline surfaces, NURBS [9], subdivision surfaces (such as Catmull-Clark [3] and Loop [7] subdivision surfaces), and also in the domain of T-spline surfaces.

Approaches developed for solving the interpolation problem range from exact interpolation, *via* the traditional system of linear equations with many variables [5], to approximation methods (also known as fitting), and lately through the use of polygonal complexes [1].

The interpolation problem is commonly posed as follows: given a curve (C) together with a surface (S), the idea is to modify (S) minimally (but sufficiently) for the modified version (S') of (S) to interpolate (C).

Furthermore, in the particular domains of B-spline and NURBS surfaces, the solution fundamentally assumes (C) to be *iso-parametric* with respect to one or the other of the parameters controlling (S), which turns out to be a considerable simplification of a more difficult problem.

The particular choice of the research direction reported in this paper is motivated by the problem of modifying an initial B-spline surface to interpolate a non-constant-parameter B-spline curve [4]; a problem that is still defying an exact solution, even after a few decade of research. Moreover, the approximate solutions that have so far been developed for that problem are not

sufficiently efficient for practical purposes. In this respect, the particular wording of the title of this paper is also selected in the spirit of the title of the above reference.

In this regard, it may be argued that the main difficulty standing in the way of an exact and practical solution of the problem results from the inability to incorporate the control points of the given curve as an integral part of those of the given surface, without perhaps destroying the regular B-spline nature of that surface, and therefore the starting premise of the problem itself. This may also be the source of the high degree of the curve-on-surface emanating from the original curve, and resulting in the inability of exactly matching the original curve with the corresponding curve-on-surface [6], and consequently resulting in the inability of obtaining an exact solution of the problem.

In view of these difficulties, the present paper reduces the problem to that of constructing a Catmull-Clark [3] subdivision surface that can interpolate an initial *regular* cubic B-spline curve. However, the relationship between the original problem posed in [4] and the problem addressed in the present paper is still maintained by insisting that the control mesh corresponding to the initially given subdivision surface is regular, and that the control polygon corresponding to the curve is not necessarily equal in size to any of the polygons constituting the control mesh of the initial surface nor running along any of the parameter lines controlling that surface.

CONTACT Abdulwahed Abbas  abbas@balamand.edu.lb

Ahmad H. Nasri's new affiliation: Current address: Fahad Bin Sultan University, Tabuk, PoB 15700, KSA

In this sense, the research reported in this paper represents yet again another demonstration that an alternative representation of the given of a problem might render a defiant solution possible.

In summary, this paper presents an algorithm that is able to embed a cubic B-spline curve on a *regular* quad-based Catmull-Clark subdivision surface along any arbitrary direction that does not necessarily *run along* any of the polygons of the mesh controlling the surface. The usefulness of this approach relies on the ability of *iso-parametricity* (or otherwise) of the given curve to *retain* some *sensible* meaning with respect to a subdivision surface, precisely when the control mesh associated with that surface is *regular*.

The rest of this paper is structured as follows. Section 2 reviews the basics of Catmull-Clark subdivision surfaces. Section 3 reviews the associated notion of polygonal complexes and its role in achieving curve interpolation. Section 4 presents an algorithm that attains non-iso-parametric curve interpolation and further discusses some of its possible consequences, while section 5 concludes the paper with a summary and some directions for further work.

2. Catmull-Clark subdivision surfaces

A Catmull-Clark subdivision surface [3] is a generalization of the uniform bi-cubic B-spline surfaces where, given an initial control mesh, composed of vertices, edges and faces, one subdivision step proceeds by determining:

- For each face, a new *F-vertex*, that is the average of the constituent vertices of this face.
- For each inner edge, a new *E-vertex*, that is the average of the constituent pair of vertices of this edge together with the *F-vertices* of the two directly-adjacent faces
- For each inner vertex *V*, a new *V-vertex*, that is determined by the following expression:

$$\frac{(n-2)V + \frac{R+S}{n}}{n} \quad (1)$$

where

- *n* is the valence of the vertex *V* of the mesh.
- $R = \sum_{i=1}^n V_i$ and $S = \sum_{i=1}^n V_{f_i}$ where V_i is a vertex adjacent to *V* via an edge and V_{f_i} is an *F-vertex* f_i embodying *V*.

The next mesh in the subdivision sequence is obtained by linking together the various F-, E- and V-vertices obtained by the above refinement process, in such a way that each F-vertex and V-vertex is connected to their directly-adjacent E-vertices (see Fig. 1).

This way, when enough of those refinement steps are performed, the initial control mesh gradually transforms

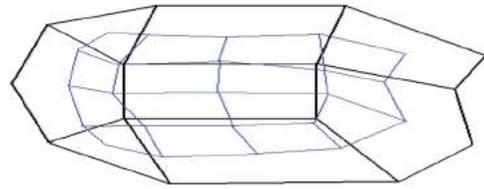


Figure 1. A Catmull-Clark subdivision step.

into a final smooth limit surface. However, the existence of non-quad faces in the initial control mesh gives rise to so-called irregular vertices, which might have negative impact on the overall smoothness of the final limit surface.

3. Catmull-Clark polygonal complexes

A *simple* Catmull-Clark (CC) polygonal complex [1] may be seen as a sequence of pairs of rectangular faces; where each pair of this sequence has a common edge and each two consecutive pairs have common respective edges (see Fig. 2).

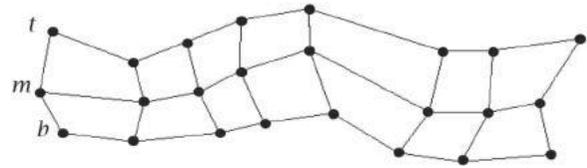


Figure 2. A simple Catmull-Clark polygonal complex.

This may be represented by a $3 \times n$ matrix *M* of vertices representing three control polygons: top (t_i), middle (m_i) and bottom (b_i), all having the same number *n* of vertices.

The interest in polygonal complexes arises from the observation that, after each step of the subdivision process, the polygonal complex gets narrower and narrower till, at the limit, it becomes a curve. Even more interestingly perhaps, this limit curve is automatically interpolated by the surface limit of any control mesh embodying this complex (see Fig. 3).

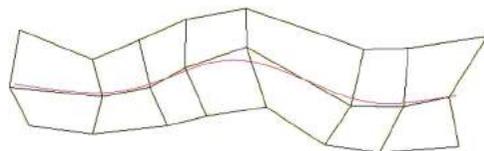


Figure 3. Limit curve of a polygonal complex.

Deviating now from regularity, a *general* polygonal complex is encountered when the polygons (t_i), (m_i) and (b_i) do not all have the same size (see Fig. 4).

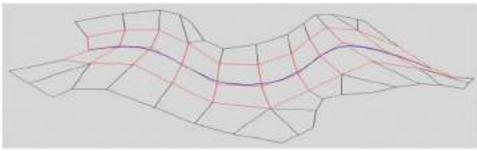


Figure 4. A general polygonal complex.

That is, though each inner vertex is regular in the sense that it connects exactly four edges, the corresponding faces may not be regular at the outer edges. However, a general complex reduces to a simple one after a single subdivision step (see inner colored complex in Fig. 4, again).

In addition to that, the limit of a simple Catmull-Clark complex M is a cubic B-spline curve whose control polygon (P) is determined by the following formula [8]:

$$\frac{1}{6} \times [1 \quad 4 \quad 1] \times M \quad (2)$$

Conversely, if the mid-polygon m of a simple (CC) complex M is substituted by the polygon [8]:

$$m' = \frac{1}{4} \times [-1 \quad 6 \quad -1] \times M \quad (3)$$

the limit curve of the resulting complex M' will be a B-spline curve identical to that of m . Accordingly, any given curve defined by a control polygon (m_i) , can be turned into a polygonal complex M by adding two more rows of points (t_i) and (b_i) to either side of it and then, through the application of equation (3), guarantees that any mesh embodying the complex M' thus constructed in fact admits a limit surface interpolating the original curve corresponding to (m_i) .

4. The interpolation algorithm

This algorithm takes, as input, a polygon and a regular control mesh of the kind depicted in Fig. 5. Such a figure has frequently been taken by many authors (as in [6], for example) to illustrate the projection of the given curve in the parameter domain of the surface, to show that

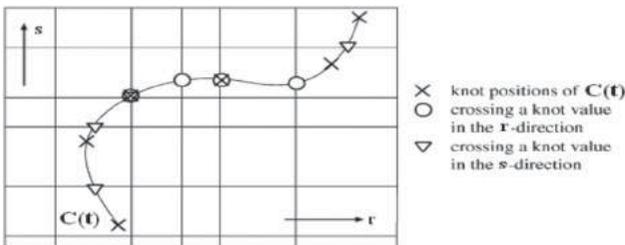


Figure 5. A Non-iso-parametric curve.

the curve is clearly non-iso-parametric with respect to the underlying regular mesh corresponding to a B-spline surface.

However, a word of caution is due here, just to avoid any possible mis-interpretation. In fact, Fig. 5 is assumed here to represent the actual curve and surface as modeled in 3D space. As a consequence, all references made below to *horizontal* or *vertical* edges of the mesh in figures like Fig. 5, should be understood with respect to the orientation of those edges as seen on the actual page of this paper, and not with respect to the actual mesh as modeled in 3D space.

4.1. Algorithm behavior

Starting from the control polygon (P) of the initial curve (C) and the initial control mesh (M) of the surface (S), the following steps describe in details the interpolation process, which is simply a re-meshing process that, starting from the mesh (M), leads to a final control mesh (M') whose associated Catmull-Clark subdivision surface (S') will end up interpolating the initially-given curve (C).

The detailed behavior of the algorithm is described through a trace of the following steps:

Step 1: the first step is a mesh editing phase that integrates the polygon (P) with the control mesh (M). With reference to Fig. 5, this has the goal of ensuring that as many as possible of the initial control points of (P) sit on some of the corresponding *horizontal* or *vertical* edges of control mesh (M), or perhaps on some of its vertices (see Fig. 6).



Figure 6. Initial control points of polygon on initial edges or junctions of mesh.

Step 2: this step adds more *horizontal* or *vertical* edges to the new mesh, sufficiently to ensure that every edge of the curve splits a face of the control mesh into two faces and, at the same time, being careful that newly inserted edges do not interfere with existing edges of the curve (see Fig. 7).

Consequently, each inner control point of the curve should now be the meeting point of no more than four incoming edges; two of these are its own edges alternating with two other edges of the modified control mesh.

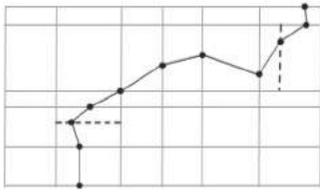


Figure 7. Adding more *horizontal* and/or *vertical* edges to the mesh.

Step 3: any other interfering edge of the control mesh (in the sense of disobeying the constraints stated in *Step 2*) should therefore be removed (see Fig. 8).

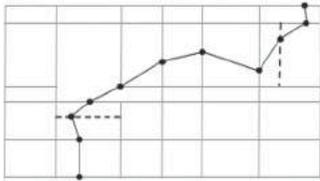


Figure 8. Removing interfering edges.

At the end of this step, every edge of the curve should now be adjacent to exactly two faces of the mesh; which gives rise to a *general* complex, in the sense of Fig. 4 above.

Step 4: both curve and mesh are now subdivided separately one step each, resulting in a *simple* Catmull-Clark polygonal complex embedded in the control mesh of the surface (see Fig. 9).

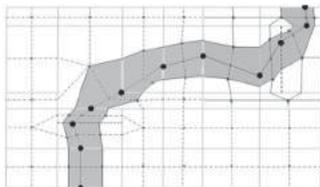


Figure 9. The (shaded) polygonal complex.

Step 5: the mid-polygon of the resulting polygonal complex is now replaced by the refined control polygon of the curve; point for point, and in the obvious way.

Step 6: the control points of the mid-polygon of the resulting complex are now repositioned through a single application of Eqn. (3).

This way, after subdividing the curve and the resulting mesh separately and up to the limit, the original curve will end up interpolated by the limit surface.

In conclusion, a couple of points are to be noted here:

- the initial curve (C) is never altered at any stage of the modification process described above. Rather, the above steps are designed to alter the initial control mesh (M) only.

- the above steps introduce quite a few extraordinary vertices to the initially-regular control mesh (M), precisely due to non-iso-parametric nature of the initial curve (C). In turn, the presence of these extraordinary vertices on the final control mesh (M') are the main reason for selecting Catmull-Clark subdivision to generate the final interpolating surface (S'), simply because Catmull-Clark subdivision does not mind the presence of those extra-ordinary vertices on that control mesh.

4.2. Algorithm summary

The steps of the interpolation process detailed in the previous section may be summarized by the following numbered items. Each item summarizes the step with the same number in section 4.1.

Given are an initial polygon C_0 and an initial regular mesh M_0

- 1- $M_1 =$ integrate C_0 within M_0 . This involves adding the vertices and edges of C_0 to those of M_0 and also splitting some faces of M_0 accordingly
- 2- $M_2 =$ the mesh resulting from adding more edges to (and maybe removing interfering edges from) M_1 . This might also involve combining faces of M_1 together. This gives rise to initial possibly general polygonal complex P_0 within M_2
- 3- $M_3 =$ subdivide M_2 one step. This gives rise to simple polygonal complex P_1 out of P_0 with C_1 as the mid-polygon of P_1 corresponding to C_0 within M_3
 $C_2 =$ subdivide C_0 one step separately
- 4- $M_4 =$ replace the vertices of C_1 by corresponding ones from C_2 . This also changes P_1 to P_2
- 5- $M_5 =$ reposition the vertices of mid-polygon C_2 of P_2 with respect to the vertices of corresponding wing-polygons of P_2 using Eqn. (3)
- 6- Subdivide C_2 up to the limit curve C and subdivide M_5 up to the limit surface S

Result: the curve C will end up interpolated by the surface S

The working of the steps 1 to 6 taken by the above algorithm is illustrated in Fig. 10 as well as in all subsequent figures up to Fig. 14.

However, some clarifications are due to make sure that these figures are correctly interpreted. In fact, each figure is composed of a sequence of frames numbered: (a), (b), (c), etc, intended to be seen in association with the execution of steps 1 to 6 of the above algorithm.

In Fig. 10, for instance, frames (a) and (b) are intended to show the regular nature of the initial control mesh, frame (c) is intended to illustrate the integration of the initial control polygon with the initial control mesh (i.e. step 1 of the algorithm).

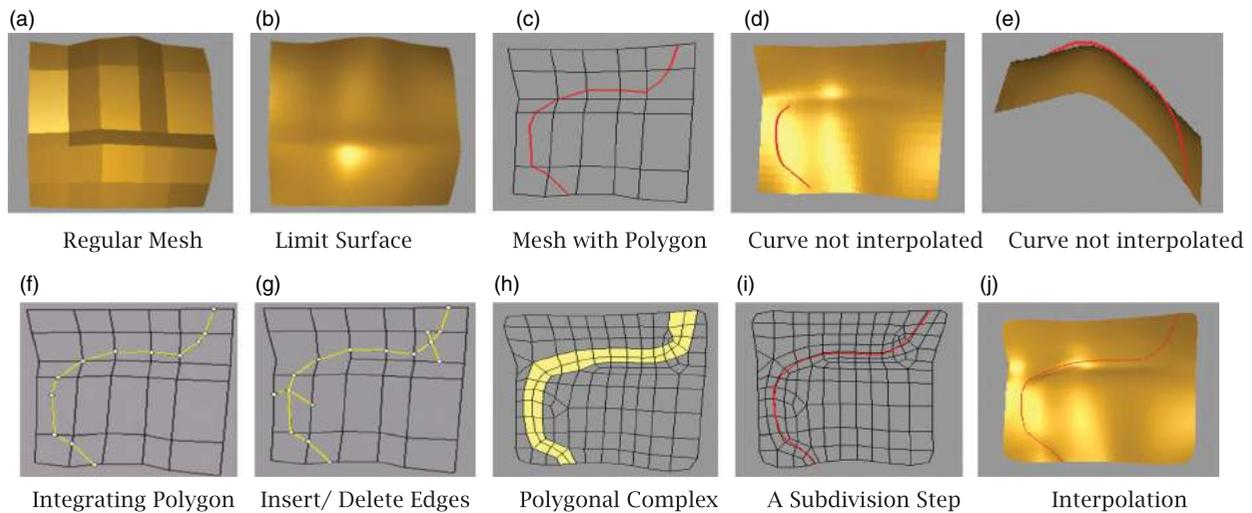


Figure 10. A 3D Reproduction of Fig. 5.

Frames (d) and (e) are intended to illustrate that the initial surface does not interpolate the initial curve. In fact, while this is clear in frame (d), the change in perspective in frame (e) is included as a further illustration of this particular fact.

Frames (f) to (i) are intended to depict the re-meshing process that goes on in step 2 up to step 6 of the above algorithm and in the same sequence. Finally, frame 11 is intended to illustrate the final state of the algorithm where exact interpolation is finally reached.

This strategy (and all the decisions that come with it) has the benefit of making interpolation exact (rather than approximate). Moreover, so far, there appears to be no need for any modification of the subdivision coefficients or rules being employed at any stage of the subdivision process.

However, note here that even though exact interpolation is always achieved following this approach, the addition and removal of edges and faces will result in the appearance of extra-ordinary vertices on the final surface which, as mentioned earlier, will have an obvious impact on the overall quality and the smoothness of the resulting surface.

These extra-ordinary vertices arise as a result of irregular faces appearing during the so many transformations that the, initially regular, faces have to go through during the re-meshing process. The irregularity of those faces is mainly due to the non-iso-parametric nature of the initial control polygon, which cannot be entirely eliminated, since it is the initial premise of the problem at hand.

However, these undesirable effects can somewhat be minimized by a careful design of the initial control mesh, which can always be done at minimal cost. In fact, in the case where the initial control polygon avoids the junctions of the control mesh, there will be no need for

the removal of any of the existing edges of the control mesh. In turn, this will suppress most of the T-shaped junctions and will therefore remove plenty of the badly-shaped irregular faces.

In this respect, in the situation where each vertex of the initial control polygon is sitting on a separate edge of the control mesh without coinciding with any of its vertices (see Fig. 12(d), for an illustration); the following observations may be recorded:

- The initial curve is distinctly removed from the initial surface (see Fig. 11(e)).
- No additional edges need to be inserted into the mesh.
- None of the existing edges of the mesh need to be removed.

This implies an absolute minimum of extraordinary faces (the white ones in Fig. 11(d)) and a total absence of superfluous degrees of freedom.

In this context, it would be possible to interpolate intersecting non-iso-parametric curves by the given regular surface (see Fig. 12). Here, the intersection points on the control mesh are necessarily irregular. As such, they will give rise to the so-called X-configurations (see center of Fig. 12(c)) at the next subdivision level, which have to be subdivided differently so as to insure interpolation at and around those points (more details on that may be found in [2]).

Note here that, in contrast with the other figures presented in this paper, Fig. 12 has more involved features, in the sense that the initial control mesh is comparatively composed of much more control points. Nevertheless, exact interpolation is still maintained in the final state of the mesh.

Furthermore, the given regular surface is also able to interpolate partial non-iso-parametric curves (see Fig. 13) (i.e. curves that do not stretch to the border of

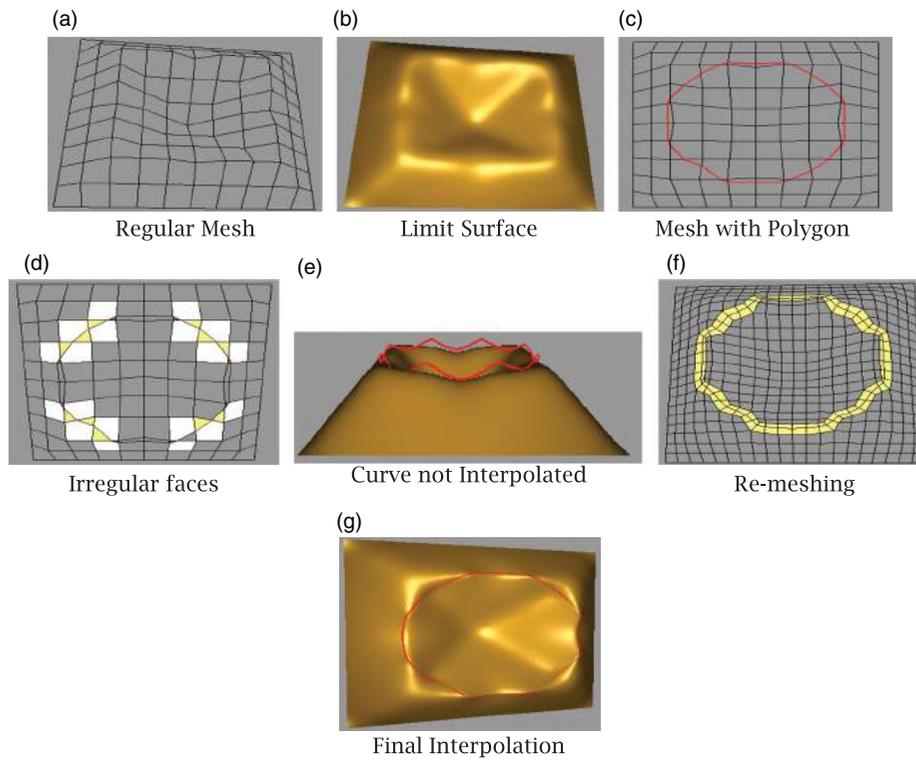


Figure 11. An Ashtray.

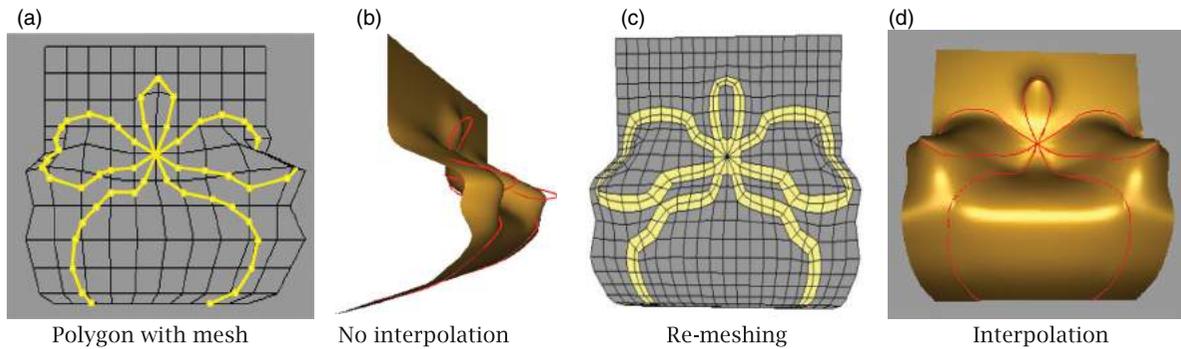


Figure 12. Interpolating Intersecting Curves.

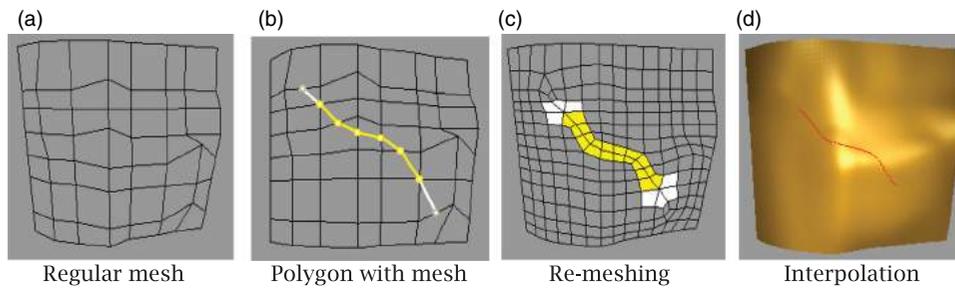


Figure 13. Interpolating a partial non-iso-parametric curve.

the interpolating surface). In fact, the end points of the curves would simply be designed in the same way as the meeting points of the intersecting curves mentioned earlier.

In Fig. 13(b), two extra white edges are inserted next to the partial curve (in yellow), which is necessary for the automatic appearance of X-configurations at the extremities of the partial curve at the next subdivision level

(Fig. 13(c)). These structures are designed for interpolating the extremities of the curve by the limit surface (Fig. 13(d)).

5. Summary and directions for further work

This paper presents an algorithm for the interpolation of non-iso-parametric curves, which relies on subdivision surfaces and polygonal complexes. This approach regards the solution as essentially a control mesh reconfiguration process. In contrast with previous research work that could be judged as relevant to the subject, the present approach results in precise (not approximate) interpolation, which does not require any changes to the subdivision coefficients except in well specified situations. Moreover, following this approach, efficiency does not present itself as an issue in the overall process.

Finally, the area of the research, that has not received any attention in this paper (and therefore requires further work), is the quality of the resulting surface. This is obviously related to the particular decisions that should be taken when choosing the extra points needed during the re-meshing process.

In this respect, in all the figures presented in this paper, given that no particular care is exercised when the initial control points of the mesh are selected, the changes affecting the initial state of the mesh to reach the final state seem to be tolerable. This may be attributed to the relative proximity assumed to exist between the initial polygon and the initial control mesh. Thus, if the integration phase (i.e. *Step 1*) of the algorithm passes with no anomalies, the rest of the re-meshing process should not cause any visible artifacts to appear on the final state of the interpolating surface. This observation should require a more rigid verification, however.

Acknowledgement

The research work reported in this paper is supported by a grant from the Lebanese Council for Scientific Research for the academic year 2014–2015.

ORCID

Abdulwahed M. Abbas  <http://orcid.org/0000-0002-3954-7076>

Ahmad H. Nasri  <http://orcid.org/0000-0002-2047-6693>

References

- [1] Abbas, A.: Generalizing Polygonal Complexes across Modeling Domains, Computer Graphics International, Bournemouth, June 12–15, U.K., 2012.
- [2] Abbas, A.: A Minimally-Constrained Subdivision Surface Interpolating Arbitrarily-Intersecting Network of Curves, Computer Graphics International, Nanyang Technological University, Singapore, June 8–11, 2010.
- [3] 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/10.1016/0010-4485\(78\)90110-0](http://dx.doi.org/10.1016/0010-4485(78)90110-0).
- [4] Ferguson, D. R.; Grandine, T. A.: On the construction of surfaces interpolating curves: A method for handling nonconstant parameter curves, ACM Transactions on Graphics, 9(2), 1990, 212–225. <http://dx.doi.org/10.1145/78956.78961>.
- [5] Halstead, M.; Kass, M.; DeRose, T.: Efficient, fair interpolation using Catmull-Clark surfaces, In Proceedings of SIGGRAPH 93, Computer Graphics Proceedings, Annual Conference Series, August, 1993, 35–44.
- [6] Hu, Y.-P.; Sun, T.-C.: Moving a B-spline surface to a curve—a trimmed surface matching algorithm, Computer-Aided Design, 29(6), 1997, 449–55. [http://dx.doi.org/10.1016/S0010-4485\(96\)00086-3](http://dx.doi.org/10.1016/S0010-4485(96)00086-3).
- [7] 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>.
- [8] Nasri, A.; Abbas, A.; Hasbini, I.: Skinning Catmull-Clark Subdivision Surfaces with Incompatible Cross-Section Curves, Pacific Graphics, 2003, 102–111.
- [9] Piegl, L.; Tiller, W.: The NURBS book (2nd Ed.), Springer-Verlag, New York, 1997. <http://dx.doi.org/10.1007/978-3-642-59223-2>.
- [10] Schaefer, S.; Warren, J.; Zorin, D.: Lofting curve networks using subdivision surfaces, Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing, ACM New York, NY, USA, 2004, 103–114.