www.cadanda.com Computer-Aided Design and Applications

Reconstruction of Branched Surfaces: Experiments with Disjoint B-spline Surface

Amba D. Bhatt[1], Archak Goel[2], Ujjaval Gupta[3] and Stuti Awasthi[4]

[1]Motilal Nehru National Institute of Technology, Allahabad, amba_bhatt@yahoo.com
[2]Motilal Nehru National Institute of Technology, Allahabad, archakgoel90@gmail.com
[3]Indian Institute of Science, Bangalore, ujjaval88@gmail.com
[4]Motilal Nehru National Institute of Technology, Allahabad, awasthi.stuti@gmail.com

ABSTRACT

Reconstruction of branched surfaces is an important class of reconstructing problems with wide applications in modeling of branched structures in fields like biomedical and automotive. For multiple branching, the sections are normally segmented, reconstructed and then combined together, and thus involve serial processes that are computationally expensive. Moreover, maintaining continuity between such reconstructed patches while preserving the topological features is also difficult. The present work reconstructs the disjoint surface with the help of a single equation from sectional data. At the same time it addresses the requirements of continuity, geometric and topological complexities. Behaviour of surface contours by varying control points is studied and then by electing an appropriate arrangement of control points and manipulating the control polyhedron a continuous surface is generated. The systematic development of the method is discussed with the help of experiments with excellent results for bifurcations and multiple-bifurcations.

Keywords: surface reconstruction, B-spline surface, branching, reverse engineering.

1. INTRODUCTION

Surface reconstruction is a process where a desired shape is constructed using the scanned data. Some of the applications are modeling of human airway tree, femur, human vasculature, automobile parts (like branched manifolds and tubular frame parts) and terrain reconstruction. This reconstruction, at times, involves branching (furcation) i.e. creation of disjoint surfaces from one contour to two or more contours in adjacent plane. This branched surface may have various geometrical (like continuity and planarity) and topological (like shape and multi-furcation) complexities. In absence of branching the problem is greatly simplified and can be handled with simple surface interpolation techniques, depending upon the accuracy required and computation that can be handled; however branching makes this problem more difficult.

1.1. Previous Related Work

Different techniques in the past have addressed the problem to varying levels of complexities. Methods based on triangulation [7,11] have also been developed to reconstruct branched surfaces but continuity requirements could not be met. For achieving better surface quality and handling more features, a lot of work [1,3,5,9,10] has been done towards stepwise surface construction and then gluing them together to get a single surface. Chai et al. [1] used PDEs to reconstruct C^1 continuous surface from contour map to solve with branching problem in terrain reconstruction. Sub-surfaces bounded between contours at two neighboring height levels were generated. But an extra step to integrate these sub-surfaces as a single C^1 continuous surface was required. In medical field, Volkau et al. [9] reconstructed surface to human normal cerebral arterial system by creating the furcation surface and two tubular branches separately. Also, they had to use different methods for generating the furcation and the tubular surface. Again, gluing of these three surfaces was required in their method. Kim et al. [5], generated C^2 continuous branched surface using non uniform B-spline but they created surfaces of two branches separately and then joined them to obtain a single surface. Approaches based on joining of the surfaces leads to more complexities as more parameters are extracted and processed. For example, Chai et al. [1] needed

Taylor & Francis
Taylor & Francis Group

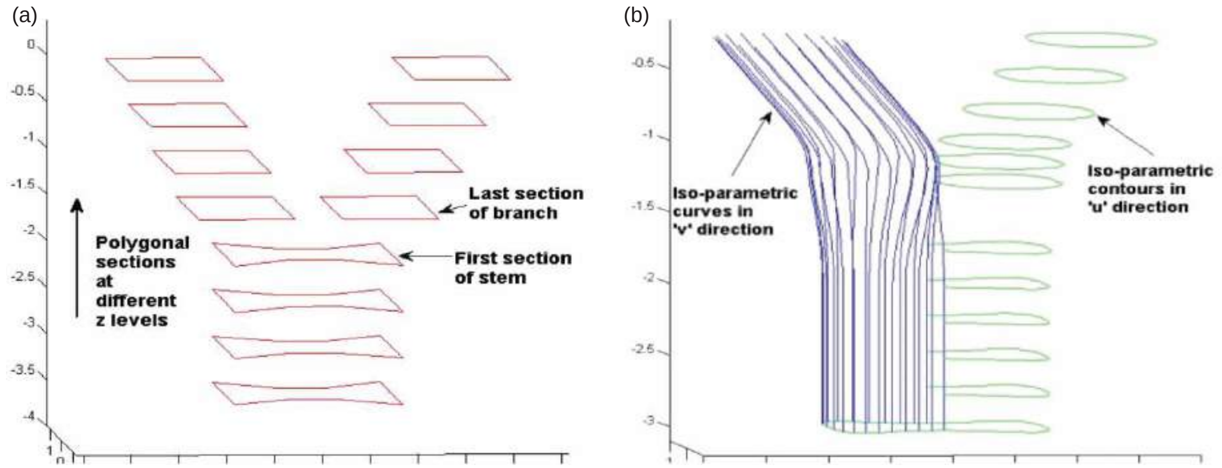

Fig. 1: Terminology and representation: (a) shows polygonal sections at different z levels, (b) shows two parametric directions in relation to the disjoint surface model.

to estimate the gradients at the contours to ensure continuity between two surfaces being joined. This information is not given explicitly by the contour map and requires an extra estimation process. The method proposed in this paper requires no such separate steps to build different parts separately, which saves computation cost and simplifies the process.

Some of the techniques [2–4] in the direction to get smooth surfaces required an additional hole filling step. Guo et al. [3] used parallel transport frame to obtain half tube sweeped surfaces which had to be patched with additional Bezier surfaces. Moreover, their method could not solve completely the problem of frame blending required to join the surfaces as one single surface. Periodic B-spline was used by Jaillet et al. [4] to create the branched surface but there was tearing of surface towards the stem. These holes were filled using an additional Coons patch. Subsequent hole filling using Gordon Coons patch was also used by Gabrielides et al. [2]. They split the problem in a sequence of local Hermite surfaces and integrated them with G^1 continuity, after which the hole was patched. Unlike these methods, hole-filling and joining steps are not required in our method. These serial processes become complex and involve large computations especially for applications like modeling of human airway tree [6].Therefore, in this paper we present a new and simpler method to reduce the steps, by adopting a single equation to create disjoint B-spline surface which can have different orders in both parametric directions (u and v).

Moreover, the techniques proposed by [3,9,10] are limited to cross-sections with circular profile. Method proposed in [9] is limited to bifurcations and Guo et al. [3] had to modify their method for bifurcation and trifurcation cases. The method discussed here can address these limitations.

1.2. Overview of the Paper

The present work is related to reconstructing disjoint surfaces from the data points available. The data points are presumed to be control points of the B-spline surface that will approximate the control polyhedron. In this paper all the experiments are carried out for B-spline surface of order 3, in both the parametric directions.

Fig. 1(a) shows the input sections used for surface interpolation. Here, the control polyhedron sections are taken as simple polygons for ease of understanding. These sections lie in parallel planes and are stacked one above the other at certain distance in z direction.

At some z levels we have multiple polygonal sections and at others we have single. The part of polyhedron containing multiple polygonal sections is called branch while the other part having single polygons, is called stem. Further, these sections act as the "control polyhedron" for the disjoint surface. First, by experimentation on curves we found a generalized formula to define the range of parameter u to get disjoint B-spline curves. This technique was extended to surfaces with election of an appropriate arrangement of control points and manipulation of control polyhedron. In the subsequent subsections the adopted methodology is explained with conceptual figures.

2. METHODOLOGY

In this work it has been assumed that the control polyhedron has been provided. For actual problems, where the curve has to be interpolated on actual data, the control polyhedron can be derived from the data set by using methods described in [8]. This polyhedron is now used to generate surface using B-splines.

2.1. Disjoint B-spline Curves

The surface developed in this experiment, using single equation, is disjoint in one direction (u). Generating bi-parametric surface can be inferred in a way that, while traversing along v, surface contours/curves in u direction are generated at every v. Hence, the methodology to obtain disjoint B-spline curves using single equation is discussed in this section. For creating 'n' number of disjoint curves from one set of control points, 'n' number of range sets were selected out as explained as below.

$$
\begin{aligned}
u_{min} &= x_k \\
u_{max1} &= x_{((k+(n1+1-(k-1))))} \\
&= x_{(n1+2)} \\
u_{min2} &= x_{((k+(n1+1-(k-1)))+(k-1))} \\
&= x_{((n1+2)+(k-1))} \\
u_{max2} &= x_{((k+(n1+1-(k-1)))+(k-1))+(n2+1-(k-1))} \\
&= x_{((n1+2)+(n2+2))}
\end{aligned}
$$

where $n1+1$, $n2+1$ are the control points which make up the first and second disjoint curve respectively. Here, x and y are matrices of knot vectors corresponding to u and v direction respectively. There are no additional points between the two disjoint curves. Hence, the range of parameter u for first disjoint curve is $u_{min1} \le u_1 \le u_{max1}$; and similarly for second one is $u_{min2} \le u_2 \le u_{max2}$. These ranges are then concatenated in one set (Eqn. (2.1)) and used for generating all disjoints curves/contours of the surface with single equation.

$$
u = \bigcup_{i=1}^{n} u_i \tag{2.1}
$$

Fig. 2 shows the results of the methodology adopted for generating disjoint curves. This technique is applied to develop disjoint closed B-spline curves and then extended to surfaces. This method preserves the original nature of curve which can be seen in the figure.

2.2. Pseudo Code for Generating Disjoint B-spline Surface

The technique described in last section is implemented to obtain branched surface using following algorithm. The variables used are same as described in previous discussions. Matrix "D" contains information about no. of disjoint curves making up contour(s) at one z level and number of vertices for generating each disjoint curve.

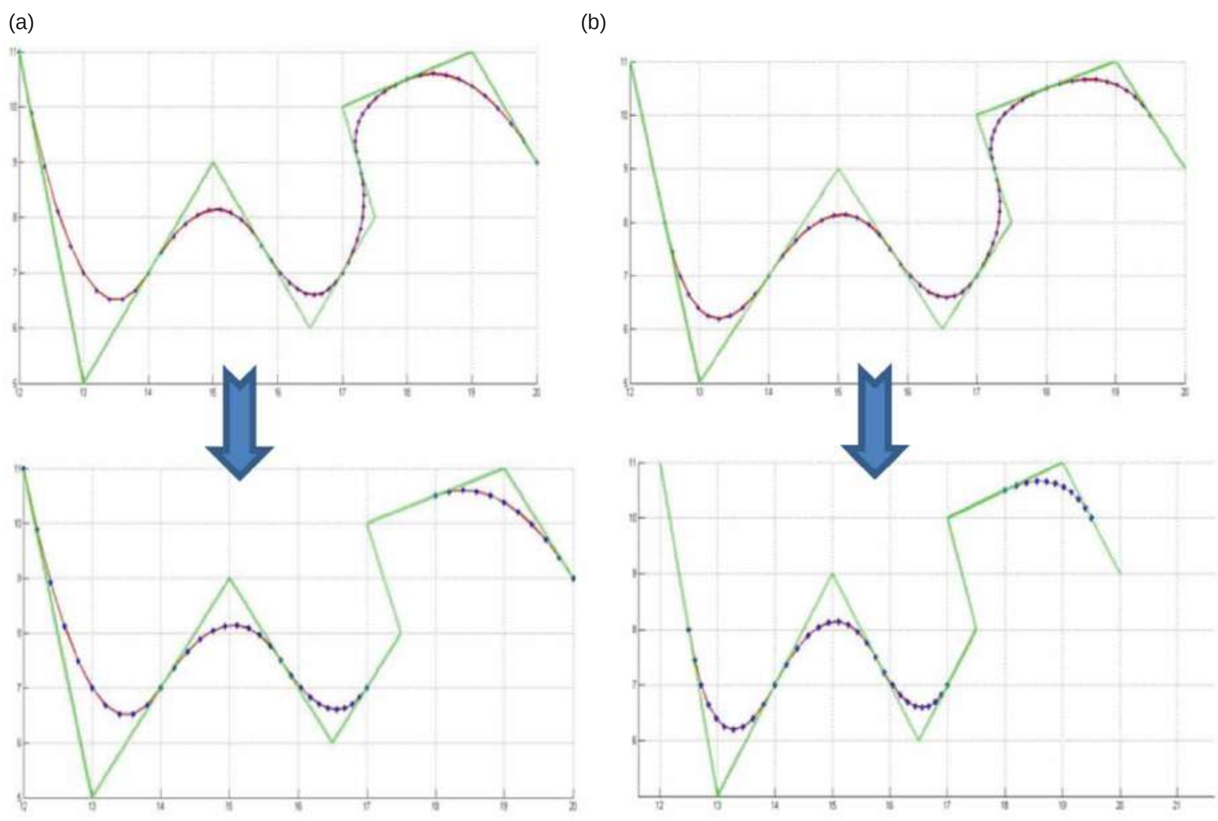

Fig. 2: Disjoint B-spline curves. Curve is shown in blue and control polygon in green. In red color are points along the curve: (a) and (b) shows original and result figure for open uniform and periodic B-spline respectively.

Input-
Degree of surface required k (along u) and l (along v)
$D = [D(1)\, D(2) \cdots D(t)]$
Calculating knot vectors for v and (matrix y).
Calculating range of v using standard method for B –spline as in [8] and storing all values in V matrix
Calculating Basis Functions $M_{j,l}$ *at each element i.e.* $v = v_i$
Obtaining range set for disjoint contours in u direction:
Initialize $u_{min} = y_k$
While condition (until range set of u for all disjoint curves is not calculated)
$u_{max} = u_{min} + D(t) - (k - 1)$
Producing all values from u_{min} *to* u_{max} *with a fix interval and inserting in common matrix* (U) *for taking union with all range sets.*
Calculating u_{min} *and* u_{max} *for next set of range:*
$u_{min} = u_{max} + k - 1$
$t = t + 1$
end while loop
Using U as the input matrix for calculating Basis Functions at each element $u = u_j$
Obtaining final surface contour elements using B-spline surface equation as in [8]

$$S(u, v) = \sum_{(i=1)}^{(n+1)} \sum_{(j=1)}^{(m+1)} B_{(i,j)} N_{(i,k)}(u) M_{(j,l)}(v) \qquad (2.2)$$

End of For loop
End of program

3. DISCUSSION OF EXPERIMENTS

Sequential experiments lead to the development of the final technique. To understand, the logic of the methods used and the possible cases where this method can be applied, the experiments carried out are discussed.

3.1. Polyhedron Representation and Definition

Fig. 3 demonstrates the manner in which control points of branch's polygon will be connected to the control polygon of the stem. For example the point labeled '1' on left branch will be connected to the corresponding point labeled '1' on the stem's polygon of first model (subsection 3.2).

Similarly, point labeled '7' on right branch's polygon will be connected to the corresponding point on stem's polygon .The multiple labels on a point imply that they have multiple connectivity with stem's control points in accordance with the given label. A label given '7' and '11' will be connected to '7' and '11' on the stem. This way entire control polyhedron is defined. Notice that sum of points of stem is equal to total number of points on both the branches. The arrows are the guides depicting the direction in which the side is traversed and the number of times it has been traversed. In Fig. 3. the double arrows on stem's sides indicate that the side has been traversed twice in definition of control polygon.

3.2. First Model

In the first experiment (Fig. 3.), the control polyhedron of the surface was produced by repeating the control polygons at each level in a particular fashion. Here, the branch polygon is made of four distinct geometric points which are wrapped to get closed contours. The control polygon of stem is geometrically a rectangle of four distinct points but actually two polygons (as that of the branch) are superimposed because number of control points at each level has to be constant.

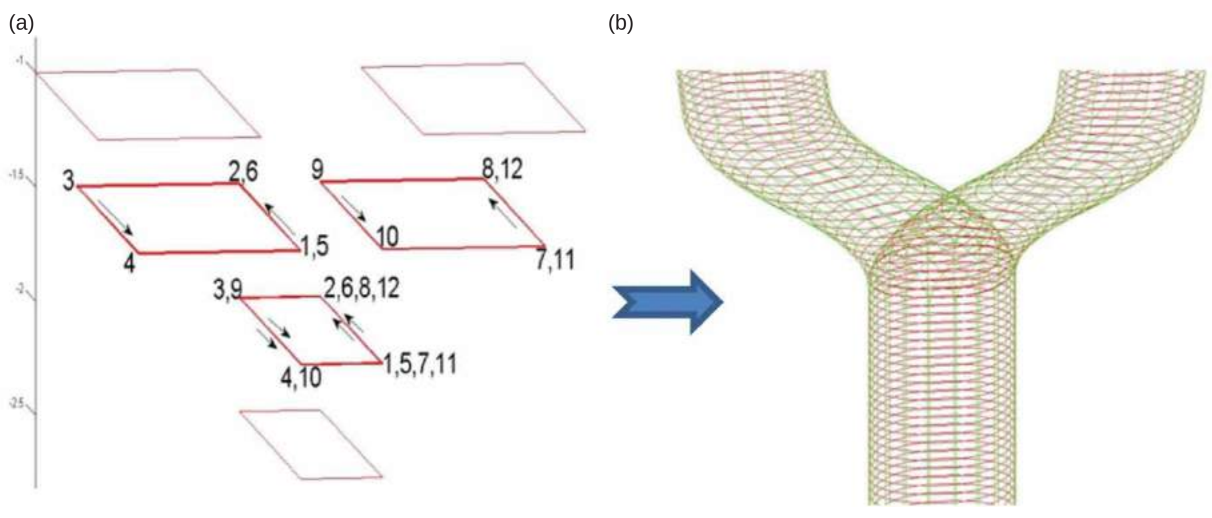

Fig. 3: Polygon section geometry (a) and relative connectivity explained using first model and (b) the surface generated by this arrangement (b).

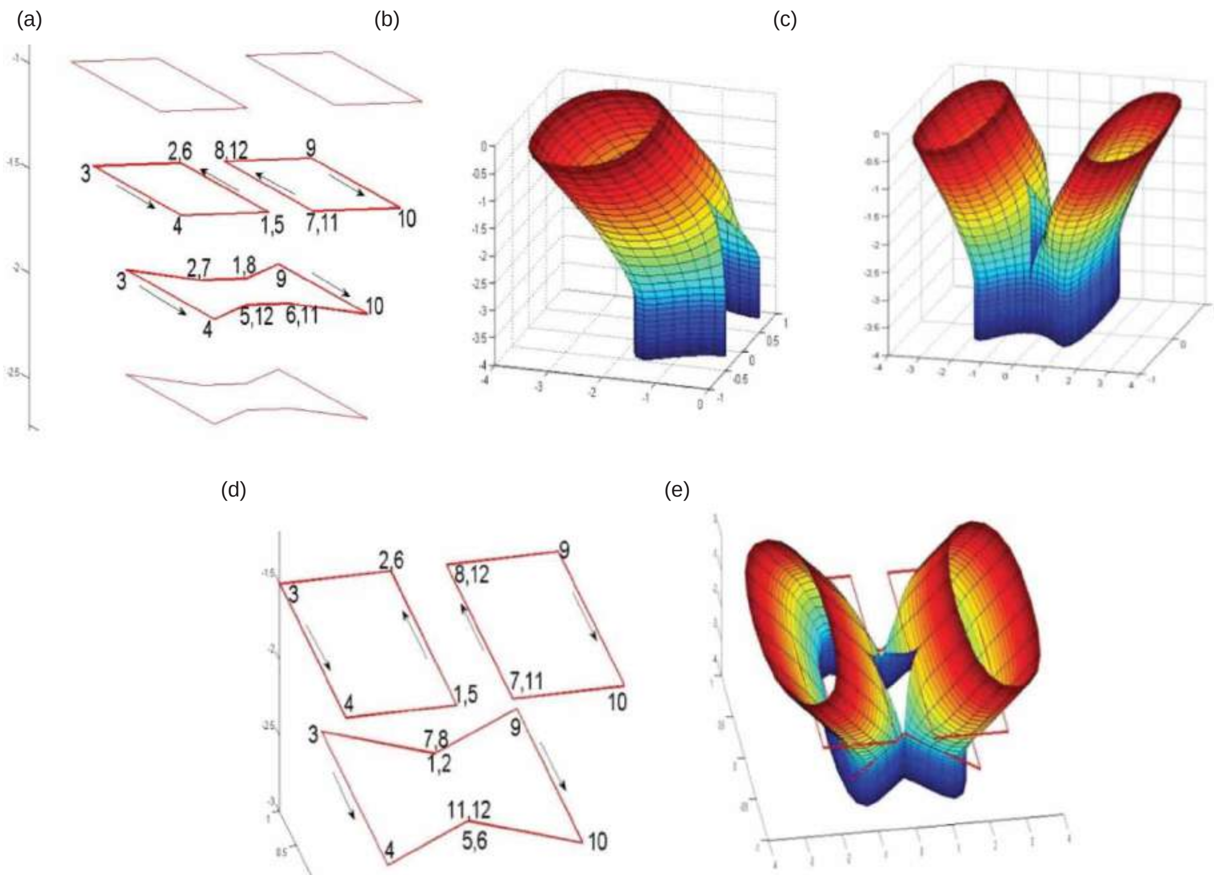

Fig. 4: Second model and surface interpolation results: (a) shows the arrangement of control points (b) shows dissected branch's surface model (c) shows the complete surface model of branch. (d) shows change in control points for second model and change in result (e).

Although the surface is C^1 continuous along u and v direction but there are intersecting iso-parametric curves. This leads to non-hollow disjoint surface or noise at the junction area, which is not required. Therefore, to avoid such inconsistency the control point arrangement was manipulated in the second model, with the aim to ensure separation of branches, when they join at stem.

3.3. Second Model

The arrangement of points was changed to follow a different approach. The stem's polygon would be made of two distinct open polygons such that they join to appear one polygon. The fact that periodic B-splines are tangent at mid-point of every segment for order 3 is used to maintain C^1 continuity.

Here, two arrangements are shown for the second model. In the first arrangement (Fig. 4(a).) the control points '1' and '2' at stem are wrapped while in second arrangement (Fig. 4(d).) these control points are repeated to clamp the B-spline curve. Consequently in the later, the continuity at junction of left and right side branch, along all the stem contours, was reduced to C^0 (in object space) in u direction. Continuity in v direction is still C^1.

In comparison to previous experiment (Fig. 3.) the branch has been successfully separated at the junction in addition to the C^1 continuity in both directions (Fig. 4(b). and Fig. 4(c).). But a new problem of tearing of the branch along v direction is evident. This led to opening of surfaces in some portion (as shown in dissection of branch Fig. 4(b).)

It was observed that the opening always starts between the last and second last sections of the branch, for case of third order. This observation will be utilized in subsequent sections to produce more successful results.

If second arrangement is to be used then, at the junction (position of '1' and '2' points in Fig. 5(d).) of left and right polygons of the stem, two additional control points are required–one each on either side. They have to be arranged such that these three geometric positions form a straight line. This arrangement will ensure that curves corresponding to both polygons will be aligned at 0 or 180 degrees with respect to each other.

These experiments necessitated the understanding of reasons for the tearing of surface so that they can be circumvented to resolve the problem and generate a Y shaped disjoint surface with single equation.

4. UNDERSTANDING THE PROBLEM OF SURFACE INTERPOLATION

Fig. 5. shows the curves obtained after multiplication of each set of basis functions. The user given control polyhedron is in red color. As per Eqn. (2.1), after multiplying $M_{(j,l)}$ to the input control polyhedron $B_{(i,j)}$, blue curves are obtained. The dotted lines show the intersecting curves, which lead to formation of hole. When this curve was multiplied second time by basis functions of u direction $N_{(i,k)}$, the final surface contours (green color, Fig. 5(b).) were obtained. Therefore, the polyhedron in red acts as control polyhedron for the blue colored curves and the points on blue curves act as control polyhedron for the curves in green color (actual surface defining contours).

In this figure we are obtaining closed surface contours along u direction (in green color), at $v = 1$ because, the blue curve 1 and 2 does not diverge from blue curve 5 and 6 respectively. But at $v = 2$, these curves start to diverge. As a result, the mid-point of first and the last edge of their (surface contours) control polygon are not coincident which is required for getting closed B-spline curves of 3^{rd} order. This leads to opening of final surface contours along u direction. In wrapping method for generating closed periodic B-spline curve, the first and last edge of polygon is overlapped to get one line segment. It was observed that the surface contours open about midpoint of this line segment and in a direction parallel to it. Once this behavior is understood, efforts were made to somehow avoid the blue curves to diverge by manipulating the control polyhedron.

5. FURTHER EXPERIMENTATION

Experiments based on the findings and understanding from above section have been discussed here.

5.1. Closed Surface with C^0 Continuity at Junction

It is known that repeating the control point at a position, according to the order of curve, forces the curve to pass through that geometric position. Here, the last two sections of the branch were superimposed to give one geometrical section. This approach was used to constrain the blue curves (Fig. 5.) to pass through the user defined control points and ensure their wrapping. Hence, the generated surface contour opened only after the last section of branch. Moreover, the edges having overlapping first and last segment, in each of the polygons at last section (as in Fig. 4(a).), were overlapped with each other (Fig. 6(c).). Now, as the points about which the surface contours of left and right branch open are coinciding and the opening direction for both is parallel, these two open contours join together to give one closed surface contour. Hence, a closed surface is generated.

Some contours near the junction are C^0 continuous, where both branches meet. All other contours have C^1 continuity.

These are not final results as this method is further improved to get enhanced continuity.

5.2. Next Successful Experiment with G^1 Continuity.

Here the fact obtained from section 3.3 is utilized, that the hole starts between last and second last sections of the branch. So, one edge from each of the last two polygonal sections of both the branches, is made overlapping (Fig. 7(a).). The sections which were superimposed in subsection 5.1, are

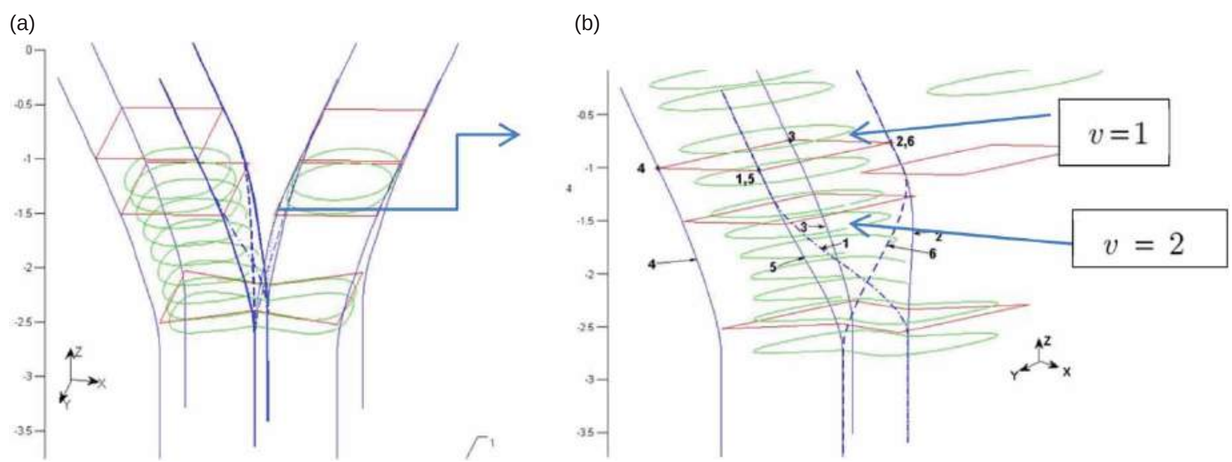

Fig. 5: The wireframe of the generated surface. (a) shows the control polygon(in red) of the branch and the subsequent curves generated (b) shows enlarged view of one side of branch. The labels denote the order in which the points from blue curve are taken by the algorithm to plot the surface contours.

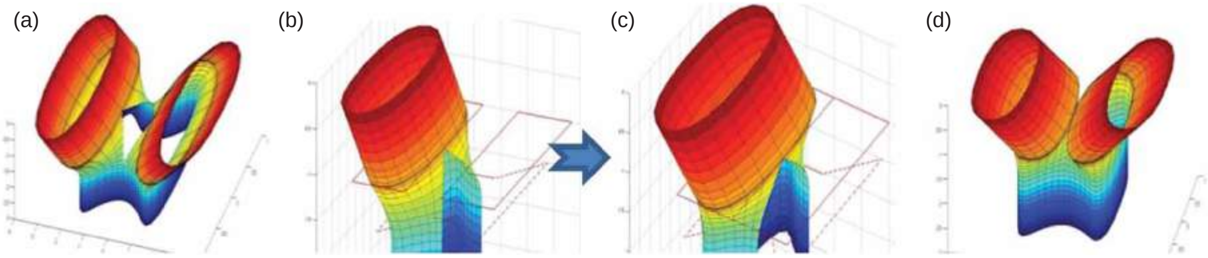

Fig. 6: Depiction of unsuccessful (a) and successful (c) surface generation (dissection and full surface). (b) is dissection of (a) and when (b) is changed to give last section like (c) a closed surface is generated (d). The dotted lines in red are control polygon at stem area.

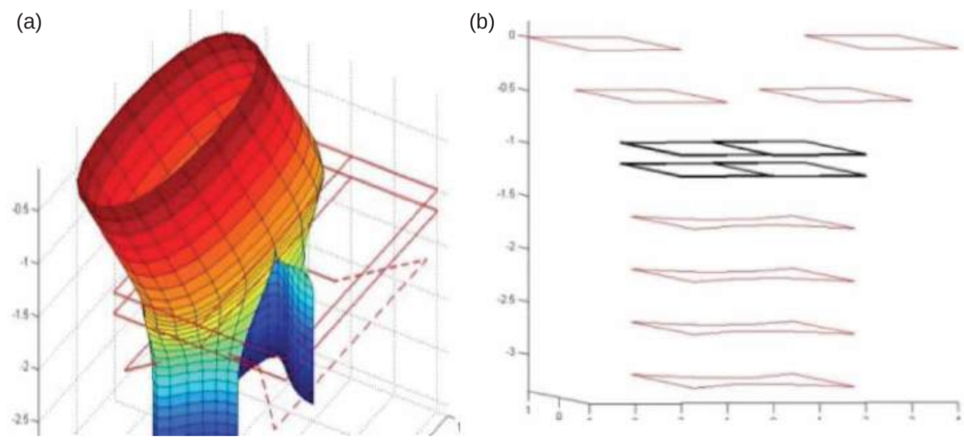

Fig. 7: The modifications in control polyhedron. (a) shows disjoint surface with last two sections of the branch, in which polygons share a common edge. (b) shows the sections/slices to be inserted in actual application.

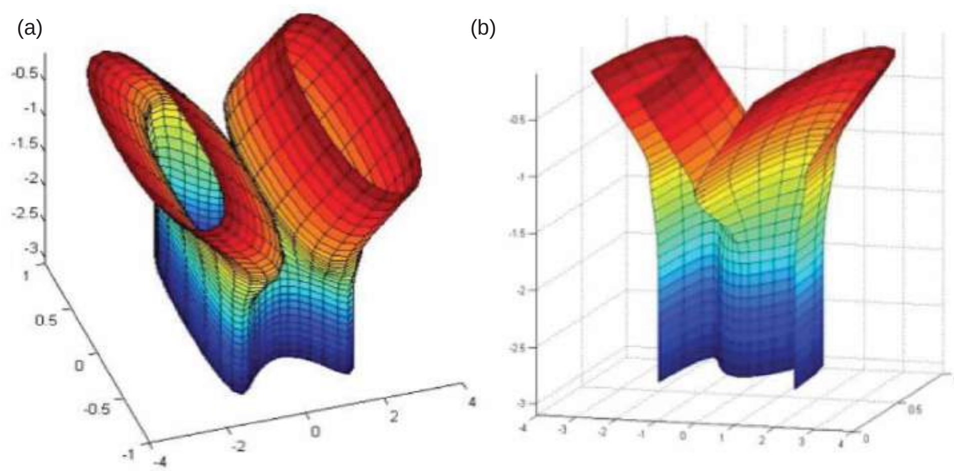

Fig. 8: G^1 continuous branch and its open section.

not superimposed here. The arrangement of control points is that of second model (Fig. 4(a).).

Above discussion implies that in actual problem two additional slices need to be inserted as shown in Fig. 7(b). They should have same arrangement of points as in Fig. 4(a).

The surface is closed. The branched surface (Fig. 8(a)). is improved from previous experiment and now it is G^1 continuous in parametric space.

6. ASYMMETRIC BRANCHES

In branched surfaces, it is possible that at the furcation, surface on one of the side of the furcation may be covering a larger area with a different shape than the surface on other side. This may constrain the overlapping edges of inserted slices' polygons to not be able to overlap end to end. Whereas for cases discussed uptil now, the nature of furcation region

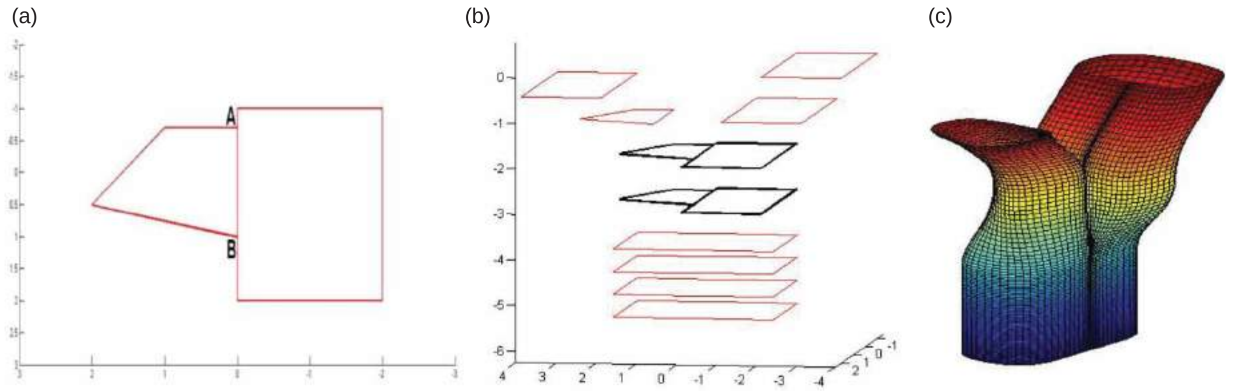

Fig. 9: The asymmetric model: (a) shows top view of one of the inserted sections. (b) shows the control polygon with inserted sections (in black) (c) is the reconstructed surface.

was such that the common edges of both polygons, at the inserted slices, were overlapping end to end. This will lead the midpoints (about which the contours of surface open) of the overlapping edges to be non-coincident. Therefore the contours of left and right branches will not be able to join to give a closed surface. To address the problem of this new possibility it requires a slight change in the methodology which is discussed in this section.

To create disjoint surface in this case, two control points are repeated, one each at start and at end, in each polygon, at any one position along the common line AB (Fig. 9(a).). This constrains the contours of both the branches to open symmetrically about one single point and when they meet at junction, a closed surface is obtained. Fig. 9(c) is the result of the new arrangement with G^1 continuity along u. This is the advantage of method discussed in this paper that it is flexible to fulfill shape requirements of the branched surface and simply the arrangement of control points has to be varied.

7. SURFACE WITH UNIFORM KNOT VECTORS

With the arrangement of points as in 6.2, and using uniform knot vectors for u and v direction, led to generation of same overall shape (Fig. 10(a).) but more contours in v direction. If knot vectors for v direction are changed to uniform knot vectors while that in u are kept periodic, there was no change in overall shape. There was difference in surface contours position only at ends (Fig. 10(b).). Uniform knot vectors in v direction can be more suitable where the start and end of surface is required at a predetermined position. For order of surface higher than 3, in v direction, requires that $(order - 1)$ consecutive slices are to be introduced in stem section with same

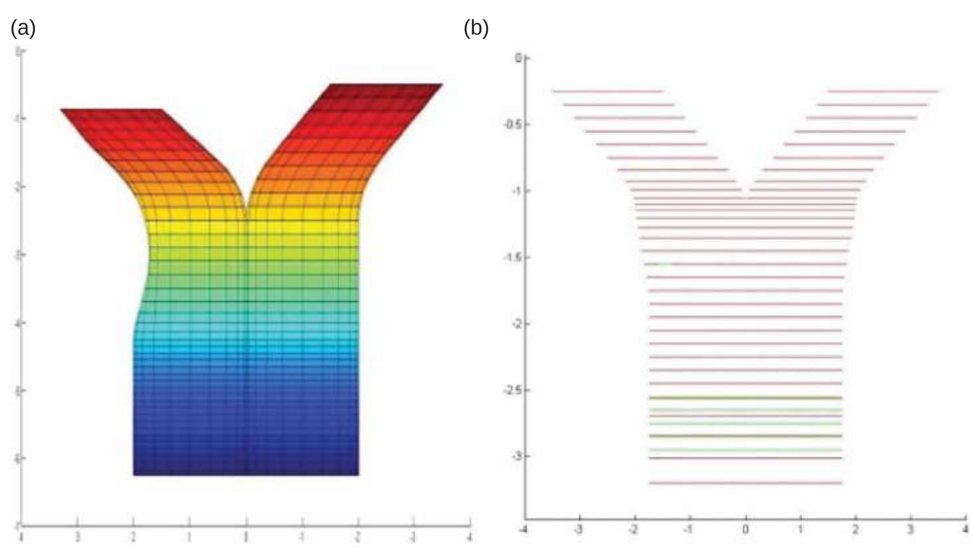

Fig. 10: (a) is branch with uniform knot vectors. For a different branch (b) is image of super-imposed surface contours with periodic (in green) and uniform (red) knot vectors only along V.

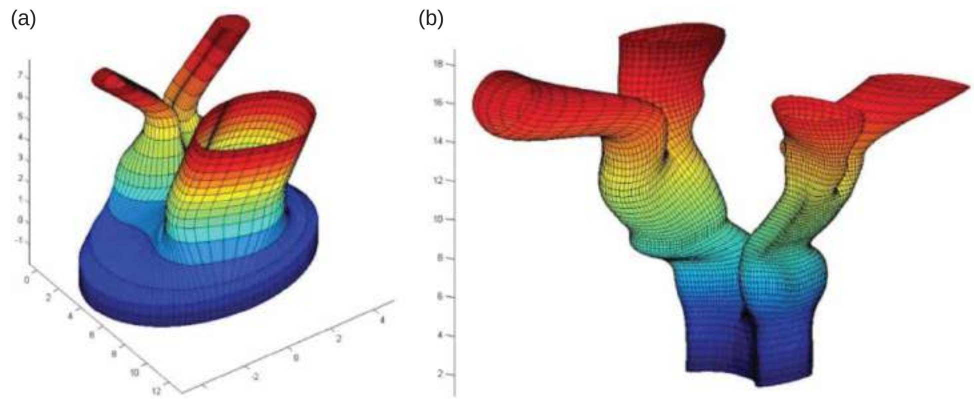

Fig. 11: Multiply branched surface model.

scheme of arrangement of points as in 6.2 or 5.2 (as per shape requirements). This can be used to meet higher continuity requirements.

8. MULTIPLE BRANCHING USING SINGLE EQUATION

Multiple branches are found in trees, lungs, human body (airway tree in lungs) and many other objects of interest. Multiple branches were created successfully using the same algorithm as mentioned in 2.2. The arrangement of control points in Fig. 11(a). is developed using the same scheme as in section 6. Here, two branches emanate from the stem and one of the branches is further getting branched in two more branches.

Fig. 11(b) is generated using control point arrangement as in section 5.2. But owing to shape requirements the polygons or sections here are not stacked parallel rather they are aligned along the cross-sections. This is a two stage bifurcation, generated using single equation of B-spline surface, for order 3.

9. CONCLUSION

The method discussed in this paper has never been used before and opens a new direction for constructing branched surfaces using single equation of B-spline surface. Also being a parametric model, meshing is almost automatic and can be used for finite element analysis. Thus, it proves to be easier than other approaches. Moreover, not being restricted to a particular shape of cross-section, it can handle the requirements of irregular cross-sections, asymmetric branching and varying cross-section profiles. Since the method gives a minimum of G^1 continuous (in parametric space) surfaces, it removes additional steps of stitching. The continuity here can be higher than C^1 in v direction. Developing multiple bifurcations further minimizes steps required to integrate each bifurcation with others. Thus, it is a tool which can address topological and geometrical requirements while minimizing the steps and complexity involved. Some of the applications can be design of human airways, automobile manifolds and frame components.

10. FUTURE WORK

This method has given us encouraging results and it can be extended to multi-furcations and topologically other types of branches for which work is going on. Surface reconstruction is a process involving series of steps like data extraction and processing, slice insertion, and fitting. Most of steps have pre-existing techniques but specific methods to comply with this technique remain to be experimented and implemented. For e.g. algorithm for arrangement of the control points in each polygon has to be automatized for multiple as well as singly bifurcated surfaces. The method for slice insertion position in the actual data remains to be experimented to get a good estimate of branching position. Thereafter, it will be applied to data extracted from real world models.

REFERENCES

[1] Chai, J.; Miyoshi, T.; Nakamae, E.: Contour interpolation and surface reconstruction of smooth terrain models, In Visualization'98. Proceedings IEEE, 1998, 27–33.

[2] Gabrielides, N. C.; Ginnis, A. I.; Kaklis, P. D.; Karavelas, M. I.: G1-smooth branching surface construction from cross sections, Computer-Aided Design, 39(8), 2007, 639–651. doi:10.1016/j.cad.2007.05.004

[3] Guo, J.; Li, S.; Pan Chui, Y.; Qin, J.; Ann Heng, P.: Mesh quality oriented 3D geometric vascular modeling based on parallel transport frame, Computers in Biology and Medicine, 2013. doi:10.1016/j.compbiomed.2013.04.004

[4] Jaillet, F.; Shariat, B.; Vandorpe, D: Smooth reconstruction of branching objects with B-splines, Revue internationale de CFAO et d'informatique graphique, 16(2), 2001, 183–201.
[5] Kim, H. S.; Kim, Y. H.; Choe, Y. H.; Kim, S. M.; Cho, T. S.; Mun, J. H.: A new modeling method for objects with branching problem using non-uniform B-spline, In Computational Science-ICCS 2004, Springer Berlin Heidelberg, 2004, 1095–1102. doi:10.1007/978-3-540-25944-2_142
[6] Kitaoka, H.; Takaki, R.; Suki, B.: A three-dimensional model of the human airway tree, Journal of Applied Physiology, 87(6), 1999, 2207–2217.
[7] Reinhard, K.; Schilling, A.; Straßer, W.: Reconstruction and simplification of surfaces from contours, Computer Graphics and Applications, Proceedings. Seventh Pacific Conference on. IEEE, 1999, 198–207.
[8] Rogers, D.F.; Adams, J. A.: Mathematical Elements for Computer Graphics, Chapter-6, Tata McGraw-Hill, New Delhi, 2002, 445–458.
[9] Volkau, I.; Zheng, W.; Baimouratov, R.; Aziz, A.; Nowinski, W. L.: Geometric modeling of the human normal cerebral arterial system, Medical Imaging, IEEE Transactions, 24(4), 2005, 529–539. doi:10.1109/TMI.2005.845041
[10] Ye, X.; Cai, Y. Y.; Chui, C.; Anderson, J. H.: Constructive modeling of G1 bifurcation, Computer Aided Geometric Design, 19(7), 2002, 513–53. doi:10.1016/S0167-8396(02)00131-0
[11] Zhiyi, Z.; Konno, K.; Tokuyama, Y.: 3D terrain reconstruction based on contours, Computer Aided Design and Computer Graphics, Ninth International Conference on IEEE, 2005, 6.