

Aspects to be considered in case of variable surfaces modelling

Andrei Mihalache*, Gheorghe Nagit, and Vasile Merticaru

Gheorghe Asachi Technical University of Iasi, Department of Machine Manufacturing and Industrial Management, Blvd. Dimitrie Mangeron no. 59A, Iasi, Romania

Abstract. Proper triangulation will bring benefits regarding the numerical conditioning of nodes within the point cloud but will also allow certain nodes to be able to move around those that are neighboring them. In this way, surface features as chained regions or curves will have a certain degree of freedom which will allow them to slide one towards another. At a more careful evaluation there is a major drawback in terms of lack of possibility to take into consideration the operations which depend upon approximate shapes inside the ideal shape evaluation process. In order to be able to control the accuracy of the process there is an impetuous need for all surfaces to be described functionally. Any triangulation procedure has to be able to allow control over the analyzed surface topology and any other after modifications: dividing the surface along an incorporated curve in order to define a new edge or sticking two surfaces together in order to form a mutual boundary edge.

1 Introduction

This paper's goal is to present an identification procedure of points within a point cloud that represent the boundaries and in such way to be able to define the entire surface based on those points. This procedure would move the nodes inside the point cloud around the neighboring already considered points. This technique is somehow similar to general triangulation procedures that are presented in the scientific literature. The proposed procedure will not only lead to a noise reduction inside the point cloud and point randomly distributed to be ordered but will also maintain a higher quality of the triangulation process of nodes from the point cloud as they are identified and thus considered. A proper triangulation will bring benefits regarding the numerical conditioning of nodes within the point cloud but will also allow certain nodes to be able to move around those that are neighboring them. In this way, surface features as chained regions or curves will have a certain degree of freedom which will allow them to slide one towards another. As other authors state it has been noticed that by combining a Laplacian procedure in finishing of neighboring nodes with a Delaunay triangulation we may achieve satisfying results [1]. In order for the proposed procedure to work we will assume that instead of considering a surface being described by a single empty circumference circle the authors will consider it

* Corresponding author and co-author(s): andrei.mihalache@tuiasi.ro

as described by a sum of empty circumference spheres. In such way we will maximize the minimal value of the considered angle inside triangulation process (measured in 3D space). The procedure proposes to triangulate in an optimum way any type of surface through a series of edge flip operations. Those operations should keep and maintain the initial topology of any surface in such a way that will always be a valid mesh which would allow re-triangulations of the same surface as the process advances [2].

2 The triangulation procedure

In order to be able to approximate a free surface, as others have mentioned, we will need a very precise description of the surface's 3D limits within the point cloud. Also a very significant reason to succeed is to have a rigorous view upon the surface boundaries no matter the chosen approach: either a point interpolation which will take into account the curve radiuses of the considered boundaries or an approach based on the consideration that limits are used for cutting through the overlapped and contour fitted surface [3]. The method proposes a procedure for point detection that is representing the boundaries. It aims to fit a cylindrical section along all boundaries of the surface that has been defined by the point cloud in such a way that the cylindrical surface would touch at any time two points within the point cloud between two rotations and the direction of the cylinder's axis to be parallel at the normal of the point cloud (Figure 1).

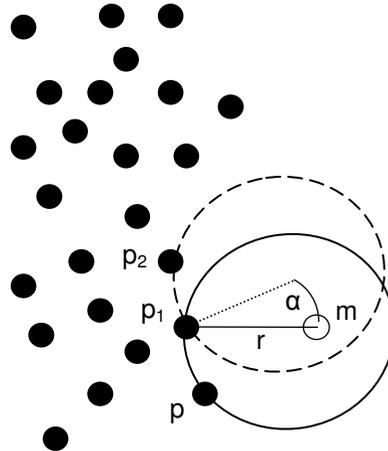


Fig. 1. The rolling process of a cylindrical surface in order to detect the boundaries of a surface described by a point cloud.

An express requirement of the proposed method is that the interior of the cylindrical surface to be free at any time. That implies that it should not contain any point from the point cloud which will lead eventually to the description of a polyline especially drafted for the boundaries of the analyzed surface if the order of connecting the boundary points is respected. The procedure will identify a new boundary point when two points already are in touch with the cylindrical surface as depicted in Figure 2.

The point that we are looking for, touching the cylindrical surface must be in an area that would equal the $2r$ radius of a sphere that would in take the last point that has been identified in such a way [4]. To ease the computation process the authors have appealed to a geometric description that allows a quicker identification of points that are in the neighbor by using a 3D grid. The grid will have a search radius of $2r$. As illustrated in Figure 2 we will rotate the cylinder around the last identified boundary point which has the direction of the cylinder axis defined as rotational.

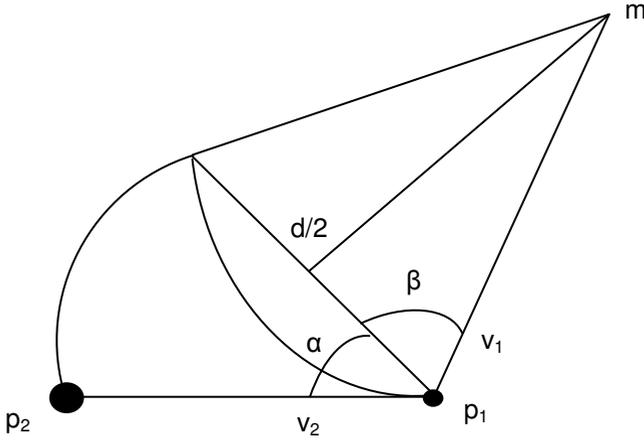


Fig. 2. The computing procedure of α in order to connect p_2 with the cylindrical surface.

Thus, we will consider p_1 as the last identified boundary point and n_1 as the estimated normal to the surface in p_1 . The m point is arbitrary chosen along the cylinder axis. In order to achieve the connection of a p_2 point with the surface of the cylinder the method supposes that the cylinder is virtually rotated around p_1 having as rotational axis the n_1 [4].

The angle is determined as following: on the plane that contains the n_1 normal we project the line $m - p_1$ that will receive a v_1 badge and for the $p_2 - p_1$ from the plane that contains the normal we will have v_2 [5]. The β angle may be computed as follows:

$$\beta = \arccos\left(\frac{\|v_2\|}{2r}\right), \text{ where } \|v_2\| = \sqrt{v_2 \times v_2} \tag{1}$$

In this case the returning angle α may be computed as follows:

$$\alpha = \begin{cases} \angle(v_1, v_2) - \beta & \text{for } \det(n, v_1, v_2) > 0 \\ \angle(v_1, -v_2) - \beta + \pi & \text{for } \det(n, v_1, v_2) < 0 \end{cases} \tag{2}$$

In this manner the procedure will ensure that for all the points inside a sphere of a $2r$ radius around the p_1 previously identified boundary point will result in α turning point. The point which will result with the smaller positive value for α will be considered as the next point along the boundary line p_2 [5].

In this way we are able to approximate the surface normal for the newly identified point. In consequence the cylinder axis is being adjusted till it gets identical with the new surface normal from the p_2 point in such way that the cylinder surface will continue to remain defined in p_1 and p_2 points [6].

The whole procedure repeats itself until a loop is identified or up till it reaches the situation where there is a need for a α having a bigger value than $\frac{4}{3}\pi$ in order to identify a new boundary point. In order for the proposed procedure to work the user must specify the r radius, the first point and eventually the initial direction in the case that edge detection is the main concern.

The dimension of the geometrical description that has been used to identify all points that are contained inside a r radius sphere around the last identified edge point is given by the known r radius of the cylinder [6].

3 Operations on surface

The authors will present some of the base operations performed in surface modelling. The goal is that of illustrating some significant aspects that may appear in case of variable surfaces.

3.1 Control curve incorporation

A control curve of a surface may be described if we choose to consider a large number of points which may be or not neighbors on the surface which would allow the control curve to pass through. The facet dividing process is used to allow such points to be placed arbitrary on the surface. The operation adds a new node in the middle of a triangular facet and then connects it with every tip of the triangle. As we identify and then insert in the mesh these anchor points the process will result with a sequence of nodes and edges at our disposal that may be attached to the already designed ones.

As the literature states one way to describe a curve defined by two points is to identify a sequence of neighboring facets that are interconnected by these nodes [7]. By combining them we shall get a polygon which contains the curve that connects the two points. As soon as the curve has been identified another similar one is being created inside the 3D space which has its points constrained in the original positions anchored in such a way that it would allow the previously identified curve to approximate it.

3.2 Topology change

Any triangulation procedure has to be able to allow control over the analyzed surface topology and any other after modifications: dividing the surface along an incorporated curve in order to define a new edge or sticking two surfaces together in order to form a mutual boundary edge [8].

The authors take into account the fact that inside 3D space there is the possibility to coexist large discontinuities in the distribution of the points from the point cloud that are produced by occlusions. Those areas won't be described explicitly and thus small creases will appear throughout the surface. That is why the proposed algorithm has to take into account all the areas that are contained within the analyzed perimeter when combining or dividing points that are describing edges. The imposed condition in this case is that those described curves are forced to coincide.

3.2.1. Curve correspondence

In order to merge two surfaces along a pair of edges the nodes that belong to the two curves must be reordered in such a way that they will correspond. A simpler procedure may be that which takes into account the possibility to refine incrementally the curve considering lesser edges. This would be possible by dividing the longest edge up till both curves will reach approximately the same number of nodes [6]. The proposed procedure would be very useful in case the user neighbors the curves in order to merge them.

3.2.2. Surface division and merging

The premise that stands at the basis of our proposed procedure states that a surface is considered as a collection of nodes and that each one of those is randomly neighboring a series of other neighboring nodes. The circle's radius that encapsulates a certain number of

neighboring points may be bigger or smaller as the described area requires it but also depending on the topological and geometrical complexity of the analyzed surface. In order to merge two curves that are situated at the surface's boundary after they have suffered operations that made them correspondent, the procedure will move the inner edges of a vertex towards another for each pair of nodes in the order they have been identified. This will transform the last nodes from edge points into mapped points and thus the previous edge curve will remain totally disconnected in respect with the resulted merged surface. This will determine a wipe out of its nodes and edges.

Dividing a surface along of a closed curve which has already been mapped represents the opposite process: edges that belong to a node are being classified as part of the interior, exterior or some region's boundary line. We then copy the boundary nodes and edges and then delete the inner edges connections from a copy of the surface's limits and also the outer edges connections from the other copy [7].

3.3 Surface intersection

A solution may be the one considering the intersection curves as self-considered edges of the resulted composed surface. In order to intersect two varying surfaces we firstly consider a curve that approximates the shape of the intersection curve of those two considered surfaces. Then those surfaces are redefined in such a manner that they would retain or interpolate this independent curve as a new edge. This manner of combining surfaces will lead to simplification in the implementation processes of common operations that are based on intersections.

3.4 Control tools for outer shapes

In order to be able to control the accuracy of the process there is an impetuous need for all surfaces to be described functionally. But this tedious work will only harden even more the process by accumulating an even larger number of operations in the case of free form surfaces [8].

That is why the authors consider that by controlling outer shapes that were previously represented we may also be able to control surface's merged regions shape. The nodes that are along edges inside an externally controlled region and those from an already mapped region will represent a blend of both neighboring and free nodes constrained topologically speaking.

This would allow that the shape of the surface from the controlled area to influence the shape of the mapped area in such a manner that the information regarding tangents or its curvature's propagation along edges.

3.5 Structured models construction

The process of combining some parameterized shapes depending on the interlaced and merged regions leads to the construction and definition of models that contain free form surfaces. The control tools types of these shapes may be organized into libraries which would allow a hierarchical distribution of shapes depending also on their ability to deform [6].

This study has led to the conclusion that control tools resulted collection serves at the construction of a similar mesh that clothes the skeleton. As the model gets refined its edges are recomposed in order to accommodate the imposed changes with respect to keeping the same general shape.

4 Conclusions

The procedure proposes an algorithm for triangulation. Applying it takes us to some considerations about the required operations that may be needed in surface modeling. Considering each surface as a collection of nodes that are neighboring randomly other nodes allows us to properly close the gaps between surfaces. Also by considering previously mapped forms or primitives we are able to analyze surfaces in such a way that many may be appropriate with the ones already considered thus making it easier to process large volumes of information.

Our approach considers the surface as a sum of hollow circumference spheres. From the triangulation's point of view it's easier to map the surface by means of known algorithms as it is more accurate. It also allows more nodes to migrate towards each other thus making it possible for them to unite inside the procedure. As we rotate the theoretical cylinder around considered points from the surface we are able to approximate the surface normal making it possible to move the cylinder until its revolution axis becomes identical with the normal.

An immediate application of the proposed procedure may be the one that identifies points that are randomly scattered inside a point cloud by using a cylindrical surface which may be rolled out in order to describe in space a curve. In such a case instead of considering the normal by means of the proposed procedure we may estimate tangents by aligning the cylinder axis with a perpendicular arbitrary direction to the considered tangent one.

At a more careful evaluation upon the proposed method we may observe that a major drawback would be the lack of possibility to take into consideration the operations which depend upon approximate shapes inside the ideal shape evaluation process. For example we cannot base ourselves on the points that are at the intersection of two approximated surfaces in order to be able to formulate a conclusion regarding their ability to vary despite the fact that they are intersected.

Taking into account also the introduced error by the meshing process we cannot say for certain that the way two approximated surfaces do intersect or not and also the manner in which they do it gives us any insights about the true intersection area topologically speaking. From this point of view it appears that the intersecting surfaces are being cut along edges when they are merged along their intersection curves.

References

1. M.Y. Li, W.B. Ding, *Procedia CIRP*, The 9th International Conference on Digital Enterprise Technology – Intelligent Manufacturing in the Knowledge Economy Era, **56**, 178-181 (2016)
2. T. Sovany, Z. Tisler, K. Kristo, A. Kelemen, G. Regdon Jr. , *European Journal of Pharmaceutics and Biopharmaceutics*, 7th International Granulation Workshop: Granulation across the Length Scales, **106**, 79-87 (2016)
3. G. Trummer, C. Marte, P. Dietmaier, C. Sommitsch, K. Six, *Wear*, **352-353**, 136-145 (2016)
4. A. Needleman, *Procedia IUTAM*, Mechanics for the World: Proceedings of the 23rd International Congress of Theoretical and Applied Mechanics, ICTAM2012, **10**, 221 – 246 (2014)
5. G. Xu, Q. Zhang, *Computer Aided Geometric Design*, **25**, Iss. 3, 181–202 (2008)
6. P. Zhang, Z. Liu, *Measurement*, **94**, 37–45 (2016)
7. A. Chida, H. Masudan, *Journal of Computational Design and Engineering*, **3**, Iss. 4, 322-329 (2016)
8. W. Peng, C. Xu, Z. Feng, *Neurocomputing*, **179**, 228–237 (2016)