# 3B.2

# A MESH MORPHING TECHNIQUE FOR GEOMETRICALLY DISSIMILAR TESSELLATED SURFACES

Radhika Vurputoor[1], **Nilanjan Mukherjee[2], Jean Cabello[3] & Michael Hancock[4]

*Meshing & Abstraction Group,*
*Digital Simulation Solutions*
UGS PLM Software
**SIEMENS**
*2000 Eastman Dr, Milford, OH 45150, USA*

*Email: 1 [radhika.vurputoor@siemens.com]*
*2 [mukherjee.nilanjan@siemens.com]*
*3 [jean.cabello@siemens.com]*
*4 [hancock.michael@siemens.com]*

** Author to whom all correspondence should be addressed

**Abstract.** Many industrial structural mechanics applications require topologically similar meshes on topologically similar surfaces that might differ in shape or size. Rotationally symmetric mesh models requiring cyclic symmetry boundary conditions, contact meshes that connect large structural assemblies and automotive and space panels with minor structural variations are some common challenges. The present paper discusses a generic technique that assembles several simple mapping /morphing tools to map a mesh from a source surface to a topologically similar target surface that might vary in shape or size. A loop/edge based relationship is used to define the dependency. The boundary discretization method adopted maps m nodes on a loop with n edges to a target loop with p edges. A two-dimensional boundary constrained sample mesh is constructed on the related surfaces, which is efficiently used to map the mesh on the source surface to the target using linear iso-parametric mapping techniques. Local distortions of the target mesh are checked for and corrected by a mesh relaxation procedure. Although the algorithm works for both parametric NURBS and discrete surfaces, the present discussion focuses on tessellated surfaces. Results include some key examples of practical interest.

## 1. INTRODUCTION

In the finite element analysis world, several structural mechanics problems require the user to create topologically similar meshes on topologically similar surfaces for more than one purpose. An elementary sub-structuring approach involves scooping out a radial pie-piece of a large rotationally symmetric structure for various analyses. In order to simulate the global environment, these mesh models require cyclic symmetry boundary conditions to be created on the cut-faces. The mesh on the cut-faces must be topologically similar such that a node-to-node/element-to-element correspondence exists.

Another interesting problem deals with large assembly structures where contact meshes are used to connect faces of different bodies. These faces are usually topologically congruent but might vary in shape or size, thus implying geometric incongruency. Topologically identical meshes need to be generated on these faces such that contact elements can be created between matching nodes and elements.

In the automotive and aerospace industry, similar problems could be found. It is a standard design procedure today to reuse legacy FE models of car-body panels with minor structural variations. This produces surfaces that have a legacy quad mesh on them but have been minutely modified geometrically. The modifications mostly imply repositioning or resizing of existing features. Thus, the modified surface is a topological twin of the original surface although its geometry is now different. If the same mesh needs to fit the modified geometry, it needs to be morphed to reflect the geometric change. The scope of the present paper is to lay down a technique to morph triangular and quadrilateral meshes to surfaces that are topologically similar but geometrically dissimilar.

## 2. PAST RESEARCH

Many investigations have been reported on 2D and 3D mesh morphing especially in the area of multiresolution mesh morphing [2-5]. Similarly, many investigations exist in the area of moving boundary meshes [6-7]. However, in most of these proposed methods, the required topological proximity between the source and the target is stringent (faces need to have the same number of loops and edges). Furthermore, the need to preserve the topology of the original mesh is not as critical as the present case. The quality of the finite element mesh is also vastly important, as opposed to meshes for graphic use, as these meshes need to solve. Staten et al [1], in

their BM Sweep technique, presented an attractive method to locate interior nodes during sweep mesh generation. The method is used in a completely different context – Hexahedral sweep meshing and does not aim at morphing a surface mesh to another geometrically dissimilar surface. Accordingly, it reveals certain bottlenecks when applied to solve the present problems. One limitation of the method is that it morphs nodes from the source face to volume interior layers, which are usually planar. In addition to that, it could not be used to move nodes or morph meshes across periodic surfaces. To solve similar problems in the hex mesh generation area, Shih & Sakura [8] and Lai & Benzley [9] provided solutions to place volume interior nodes. These algorithms suffer from very similar limitations.

## 3. PROBLEM STATEMENT

The present paper aims at laying down a robust, general-purpose algorithm for morphing 2D meshes across surfaces that are topologically identical but geometrically incongruent. The approach presented works for both NURBS surfaces as well as discrete, tessellated geometry. However, with discrete geometry several analytical disadvantages could be overcome. This paper highlights some of those virtues. The method proposed is general-purpose and particularly addresses three classes of structural problems, namely

1. Cyclic symmetry situations where the source and target faces are part of the same rotationally symmetric body and are congruent. The morphed mesh on the target face must be a pure transformation of the source face and requires stringent positional accuracy. Cyclic symmetry boundary conditions are then created on these meshes. A typical example is shown in Figure 1.

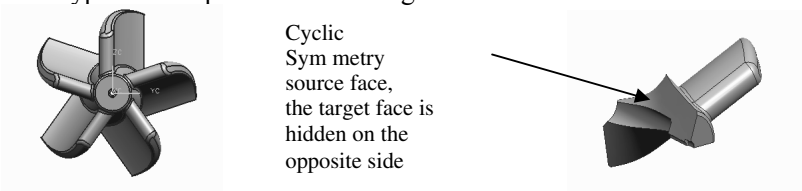


**Fig. 1. A sector of a rotor showing faces where cyclic symmetry boundary conditions need to be applie.**

2. Mesh assembly situations where contact elements are created between two bodies through certain faces. These faces are usually geometrically incongruent but topologically identical. An example is shown in Fig. 11.

3. Nearly identical sections of automotive body or space panels that contain large multi-featured faces that have some minute geometric dissimilarity. Either certain dimensions of these faces change or some features are modified to a minor degree. These classes of problems also extend to handling legacy FE data while minutely updating old automotive designs to create new ones. Car doors or excavator carriage frames are perfect examples. Figures 2. & 3 exemplify a simplified situation where a hole has been translated on the target face to a different location. $t^S$ and $t^T$ ($t^S \neq t^T$) are the margin width of the holes on the source and the target faces respectively. The loops on the source and the target face are defined in the opposite directions. The morphed mesh on the target face is shown in Fig. 3.

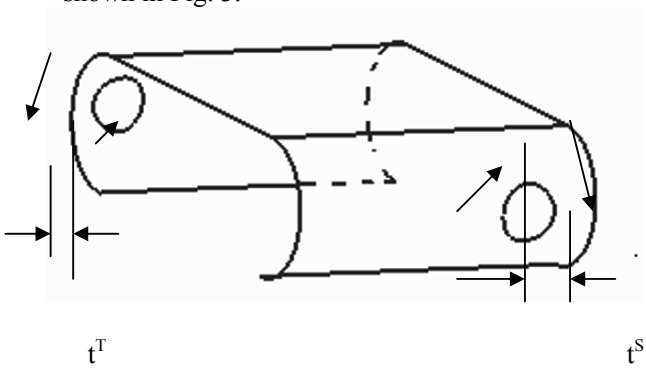


**Fig. 2. Topologically similar source and target faces**

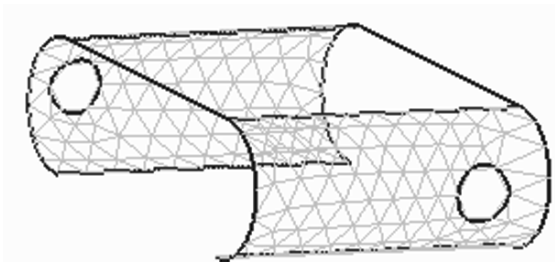

**Fig. 3. Triangular mesh on source is morphed where a feature (hole) is displaced on to the target.**

## 4. MORPHING THE FACE BOUNDARIES

Given a source face and a corresponding target face that are topologically congruent i.e. that have the same number of loops, we first define corresponding pairs of loop between the source and the target as well as, for each pair, a start vertex and direction on the source (similarly a corresponding start vertex and direction on the target). The goal is to map with the least distortion the boundary node distribution on the source to a corresponding boundary node distribution on the target.

Let us use the following notations:

$E_i$ : edge bounded by vertices $V_i$, $V_{i+1}$.

$E_i^S$ : edge on the source, resp. $E_i^T$ edge on the target.

$L_i^S$ : loop on the source, resp. $L_i^T$ loop on the target.

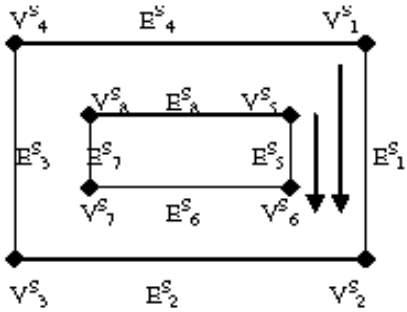


**Fig 4a. Source face, loop, edges, vertices**

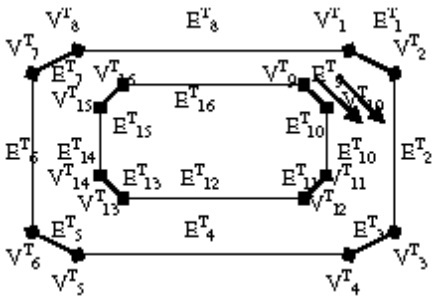


**Fig 4b. Target face, loop, edges, vertices**

In the case depicted by Fig.4a & Fig 4b., we have :

$$L^S{}_1 = (E_1, E_2, E_3, E_4) \rightarrow L^T{}_1 = (E_1, E_2, E_3, E_4, E_5, E_6, E_7, E_8)$$

$$L^S{}_2 = (E_5, E_6, E_7, E_8) \rightarrow L^T{}_2 = (E_9, E_{10}, E_{11}, E_{12}, E_{13}, E_{14}, E_{15}, E_{16})$$

where the upperscripts have been dropped for the edges. Let us consider the first loop, $L_1$ for both source and target.

Defining :

$AL_i$ : total arc length for edge $E_i$

$AL_i(P)$ : relative local arc length to a point P along edge $E_i$ .

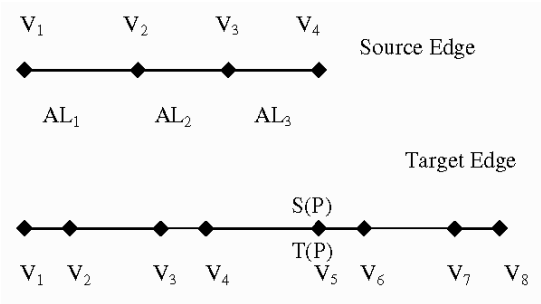


**Fig.5 : Normalized perimeter based parameterization (middle) of the source (top) and target (bottom) loop.**

$$AL_i = AL_i(V_{i+1}) \tag{1}$$

$NAL_i(P)$ : normalized relative local arc length of a point P belonging to edge $E_i$ .

$NAL_i(P) = AL_i(P) / AL_i$

$NAL_i(V_i) = 0$

$NAL_i(V_{i+1}) = 1$

$AL(P)$ : Normalized cumulative arc length along the loop at point P belonging to edge $E_i$

Assume the loop is made up of m edges.

$$AL(P) = (\sum_{j=1}^{j=i-1} AL_j + AL_i(P)) / (\sum_{j=1}^{j=m} AL_j)$$

$$\forall P \in E_i \tag{2}$$

$0 \leq AL(P) \leq 1 \quad \forall \quad P$ on the loop

$$L = \bigcup_{j=1}^{j=m} E_j \tag{3}$$

The affine function AL(P) defines a "normalized perimeter based" parameterization of a loop made up of m edges obtained by accumulation of the arc length while traveling along the loop. Such a parameterization can be defined both for the source and target loop that are in correspondence. The parameterization for the source loop is denoted $S(P)$ and for the target loop $T(P)$ and provides a common reference parameterization to distribute the nodes along their respective loops. This is an extension of the normalized arc length parameterization along an edge to a loop. Using this common parameterization for both source and target loops, the vertices of the target loop can be positioned along the source loop. For example, $V_2^T$ falls on edge $E_1^S$.

Next, the boundary node discretization of the source edge is performed. Each boundary node $N_j$ has parameter location $S(N_j)$ along the source loop given by :

$$N_j \in E_i \tag{4}$$

$$S(N_j) = AL(N_j) \tag{5}$$

Assuming that the number of edges that make up the target loop is smaller than the number of boundary nodes on the source, the algorithm proceeds by first snapping the target vertices locations $T(V_j)$ to the closest existing boundary node. This operation defines a piecewise decomposition of the source loop. A target edge $E_i^T$ (parameter location $[T(V_i) : T(V_{i+1})]$) is then mapped to a segment $[S(N_j), S(N_k)] \quad k > j$ .

$$T(V_i) \cong S(N_j) \text{ and } T(V_{i+1}) \cong S(N_k) \tag{6}$$

The k-j-1 interior nodes on the source are then created on the target edge according to their relative local arc length. For example, an interior source node $N_{jo}$ with parameter location $S(N_{j0})$ on the source satisfies the following relations:

$$j < j_0 < k \tag{7}$$

$$0 \le S(N_j) < S(N_{j0}) < S(N_k) \le 1 \tag{8}$$

$$S(N_{j0}) = AL^S(N_{j0}) \tag{9}$$

Its corresponding target node $N_{jo}$ verifies the equations:

$$T(N_{jo}) = S(N_{j0}) \quad (10)$$

$$T(N_{j0}) = AL^T(N_{j0}) \quad (11)$$

$$0 \le T(V_i) < T(N_{j0}) < T(V_{i+1}) \le 1 \quad (12)$$

The parameter location of source node $N_{j0}$ is computed using equations (9) and (2). Equation (11) provides the parameter location of the corresponding target node location. The inverse of the function defined by equation (2) is used to derive the normalized relative local arc length from the normalized perimeter based arc length i.e.

$$AL^T_{Loop} = \sum_{j=1}^{j=m} AL_j \quad (13)$$

$$NAL^T_i(N_{j0}) = [S(N_{j0}) * AL^T_{Loop} - \sum_{j=1}^{j=i-1} AL_j] / AL^T_i \quad (14)$$

In the special case when the number of edges on the source and the target is equal, i.e. $n = m$ the algorithm is modified so that the step that involves finding the closest boundary node to snap to a target vertex is replaced by mapping target vertices to their corresponding source vertices. A source node belonging to an edge $E^S_i$ is then mapped to a target node belonging to an edge $E^T_i$ using equations (13) and (14) above. Notice than in this special case, equation (10) does not hold when the edges on the source and target have different arc lengths. Piecewise identity, contraction or dilation are uniformly applied to each source edge nodal distribution to get the corresponding target edge nodal distribution.

It is important to state here that the orientation and the start nodes of the source and target loops are critical for obtaining a valid, good quality map.

## 5. MORPHING THE MESH TO THE TARGET FACE

Once the boundary of the source and target faces are discretized, the next major task is to morph the mesh to the target face. The essential steps involved in the process are –

a. Flatten both the source and target faces into a non-intersecting planar domain using a procedure described by Beatty [11].
b. Generate a boundary constrained 2D triangular mesh on the source face. This mesh is boundary conforming and serves as a background mesh over which the 2D mesh needs to be generated.
c. Create a topologically conforming background template mesh on the target face by mapping the triangle connectivity of the source face.
d. Generate a 2D tria/quad/quad-dominant mesh (as desired by the user) of a given size on the planar domain on the source face.
e. Use the background mesh to map the interior nodes of the source face to the target face similar to the method described by Staten et al [1].
f. Complete the mesh on the target face by constructing the elements by mapping their connectivity from the source face.

The details of the procedure will vary for periodic faces over non-periodic faces.

### *5.1 Mesh Morphing On Non-periodic faces*

a. The source and the target faces dealt with for this case are open tessellated faces that are developable [11]. Two such faces are shown in Fig. 3. Several domain development techniques for discrete surfaces could be used. For NURBS surfaces, the parameter space can also be used for this purpose. We have used the following techniques [10-12] the details of which are outside the scope of the present paper.

b. Using the boundary discretization discussed in section 2 (Fig. 3), a boundary constrained background mesh is now generated on the source face. The TriaQuaMesher algorithm, previously discussed in detail [13], is employed for the purpose in a boundary-constrained mode. The resultant mesh is depicted in Fig.6a.

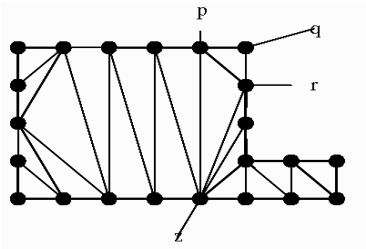


**Fig. 6a Boundary-constrained background template mesh on source surface**

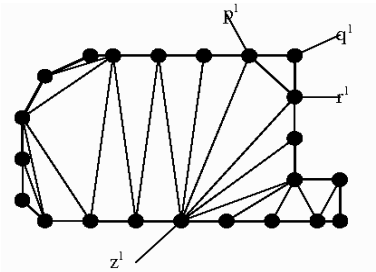


**Fig. 6b Boundary constrained background template mesh on target surface**

c. The boundary constrained background mesh on the target surface can be generated by mapping the element connectivity information from the source template as shown in Fig. 6b. For example,

$$\Delta(p,q,r) \equiv \Delta_T(p^1, q^1, r^1) \quad \text{and}$$
$$\Delta_S(p,r,z) \equiv \Delta_T(p^1,r^1,z^1) \tag{15}$$

Likewise, the source and target meshes can be correlated as

$$\bigcup_{i=1}^{N} \Delta_{Si}(n_1,n_2,n_3) \equiv \bigcup_{i=1}^{N} \Delta_{Ti}(n_1^1, n_2^1, n_3^1) \tag{16}$$

where N represents the number of triangles in the mesh

$\Delta_{Si}$ represents the i-th triangle of the source template mesh whose connectivity is defined by 3 nodes $(n_1,n_2,n_3)$.

$\Delta_{Ti}$ represents the i-th triangle of the target template mesh whose connectivity is defined by 3 nodes $(n_1^1, n_2^1, n_3^1)$ which correspond to $(n_1,n_2,n_3)$.

If any triangle $_{Ti}$ gets inverted, the domains and thus the source and target faces are deemed largely dissimilar. This explains the limit of geometric dissimilarity that can be handled by the algorithm.

d. The mesh on the source face is generated by the TriQuaMesher (embedded with transfinite meshing) algorithm first proposed decades back by Sluiter et al [13] and later modified by Cabello [14]. TQM is a recursive domain subdivision algorithm that starts with discretized loops defining the mesh area. The split line locations are governed by the included angles of the loop segments. The splitting algorithm aims at recursively decimating the area into convex regions. Hermite polynomials (and alternatively a discrete background sample triangular mesh) are used to represent the split lines. A boundary-blended transfinite meshing algorithm, discussed by Mukherjee [15], is optionally employed in the sub-areas when they become convex and 4-sided.

e. In this stage, the mesh on the source face is done and is morphed to the target. To accomplish this, each face-interior node of the source face and the triangle containing it, is used to map its location on the corresponding triangle of the target. The following equation family (17-18) explain the mapping procedure [refer Fig. 7(a-b)].

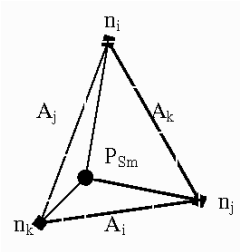


**Fig 7a. Source triangle $\Delta_{Sm}$ containing node $P_{Sm}$**

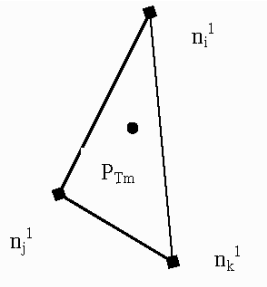


**Fig 7b. Corresponding target triangle $\Delta_{Tm}$ containing node $P_{Tm}$**

$$P_{Tm}(u,v) = N_i.n_i^1(u,v) + N_j.n_j^1(u,v) + N_k.n_k^1(u,v) \quad (17)$$

where

$P_{Sm}(u,v)$ represent the location of the m-th interior node $P_{Sm}$ on the source

$P_{Tm}(u,v)$ represent the location of the m-th interior node $P_{Tm}$ on the target

$N_i$, $N_j$, $N_k$ represent the shape functions of source-face background triangle $\Delta_{Sm}$ that contain the source node $P_{Sm}$

$n_i^1$, $n_j^1$, $n_k^1$ are three nodes of the target background triangle $\Delta_{Tm}$ (corresponds to $\Delta_{Sm}$) containing the target node $P_{Tm}$ .

The shape functions $N_i$, $N_j$, $N_k$ can be expressed as

$$N_i = A_i/A_{Sm}; \quad N_j = A_j/A_{Sm}; \quad N_k=A_k/A_{Sm} \quad (18)$$

where $A_i$, $A_j$, $A_k$ are the sub-areas as shown in Fig 5a.

$A_{Sm}$ represents the area of the m-th source triangle $\Delta_{Sm}$

After all the interior nodes are morphed, the mesh elements are constructed on the target face using the connectivity information of the source face. Fig. 6(a, b) depict a structured quad mesh on the source morphed to a topologically compatible mesh on a geometrically dissimilar target face. The i-th quad element of the target face can be constructed using the relation given below

$$\square_{Ti}(p^1, q^1, r^1, s^1) \equiv \square_{Si}(p, q, r, s) \quad (19)$$

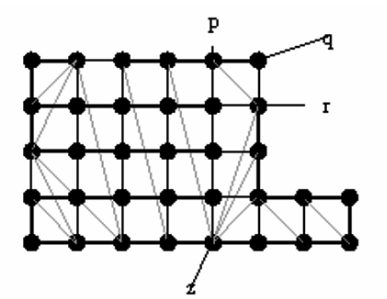


**Fig. 8a Final mesh created on the source**

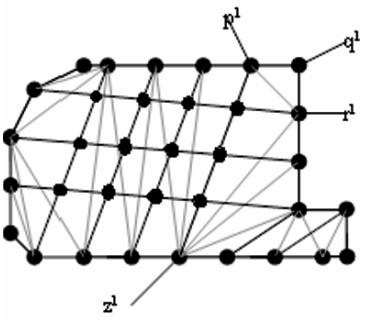


**Fig. 8b Final mesh morphed on the target face face over the background mesh**

Depending on the geometric dissimilarity of the target face from the source, the mesh could suffer distortion (Fig. 8b). A mesh relaxation or smoothing procedure that does not alter the mesh topology could be used in such situations to reduce mesh distortion. To measure the mesh distortion caused on the target face, an element quality metric, $\lambda$, is used, where

$\lambda_i$ (for the i-th triangle) $= \alpha_{iT} / \alpha_{iS}$;

where $\alpha = 2\sqrt{3}\ (\overline{BC} \times \overline{AC}) / (\|AB\|^2 + \|BC\|^2 + \|CA\|^2)$ as proposed by Lo [16]

for a triangle ABC (20)

$\lambda_i$ (for the i-th quadrilateral) =

$( |J_{min}|/|J_{max}| )_{iT} / ( |J_{min}|/|J_{max}| )_{iS}$ (21)

$( |J_{min}|/|J_{max}| )_{iT}$ denotes the Jacobian determinant ratio for the ith element of the target face

$( |J_{min}|/|J_{max}| )_{iS}$ denotes the Jacobian determinant ratio for the ith element of the source face

If $\lambda_i$ falls below a certain critical value $\lambda_{cr}$ (which may differ for element topology), it indicates that at least one element in the target mesh is severely distorted and the smoother kicks in. A variational smoothing algorithm discussed by Mukherjee [17] that selectively combines numerous iterative solutions based on nodal valency is used for this purpose.

### *5.2 Mesh Morphing On Periodic Faces*

For general structural mechanics applications like contact etc. mesh morphing of periodic faces have to be often dealt with. The BM Sweep [1]

and other past algorithms [2,4,5] did not explain handling of periodic faces. Periodic faces (e.g cylindrical, spherical faces) are more challenging to deal with, especially with NURBS geometry. Firstly, if the parameter spaces of these surfaces are overtly distorted, it is difficult to generate a good quality mesh; let alone the morphed mesh on the target face. Secondly, even if the parameter space is good, there is no guarantee that the seam/pole(s) of the target face will align with those of the source face, thus distorting the mesh on the target.

With a discrete representation, there is an advantage. Arbitrary seams (cuts) could be introduced at desired locations on the tessellated faces (both source and target) as shown below in Figure 9a-9b. The generated 2D domain is thus geometrically more uniform and congruent resulting in good quality meshes on both the source and the target face.

The algorithmic details for a cylinder are given below –

- Cylindrical faces are cut so that the facets can be unzipped to get a flattened domain. Essentially a two-loop cylinder becomes a one-loop cylinder with the cut edge being repeated twice in the loop (edge e3 as shown in Fig. 9a).
- Each cut edge in the loop is represented as a "dummy" edge (edge e4 as shown in Fig. 9a) and an edge dependency relationship is established between these edges.
- For the purpose of mapping the source nodes on the target, duplicate nodes (as shown in Fig. 10a) are introduced in the master mesh along the cut edge. This enables a one-to-one mapping with the source and the target edges.
- At this point, a periodic surface is treated just like a non-periodic face and the mapping of boundary and interior nodes is done as described in 5.1(e).
- At the point of creation of elements on the target face, duplicate nodes are eliminated based on the edge-dependency relation ship between the cut edges of the target loop.
- The details of how the best location for the seam is chosen is discussed by Beatty & Mukherjee [10]. The algorithm for generating a flattened 2D domain from a facetted 3D surface is discussed in detail by Beatty [11] and Hormann [12]. We avoid a repetition of these techniques as they are outside the scope of this paper.

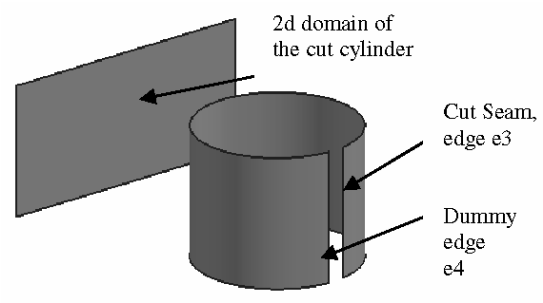


**Fig. 9a. A cylindrical surface cut along the seam**

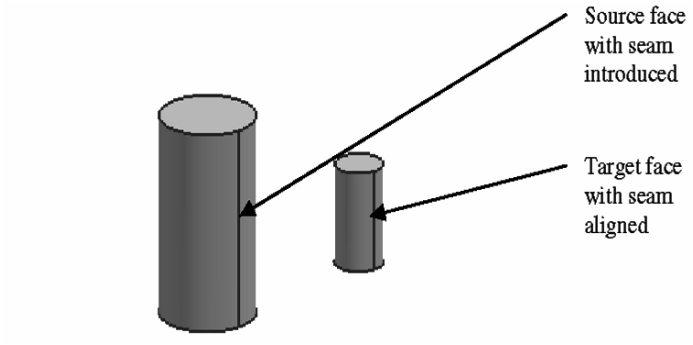


**Fig. 9b. Topologically aligned seams introduced with its 2D domain in the background source and target cylindrical surfaces.**

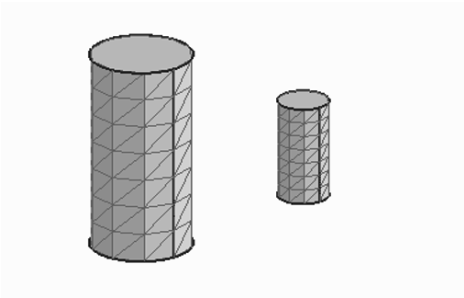

**Fig. 10b. Triangular mapped mesh generated on the source is morphed to the target.**

## 8. EXAMPLES AND DISCUSSION

Figs. 11 (a & b) presents a simple case of topological congruency with geometric dissimilarity. A triangular mesh is created using the TQM

algorithm on a four-edged quadrilateral flat face (Fig. 11a). The elliptical face in Fig. 11b, represents the target face that is still flat and 4-edged, but is elliptical in shape. The triangular mesh on the elliptical face is morphed from the mesh on the quadrilateral face. All nodes and elements of the source face have a one-to-one correspondence with the nodes/elements on the target face.

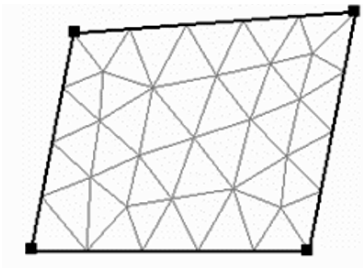

**Fig. 11a. Triangular mesh on a four-edged source face**

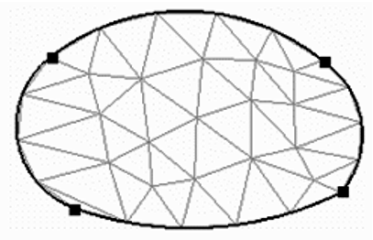

**Fig. 11b. Morphed contact mesh on a four-edged target face**

A radial sector of an Aluminium impeller as shown in Fig.1, is sub-structured with cyclic symmetry boundary conditions. Two radially cut faces, as shown in Fig. 12, represent the cyclic symmetry faces. A boundary condition needs to be applied to these faces that equate the degree of freedom of each node on these faces in the Hoop direction ( θ-direction). In order to achieve this, it is necessary to have the same number of nodes/elements on these faces. The proposed mesh morphing technique is used to morph the mesh on the source face (any one of the symmetry faces) to the target. The meshes are shown in Fig. 12. Cyclic symmetry boundary conditions are applied on the nodes of the source and target faces. Next, this radial sector of the impeller, is analyzed for centrifugal load (rotating at 3000 RPM). Table 1.0 lists the maximum displacement, maximum Von Mises and maximum Octahedral shear stress for the radial substructure in comparison with those obtained from a full model solve. The cyclic symmetry model has about a fifth of the DOFs of the full model. The results indicate the accuracy of the sub-structured finite element model.

This kind of sub-strictured analysis would not be possible without a surface mesh morphing functionality. The displacement plots are shown in Fig 13(a-b).

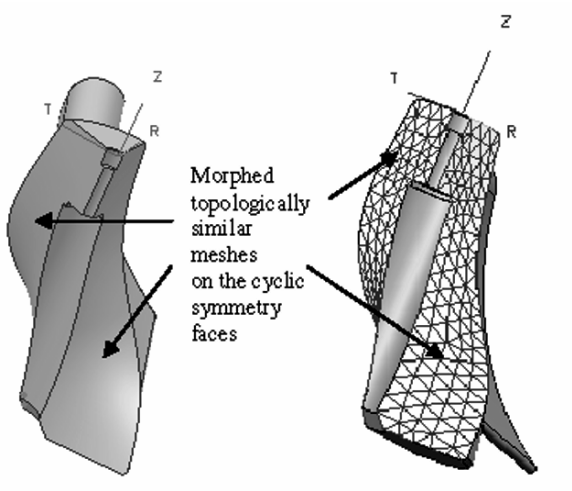


**Fig. 12. Morphed mesh on cyclic symmetry faces. The entire rotationally symmetric model is shown in Fig. 1.**

**TABLE 1.0 Stress-Deflection Result Comparison**

| **Type of Model** | **Max. Disp.** (in) | **Max. Von Mises Stress** (psi) | **Max. Tresca (Octahedral Shear) Stress** (psi) | **DOFs** |
|---|---|---|---|---|
| Full-Model | 1.014e-5 | 13.13 | 6.19 | 75,238 |
| Cyclic Symmetry Model | 1.141e-5 | 15.28 | 6.25 | 14,970 |

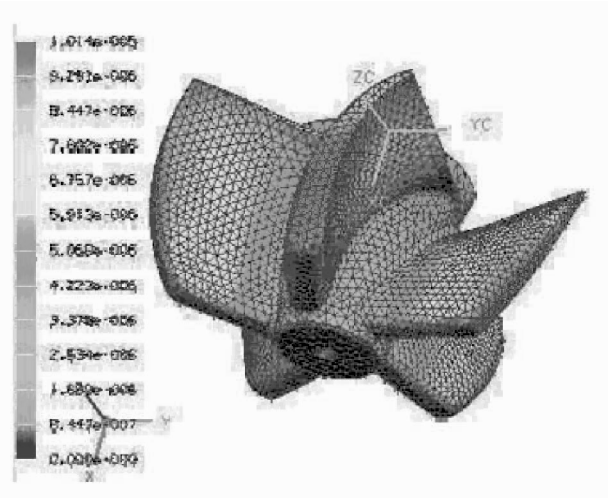


**Fig. 13a. Displacement plot of the full model of the impeller under centrifugal load.**

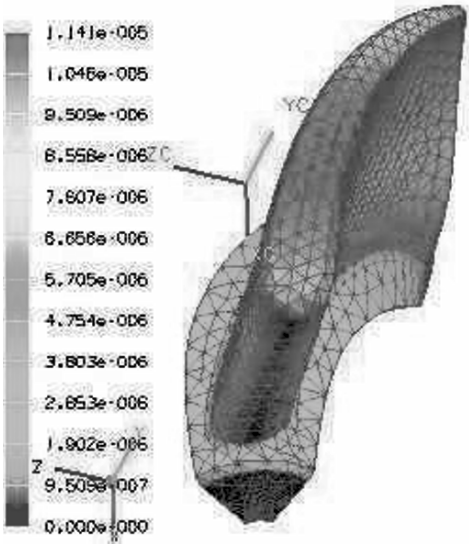


**Fig. 13a. Displacement plot of the sub-structured model of the impeller (with cyclic symmetry boundary conditions on the mesh-morphed source and target faces) under centrifugal load.**

Fig. 14 shows an interesting contact-mesh problem involving two hollow cylinders. The outer and the inner cylinders share a common contact face. This is represented by a pair of faces - the outer cylindrical face of the inner

cylinder and the inner cylindrical face of the outer cylinder. A geometry abstraction operator, that is intrinsic to this algorithm, cuts the cylindrical faces by introducing real seams that align with each other. Any one face of the pair could be used as the source and is first meshed with a transfinite meshing algorithm. This mesh is morphed on the target face using the proposed morphing strategy. The mesh nodes that correspond are used to define a contact mesh.

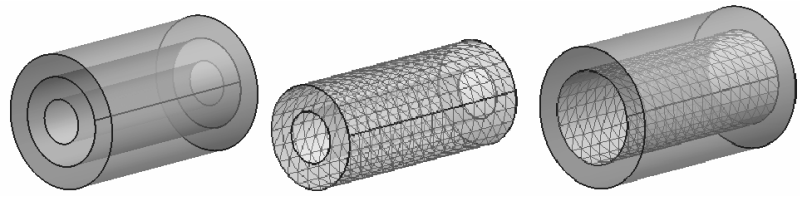

**Fig 14. A contact mesh problem showing two concentric hollow cylinders. The cylindrical (seamed) surfaces on the mating bodies define the contact surfaces.**

## 9. CONCLUSION

The present paper focuses on identifying a robust, general-purpose algorithm for morphing 2D meshes across surfaces that are topologically identical but geometrically incongruent. The approach presented is discussed for discrete, tessellated geometry although it will work well for spline geometry too. The paper presents three different categories of industrial problems that could be solved by the proposed algorithm. These are- a) rotational substructures with cyclic symmetry constraints; b) surface contact situations across finite element assemblies and c) legacy finite element models of automotive body panels with minor modification of face-interior features. The proposed method can also be elegantly applied to periodic surfaces. Several examples show the range, robustness and versatile strength of the algorithm. The algorithm is not limited by the geometric difference between topologically identical surfaces. However, the quality of the morphed mesh is limited by the degree of geometric deformation between the source and the target surfaces.

## REFERENCES

[1] M.L. Staten, S. A. Canann and S.J. Owen, “BMSweep: Locating Interior Nodes During Sweeping” *Proceedings, 7th International Meshing Roundtable*, pp.7-18 (1998).

[2] A.W.F. Lee, D.Dobkin, W. Sweldens and P. Schroder, “Multiresolution Mesh Morphing” *Proceedings of SIGGRAPH 99,* pp343-350.(1999).

[3] E. Praun, W. Sweldens and P. Schroder, “Consistent Mesh Parameterizations” *Proceedings of SIGGRAPH 2001.* pp.179-184 (2001).

[4] K. Hormann and G. Greiner “MIPS: An Efficient Global Parameterization Method” *Curve and Surface Design-Saint Malo 1999.* pp.153-162 Vanderbilt University Press (2000).

[5] P. Alliez, P. C. de Verdiere, O. Devillers and M. Isenburg, “Isotropic Surface Remeshing” *Proceedings of Shape Modelling International* (2003).

[6] T. J. Baker, “Mesh movement and metamorphosis”. *Proceedings, 10th International Meshing Roundtable*, Sandia National Laboratories pp. 387- 396, (2001).

[7] T.J. Baker, "Interpolation from a cloud of points", *Proceedings, 12th International Meshing Roundtable*, Sandia National Laboratories, pp.55-63, (2003).

[7] B. Shih and H. Sakurai, “Shape Recognition and Shape-Specific Meshing for Generating All Hexahedral Meshes” *Proceedings, 6th International Meshing Roundtable*, Sandia National Laboratories, pp.197-209, (1997).

[9] Mingwu Lai and S.E. Benzley, “Automated Hexahedral Mesh Generation by Virtual Decomposition” *Proceedings, 5th International Meshing Roundtable*,, pp 217-225, (1996).

[10] Kirk Beatty and Nilanjan Mukherjee, “Mesh Generation On Cylindrical Geometric Abstractions” *Internal Report*, *Digital Simulation Solutions, UGS* (2004).

[11] Kirk Beatty, “Strategies for Generating 2D Domains for Facetted Surfaces” *Internal Report, Digital Simulation Solutions, UGS* (2005).

[12] Kai Hormann, “Theory and Applications of Parameterizing Triangulations” *PhD Thesis,* Department of Computer Science, University of Erlangen (2001).

[13] A.J.G. Schoof, L.H.Th. M. van Buekering and M.L.C. Sluiter, “A General Purpose Two-Dimensional Mesh” *Advances in Engineering Software,* Vol.1(13) pp 131-136, (1979).

[14] Jean Cabello, “Towards Quality Surface Meshing”*, Proceedings, 12th International Meshing Roundtable*,, pp 201-213, (2003).

[15] Nilanjan Mukherjee, “High Quality Bi-Linear Transfinite Meshing with Interior Point Constraints”*, Proceedings, 15th International Meshing Roundtable*, Ed. P.P.Pebay, Springer, pp 309-323, (2006).

[16] S.H. Lo, "A New Mesh Generation Scheme For Arbitrary Planar Domains", *International Journal For Numerical Methods in Engineering*, No. 21, pp.1403-1426, (1985).

[17] Nilanjan Mukherjee, “A Hybrid, Variational 3D Smoother For Orphaned Shell Meshes”*, Proceedings, 11th International Meshing Roundtable*, pp 379-390, (2002).