

Development of Unstructured Curved Meshes with G^1 Surface Continuity for High-Order Finite Element Simulations

Qiukai Lu and Mark S. Shephard

Abstract This paper presents a curved meshing technique for unstructured tetrahedral meshes where G^1 surface continuity is maintained for the triangular element faces representing the curved domain surfaces. A bottom-up curving approach is used to support geometric models with multiple surface patches where either C^0 or G^1 geometry continuity between patches is desired. Specific parametrization approaches based on Bézier forms and blending functions are used to define the mapping for curved element faces and volumes between parametric and physical coordinate systems. A preliminary result demonstrates that using G^1 -continuity meshes can improve the solution results obtained.

1 Introduction

It is well known that high-order finite element methods are among the most powerful methods for simulating complex engineering problems [2]. In order to fully realize the benefits of the high-order methods, the mesh entities representing curved portions of the domain geometry must be curved and provide an high-enough order of geometry approximation [11, 12]. The ability to provide such a higher order of geometric approximation is facilitated by the use of greater than C^0 geometric shape continuity between elements [8, 14–16]. Although such higher order geometric continuity is being increasingly used with tensor product representations over quadrilaterals (see [6, 10]), there is also the desire to have higher than C^0 geometry continuity between elements on unstructured meshes where curved triangular finite element faces are used. The current work is intended to investigate and address the technical difficulties with developing curved meshing techniques for unstructured meshes where G^1 surface geometry continuity is maintained for the triangular element faces representing the curved domain surfaces. A preliminary result is also

Q. Lu (✉) • M.S. Shephard
Scientific Computation Research Center, Rensselaer Polytechnic Institute, 110 8th Street, Troy,
NY, USA
e-mail: luq3@rpi.edu; shephard@rpi.edu

included that shows improved solution results when G^1 surface triangulations are used.

2 Procedure to Create G^1 Curved Meshes

Many triangular patches have been developed in the CAGD community to construct G^1 continuous surface interpolations [8, 9, 14–16]. The techniques can in general be categorized into one of the two sets—polynomial based patches or rational blend based patches. The schemes using polynomials address the problem by either using single patch with relatively high polynomial degree or creating piecewise parametrization using sub-patches [13], both of which lead to more control points to be determined for the patch. The schemes using rationals are able to keep a patch complete by using blending functions [3]. Rational patches achieve G^1 with relatively low degree and require fewer control points. For the study in this paper, the rational blend based scheme is chosen because of its relatively straight-forward to construct and the data structure is similar to a regular Bézier triangle. The procedure to create G^1 curved meshes from C^0 straight-sided meshes using rational triangular patches is introduced in the following subsections. It is assumed that a straight-sided mesh is given with the set of boundary mesh entities correctly classified on a CAD model. Each mesh vertex on the model boundary is able to obtain its position and surface normal data by interacting with the CAD model.

2.1 Rational Triangular G^1 Patch

The essential part of the procedure to create G^1 curved meshes is the scheme to construct triangular G^1 patches for mesh faces that interpolate the position $x_i(\xi)$ and normal data n_j at their bounding vertices. The scheme used in this work to represent the curved geometry is based on an extension of the Gregory patch proposed by Walton et al. [16]. For each individual mesh face, each of the three bounding edges is assigned with a geometric representation of a cubic Bézier curve $B^{(3)}(\xi)$. Tangent vectors are obtained along the curve direction by taking the derivatives of the Bézier curve parametrization $t^{(2)} = \frac{\partial B}{\partial \xi}$. Cross-boundary tangent vectors are calculated by taking cross product with the surface normal given at mesh vertices $g^{(2)} = n_j \times t^{(2)}$. In order to obtain the required G^1 continuity, the cross-boundary tangent fields associated with the three mesh edges have to be satisfied simultaneously, thus requiring more degrees of freedom than a typical triangular Bézier patch. As a result, the order of the polynomials representing the surface patch is increased from cubic to at least quartic $B^{(4)}(\xi)$, which leads to a set of three surface control points. Each of the three surface control point is subsequently split into two and related together using linear blending functions. The rational blend degree-4 triangular Bézier patch

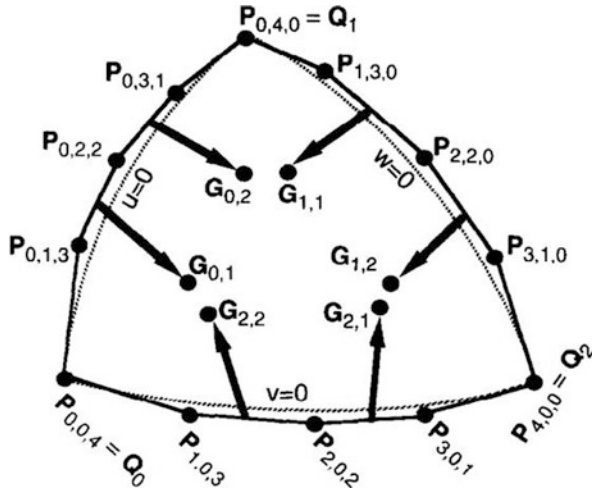


Fig. 1 Triangular Gregory patch and its control points

is defined by

$$B^{(4)}(\xi) = P_{ijk}b_{ijk}^4(\xi) \tag{1}$$

where P_{ijk} are the control points and $b_{ijk}^4(\xi)$ are 4th order Bernstein basis functions. The surface control points $P_{112}, P_{121}, P_{211}$ are affine combinations of the split surface control points $G_{i,j}$ and are calculated using $P_{1,1,2} = \frac{1}{\xi_1 + \xi_2}(\xi_1 G_{2,2} + \xi_2 G_{0,1})$, $P_{1,2,1} = \frac{1}{\xi_3 + \xi_1}(\xi_3 G_{0,2} + \xi_1 G_{1,1})$, $P_{2,1,1} = \frac{1}{\xi_2 + \xi_3}(\xi_2 G_{1,2} + \xi_3 G_{2,1})$.

Figure 1 shows an example patch and its control point set.

2.2 Surface Mesh with Mixed C^0 and G^1 Continuity

The procedure introduced in Sect. 2.1 serves the purpose of creating G^1 surface meshes for models with a single model face. In the mean time, most 3D models with challenging geometric features consist of more than one model face. Any procedure aiming to create proper surface meshes for such multi-patch models has to account for the mixture of C^0 and G^1 continuity. In this work, a bottom-up approach is adopted based on the different topological types of model entities on which a mesh entity is classified. Specifically, the mesh edges that represent the model edges where model faces join with C^0 -continuity are curved first to be G^1 along the model edge direction while maintaining C^0 in the cross-edge direction. After that, the remaining surface mesh entities that represent the rest of the model boundary are curved using the procedure discussed in Sect. 2.1. As a result, a piecewise G^1 surface mesh is created where it is G^1 within each model face as well as along the

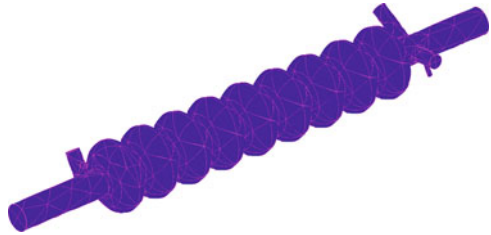
```

1 for each edge  $M_i^1$  in the mesh do
2   if  $M_i^1$  represents model edge with  $C^0$  continuity then
3     | determine edge control points to interpolate model edge tangent;
4   end
5   if  $M_i^1$  represents model face or edge with  $G^1$  continuity then
6     | determine edge control points to interpolate model face normal;
7   end
8 end
9 for each face  $M_i^2$  in the mesh do
10  if  $M_i^2 \sqsubset G_j^2$  then
11    | compute edge tangent vector  $t$ ;
12    | compute cross-edge tangent vector  $g$ ;
13    | determine face control points  $G_{i,j}$ ;
14  end
15 end

```

Algorithm 1: Algorithm for creating G^1 meshes for multi-patch CAD models

Fig. 2 Curved G^1 mesh of a linear accelerator model



bounding model edges and C^0 in the cross-boundary direction at the model edges where model faces join together. Note that in the case where two model faces join with G^1 continuity in the first place, G^1 continuity is maintained by curving the mesh edges representing the model edge in the same way as those representing model faces. The pseudo code for the overall procedure is given in Algorithm 1. Figure 2 shows an example mesh created using the algorithm.

3 Integration with Finite Element Analysis Solver

With the conventional isoparametric approach with C^0 meshes, the volumetric mapping between a standard parametric space and the physical space is constructed based on the same polynomial basis functions used for the finite element space. However, the basis functions used to represent the rational G^1 curved mesh are generally not the same as the finite element shape functions used for analysis. Therefore, a more general approach is adopted to construct the volumetric mapping in order to account for the G^1 surface geometry. The approach taken in this work is based on blending [7]. More specifically, the shapes of lower dimensional mesh

entities bounding the element volume are multiplied with linear blending functions, and the contributions are summed together to get the complete volume mapping. The equation to calculate the mapping is given in Eq. (2).

$$\begin{aligned} x_i(\xi_j) = & (1 - \xi_1)E_1(\xi') + (1 - \xi_2)E_2(\xi') + (1 - \xi_3)E_3(\xi') + (1 - \xi_4)E_4(\xi') \\ & - (1 - \xi_1 - \xi_2)F_1(\xi') - (1 - \xi_1 - \xi_3)F_2(\xi') - (1 - \xi_1 - \xi_4)F_3(\xi') \\ & - (1 - \xi_2 - \xi_3)F_4(\xi') - (1 - \xi_2 - \xi_4)F_5(\xi') - (1 - \xi_3 - \xi_4)F_6(\xi') \\ & + \xi_1 V_1(1, 0, 0, 0) + \xi_2 V_2(0, 1, 0, 0) + \xi_3 V_3(0, 0, 1, 0) + \xi_4 V_4(0, 0, 0, 1) \end{aligned} \quad (2)$$

Here, $E_j, j = 1, 2, 3, 4$ represent the four edge parametrization. Similarly, $F_j, j = 1, 2, 3, 4, 5, 6$ represent face parametrization. V_j are the vertices. It is worth noting that the blending approach is independent of the chosen face and edge parametrization, therefore can be used with other types of parametric representations of mesh faces.

With the blending based volume parametrization, coordinate mapping can be easily evaluated. Derivatives quantities $\frac{\partial x_i}{\partial \xi_j}$ can also be evaluated by applying chain rule to Eq. (2) to obtain the analytic express of the derivatives of the blending mapping. With calculated derivatives, Jacobian of the mapping and its determinant can be easily evaluated.

4 Geometric Interpolation Accuracy

To study and quantify the geometric interpolation properties of the quartic G^1 patch discussed in Sect. 2.1, a set of numerical experiments have been conducted. A series of uniformly refined meshes are generated on a CAD model representing a cylinder. The distance between the mesh faces and CAD model faces is measured for each of the uniformly refined meshes. The distance is measured in terms of the Hausdorff norm which is commonly used to measure the distance between two parametric faces [1]. The definition of Hausdorff distance is given by Eq. (3).

$$d(S, S') = \max_{p \in S} \min_{p' \in S'} \|p - p'\|_2 \quad (3)$$

As a comparison, the measurement is done for both the G^1 meshes and a set of C^0 meshes using quartic Lagrange basis functions with optimal point distribution scheme proposed by Chen and Babuska [5]. Figure 3 shows the convergence plot generated from the distance data. For the quartic G^1 meshes, 4th order interpolation accuracy is observed, and for quartic C^0 meshes, it shows 5th order interpolation accuracy. It is a well known result in 1D that the order of accuracy for polynomial interpolation is $p + 1$, where p is the highest complete polynomial order [8]. The one order difference in interpolation accuracy between the G^1 and C^0 is due to the

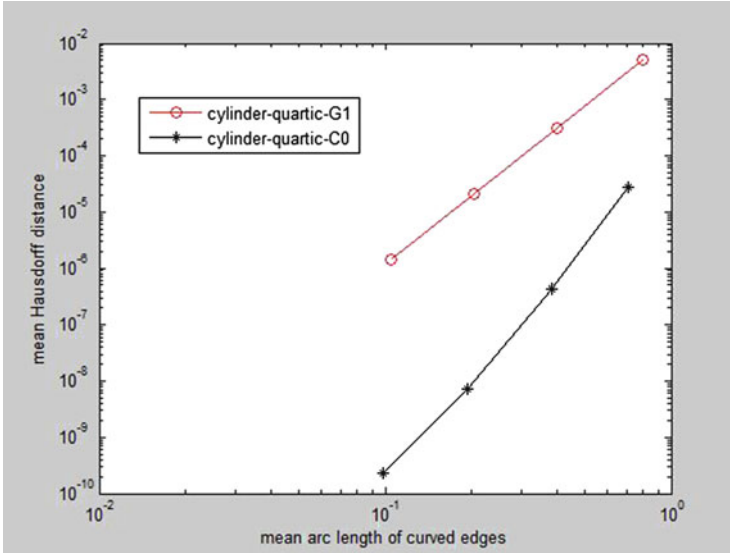


Fig. 3 Convergence of geometric approximation error

fact that certain portion of the control points of the G^1 patch have to be constrained to ensure the higher surface continuity.

5 Impact on Finite Element Solution Accuracy

The primary interest for using G^1 meshes is to see if they produce better finite element simulation results. The test problem chosen is the Poiseuille flow, which models viscous flow inside a pipe of constant circular cross-section. Governing Equation for the Poiseuille flow is defined as:

$$\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial u_z}{\partial r} \right) = \frac{1}{\mu} \frac{\partial p}{\partial z} \quad (4)$$

The fully developed flow is assumed to be incompressible, steady, laminar and has a closed form exact solution which indicates a velocity profile of a parabola. The analytic expression is given as:

$$u_z = -\frac{1}{4\mu} \frac{\partial p}{\partial z} (R^2 - r^2) \quad (5)$$

In the numerical test, it is of interest to solve for the fully developed velocity profile and compare it with the exact solution. A CAD model is constructed to represent the flow domain of a cylinder with radius $r = 0.5$. No-slip condition is set for the wall. At inlet, the velocity profile is set to be fully developed: $u_z = 0.25 - r^2$. The pressure at the outlet is set to be constant.

The finite element solver package being used to perform the analysis is *Nektar++* [4], which is a spectral/hp element framework being developed by research groups at the University of Utah and Imperial College London. It has a set of flow solvers that use high-order finite element methods. Specific modifications are made to *Nektar++* to account for the G^1 mesh construction including elemental mapping evaluation, derivatives and Jacobian calculation procedures. Two types of meshes with the same number of elements, the same order of polynomial degree, but different order of geometric continuity are used, namely, quartic C^0 curved and quartic G^1 curved meshes (See Fig. 4). A series of simulations are performed with each type of the meshes using 4th and 5th order Legendre polynomial shape functions. The error of finite element solution of the velocity field against the exact analytic solution is measured in terms of the L_2 norm and is shown in Table 1. It is observed in this test case that meshes with G^1 surface continuity achieve better solution accuracy compared with C^0 meshes for the same order of shape functions.

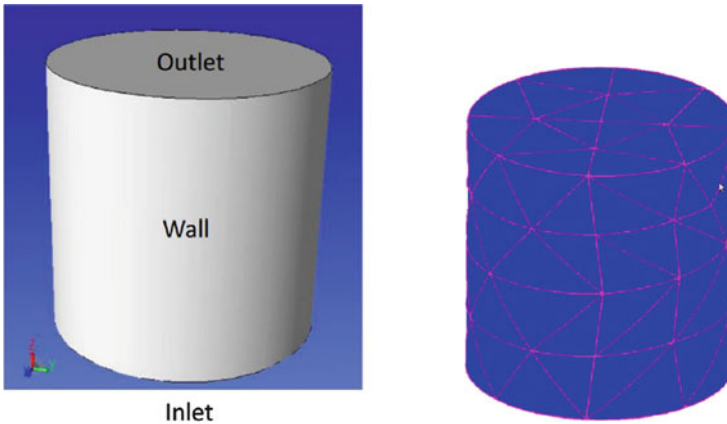


Fig. 4 CAD model and quartic G^1 mesh

Table 1 Finite element solution error for different types of curved meshes

Shape func order	Quartic C^0	Quartic G^1
4	1.29207e-3	4.33327e-4
5	5.98625e-4	9.67477e-5

6 Closing Remarks

This paper presented a procedure to create G^1 curved surface meshes for high-order finite element simulations. A method to create G^1 -continuous surface patches is introduced and an approach to integrate a G^1 mesh with existing finite element solver is presented. A preliminary test result shows the advantage of using G^1 continuous meshes, compared with conventional C^0 meshes, in terms of finite element solution accuracy of a standard integral norm. Additional studies to examine the influence of G^1 continuity on more problems and for other solution norms must be carried out. There is particular interest to examine solution parameters more local to the surface. For future developments, it is of interest to study other types of high-order surface patches. Furthermore, the capability of using the CAD model surface parametrization to define exact geometric mapping is to be developed. In order to support adaptive simulations, extensions of existing mesh modification operations and mesh adaptation procedure [11] will be needed to account for high-order curved meshes.

References

1. N. Aspert, D. Santa-Cruz, T. Ebrahimi, Mesh: measuring errors between surfaces using the Hausdorff distance, in *IEEE International Conference on Multimedia and Expo*, vol. 1, pp. 705–708 (2002)
2. I. Babuska, B.A. Szabo, I.N. Katz, The p-version of the finite element method. *SIAM J. Numer. Anal.* **18**(3), 515–545 (1981)
3. M. Boschioli, C. Funfzig, L. Romani, G. Albrecht, G^1 rational blend interpolatory schemes: a comparative study. *Graph. Model.* **74**(1), 29–49 (2012)
4. C.D. Cantwell, S. Yakovlev, R.M. Kirby, N.S. Peters, S.J. Sherwin, High-order spectral/hp element discretisation for reaction-diffusion problems on surfaces: application to cardiac electrophysiology. *J. Comput. Phys.* **257**, Part A(0), 813–829 (2014)
5. Q. Chen, I. Babuška, Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle. *Comput. Methods Appl. Mech. Eng.* **128**(3), 405–417 (1995)
6. L. Demkowicz, P. Gatto, W. Qiu, A. Joplin, G^1 -interpolation and geometry reconstruction for higher order finite elements. *Comput. Methods Appl. Mech. Eng.* **198**(13), 1198–1212 (2009)
7. S. Dey, M.S. Shephard, J.E. Flaherty, Geometry representation issues associated with p-version finite element computations. *Comput. Methods Appl. Mech. Eng.* **150**(1–4), 39–55 (1997). Symposium on Advances in Computational Mechanics
8. G.E. Farin, Triangular bernstein-bezier patches. *Comput. Aided Geom. Des.* **3**, 83–127 (1986)
9. S. Hahmann, G.P. Bonneau, Triangular g^1 interpolation by 4-splitting domain triangles. *Comput. Aided Geom. Des.* **17**(8), 731–757 (2000)
10. T.J.R. Hughes, J. Cottrell, Y. Bazilevs, Isogeometric analysis: cad, finite elements, nurbs, exact geometry and mesh refinement. *Comput. Methods Appl. Mech. Eng.* **194**(39–41), 4135–4195 (2005)
11. Q. Lu, M. Shephard, S. Tendulkar, M. Beall, Parallel mesh adaptation for high-order finite element methods with curved element geometry. *Eng. Comput.* **30**(2), 271–286 (2014)
12. X. Luo, M.S. Shephard, J.F. Remacle, R.M. O'bara, M.W. Beall, B. Szabo, R. Actis, p-version mesh generation issues, in *Proceedings of the 11th Meshing Roundtable, Ithaca, NY*, pp. 343–354, Ithaca, NY (2002)

13. S. Mann, C. Loop, M. Lounsbery, D. Meyers, J. Painter, T. DeRose, K. Sloan, A survey of parametric scattered data fitting using triangular interpolants, *Curve and Surface Design*, vol. 29, pp. 145–172 (1992)
14. J. Peters, Biquartic C^1 -surface splines over irregular meshes. *Comput. Aided Des.* **27**(12), 895–903 (1995)
15. B.R. Piper, Visually smooth interpolation with triangular bezier patches, in *Geometric Modelling: Algorithms and New Trends*, ed. by G. Farin (SIAM, Philadelphia, 1987), pp. 221–234
16. D. Walton, D. Meek, A triangular G^1 patch from boundary curves. *Comput. Aided Des.* **28**(2), 113–123 (1996)