G1-smooth Biquintic Approximation of Catmull-Clark Subdivision Surfaces

Michelangelo Marsala, Angelos Mantzaflaris, Bernard Mourrain

G^1-smooth Biquintic Approximation of Catmull-Clark Subdivision Surfaces

Michelangelo Marsala*, Angelos Mantzaflaris, Bernard Mourrain

Inria Sophia Antipolis-Méditerranée, Université Côte d'Azur, France

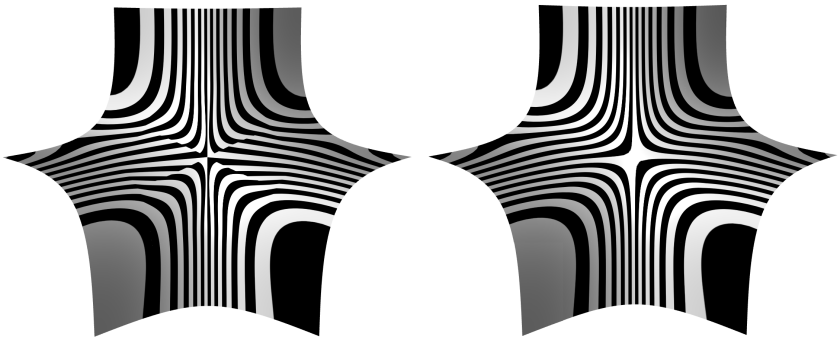


Figure 1: Isophotes on approximate bicubic (ACC_3) patches and on our biquintic G^1 construction around a valence 6 vertex.

Abstract

In this paper a construction of a globally G^1 family of Bézier surfaces, defined by smoothing masks approximating the well-known Catmull-Clark (CC) subdivision surface is presented. The resulting surface is a collection of Bézier patches, which are bicubic C^2 around regular vertices and biquintic G^1 around extraordinary vertices (and C^1 on their one-rings vertices). Each Bézier point is computed using a locally defined mask around the neighboring mesh vertices. To define G^1 conditions, we assign quadratic *gluing data* around extraordinary vertices that depend solely on their *valence* and we use degree five patches to satisfy these G^1 constraints. We explore the space of possible solutions, considering several projections on the solution space leading to different explicit formulas for the masks. Certain control points are computed by means of degree elevation of the C^0 scheme of Loop and Schaefer [22], while for others, explicit masks are deduced by providing closed-form solutions of the G^1 conditions, expressed in terms of the masks. We come up with four different schemes and conduct curvature analysis on an extensive benchmark in order to assert the quality of the resulting surfaces and identify the ones that lead to the best result, both visually and numerically. We demonstrate that the resulting surfaces converge quadratically to the CC limit when the mesh is subdivided.

Keywords: Bézier patches, gluing data, subdivision surface, extraordinary vertices, geometric continuity

1. Introduction

Describing accurately a complex shape in terms of its main geometric features is a major challenge in geometric modeling. It aims at providing compact and efficient models for approximating or representing precisely geometric objects, favoring the exploitation of such models in simulation and optimization processes.

B-Spline representations make a step toward this objective, by describing smooth shapes in terms of control points, which express the parametric features of such parametrized surfaces. However, for the representation of complex shapes, for instance those appearing in Computer-aided Design and Manufacturing (CAD-CAM), these parametric surfaces are usually trimmed and assembled in patchworks to define the

*Corresponding author

Email addresses: michelangelo.marsala@inria.fr (Michelangelo Marsala), angelos.mantzaflaris@inria.fr (Angelos Mantzaflaris), bernard.mourrain@inria.fr (Bernard Mourrain)

complete geometric object. This results in shape descriptions, which may not be exact, that is, they exhibit leaking patches, stretched elements, etc., which do not adhere to the intended geometry of the objects.

Another approach that has been investigated to address this challenge involves subdivision surfaces. Starting from a coarse mesh, which controls the geometric features of the surface, a subdivision process is applied iteratively to obtain finer and finer meshes, which converge to the limit surface. The Catmull-Clark scheme [11] is a well-known subdivision scheme, which operates on quadrilateral meshes. The limit surface is smooth, but it corresponds to an infinite number of bicubic B-Spline surfaces nearby an Extraordinary Vertex (EV) of the initial mesh. Moreover, the surface exhibits unwanted oscillations around EVs.

In [10] the author makes use of non-uniform B-spline knot insertion rules and spectral analysis to obtain subdivision schemes with bounded curvature around EVs. In [22], an approximation of Catmull-Clark subdivision surfaces by bicubic B-spline patches was proposed. It can be used to render efficiently such surfaces, but it also provides a compact geometric model description, composed of a collection of Bézier patches that can be directly used in CAD-CAM. The control points of the Bézier patches are obtained from the initial coarse mesh using explicit masks, which depend only on the valence of the vertices. Unfortunately, the bicubic Approximate Catmull-Clark surface (ACC_3) is not G^1-smooth nearby an EV: the tangent planes of patches at a point of a common edge through an EV are distinct (see Fig. 1). For a mathematical definition of geometric continuity or G^1-smoothness via connecting diffeomorphisms, we refer to [16].

The construction of G^1-smooth B-spline surfaces has been extensively investigated over the last decades. To cite a few works, in [14] and [27], G^1 bicubic macro-patch surfaces for quad-meshes are studied. The minimal number of splits of these macro-patches is analyzed for linear gluing data between two adjacent patches. In [15], bicubic 4-split macro patches and linear gluing data are studied. In [6], the space G^1-smooth 4-split splines on a quad-mesh with quadratic gluing data is analyzed, providing dimension formulae and basis constructions.

The article [25] proposed a construction of a general G^1 surfaces with biquartic B-spline defined on the mid-point refined mesh. The control points are computed as explicit combinations of the initial vertices and of face points deduced from blending ratios of the face corners. Differently from our approach, this construction does not ensure the interpolation of the Catmull-Clark limit surface on the mesh's vertices.In [26], a construction of bicubic B-spline with 6 internal knots along each direction is proposed to replace the neighborhood of an EV on a Catmull-Clark subdivision surface by a G^1-smooth B-spline surface. The solution is given implicitly by describing and solving the linear constraints. Moreover, this construction requires more than one initial applications of Catmull-Clark subdivision on the input mesh; this requirement is not needed in our construction.

In [30], arbitrary G^1-smooth 4-split biquintic B-spline surfaces with a single interior knot on quadrilateral meshes are constructed by solving the so-called tangent and twist derivatives conditions at the vertices as well as some linear relations on the interior control points along an edge. The control points are computed as the solution of a linear system, in contrast to the present work, where a local and explicit computation for the control points is provided by smoothing masks. Indeed, instead of solving the linear system that expresses the regularity of the functions, we work directly with explicit masks for the Bézier control points, therefore removing the need of a linear solver.In [7], biquartic 4-split macro-patches are constructed using quadratic gluing data. In these works, the control points are explicitly described in terms of some free control points and some shape parameters; moreover, the G^1 conditions are obtained by solving the global linear system obtained imposing directly the G^1 constraints between adjacent patches. The remaining degrees of freedom are fixed by imposing the minimization of the thin-plate energy. Differently from that work, we provide a complete and explicit construction for the surface without solving any global system. In [21] a guided subdivision approach is developed to construct a bi-3, bi-4 or bi-5 C^1 Bézier surface. This approach, in which a face split is carried out, is intended to obtain a final surface with regular isophotes across extraordinary vertices; to do that, free parameters are introduced and fixed using minimization of a functional to obtain a better result, whereas we provide explicit mask formulas for the construction of the patches.

The research on smooth splines over unstructured meshes is a revived topic due to its importance for isogeometric analysis discretizations [17]. In [3], the space on G^1-smooth Bézier patches of bidegree (n, n) is studied including dimensional analysis and basis constructions. The gluing data are built from bilinear parametrization of quadrilateral faces of the planar mesh. The idea of gluing data inferred from an initial C^0 multipatch B-spline model has also been explored for multipatch B-spline models in [13, 20, 19], and also in relation to locally refined splines [8, 9]. In [29] a construction of biquartic G^1 basis functions over T-meshes is presented. Te latter is devoted on basis computation for IGA simulation, rather than surface reconstruction. The basis functions are obtained solving a constrained optimization problem involving $20N + 1$ constraints, with N the valence of the extraordinary vertex, which can be computationally expensive.Recent approaches also include the D-patch method [31] or the approximate C^1 construction in [32]. For the case of subdivision surfaces, analysis methodologies have been explored eg. in [1, 2, 12, 28, 34], as well as related issues, notably efficient quadrature [2, 18].

Contributions. We present a new construction of G^1 surfaces from quad-meshes; for our development we will assume that the input mesh features isolated EVs. The constructed surface, made of biquintic Bézier patches, converges quadratically to the Catmull-Clark subdivision surface of the quad-mesh, interpolating it at extraordinary vertices and reproducing it with C^2 (resp. C^1) regularity around regular vertices (resp. vertices connected to an EV). By analyzing the conditions of G^1-regularity between two adjacent Bézier patches with the use of quadratic gluing functions, we present new explicit masks to compute the control points of the Bézier patches of the G^1 surface. We exploit the symmetry of the masks around (extraordinary) vertices to explicitly solve the linear systems obtained in the various steps of our construction, addressing the degeneracies of subsystems for even valence and introducing suitable degrees of freedom in the computation of the masks. Contrary to the classical subdivision approach, the method generates few number of control points (16 or 36 for each patch), which allows us to reconstruct directly the final surface; for this reason, the entire construction is computationally efficient. Moreover, a quantitative analysis of the G^1 surfaces corroborate the geometric quality of the construction.

Outline. After introducing some definitions and notation in Section 2, we recall the work of Loop and Schaefer [22] in Section 3. By applying twice the degree elevation algorithm to Loop and Schaefer's bicubic surface and making use of gluing data functions we obtain sufficient degrees of freedom to impose G^1 condition introduced around extraordinary patches (Section 4). In Section 5, we analyze the linear system induced by the equations defining the G^1 regularity along edges and provide closed formulas for its inverse that yield masks associated to the control points of the Bézier representation. The system is treated by higher-order Taylor expansion around EVs and explicit solutions are presented in all cases. In Section 6, we analyze the quality of the families of solutions obtained by the above constructions. We present isophote and curvature analysis, as well as numerical results for four variants of our scheme. Furthermore, we compare with the CC limit surface and show that our construction converges quadratically to that limit. Section 7 concludes the present work.

2. Preliminaries

Before going into the construction of the masks of the G^1 surface, we need to introduce some definitions that we will use throughout the paper. First, we introduce the definition of Bézier patch:

Definition 1 (Bézier Patch)**.** *A Bézier (quad) Patch is a surface defined as*

$$Q(u,v) = \sum_{i,j=0}^{n,m} \mathbf{b}_{i,j} B^{i,j}_{n,m}(u,v)\,, \quad (u,v) \in [0,1]^2\,,$$

where $\mathbf{b}_{i,j} \in \mathbb{R}^3$ *are the Bézier control points and* $B^{i,j}_{n,m}(u,v) = B^i_n(u) B^j_m(v)$ *are the bivariate Bernstein polynomials of bidegree* (n,m)*, that is,* $B^k_d(u) = \binom{d}{k} u^k (1-u)^{d-k}$.

Definition 2 (Mesh)**.** *A mesh* $\mathcal{M}$ *is a collection of vertices, edges and faces that defines the shape of a polyhedral object. We call Bézier face point, a point which corresponds to the interior of a mesh face; a point which corresponds to the interior of an edge is a Bézier edge point and a Bézier vertex point is a point which is shared with more than one edge.*

Let p be a vertex of a mesh: the valence N of the vertex is defined as the number of edges (or faces) to which it belongs: if $N = 4$ we will refer to p as a regular vertex, if $N \neq 4$ as an extraordinary vertex (EV). A patch presenting an EV among its vertices is called extraordinary patch. Moreover, we call first neighborhood of p the set of points directly connected to p by an edge and second neighborhood of p the set of points belonging to an extraordinary patch which are opposite to the EV (see Fig. 2).

Given an input mesh $\mathcal{M}$, we will use a *smoothing scheme* to construct a smooth surface:

Definition 3 (Smoothing Scheme)**.** *Given a mesh* $\mathcal{M}$ *we define a smoothing scheme through its smoothing matrix* $S = [\mathrm{M}^{(k)}_{i,j}]$ *as an operator such that*

$$\mathbf{b} = \mathbf{p}\, S\,, \quad \mathbf{b}^{(k)}_{i,j} = \sum_{\ell} \mathbf{p}_\ell \left(\mathrm{M}^{(k)}_{i,j}\right)_\ell\,,$$

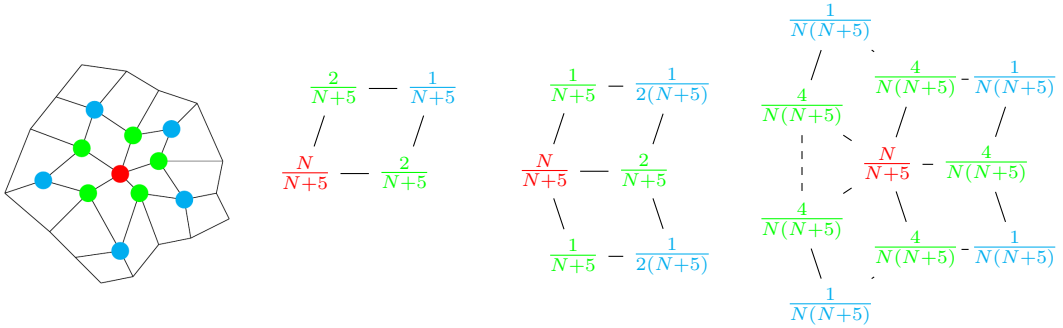
where $\mathbf{p} = [\mathbf{p}_1, \ldots, \mathbf{p}_m] \in \mathbb{R}^{3\times m}$ *is the set of mesh vertices and* $\mathbf{b} = [\mathbf{b}^{(k)}]$*, where* $\mathbf{b}^{(k)}$ *is the matrix containing the control points of the k-th patch. The smoothing matrix contains in its columns the different masks* M

defining the scheme: we have vertex point masks, edge point masks and face point masks. A mask is a set of coefficients (weights) returning the rules with whom the control points are computed from the mesh vertices: the weights of a smoothing (or Bézier) mask must sum to 1 and they can be, possibly, negative.

Usually, the size of the smoothing matrix S depends solely on the valence of the vertex to whom it is attached and to its first and second neighborhood; we will refer to smoothing masks using the notation $\mathrm{M}=[\omega_\bullet, \boldsymbol{\omega}, \boldsymbol{\omega'}]^T$, where $\omega_\bullet$ is the weight for the vertex point and $\boldsymbol{\omega} = \{\omega_i\}_{i=1}^N$ and $\boldsymbol{\omega'} = \{\omega'_i\}_{i=1}^N$ are the weights for its first and second neighborhoods, respectively. Here we will work with quad-meshes and we will provide masks which will return control points for each face of the mesh, therefore defining a collection of Bézier patches. Moreover, we will consider meshes with isolated EVs, i.e. surrounded by regular vertices. This assumption is not restrictive since any linked EVs can be decoupled with one subdivision of the mesh.

3. Bicubic Approximate Catmull-Clark ($\mathrm{ACC_3}$) and degree elevated ($\mathrm{ACC_5}$)

The bicubic approximate Catmull-Clark surface of [22] is a smoothing scheme reconstructing a bicubic surface starting from a quad-mesh which approximates the limit surface of the classical Catmull-Clark subdivision scheme; the application of this scheme to a mesh returns, on each face, 16 control points defining a bicubic Bézier patch. The surface obtained from $\mathrm{ACC_3}$ is C^2 everywhere except in a neighborhood of the EVs, in which it is only C^0 (see. Fig.1); moreover, the vertices generated by $\mathrm{ACC_3}$ interpolate the Catmull-Clark limit surface.



(a) (b) (c) (d)

Figure 2: (a): first neighborhood (green) and second neighborhood (blue) of a vertex (red) with valence $N = 5$. (b–d): $\mathrm{ACC_3}$ masks proposed in [22] for face ($\widehat{\mathbf{b}}_{1,1}$), edge ($\widehat{\mathbf{b}}_{1,0}$ and $\widehat{\mathbf{b}}_{0,1}$) and vertex ($\widehat{\mathbf{b}}_{1,1}$) Bézier points, respectively.

To recover regularity around these EVs without affecting the adjacent regular vertices, i.e. without compromising the interpolatory property and C^1 smoothness of the surrounding regular vertices, cubic or even quartic patches do not suffice. Consequently, we will work with biquintic patches, defined by 36 Bézier points each. As a starting point for our construction we use the degree elevated masks $\mathrm{ACC_5}$, which are obtained after applying twice the degree elevation algorithm to $\mathrm{ACC_3}$ masks; we will use underlined symbols to denote $\mathrm{ACC_5}$ masks i.e. $\underline{\mathrm{M}} = [\underline{\omega}_\bullet, \underline{\boldsymbol{\omega}}, \underline{\boldsymbol{\omega}}']^T$. For example, the $\mathrm{ACC_5}$ masks $\underline{\mathrm{M}}_{2,2}$ and $\underline{\mathrm{M}}_{4,0}$ returning the Bézier points $\mathbf{b}_{2,2}$ and $\mathbf{b}_{4,0}$, respectively, are given by

$$\underline{\mathrm{M}}_{2,2} = \frac{1}{100}\left(\widehat{\mathrm{M}}_{0,0} + 6\widehat{\mathrm{M}}_{1,0} + 3\widehat{\mathrm{M}}_{2,0} + 6\widehat{\mathrm{M}}_{0,1} + 36\widehat{\mathrm{M}}_{1,1} + 18\widehat{\mathrm{M}}_{2,1} + 3\widehat{\mathrm{M}}_{0,2} + 18\widehat{\mathrm{M}}_{1,2} + 9\widehat{\mathrm{M}}_{2,2}\right),$$
$$\underline{\mathrm{M}}_{4,0} = \frac{1}{5}\left(3\widehat{\mathrm{M}}_{2,0} + 2\widehat{\mathrm{M}}_{3,0}\right),$$

where the $\widehat{\mathrm{M}}_{i,j}$ are the $\mathrm{ACC_3}$ bicubic masks (see Fig. 2-(b–d)). Note that the degree elevation is applied here directly on the considered masks to obtain the new weights.

The masks for Bézier points can be written as vectors; for instance, the corner point mask $\underline{\mathrm{M}}_{0,0}$ is given as follows:

$$\underline{\mathrm{M}}_{0,0} \in \mathbb{R}^{2N+1}, \text{ with } \underline{\mathrm{M}}_{0,0} = \frac{1}{N(N+5)}[N^2, 4, \ldots, 4, 1, \ldots, 1]^T = \frac{1}{N(N+5)}[N^2, \mathbf{4}, \mathbf{1}]^T, \tag{1}$$

where the first coordinate is the weight for the EV, followed by the weights for the vertices of first neighborhood and then the second neighborhood (cf. Fig. 2-(a)). We will use this vector notation for masks throughout the paper.

4. G^1 Constraints

The approach we use to recover regularity in the neighborhood of EVs is to compute new masks of a scheme returning Bézier control points (of a biquintic patch) s.t. the induced surface is G^1–continuous.

Definition 4 (Multipatch surface)**.** *A surface defined on a mesh $\mathcal{M}$ is a collection of patches $f = (f_\sigma)_{\sigma\in\mathcal{M}}$ where σ is a face of the mesh and f_σ is called the restriction of f on the face σ.*

A surface is said to be G^1 if for every point on it there exists a unique tangent plane; using multipatch surfaces it is easy to understand that, without any constraints, this kind of regularity is not automatically ensured along a common edge shared by two adjacent patches. Therefore, to introduce the G^1–regularity between two adjacent patches we use gluing data functions:

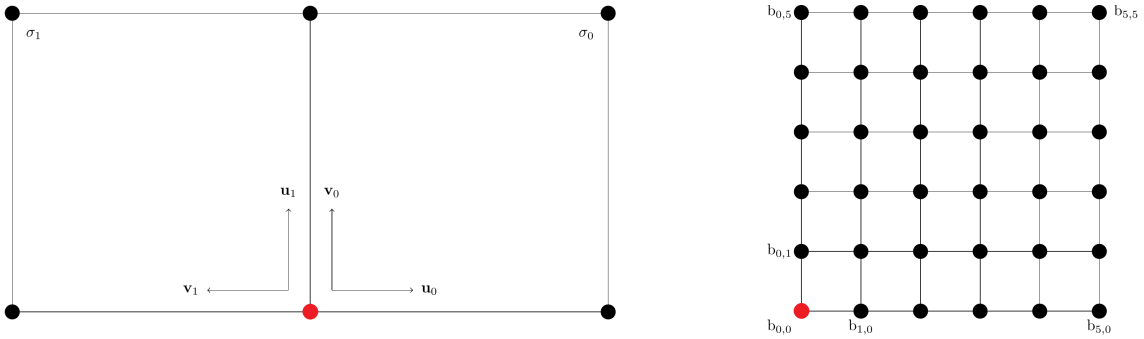


Figure 3: Left: local coordinate systems between two patches. Right: control point labeling in a biquintic patch.

Definition 5 (Gluing data)**.** *We call gluing data the functions $a_{N,N'}, b_{N,N'}$ s.t.*

$$a_{N,N'} : e \longrightarrow \mathbb{R}\,, \quad b_{N,N'} : e \longrightarrow \mathbb{R}\,,$$

where e is the common edge shared between two adjacent extraordinary patches of the mesh $\mathcal{M}$ and connecting a vertex of valence N with one of valence N'.

From [16], a differentiable function $f = (f_\sigma)_{\sigma\in\mathcal{M}}$ on a mesh $\mathcal{M}$ has to verify the following conditions:

$$\begin{cases} f_1(u_1, 0) = f_0(0, u_1)\,, \\ \dfrac{\partial f_1}{\partial v_1}(u_1, 0) = b_{N,N'}(u_1)\dfrac{\partial f_0}{\partial u_0}(0, u_1) + a_{N,N'}(u_1)\dfrac{\partial f_0}{\partial v_0}(0, u_1)\,, \end{cases} \qquad u_1 \in [0, 1]\,, \tag{2}$$

where f_0, f_1 are the functions defined on faces $\sigma_0, \sigma_1 \in \mathcal{M}$ sharing an edge e (see Fig. 3) and $[a_{N,N'}, b_{N,N'}]$ are the gluing data along the edge e. Since our construction is developed in the parametric domain, we take $e = [0, 1]$. The functions satisfying eq. (2) for all shared edges belong to the linear space of G^1 B-spline functions on $\mathcal{M}$. In this work, we use quadratic symmetric gluing data [7, 15, 16] defined as

$$\begin{aligned} &a_{N,N'}(u) = a_0 B_2^0(u) - a_2 B_2^2(u)\,, \quad \text{where} \quad a_0 = 2\cos\left(\frac{2\pi}{N}\right)\,,\ a_2 = 2\cos\left(\frac{2\pi}{N'}\right)\,, \\ &b_{N,N'}(u) = -1\,. \end{aligned} \tag{3}$$

In the cases we study, we will assume that $N' = 4$ so that $a_2 = 0$, that is an EV is surrounded by regular vertices. Translating eq. (2) in terms of Bézier patches and using (3), we obtain

$$\begin{cases} \displaystyle\sum_{i=0}^{5} \mathbf{b}_{i,0}^{(1)} B_5^i(u) = \sum_{i=0}^{5} \mathbf{b}_{0,i}^{(0)} B_5^i(u)\,, \\ \displaystyle\sum_{i=0}^{5} \left(\mathbf{b}_{i,1}^{(1)} - \mathbf{b}_{i,0}^{(1)} + \mathbf{b}_{1,i}^{(0)} - \mathbf{b}_{0,i}^{(0)}\right) B_5^i(u) = a_0 B_2^0(u) \left(\sum_{i=0}^{4} \left(\mathbf{b}_{0,i+1}^{(0)} - \mathbf{b}_{0,i}^{(0)}\right) B_4^i(u)\right)\,. \end{cases} \tag{4}$$

If we call $\mathrm{M}_{i,j}^{(k)}$ the smoothing mask returning the control point $\mathbf{b}_{i,j}^{(k)}$ belonging to the patch k, solving eq. (4) with respect to $\mathrm{M}_{0,1}^{(1)}, \mathrm{M}_{1,1}^{(1)}, \mathrm{M}_{2,1}^{(1)}, \mathrm{M}_{3,1}^{(1)}, \mathrm{M}_{4,1}^{(1)}, \mathrm{M}_{5,0}^{(1)}, \mathrm{M}_{5,1}^{(1)}$, we obtain after some reductions the following relations to be satisfied:

$$\begin{aligned}
\mathrm{M}_{0,1}^{(1)} + \mathrm{M}_{1,0}^{(0)} &= \bar{a}_0 \mathrm{M}_{0,0}^{(1)} + \ a_0 \mathrm{M}_{1,0}^{(1)}\,, && (5)\\
5(\mathrm{M}_{1,1}^{(1)} + \mathrm{M}_{1,1}^{(0)}) &= a_0 \mathrm{M}_{0,0}^{(1)} + 5\bar{a}_0 \mathrm{M}_{1,0}^{(1)} + \ 4a_0 \mathrm{M}_{2,0}^{(1)}\,, && (6)\\
10(\mathrm{M}_{2,1}^{(1)} + \mathrm{M}_{1,2}^{(0)}) &= -a_0 \mathrm{M}_{0,0}^{(1)} + 5a_0 \mathrm{M}_{1,0}^{(1)} + 10\bar{a}_0 \mathrm{M}_{2,0}^{(1)} + \ 6a_0 \mathrm{M}_{3,0}^{(1)}\,, && (7)\\
10(\mathrm{M}_{3,1}^{(1)} + \mathrm{M}_{1,3}^{(0)}) &= a_0 \mathrm{M}_{0,0}^{(1)} - 5a_0 \mathrm{M}_{1,0}^{(1)} + 10a_0 \mathrm{M}_{2,0}^{(1)} + 10\bar{a}_0 \mathrm{M}_{3,0}^{(1)} + 4a_0 \mathrm{M}_{4,0}^{(1)}\,, && (8)\\
\mathrm{M}_{4,1}^{(1)} + \mathrm{M}_{1,4}^{(0)} &= 2\mathrm{M}_{4,0}^{(1)}\,, && (9)\\
\mathrm{M}_{5,1}^{(1)} + \mathrm{M}_{1,5}^{(0)} &= 2\mathrm{M}_{5,0}^{(1)}\,, && (10)\\
10(\mathrm{M}_{3,0}^{(1)} - \mathrm{M}_{2,0}^{(1)}) &= \mathrm{M}_{0,0}^{(1)} - 5\mathrm{M}_{1,0}^{(1)} + 5\mathrm{M}_{4,0}^{(1)} - \mathrm{M}_{5,0}^{(1)}\,, && (11)
\end{aligned}$$

with $\bar{a}_0 = 2 - a_0$.

Masks verifying eq. (5)-(11) generate a family of G^1 smoothing schemes. From now on the dependency of the mask from the patch will be specified only when it is different from patch 1. Particularly interesting is eq. (11): if we use quadratic gluing data, the degree of the left-hand side of (2) is ≤ 5 while the right-hand side has degree $4 + 2 = 6$. Equation (11) reflects the constrain that the left-hand side has degree ≤ 5: in fact it is saying that even if we are working with biquintic patches, along the edge we have a quartic curve. Moreover, we notice that eq. (9) and (10) are always satisfied by ACC_5 masks: since we are in the vicinity of a regular vertex, we know that the ACC_5 surface is C^2 there; it means that the first derivatives along opposite directions across the regular vertex are the same, i.e. $\mathrm{M}_{1,5}^{(0)} - \mathrm{M}_{0,5}^{(0)} = -(\mathrm{M}_{5,1}^{(1)} - \mathrm{M}_{5,0}^{(1)})$ as $\mathrm{M}_{0,5}^{(0)} = \mathrm{M}_{5,0}^{(1)}$, which is in fact eq. (10). Then, since the limit surface is also at least C^1 across regular vertices (or C^2 in some cases), we have, additionally, equal second order derivatives along opposite directions i.e. $\mathrm{M}_{1,4}^{(0)} - \mathrm{M}_{0,4}^{(0)} = -(\mathrm{M}_{4,1}^{(1)} - \mathrm{M}_{4,0}^{(1)})$ that is exactly eq. (9) since $\mathrm{M}_{0,4}^{(0)} = \mathrm{M}_{4,0}^{(1)}$.

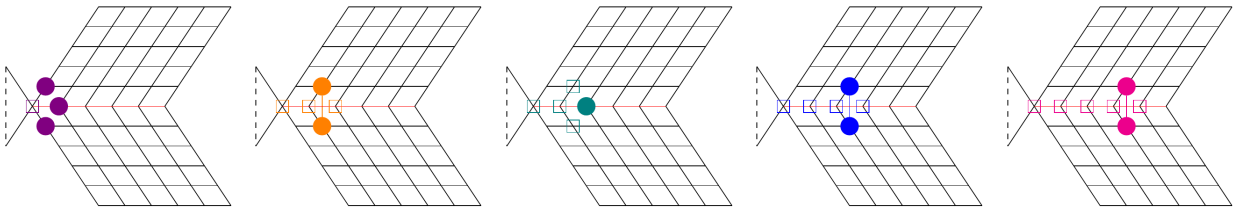
5. Computing the Bézier masks

In this section, we solve the system (5)-(11) to obtain masks for the Bézier patches ensuring G^1 regularity of the resulting surface. The process is done incrementally with respect to the order of the Bézier mask where the order refers to the sum of the subscripts of the Bézier mask. In particular, for Bézier masks of order zero, i.e. vertex points, we take $\mathrm{M}_{0,0} = \underline{\mathrm{M}}_{0,0}$ as in (1). For Bézier points of order one we will solve eq. (5); since the latter involves Bézier points on different patches all around the EV, eq. (5) translates into a circulant system to be solved. Eq. (6) presents constraints regarding masks of second order and it can be solved in two different ways; if we want to solve it with respect to $\mathrm{M}_{1,1}$, since it is related to points around the EV in different patches similarly to eq. (5), it translates into a circulant system. On the other hand, if we want to solve it with respect to the mask $\mathrm{M}_{2,0}$ it can be directly obtained from ACC_5 masks. Then, eq. (7)-(8) regarding Bézier points of order three and four, respectively, can be solved by direct computation in either a symmetric or asymmetric approach. Moreover, the masks $\mathrm{M}_{4,0}, \mathrm{M}_{5,0}, \mathrm{M}_{4,1}, \mathrm{M}_{5,1}, \mathrm{M}_{2,2}, \mathrm{M}_{3,2}, \mathrm{M}_{4,2}, \mathrm{M}_{5,2}, \mathrm{M}_{2,3}, \mathrm{M}_{3,3}, \mathrm{M}_{4,3}, \mathrm{M}_{5,3}$ as well as $\mathrm{M}_{0,4}, \mathrm{M}_{1,4}, \mathrm{M}_{2,4}, \mathrm{M}_{3,4}, \mathrm{M}_{4,4}, \mathrm{M}_{5,4}, \mathrm{M}_{0,5}, \mathrm{M}_{1,5}, \mathrm{M}_{2,5}, \mathrm{M}_{3,5}, \mathrm{M}_{4,5}, \mathrm{M}_{5,5}$ are equal to those of ACC_5.

Throughout the construction we assume that the first neighborhood of an EV is composed solely by regular vertices; this assumption is not restrictive, since it is satisfied by applying at most one CC subdivision step to the mesh. Moreover, biquintic Bézier points will be computed only in the faces adjacent to EVs, otherwise we will use the standard bicubic patch of ACC_3. First we focus on inner EVs and postpone the discussion of boundaries until Section 5.4.

5.1. Bézier masks of order one

We compute first the masks $\mathrm{M}_{1,0} = \mathrm{M}_{1,0}^{(1)}$ and $\mathrm{M}_{0,1}^{(0)}$ using eq. (5) as follows. If the valence of the EV is $N = 3$, eq. (5) is automatically satisfied by ACC_5 masks. Then, we do not need to determine new masks for this relation. Otherwise, for valences $N \geq 5$ we have to compute new masks verifying (5); given the mask for the vertex point $\mathrm{M}_{0,0} = \underline{\mathrm{M}}_{0,0} = [\underline{\alpha}_\bullet, \underline{\boldsymbol{\alpha}}, \underline{\boldsymbol{\alpha}'}]^T$, we shall compute the mask $\mathrm{M}_{1,0} = [\beta_\bullet, \boldsymbol{\beta}, \boldsymbol{\beta}']^T$. We observe



(a) (b) (c) (d) (e)

Figure 4: Bézier points involved in G^1 constraints. (a) first derivatives: disposal of the control points involved in the left-hand side (circles) and right-hand side (squares) of (13). (b)-(c) second derivatives: disposal of Bézier points regarding the left-hand side and right-hand side of (17) and (29), respectively. (d)-(e) third and fourth derivatives: disposal of the control points utilized, respectively, in the left-hand side and right-hand side of (30) and (32).

that the masks $\mathrm{M}_{0,1} = \mathrm{M}_{0,1}^{(1)}$ and $\mathrm{M}_{1,0}^{(0)}$ can be obtained by permuting the weights of $\mathrm{M}_{1,0}$. In particular, let $C = \mathrm{diag}(1, \widehat{C}, \widehat{C})$ be a $(2N+1) \times (2N+1)$ block diagonal matrix where

$$\widehat{C} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1 \\ 1 & 0 & & \cdots & 0 \end{pmatrix} = \mathrm{Circ}(0, 1, 0, \ldots, 0) \tag{12}$$

is a $N \times N$ circulant matrix; it follows that $\mathrm{M}_{1,0}^{(0)} = C\,\mathrm{M}_{1,0}$, $\mathrm{M}_{0,1} = C^{-1}\mathrm{M}_{1,0}$, where $C^{-1} = \mathrm{diag}(1, \widehat{C}^{-1}, \widehat{C}^{-1})$ and $\widehat{C}^{-1} = \widehat{C}^{N-1}$. Therefore eq. (5) implies the following system to be solved:

$$(-a_0 I + C + C^{N-1})\mathrm{M}_{1,0} = (2 - a_0)\underline{\mathrm{M}}_{0,0}\,, \tag{13}$$

where I is the $(2N+1) \times (2N+1)$ identity matrix; Fig. 4-(a) shows the position of the control points involved in this construction.

Proposition 1. *Let $A = -a_0 I + C + C^{N-1}$ be as in eq. (13). The matrix A is symmetric and singular.*

Proof. From the particular shape of A, its symmetry is straightforward since $C^T = C^{-1} = C^{N-1}$. Matrix A can be decomposed as

$$A = -a_0 I + C + C^{N-1} = C^{N-1}(C^2 - a_0 C + I) = C^{N-1}(C - z_1 I)(C - z_{N-1} I)\,,$$

where z_1, z_{N-1} are the roots of the polynomial $x^2 - a_0 x + 1$, with $a_0 = 2\cos(\theta)$, $\theta = 2\pi/N$. If $\mathtt{i} = \sqrt{-1}$ is the imaginary unit, they are obtained as $z_1 = e^{\mathtt{i}\theta}$, $z_{N-1} = e^{-\mathtt{i}\theta}$ i.e. they are the N-th roots of unity $z_k = e^{\mathtt{i}\theta k}$ corresponding to the angle θ for $k = 1$ and $k = N-1$. Now, $\det(C - z_i I) = (1 - z_i)\det(\widehat{C} - z_i \widehat{I})^2$, $i = 1, N-1$, where $\widehat{I}$ is the $N \times N$ identity matrix; since $\widehat{C}^N - \widehat{I} = 0$ we have

$$\det(\widehat{C} - \lambda \widehat{I}) = (-1)^N(\lambda^N - 1) \tag{14}$$

from which we deduce that $\det(\widehat{C} - z_i \widehat{I}) = (-z_i)^N + (-1)^{N+1} = 0$, $i = 1, N-1$. This shows that $A = C^{N-1}(C - z_1 I)(C - z_{N-1} I)$ is not invertible. □

The system (13) can be therefore solved; the following theorem ensures its solvability.

Theorem 1. *The system (13) admits a four-dimensional affine space of solutions.*

Proof. Let $\mathbf{b} = (2 - a_0)\underline{\mathrm{M}}_{0,0}$ be the right-hand side of (13); we want to show that $\mathbf{b} \in \mathrm{Im}(A)$, i.e. $W\mathbf{b} = \mathbf{0}$, where $\mathrm{Im}(W^T) = \mathrm{Leftker}(A) = \mathrm{Ker}(A)$ since A is symmetric. Recalling the identity

$$\mathbf{0} = C^N - I = \prod_{j=0}^{N-1} (C - z_j I)\,,$$

with $z_k = e^{\mathtt{i}\theta k}$ the N-th roots of unity, it is easy to prove that we can take

$$W^T = \prod_{\substack{j=0 \\ j\neq 1}}^{N-2} (C - z_j I) = \mathrm{diag}(0, \widehat{W}^T, \widehat{W}^T),$$

in which $\widehat{W}^T = \prod_{\substack{j=0 \\ j\neq 1}}^{N-2} (\widehat{C} - z_j \widehat{I})$ and $\widehat{C}$ as in eq. (12): in fact

$$WA = (AW^T)^T = \left(C^{N-1}(C - z_1 I)(C - z_{N-1} I) \prod_{\substack{j=0 \\ j\neq 1}}^{N-2} (C - z_j I) \right)^T = \left(C^{N-1} \prod_{j=0}^{N-1} (C - z_j I) \right)^T = \mathbf{0}$$

since the matrices $(C - z_j I)$ commute. Collecting the constants from the product we have

$$W\mathbf{b} = \left(\mathbf{b}^T W^T\right)^T = \left(\frac{2 - a_0}{N(N+5)} \mathrm{diag}(N^2, 4\widehat{I}, \widehat{I})(1 - z_0, \mathbf{1} - \boldsymbol{z_0}, \mathbf{1} - \boldsymbol{z_0}) \prod_{j=2}^{N-2} (C - z_j I) \right)^T = \mathbf{0}$$

since $z_0 = 1$; due to the particular block structure of A, we deduce that $\mathrm{corank}(A) = 4$. Thus the solutions of (13) form an affine space of $\dim 4$. □

In order to compute a solution of eq. (13), we add 4 extra constraints to the coefficient matrix; then, since we arrive to an overdetermined system, the solution can be obtained applying the Least Square (LS) algorithm. Let $\widehat{\mathbf{k}}_1$ and $\widehat{\mathbf{k}}_2$ be the two (column) vectors of one basis of $\widehat{W}^T$: with this choice it results that $\mathrm{Im}(W^T) = \mathrm{Span}\{\mathbf{k}_1, \mathbf{k}_2, \mathbf{k}_3, \mathbf{k}_4\}$ where $\mathbf{k}_1 = (0, \widehat{\mathbf{k}}_1^T, \mathbf{0})^T$, $\mathbf{k}_2 = (0, \widehat{\mathbf{k}}_2^T, \mathbf{0})^T$, $\mathbf{k}_3 = (0, \mathbf{0}, \widehat{\mathbf{k}}_1^T)^T$, $\mathbf{k}_4 = (0, \mathbf{0}, \widehat{\mathbf{k}}_2^T)^T$. To obtain a new mask the closest possible to $\underline{\mathrm{M}}_{1,0}$ we project their difference onto $\mathrm{Im}(W^T)$; this projection, in principle, can be executed arbitrary, i.e. choosing any direction. Let M_* be a particular solution of (13); then, the general solution of (13) is given by

$$\mathrm{M}_{1,0} = \mathrm{M}_* + \sum_{i=1}^{4} \mu_i \mathbf{k}_i \,, \quad \mu_i \in \mathbb{R} \,. \tag{15}$$

Since the rows of A sum up to $2 - a_0$, i.e. $A\,(1, \mathbf{1}, \mathbf{1})^T = (2 - a_0)\,(1, \mathbf{1}, \mathbf{1})^T$, it is immediate to notice that we can take $\mathrm{M}_* = \underline{\mathrm{M}}_{0,0}$. Here we choose to perform an orthogonal projection $(\mathrm{M}_{1,0} - \underline{\mathrm{M}}_{1,0}) \perp \mathrm{Im}(W^T)$ i.e. $\langle \mathrm{M}_{1,0} - \underline{\mathrm{M}}_{1,0} \,|\, \mathbf{k}_i \rangle = 0$, $i = 1, 2, 3, 4$, where $\langle \cdot | \cdot \rangle$ represents the usual Euclidean scalar product. This leads to the solution

$$\boldsymbol{\mu} = K^{-1} \widetilde{\mathbf{b}} \,, \tag{16}$$

where $K^{-1} = \mathrm{diag}(\widehat{K}^{-1}, \widehat{K}^{-1})$, $\widehat{K} = \left[\langle \widehat{\mathbf{k}}_i \,|\, \widehat{\mathbf{k}}_j \rangle \right]_{i,j=1,2}$ is the 2×2 orthogonal projection matrix and

$$\widetilde{\mathbf{b}} = (\mathbf{k}_1 \ \mathbf{k}_2 \ \mathbf{k}_3 \ \mathbf{k}_4)^T \, \underline{\mathrm{M}}_{1,0} \,.$$

We also observe that if we solve (15) for $N = 3$, we obtain again the ACC_5 mask $\underline{\mathrm{M}}_{1,0}$.

Example 1. In the case of an EV with valence $N = 6$, using the construction in Theorem 1 we obtain $W^T = \mathrm{Span}\{\mathbf{k}_1, \mathbf{k}_2, \mathbf{k}_3, \mathbf{k}_4\}$, $\widehat{\mathbf{k}}_1^T = (1, 0, -1, -1, 0, 1)$, $\widehat{\mathbf{k}}_2^T = (-1, -1, 0, 1, 1, 0)$. Hence, from eq. (15) we compute the coefficients for the orthogonal projection which are

$$\mu_1 = \frac{3}{110} \,, \quad \mu_2 = -\frac{3}{110} \,, \quad \mu_3 = \frac{3}{220} \,, \quad \mu_4 = 0 \,,$$

and substituting them in eq. (16) we obtain the weights for the mask $\mathrm{M}_{1,0}$

$$\mathrm{M}_{1,0} = \frac{1}{660} [360, 76, 58, 22, 4, 22, 58, 19, 10, 1, 1, 10, 19]^T \,.$$

5.2. *Bézier masks of second order*

Equation (6) can be solved both with respect to $\mathrm{M}_{1,1}$ or $\mathrm{M}_{2,0}$. Let us investigate the two cases.

5.2.1. *Deriving* $\mathrm{M}_{1,1}$ *by assigning* $\mathrm{M}_{2,0}$*: circulant system approach*

In this section we use a circulant system approach to obtain $\mathrm{M}_{1,1}$. We start by defining the mask $\mathrm{M}_{2,0}$, which is done differently for odd and even valence. Since $\mathrm{M}_{1,1}^{(0)} = C\,\mathrm{M}_{1,1}$, eq. (6) translates into the following linear system:

$$(I + C)\mathrm{M}_{1,1} = \mathbf{d}\,, \quad \text{where} \quad \mathbf{d} = \frac{1}{5}\left(a_0\underline{\mathrm{M}}_{0,0} - 5(a_0 - 2)\mathrm{M}_{1,0} + 4a_0\mathrm{M}_{2,0}\right)\,. \tag{17}$$

The involved Bézier points are highlighted in Fig. 4-(b). Let $B = I + C$ be the matrix in (17). To solve the system we have to distinguish the cases N odd and N even.

5.2.1.1. Case N odd. In this case, from eq. (11), substituting known masks we get

$$\mathrm{M}_{2,0} = [\delta_\bullet, \boldsymbol{\delta}, \boldsymbol{\delta}']^T = \frac{1}{10}\left(-\underline{\mathrm{M}}_{0,0} + 5\mathrm{M}_{1,0} + 10\underline{\mathrm{M}}_{3,0} - 5\underline{\mathrm{M}}_{4,0} + \underline{\mathrm{M}}_{5,0}\right)\,. \tag{18}$$

We have the following:

Proposition 2. *The matrix $B = I + C$ is invertible with $B^{-1} = \mathrm{diag}\left(\frac{1}{2}, \widehat{B}^{-1}, \widehat{B}^{-1}\right)$ where $\widehat{B} = \widehat{I} + \widehat{C}$ and*

$$\widehat{B}^{-1} = \frac{1}{2}\sum_{j=1}^{N}(-1)^{j-1}\widehat{C}^{j-1}. \tag{19}$$

Proof. It follows from

$$\begin{aligned}\left(\sum_{j=1}^{N}(-1)^{j-1}\widehat{C}^{j-1}\right)\left(\widehat{I} + \widehat{C}\right) &= \sum_{j=1}^{N}(-1)^{j-1}\widehat{C}^{j-1} + \sum_{j=1}^{N}(-1)^{j-1}\widehat{C}^{j} \\ &= \sum_{j=1}^{N}(-1)^{j-1}\widehat{C}^{j-1} + \sum_{k=2}^{N+1}(-1)^{k-2}\widehat{C}^{k-1} = \widehat{C}^0 + \widehat{C}^N = 2\widehat{I}\,,\end{aligned}$$

that we have relation (19). □

We are now able to compute explicitly $\mathrm{M}_{1,1}$.

Corollary 1. *The weights for the mask $\mathrm{M}_{1,1} = [\gamma_\bullet, \boldsymbol{\gamma}, \boldsymbol{\gamma}']^T$ in the odd case are given in terms of the masks $\underline{\mathrm{M}}_{0,0} = [\underline{\alpha}_\bullet, \underline{\boldsymbol{\alpha}}, \underline{\boldsymbol{\alpha}}']^T$, $\mathrm{M}_{1,0} = [\beta_\bullet, \boldsymbol{\beta}, \boldsymbol{\beta}']^T$ and $\mathrm{M}_{2,0} = [\delta_\bullet, \boldsymbol{\delta}, \boldsymbol{\delta}']^T$ by the following relations:*

$$\begin{aligned}\gamma_\bullet &= \frac{1}{5}(5 - 2a_0)\underline{\alpha}_\bullet + \frac{2}{5}a_0\delta_\bullet\,, \\ \gamma_i &= \frac{1}{10}a_0\sum_{j=1}^{N}(-1)^{j-1}\underline{\alpha}_{i+j-1} - \frac{1}{2}(a_0 - 2)\sum_{j=1}^{N}(-1)^{j-1}\beta_{i+j-1} + \frac{2}{5}a_0\sum_{j=1}^{N}(-1)^{j-1}\delta_{i+j-1}\,, \quad i = 1, \ldots, N\,,\end{aligned} \tag{20}$$

where the indices $i + j - 1$ are understood mod N *when $i + j - 1 > N$; a similar relation for $\boldsymbol{\gamma}'$ is obtained by substituting the values $\underline{\boldsymbol{\alpha}}'$, $\boldsymbol{\beta}'$, $\boldsymbol{\delta}'$ in (20) .*

Proof. The proof is obtained by substituting $\mathrm{M}_{1,1} = B^{-1}\mathbf{d}$ in (19). □

5.2.1.2. Case N even. If we have an even valence, the matrix B is not invertible and in fact $\mathrm{corank}(B) = 2$, due to $\mathrm{corank}(\widehat{B}) = 1$ since from eq. (14) with $\lambda = -1$ we have $\det(\widehat{B}) = 0$ and $\widehat{B}_{1,1} = 1$ where $\widehat{B}_{1,1}$ is the minor of $\widehat{B}$ obtained removing the 1-st row and column, since its cofactor is an upper triangular matrix with only ones on the diagonal. This implies $\mathrm{corank}(B) = 2$. It means that we must add two extra constraints in order to have a solution for the system; the obtained solution has to be meant in the LS sense. First of all we must check if system (17) admits a solution: denoting $\mathbf{w}_1 = (0, \widehat{\mathbf{w}}^T, \mathbf{0})^T$, $\mathbf{w}_2 = (0, \mathbf{0}, \widehat{\mathbf{w}}^T)^T$, where $\widehat{\mathbf{w}} = (1, -1, 1, \dots, -1)^T$, and $Y = (\mathbf{w}_1\ \mathbf{w}_2)$, it follows that $\mathrm{Im}(Y^T) = \mathrm{Span}\{\mathbf{w}_1, \mathbf{w}_2\} = \mathrm{Leftker}(B)$; in fact, one can see that $Y^T B = \mathbf{0}$. Then, the system (17) admits a solution if and only if $\mathbf{d} \in \mathrm{Im}(B)$ or, equivalently,

$$Y^T \mathbf{d} = \mathbf{0}\,. \tag{21}$$

Relation (21) is known as *Vertex Enclosure Relation* (V.E.R.).
In this section $\mathrm{M}_{2,0}^{\mathrm{even}} = [\delta_\bullet^{\mathrm{even}}, \boldsymbol{\delta}^{\mathrm{even}}, \boldsymbol{\delta}'^{\mathrm{even}}]^T = \frac{1}{10}\left(-\underline{\mathrm{M}}_{0,0} + 5\mathrm{M}_{1,0} + 10\underline{\mathrm{M}}_{3,0} - 5\underline{\mathrm{M}}_{4,0} + \underline{\mathrm{M}}_{5,0}\right)$. Checking property (21) for the masks $\underline{\mathrm{M}}_{0,0}$, $\mathrm{M}_{1,0}$ and $\mathrm{M}_{2,0}^{\mathrm{even}}$, we notice that $Y^T\underline{\mathrm{M}}_{0,0} = \mathbf{0}$, $Y^T\mathrm{M}_{1,0} = \mathbf{0}$ but $Y^T\mathrm{M}_{2,0}^{\mathrm{even}} \neq \mathbf{0}$. Thus, we need to compute a new mask $\mathrm{M}_{2,0} = [\delta_\bullet, \boldsymbol{\delta}, \boldsymbol{\delta}']^T$ ensuring (21) in order to have a solution for system (17). To do this we notice that $\mathrm{Im}(B) = \mathrm{Leftker}(\mathrm{Im}(Y))$; in fact it follows that $BY = \mathbf{0}$ due to the particular structure of B. Moreover, we want that the mask $\mathrm{M}_{2,0}$ is close to $\mathrm{M}_{2,0}^{\mathrm{even}}$. Therefore we compute the new mask $\mathrm{M}_{2,0}$ by projecting its difference with $\mathrm{M}_{2,0}^{\mathrm{even}}$ onto the left kernel B and forcing the V.E.R to be verified; since we decided to operate an orthogonal projection, the sought mask $\mathrm{M}_{2,0}$ is obtained imposing $\langle \mathrm{M}_{2,0} - \mathrm{M}_{2,0}^{\mathrm{even}} \,|\, \mathbf{r}_i \rangle = 0$, $i = 1, \dots, 2N+1$, where $\mathbf{r}_i$ are the rows of B i.e. the vectors of one basis of $\mathrm{Leftker}(\mathrm{Im}(Y))$. Using the bilinearity of the scalar product we can rewrite the projection as $\langle \mathrm{M}_{2,0} \,|\, \mathbf{r}_i \rangle = \langle \mathrm{M}_{2,0}^{\mathrm{even}} \,|\, \mathbf{r}_i \rangle$, which is in fact $B\,\mathrm{M}_{2,0} = B\,\mathrm{M}_{2,0}^{\mathrm{even}}$; finally, forcing the V.E.R. to be verified by the desired mask $\mathbf{w}_1^T\mathrm{M}_{2,0} = 0$ and $\mathbf{w}_2^T\mathrm{M}_{2,0} = 0$, we are able to condense the previous relations in the following linear system

$$\widetilde{B}\,\mathrm{M}_{2,0} = \widetilde{\mathbf{f}}\,,$$

where

$$\widetilde{B} = \begin{pmatrix} B \\ \mathbf{w}_1^T \\ \mathbf{w}_2^T \end{pmatrix} \quad \text{and} \quad \widetilde{\mathbf{f}} = \begin{pmatrix} B\,\mathrm{M}_{2,0}^{\mathrm{even}} \\ 0 \\ 0 \end{pmatrix}. \tag{22}$$

Reordering the rows of $\widetilde{B}$ we are able to achieve a diagonal block structure with blocks $(2, \bar{B}, \bar{B})$ which brings to the following

Proposition 3. *Let $\bar{B} = \begin{pmatrix} \widehat{B} \\ \widehat{\mathbf{w}}^T \end{pmatrix}$. The pseudo-inverse matrix of $\bar{B}$ is a block matrix $\bar{B}^+ = [B_1|B_2]$ where*

$$B_1 = \frac{1}{2N}\sum_{j=1}^{N}(-1)^{j-1}(N-2j+1)\widehat{C}^{j-1} \quad \textit{and} \quad B_2 = \frac{1}{N}\widehat{\mathbf{w}}\,, \tag{23}$$

with $B_1 \in \mathbb{R}^{N\times N}$ and $B_2 \in \mathbb{R}^{N\times 1}$.

Proof. By definition $\bar{B}^+ = (\bar{B}^T\bar{B})^{-1}\bar{B}^T$ therefore it suffices to show that $(\bar{B}^T\bar{B})\,[B_1|B_2] = \bar{B}^T$. Due to the block structure of $\bar{B}$ we have $\bar{B}^T\bar{B} = \widehat{B}^T\widehat{B} + \widehat{\mathbf{w}}\widehat{\mathbf{w}}^T = 2\widehat{I} + \widehat{C} + \widehat{C}^{-1} + \widehat{\mathbf{w}}\widehat{\mathbf{w}}^T$. We notice that $\widehat{\mathbf{w}}\widehat{\mathbf{w}}^T = \sum_{j=1}^{N}(-1)^{j-1}j\widehat{C}^{j-1}$. Thus, we have $\widehat{\mathbf{w}}\widehat{\mathbf{w}}^T B_1 = \sum_{j=1}^{N} N - 2j + 1 = 0$ since $\widehat{C}\widehat{\mathbf{w}} = \widehat{C}^{-1}\widehat{\mathbf{w}} = -\widehat{\mathbf{w}}$. This implies that $\left(2\widehat{I} + \widehat{C} + \widehat{C}^{-1}\right)\widehat{\mathbf{w}} = 0$. Let $H = \sum_{j=1}^{N}(-1)^{j-1}j\widehat{C}^{j-1}$. Using (23) we deduce that

$$\begin{aligned}(\bar{B}^T\bar{B})B_1 = \left(2\widehat{I} + \widehat{C} + \widehat{C}^{-1} + \widehat{\mathbf{w}}\widehat{\mathbf{w}}^T\right)B_1 &= \left(2\widehat{I} + \widehat{C} + \widehat{C}^{-1}\right)\left(\frac{1}{2}\widehat{\mathbf{w}}\widehat{\mathbf{w}}^T - \frac{1}{N}H + \frac{1}{2N}\widehat{\mathbf{w}}\widehat{\mathbf{w}}^T\right) \\ &= -\frac{1}{N}\left(2H + \widehat{C}H + \widehat{C}^{-1}H\right).\end{aligned} \tag{24}$$

Using the notation in (12), we have

$$H = \mathrm{Circ}(1, -2, 3, \dots, -N)\,, \quad \widehat{C}H = \mathrm{Circ}(-N, 1, -2, \dots, N-1)\,, \quad \widehat{C}^{-1}H = \mathrm{Circ}(-2, 3, -4, \dots, 1)\,,$$

from which $2H + \widehat{C}H + \widehat{C}^{-1}H = -N\mathrm{Circ}(1,0,0,\ldots,1) = -N\widehat{B}^T$. Substituting this in (24) we arrive at $(\bar{B}^T\bar{B})B_1 = \widehat{B}^T$. Similarly, using the same identity as above,

$$(\bar{B}^T\bar{B})B_2 = \left(2\widehat{I} + \widehat{C} + \widehat{C}^{-1} + \widehat{\mathbf{w}}\widehat{\mathbf{w}}^T\right)\frac{1}{N}\widehat{\mathbf{w}} = \frac{1}{N}\widehat{\mathbf{w}}\widehat{\mathbf{w}}^T\widehat{\mathbf{w}} = \widehat{\mathbf{w}}$$

since $\widehat{\mathbf{w}}^T\widehat{\mathbf{w}} = N$. □

From Proposition 3, we have the following:

Corollary 2. *The mask* $\mathrm{M}_{2,0} = [\delta_\bullet, \boldsymbol{\delta}, \boldsymbol{\delta}']^T$ *ensuring* (21) *is obtained from* $\mathrm{M}_{2,0}^{\mathrm{even}} = [\delta_\bullet^{\mathrm{even}}, \boldsymbol{\delta}^{\mathrm{even}}, \boldsymbol{\delta}'^{\mathrm{even}}]^T$ *as follows:*

$$\begin{aligned} \delta_\bullet &= \delta_\bullet^{\mathrm{even}}\,, \\ \delta_i &= \frac{1}{2N}\sum_{j=1}^{N}(-1)^{j-1}(N-2j+1)\left(\delta_{i+j-1}^{\mathrm{even}} + \delta_{i+j}^{\mathrm{even}}\right)\,, \quad i = 1,\ldots,N\,, \end{aligned} \tag{25}$$

where the indices $i+j-1$ *and* $i+j$ *have to be meant respectively* $\bmod N$ *when* $i+j-1 > N$ *and* $i+j > N$*; a similar relation for* $\boldsymbol{\delta}'$ *is obtained substituting in* (25) *the values* $\boldsymbol{\delta}'^{\mathrm{even}}$.

Proof. Since $\mathrm{M}_{2,0} = \widetilde{B}^+\widetilde{\mathbf{f}}$ the proof is obtained by the explicit formulas (22) and (23), using the block structure of $\widetilde{B}$. □

Hence, using mask $\mathrm{M}_{2,0}$ we are sure to have a solution for (17). To compute $\mathrm{M}_{1,1}$ we add two extra constraints because $\mathrm{corank}(B) = 2$: since we want to obtain a mask $\mathrm{M}_{1,1}$ the closest possible to $\underline{\mathrm{M}}_{1,1}$ we project their difference onto $\mathrm{Im}(Y^T) = \mathrm{Leftker}(B)$. Like in the previous cases, we operate an orthogonal projection i.e. $\langle \mathrm{M}_{1,1} - \underline{\mathrm{M}}_{1,1} \,|\, \mathbf{w}_i\rangle = 0$, $i = 1, 2$, which translates into $\langle \mathrm{M}_{1,1} \,|\, \mathbf{w}_i\rangle = \langle \underline{\mathrm{M}}_{1,1} \,|\, \mathbf{w}_i\rangle$. Joining these two additional relations to system (17) and denoting

$$\widetilde{\mathbf{d}} = \begin{pmatrix} \mathbf{d} \\ \langle \underline{\mathrm{M}}_{1,1} \,|\, \mathbf{w}_1\rangle \\ \langle \underline{\mathrm{M}}_{1,1} \,|\, \mathbf{w}_2\rangle \end{pmatrix}\,, \tag{26}$$

the mask for $\mathrm{M}_{1,1}$ satisfies the relation

$$\widetilde{B}\,\mathrm{M}_{1,1} = \widetilde{\mathbf{d}}\,.$$

Similarly to Corollary 2, we have:

Corollary 3. *The the mask* $\mathrm{M}_{1,1} = [\gamma_\bullet, \boldsymbol{\gamma}, \boldsymbol{\gamma}']^T$ *in the even case is given in terms of* $\underline{\mathrm{M}}_{0,0} = [\underline{\alpha}_\bullet, \underline{\boldsymbol{\alpha}}, \underline{\boldsymbol{\alpha}}']^T$, $\mathrm{M}_{1,0} = [\beta_\bullet, \boldsymbol{\beta}, \boldsymbol{\beta}']^T$ *and* $\mathrm{M}_{2,0} = [\delta_\bullet, \boldsymbol{\delta}, \boldsymbol{\delta}']^T$ *by the following relations:*

$$\begin{aligned} \gamma_\bullet &= \frac{1}{5}(5-2a_0)\underline{\alpha}_\bullet + \frac{2}{5}a_0\delta_\bullet\,, \\ \gamma_i &= \frac{1}{10N}a_0\sum_{j=1}^{N}(-1)^{j-1}(N-2j+1)\underline{\alpha}_{i+j-1} - \frac{1}{2N}(a_0-2)\sum_{j=1}^{N}(-1)^{j-1}(N-2j+1)\beta_{i+j-1} \\ &\quad + \frac{2}{5N}a_0\sum_{j=1}^{N}(-1)^{j-1}(N-2j+1)\delta_{i+j-1} + \frac{1}{N}\widehat{\mathrm{w}}_i\cdot\left\langle \underline{\boldsymbol{\gamma}} \,|\, \widehat{\mathbf{w}}\right\rangle\,, \quad i = 1,\ldots,N\,, \end{aligned} \tag{27}$$

where the indices $i+j-1$ *have to be meant* $\bmod N$ *when* $i+j-1 > N$*; a similar relation for* $\boldsymbol{\gamma}'$ *is obtained substituting in* (27) *the values* $\underline{\boldsymbol{\alpha}}'$, $\boldsymbol{\beta}'$, $\boldsymbol{\delta}'$.

Proof. Because $\mathrm{M}_{1,1} = \widetilde{B}^+\widetilde{\mathbf{d}}$ and the block shape of $\widetilde{B}$, using eq. (23) and (26) we deduce relation (27). □

Example 2. For an EV with valence $N = 6$, and using the formulas in Corollary 2, the mask $\mathrm{M}_{2,0}$ satisfying (21) is given by

$$\mathrm{M}_{2,0} = \frac{1}{1320}[592, 294, 130, -8, 18, -8, 130, 79, 8, -1, -1, 8, 79]^T . \tag{28}$$

Thus, using eq. (28) in Corollary 3, the cross derivative mask is

$$\mathrm{M}_{1,1} = \frac{1}{9900}[5016, 1512, 1512, 258, 144, 144, 258, 604, 185, 43, -4, 43, 185]^T .$$

5.2.2. Deriving $\mathrm{M}_{2,0}$ by assigning $\mathrm{M}_{1,1}$: direct approach based on ACC_5

In this case, substituting the known masks in (6), we directly obtain

$$\mathrm{M}_{2,0} = \frac{1}{4a_0}\left(-a_0\underline{\mathrm{M}}_{0,0} + 5(a_0-2)\mathrm{M}_{1,0} + 5(\underline{\mathrm{M}}_{1,1} + \underline{\mathrm{M}}^{(0)}_{1,1})\right) , \tag{29}$$

that is well defined since $a_0 \neq 0$ for $N \neq 4$, which is the case for the EVs we are analyzing. In Fig. 4-(c) is figured the location of the treated control points.

5.3. Third and fourth order Bézier masks

First of all, depending on the previous solving strategies, from eq. (11) we define the new mask

$$\mathrm{M}_{3,0} = \frac{1}{10}\left(\underline{\mathrm{M}}_{0,0} - 5\mathrm{M}_{1,0} + 10\mathrm{M}_{2,0} + 5\underline{\mathrm{M}}_{4,0} - \underline{\mathrm{M}}_{5,0}\right) .$$

When we modify the control point related to the mask $\mathrm{M}_{3,0}$ the regularity relation for C^2 smoothness in Bézier surfaces across a vertex does not hold anymore. The resulting surface will be C^1 smooth around regular vertices linked to EVs instead of the initial C^2 regularity. From eq. (7), it follows that

$$\mathrm{M}_{2,1} + \mathrm{M}^{(0)}_{1,2} = \mathbf{r} , \quad \text{where} \quad \mathbf{r} = \frac{1}{10}\left(-a_0\underline{\mathrm{M}}_{0,0} + 5a_0\mathrm{M}_{1,0} - 10(a_0-2)\mathrm{M}_{2,0} + 6a_0\mathrm{M}_{3,0}\right) , \tag{30}$$

where Fig. 4-(d) presents the location of the involved Bézier points. By using the reparametrization $\mathrm{M}^{(0)}_{1,2} - \underline{\mathrm{M}}^{(0)}_{1,2} = \theta\left(\mathrm{M}_{2,1} - \underline{\mathrm{M}}_{2,1}\right)$ we obtain the one parameter family of solutions of eq. (30)

$$\begin{cases} \mathrm{M}_{2,1} &= \dfrac{1}{\theta+1}\left(\mathbf{r} + \theta\underline{\mathrm{M}}_{2,1} - \underline{\mathrm{M}}^{(0)}_{1,2}\right) , \\ \mathrm{M}^{(0)}_{1,2} &= \mathbf{r}\dfrac{\theta}{\theta+1} - \dfrac{1}{\theta+1}\left(\theta\underline{\mathrm{M}}_{2,1} - \underline{\mathrm{M}}^{(0)}_{1,2}\right) , \end{cases} \tag{31}$$

where $\theta \in \mathbb{R} \setminus \{-1\}$ has to be fixed. If we take $\theta = 1$, eq. (31) will return a symmetric solution; indeed, this choice reflect the fact that we are obtaining the solution by orthogonal projection of the ACC_5 masks $(\underline{\mathrm{M}}_{2,1}, \underline{\mathrm{M}}^{(0)}_{1,2})$ onto the G^1 solution space. For all the other values of $\theta \setminus \{-1, 1\}$ we will obtain a non symmetric solution for (31); in particular, for $\theta = 0$, we obtain the asymmetric solution by projecting along the direction $\underline{\mathrm{M}}^{(0)}_{1,2}$. Moreover, with $\theta = \pm\infty$ we mean the projection along the direction of $\underline{\mathrm{M}}_{2,1}$; as expected from eq. (31), we don't have any solution for the value $\theta = -1$ because it translates into a projection along a direction parallel to the G^1 solution space. For the fourth order Bézier masks, from eq. (8) we have

$$\mathrm{M}_{3,1} + \mathrm{M}^{(0)}_{1,3} = \widetilde{\mathbf{r}} , \quad \text{where} \quad \widetilde{\mathbf{r}} = \frac{1}{10}\left(a_0\underline{\mathrm{M}}_{0,0} - 5a_0\mathrm{M}_{1,0} + 10a_0\mathrm{M}_{2,0} - 10(a_0-2)\mathrm{M}_{3,0} + 4a_0\underline{\mathrm{M}}_{4,0}\right) , \tag{32}$$

which can be solved as above, replacing $\mathbf{r}$ in (31) with $\widetilde{\mathbf{r}}$ in (32); the positioning of the considered control points is shown in Fig. 4-(e).

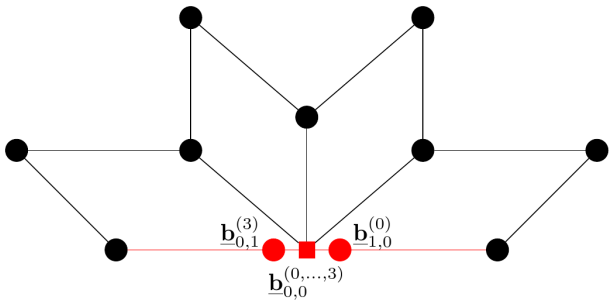


Figure 5: Example of a boundary EV with four incident faces. Red square: Bézier point for the border EV $\underline{\mathbf{b}}^{(0,\dots,3)}_{1,0}$. Red circles: initial ($\underline{\mathbf{b}}^{(0)}_{1,0}$) and final ($\underline{\mathbf{b}}^{(3)}_{0,1}$) Bézier points for the G^1 relations around a boundary EV.

5.4. Treatment of boundaries

Meshes are not always closed and, therefore, they can presents boundaries in which EVs can eventually lay. Bézier control points returning G^1 smoothness around border situations can be obtained by applying equations (5)-(11) starting from the boundary masks presented in [22], after degree elevation. Since these masks ensure the interpolation of boundary edges, we are not going to modify them; we will recover the G^1 regularity by imposing G^1 relations only in the inner regions and using the ACC_5 boundary masks as initial and final value (see Fig. 5). Let κ be the numbers of patches sharing the boundary vertex: for the Bézier masks of order one, imposing eq. (5) $(\kappa-1)$ times across the inner extraordinary edges we obtain the following linear system:

$$(-a_0 I + C + C^{N-1})\,\mathbf{M}_{1,0} = \mathbf{h}\,, \quad \text{where} \quad \mathbf{h} = \begin{pmatrix} (2-a_0)\underline{\mathrm{F}}^{(0)}_{0,0} - \underline{\mathrm{F}}^{(0)}_{1,0} \\ (2-a_0)\underline{\mathrm{F}}^{(1)}_{0,0} \\ \vdots \\ (2-a_0)\underline{\mathrm{F}}^{(\kappa-2)}_{0,0} \\ (2-a_0)\underline{\mathrm{F}}^{(\kappa-1)}_{0,0} - \underline{\mathrm{F}}^{(\kappa-1)}_{0,1} \end{pmatrix}, \tag{33}$$

$\mathbf{M}_{1,0}=\left(\mathrm{M}^{(1)}_{1,0},\dots,\mathrm{M}^{(\kappa-1)}_{1,0}\right)$, $\underline{\mathrm{F}}^{(0,\dots,\kappa-1)}_{0,0}$, $\underline{\mathrm{F}}^{(0)}_{1,0}$ and $\underline{\mathrm{F}}^{(\kappa-1)}_{0,1}$ are, respectively, the ACC_5 masks for the boundary (or frontier) control points $\underline{\mathbf{b}}^{(0,\dots,\kappa-1)}_{0,0}$ $\underline{\mathbf{b}}^{(0)}_{1,0}$ and $\underline{\mathbf{b}}^{(\kappa-1)}_{0,1}$, and $-a_0 I + C + C^{N-1} = A \in \mathbb{R}^{(\kappa-1)\times(\kappa-1)}$ as in eq. (13). System (33) can be solved with a similar strategy like in Section 5.1. Regarding the second order masks, even in this case, from eq. (11) and as in eq. (18), we have

$$\mathrm{M}^{(i)}_{2,0} = \frac{1}{10}\left(-\underline{\mathrm{F}}^{(i)}_{0,0} + 5\mathrm{M}^{(i)}_{1,0} + 10\underline{\mathrm{M}}^{(i)}_{3,0} - 5\underline{\mathrm{M}}^{(i)}_{4,0} + \underline{\mathrm{M}}^{(i)}_{5,0}\right), \quad i = 1,\dots,\kappa-1\,.$$

Now, for the cross derivatives, imposing eq. (6) $(\kappa-1)$ times we get the system:

$$\begin{pmatrix} 1 & 1 & 0 & \dots & 0 \\ & 1 & 1 & & \\ & & & \ddots & \\ 0 & \dots & 0 & 1 & 1 \end{pmatrix} \mathbf{M}_{1,1} = \mathbf{g}\,, \tag{34}$$

where

$$\mathbf{g} = \frac{1}{5}\left(a_0\underline{\mathrm{F}}^{(i)}_{0,0} + 5(2-a_0)\mathrm{M}^{(i)}_{1,0} + 4a_0\mathrm{M}^{(i)}_{2,0}\right), \quad i = 1,\dots,\kappa-1\,,$$

and $\mathbf{M}_{1,1}=\left(\mathrm{M}^{(0)}_{1,1},\dots,\mathrm{M}^{(\kappa-1)}_{1,1}\right)$. Let $R \in \mathbb{R}^{(\kappa-1)\times\kappa}$ be the rectangular matrix in (34); since $\mathrm{corank}(R) = 1$ we have a degree of freedom for the solution. If we want the masks to be close to ACC_5, we can fix this free degree, for example, projecting orthogonally the solution of (34) onto the ACC_5 space; this translates into the solution

$$\mathbf{M}_{1,1} = \underline{\mathbf{M}}_{1,1} + R^T\boldsymbol{\rho}\,,$$

where $\boldsymbol{\rho}$ is the vector containing the coefficients for the orthogonal projection given by

$$R\,R^T\boldsymbol{\rho} = -R\,\underline{\mathbf{M}}_{1,1} + \mathbf{g}\,.$$

The third and fourth order Bézier points are obtained in the same way as the case for an inner EV using eq. (30) and eq. (32) along the inner extraordinary edges.

6. Analysis of the solutions and numerical results

From the constructions presented in Section 5 it turns out that we can solve system (5)-(11) in more than one way, obtaining a family of G^1 schemes; we are now going to analyze this family and identify the procedure returning the most regular surface. First, we will describe the results obtained using a set of test meshes [24] (cf. Fig 6) which contains shapes with EVs of different valence and particular geometric structure; from the results obtained by this procedure we will find out the solving strategy, i.e. the element of the family, returning the smoothest surface. Then, we will check the strength of the previous-found best strategy by applying it to the more complex meshes in Fig. 12 and showing numerical results for the G^1 continuity.

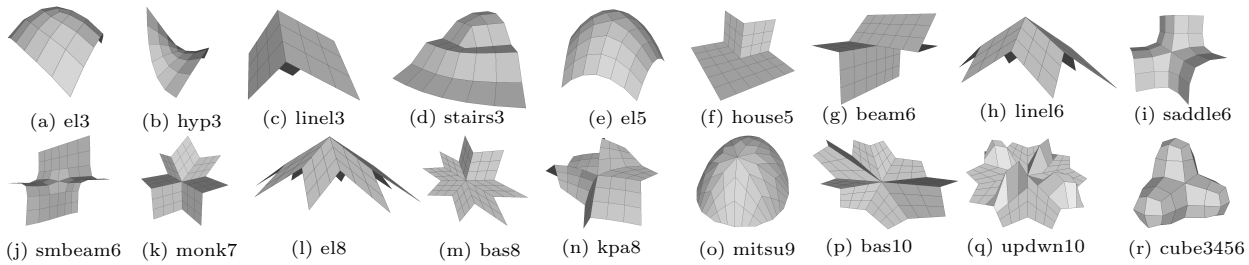


Figure 6: Test meshes in [24]: the numbers in the names refers to the EVs valences of the meshes.

6.1. Comparing the different schemes

Let us call CS (circulant system) the solving strategy in Section 5.2.1, NCS (no circulant system) the strategy in 5.2.2 and S (symmetric) and AS (asymmetric) the solutions for the higher derivatives obtained in 5.3 respectively with $\theta = 1$ and $\theta = 0$; then, we can define four global solving strategies as: NCS-S, NCS-AS, CS-AS and CS-S. To identify the smoothest surface we carried out an isophote and curvature analysis together with numerical evaluation of the G^2 errors on the surfaces obtained by each strategy: in Fig. 7-8 the results for two meshes with EV of valence $N = 5$ and $N = 8$ are shown. In strategy NCS-S we notice that for both valences, isophotes are deviated away from the EV creating an unwanted white spot around it, as also underlined by the curvatures. For the asymmetric strategies NCS-AS and CS-AS we easily realize that the asymmetry of the Bézier points translates into a rotation of the patches around the EVs, which implies irregularities in the isophotes and unpleasant curvatures. Instead, in strategy CS-S, we see that both isophotes and curvatures related to the two valences are smooth and regular in a neighborhood of the EVs. To evaluate numerically the results we use the quantities

$$[\![\widehat{\mathbf{n}}]\!] := \sqrt{\int_{\mathcal{E}} \|\widehat{\mathbf{n}}^L - \widehat{\mathbf{n}}^R\|_2^2}\,, \quad (\!|c_G|\!) := \max_{e\in\mathcal{E}} \left(\max_{u\in e} |c_G^L - c_G^R| \right)\,, \quad (\!|c_M|\!) := \max_{e\in\mathcal{E}} \left(\max_{u\in e} |c_M^L - c_M^R| \right)\,,$$

where $\int_{\mathcal{E}} := \sum_{e\in\mathcal{E}} \int_e$ with $\mathcal{E}$ the set of all edges e of the mesh $\mathcal{M}$ and $\widehat{\mathbf{n}}^L$, c_G^L, c_M^L, $\widehat{\mathbf{n}}^R$, c_G^R, c_M^R are respectively the unit normal vector, the Gauss curvature and the mean curvature of, respectively, the left and right patch sharing the common edge e. The G^1 quality is evaluated by $[\![\widehat{\mathbf{n}}]\!]$, which is a measure of the distance of our surface from the set of purely C^1 parametric surfaces. Indeed, this measure is zero on any C^1 interface, while it expresses the discrepancy or "jump" of the (non-unit) normal vector when there is a discontinuity across an interface. Moreover, the quantities $(\!|c_G|\!)$ and $(\!|c_M|\!)$ return, respectively, the maximum difference between the Gauss curvatures and mean curvatures on the patch interfaces: Table. 1 shows the values for these quantities of each solving strategies carried out on the test meshes [24] in Fig. 6. At this point, using both the numerical and visual approach, we can state that in any case the asymmetric approach AS is never

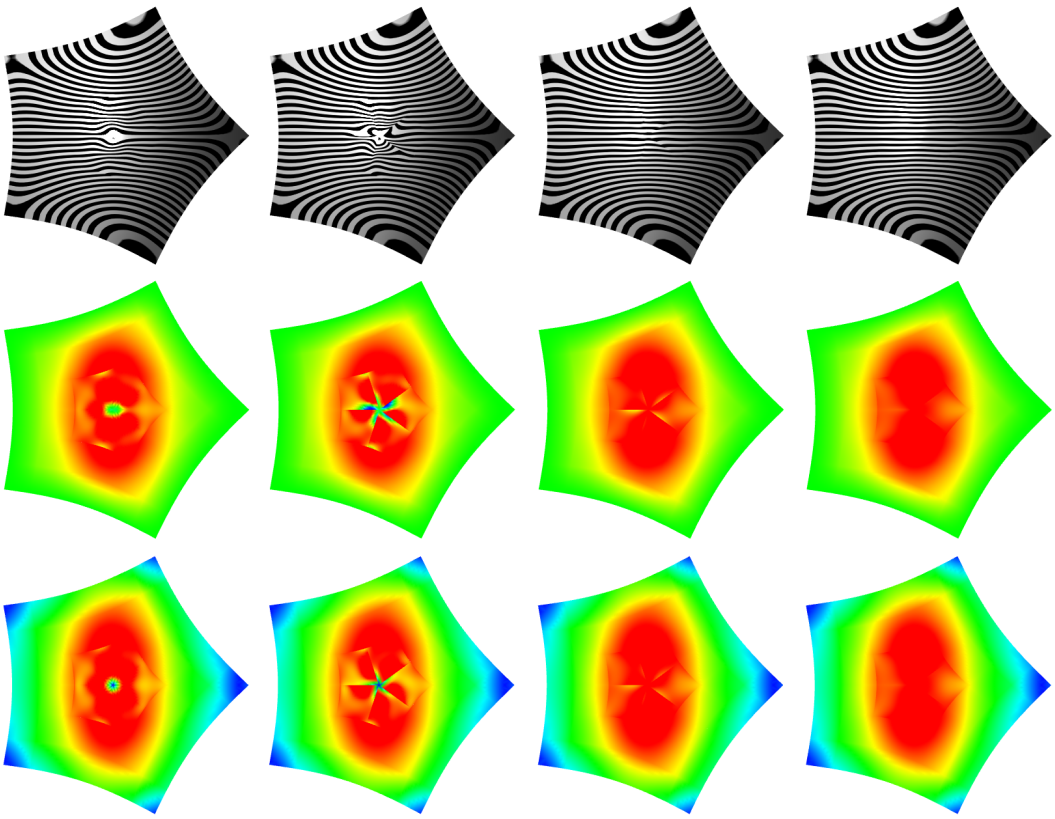


Figure 7: Mesh in Fig. 6-(e). From left to right: solving strategies NCS-S, NCS-AS, CS-AS, CS-S. Upper row: isophotes. Middle row: Gauss curvature. Lower row: mean curvature.

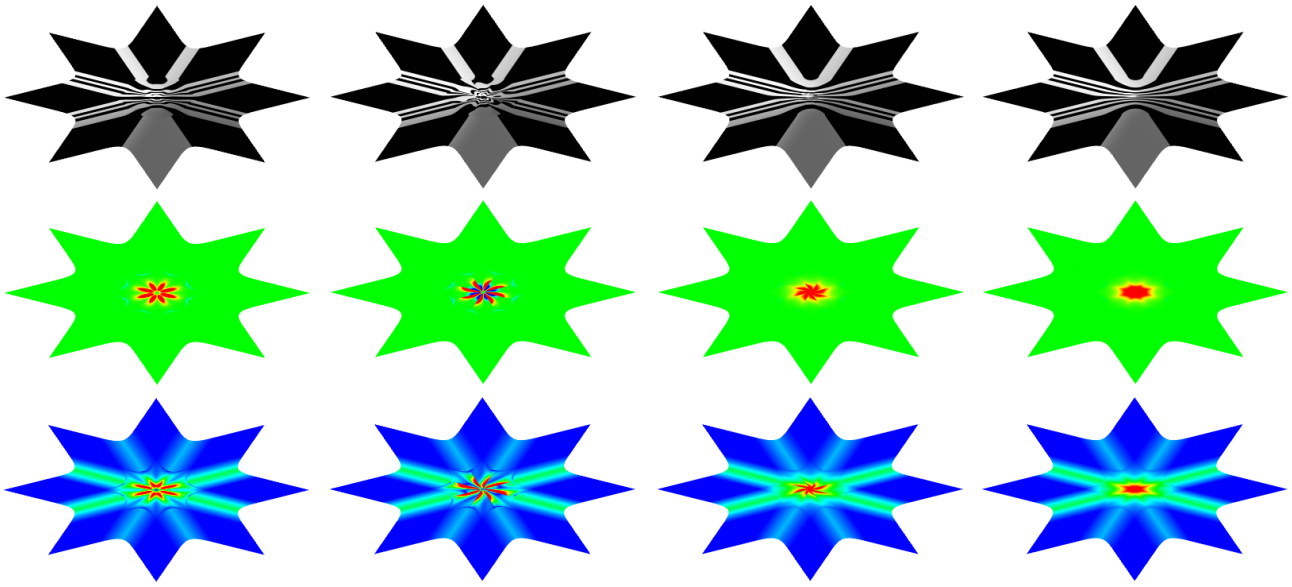


Figure 8: Mesh in Fig. 6-(l). From left to right: solving strategies NCS-S, NCS-AS, CS-AS, CS-S. Upper row: isophotes. Middle row: Gauss curvature. Lower row: mean curvature.

returning very regular surfaces; on the other hand we see that in most cases the best approach is given by the strategy CS-S. From the numerical results presented in Table. 1, we notice that for meshes with very quick and sudden change of slope of patches around EVs, strategy NCS-S returns slightly better surfaces in terms of curvature compared to CS-S.

		el3	**hyp3**	**linel3**	**stairs3**	**el5**	**house5**	**beam6**	**linel6**	**saddle6**
NCS-S	$[\![\widehat{\mathbf{n}}]\!]$	4.4e-15	3.4e-15	8.5e-15	3.3e-15	9.9e-15	7.0e-16	2.2e-15	1.5e-14	3.9e-15
	$(\!(c_G)\!)$	1.9e-00	3.1e-00	3.4e-00	2.7e-00	1.2e-00	11.3e-00	84.7e-00	8.1e-00	1.3e-00
	$(\!(c_M)\!)$	0.9e-00	2.5e-00	0.3e-00	3.5e-00	0.7e-00	1.4e-00	2.4e-00	2.8e-00	0.5e-00
NCS-AS	$[\![\widehat{\mathbf{n}}]\!]$	4.4e-15	3.4e-15	8.9e-15	3.3e-15	9.9e-15	7.5e-16	2.2e-15	1.5e-14	3.9e-15
	$(\!(c_G)\!)$	13.6e-00	3.4e-00	132.8e-00	11.0e-00	7.9e-00	16.5e-00	92.1e-00	75.2e-00	1.6e-00
	$(\!(c_M)\!)$	4.2e-00	2.7e-00	23.7e-00	7.0e-00	4.1e-00	4.9e-00	7.3e-00	23.7e-00	1.4e-00
CS-AS	$[\![\widehat{\mathbf{n}}]\!]$	4.4e-15	3.4e-15	7.0e-15	3.3e-15	9.9e-15	1.0e-15	2.2e-15	1.5e-14	4.0e-15
	$(\!(c_G)\!)$	5.0e-00	1.3e-00	33.0e-00	4.0e-00	1.3e-00	3.7e-00	43.3e-00	16.0e-00	4.1e-00
	$(\!(c_M)\!)$	2.0e-00	1.4e-00	8.2e-00	4.2e-00	0.9e-00	1.5e-00	16.7e-00	5.7e-00	3.5e-00
CS-S	$[\![\widehat{\mathbf{n}}]\!]$	4.4e-15	3.4e-15	6.9e-15	3.3e-15	9.9e-15	9.5e-16	2.6e-15	1.5e-14	4.4e-15
	$(\!(c_G)\!)$	0.8e-00	1.3e-00	6.8e-00	3.9e-00	0.1e-00	3.4e-00	17.4e-00	0.9e-00	1.5e-00
	$(\!(c_M)\!)$	0.4e-00	1.4e-00	0.6e-00	4.0e-00	0.1e-00	0.8e-00	0.9e-00	0.3e-00	0.6e-00

		smbeam6	**monk7**	**el8**	**bas8**	**kpa8**	**mitsu9**	**bas10**	**updwn10**	**cube3456**
NCS-S	$[\![\widehat{\mathbf{n}}]\!]$	2.9e-15	6.2e-15	3.0e-14	5.6e-15	1.0e-14	1.9e-14	1.8e-14	1.0e-14	3.6e-14
	$(\!(c_G)\!)$	45.6e-00	8.3e-00	7.5e-00	7.7e-00	56.6e-00	1.6e-00	50.0e-00	70.4e-00	0.7e-00
	$(\!(c_M)\!)$	4.3e-00	6.4e-00	2.9e-00	5.5e-00	6.5e-00	1.4e-00	11.8e-00	19.1e-00	0.5e-00
NCS-AS	$[\![\widehat{\mathbf{n}}]\!]$	2.9e-15	6.2e-15	3.0e-14	5.6e-15	1.0e-14	1.9e-14	1.8e-14	1.0e-14	3.2e-14
	$(\!(c_G)\!)$	49.5e-00	38.3e-00	107.5e-00	12.1e-00	63.0e-00	23.4e-00	91.8e-00	106.2e-00	3.3e-00
	$(\!(c_M)\!)$	5.1e-00	10.6e-00	40.8e-00	13.6e-00	17.6e-00	12.8e-00	27.6e-00	29.7e-00	1.9e-00
CS-AS	$[\![\widehat{\mathbf{n}}]\!]$	2.9e-15	6.1e-15	2.6e-14	5.7e-15	1.1e-14	1.9e-14	1.8e-14	1.0e-14	4.0e-14
	$(\!(c_G)\!)$	35.1e-00	92.0e-00	43.6e-00	16.1e-00	27.6e-00	14.6e-00	87.9e-00	25.3e-00	1.3e-00
	$(\!(c_M)\!)$	7.8e-00	39.0e-00	16.6e-00	16.8e-00	10.7e-00	8.3e-00	49.5e-00	87.4e-00	1.9e-00
CS-S	$[\![\widehat{\mathbf{n}}]\!]$	2.9e-15	6.1e-15	2.6e-14	5.7e-15	1.1e-14	1.9e-14	1.8e-14	1.1e-14	4.5e-14
	$(\!(c_G)\!)$	6.0e-00	91.4e-00	0.9e-00	15.1e-00	27.0e-00	1.0e-00	83.3e-00	24.5e-00	0.7e-00
	$(\!(c_M)\!)$	1.2e-00	38.8e-00	0.3e-00	15.8e-00	6.0e-00	1.0e-00	47.5e-00	84.4e-00	0.6e-00

Table 1: G^1 smoothness $[\![\widehat{\mathbf{n}}]\!]$, Gauss curvature difference $(\!(c_G)\!)$ and mean curvature difference $(\!(c_M)\!)$ obtained from strategies NCS-S, NCS-AS, CS-AS, CS-S for every test mesh in Fig. 6.

6.2. Complex meshes

Using the best strategy CS-S obtained in Section 6.1, we developed a G^1 continuity analysis on the complex meshes in Fig. 12; these meshes present a very complicate geometry together with a very high number of faces, vertices and, obviously, extraordinary vertices of several valence. In Table 2 we present a detailed description of each mesh: we show the number of EVs of any valence, the total number of EVs, the number of vertices and faces of the mesh and the numerical evaluation of the G^1 regularity. From the results in Table 6 and Table 12, we can affirm that this construction returns really smooth surfaces, with a G^1 error in the range of 10^{-11}-10^{-15}, i.e. numerically identical to zero.

	N=**3,**	**5,**	**6,**	**7,**	**8,**	**9,**	**10,**	**11,**	**12,**	**16,**	**45**	**Total EVs**	**Vertices**	**Faces**	$[\![\widehat{\mathbf{n}}]\!]$
alien	858,	82,	156,	102,	30,	6,	0,	0,	0,	0,	0	1234	5126	5124	9.0e-12
bird	1135,	147,	179,	84,	45,	14,	7,	10,	1,	0,	0	1622	6782	6780	2.7e-11
dinosaur	501,	57,	134,	34,	9,	6,	0,	0,	0,	0,	0	741	3002	3000	7.2e-12
disk	270,	90,	45,	0,	0,	0,	0,	0,	0,	0,	2	407	2214	1620	7.8e-12
dolphin	840,	89,	203,	74,	15,	7,	1,	2,	0,	0,	0	1231	5018	2016	2.7e-11
gear	1007,	147,	160,	64,	37,	9,	8,	9,	4,	1,	0	1446	6000	6000	6.4e-12
hammer	502,	73,	108,	34,	10,	6,	3,	1,	0,	0,	0	737	3004	3000	8.6e-12
hand	768,	36,	302,	27,	7,	1,	10,	0,	0,	0,	0	1142	4610	4608	5.7e-12
head	1383,	184,	241,	157,	38,	10,	4,	0,	0,	1,	0	2018	8270	8268	3.7e-11
rabbit	904,	134,	154,	88,	24,	9,	7,	1,	0,	0,	0	1321	5414	5412	6.4e-12
skull	1252,	151,	255,	124,	39,	7,	1,	2,	1,	0,	0	1832	7474	7476	7.8e-12
venus	502,	65,	113,	48,	11,	3,	0,	0,	0,	0,	0	742	3002	3000	2.8e-12

Table 2: Detailed features of the meshes in Fig. 11-12

6.3. Comparison with ACC_3 surface and Catmull-Clark limit surface

In order to investigate the accuracy of our G^1 approximation of the Catmull-Clark limit surface we analyse on the distance between the two surfaces; the two quantities we take into account are the geometry error

and the normal error. For each point laying on the Catmull-Clark limit surface, to compute the geometry error we look for their closest points on the G^1 surface: formally, if P_{CC} is a point on the Catmull-Clark surface, the correspondent geometry error geom_{err} is obtaining by minimizing the distance

$$\text{dist}(P_{CC}, \mathcal{G}^1) = \min_{P \in \mathcal{G}^1} \|P_{CC} - P\|_2 \ ,$$

where we define with $\mathcal{G}^1$ to be our CS-S surface. To evaluate the normal error, we first compute the orthogonal projection $P_\perp$ of each point P_{CC} onto $\mathcal{G}^1$. Then the normal error norm_{err} is obtained evaluating the 2-norm of the corresponding unit normals attached to them

$$\text{norm}_{err} = \|\widehat{\mathbf{n}}(P_{CC}) - \widehat{\mathbf{n}}(P_\perp)\|_2 \ .$$

The same study is conducted for the ACC_3 surface in order to provide a comparison with our improved construction and the obtained result are shown in Fig. 9. We also estimate the convergence rates of geom_{err} and norm_{err} for both the surfaces ACC_3 and $\mathcal{G}^1$. Starting from different Catmull-Clark subdivision levels of the same input mesh, we construct on each of them the two surfaces and we evaluate the maximum of the two errors over a dense sampling of the surface. Interestingly, geom_{err} converges quadratically while norm_{err} converges with a rate of 0.7 (we also observed a convergence rate of 1.5 for the non-normalized normal vector). On the other side, in the ACC_3 construction we observe that geom_{err} converges with a rate of 1.5 while norm_{err} presents an oscillatory non-convergent behavior. Fig. 10 summarizes these results.

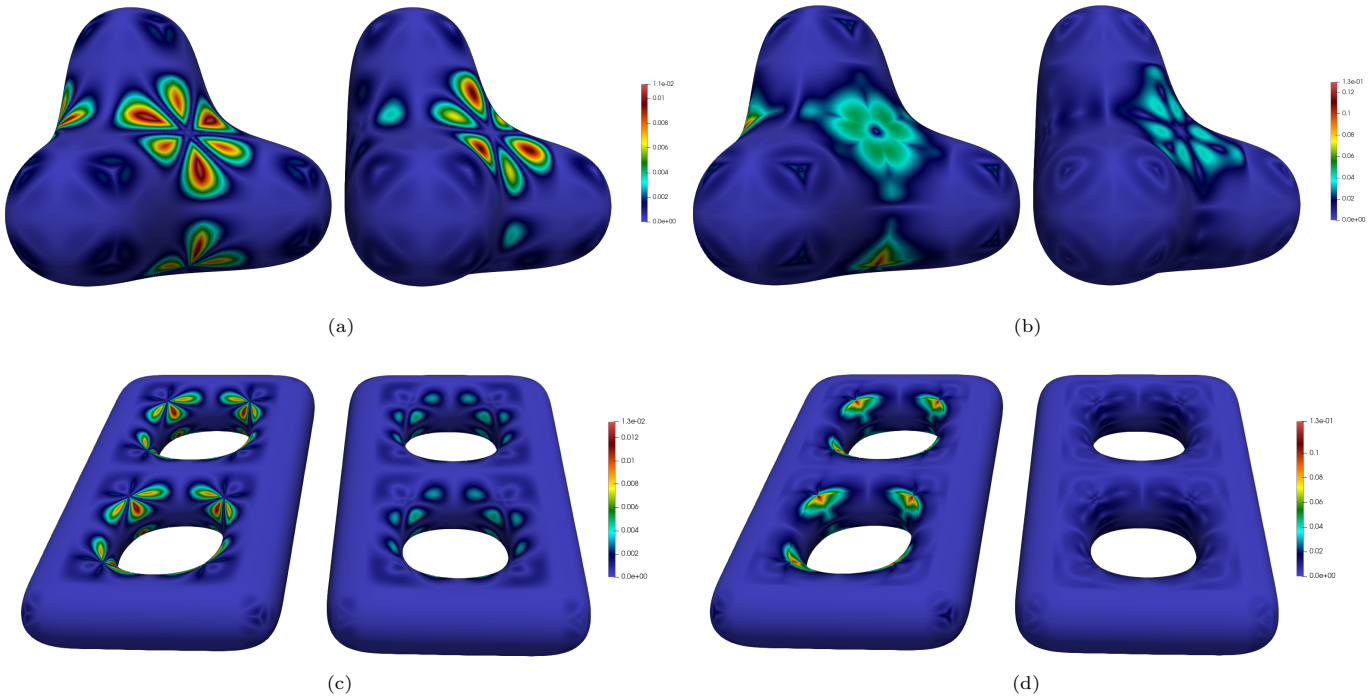


Figure 9: Catmull-Clark limit surfaces with error colormap representing the geometry error (a)-(c) and normal error (b)-(d) between the ACC_3 surface (left meshes) and our G^1 surface (right meshes) using the data in [22, p. 8].

Remark. From the numerical results we also noticed that this construction returns unit normals at the vertices which are close to the real normal of the Catmull-Clark surface. Let $\mathrm{T}_{\mathbf{u}}, \mathrm{T}_{\mathbf{v}} = C\,\mathrm{T}_{\mathbf{u}}$ be the masks returning the tangents of the limit Catmull-Clark surface with weights described in [22, pag. 3,6] and $\partial_{\mathbf{u}} = \mathrm{M}_{1,0} - \underline{\mathrm{M}}_{0,0}, \partial_{\mathbf{v}} = \mathrm{M}_{0,1} - \underline{\mathrm{M}}_{0,0} = C\,\partial_{\mathbf{u}}$ the masks returning the derivative directions of our surface obtained from (15). Denoting D the matrix with columns $\mathrm{T}_{\mathbf{u}}, \mathrm{T}_{\mathbf{v}}, \partial_{\mathbf{u}}, \partial_{\mathbf{v}}$, we observe that two trailing singular values $\sigma_i(D)$ of D are small, which means the four vector are almost coplanar i.e. the tangent planes generated by $\text{Span}\{\mathrm{T}_{\mathbf{u}}, \mathrm{T}_{\mathbf{v}}\}$ and $\text{Span}\{\partial_{\mathbf{u}}, \partial_{\mathbf{v}}\}$ are approaching each other, also verified by the converge of the unit normal (cf. Fig. 10).

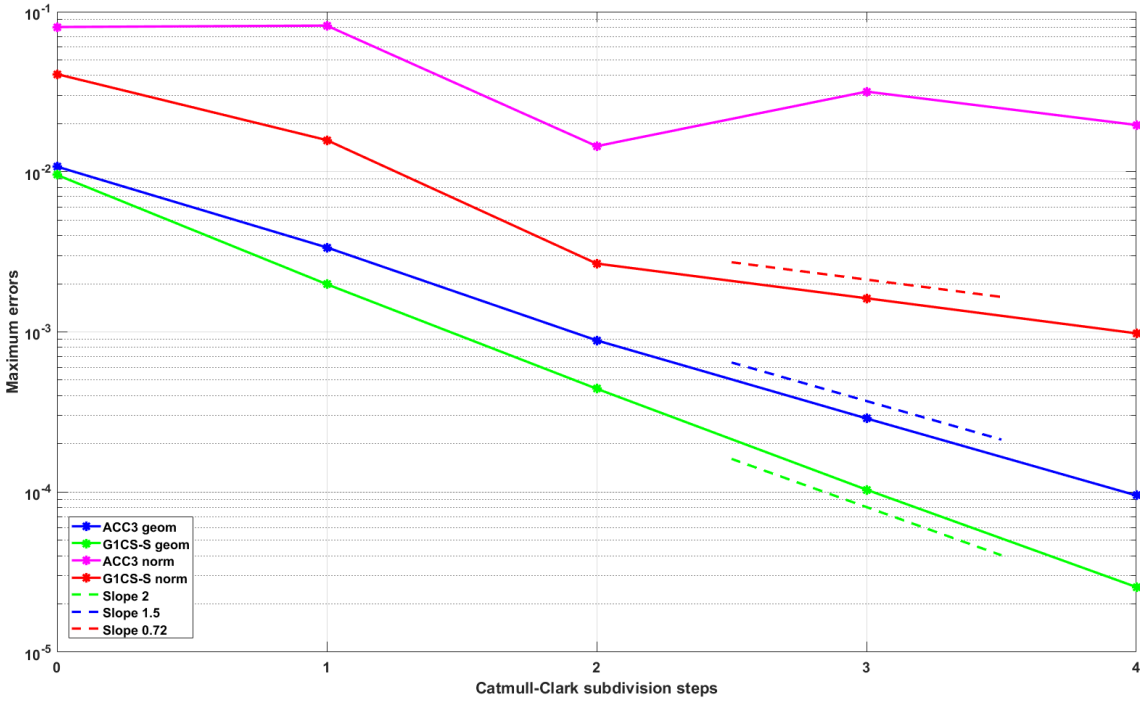
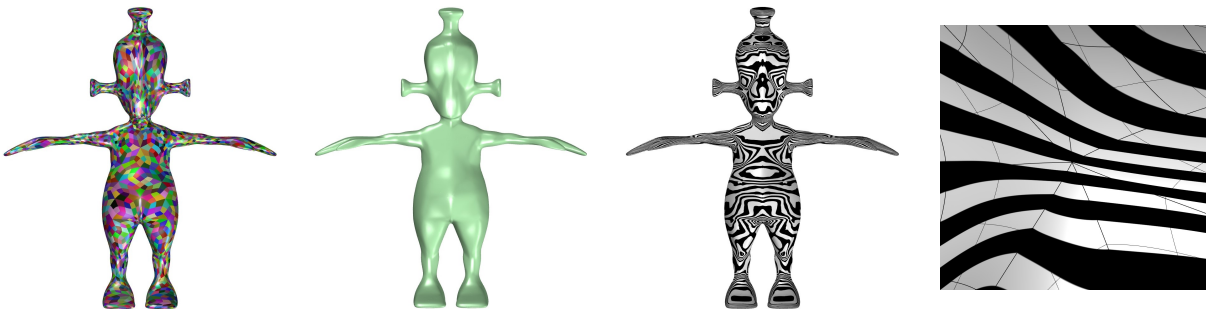
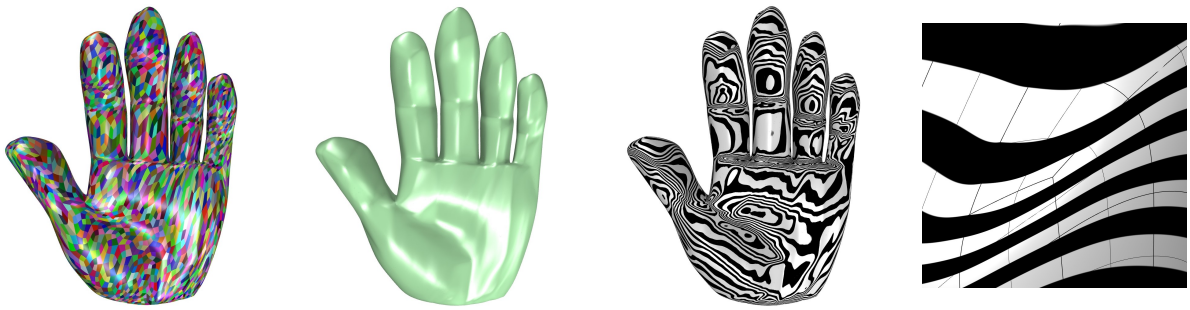


Figure 10: Experimental convergence rates of the geometry and normal errora for ACC_3 and G^1CS-S under successive Catmull-Clark subdivisions.

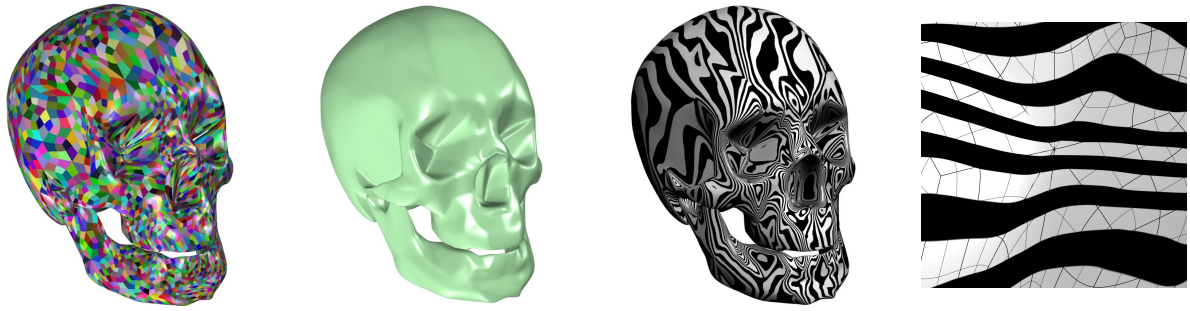


(a) alien



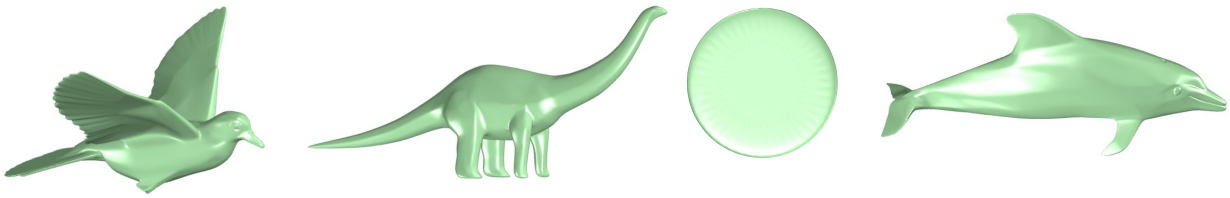
(b) hand

(c) rabbit



(d) skull

Figure 11: From left to right: multipatch color, solid color, isophotes and zoom on EVs of valences, respectively, 7, 3, 6 and 8, of some surfaces obtained by complex meshes using the solving strategy CS-S.



(a) bird (b) dinosaur (c) disk (d) dolphin

(e) gear (f) hammer (g) head (h) venus

Figure 12: Surfaces obtained by complex meshes using the solving strategy CS-S.

7. Conclusion

In this work we presented an explicit construction of a G^1 smoothing scheme providing smooth globally biquintic surfaces. Starting from the work of Loop and Schaefer [22] and making use of Bézier degree elevation algorithm we impose G^1 constraints in order to obtain a set of equations defining a family of surfaces with the desired smoothness; the solutions of these equations are analyzed case by case with respect to the order of Bézier masks and presented in such a way as to have explicit smoothing masks for any valence of the EVs. Moreover, we exhibit a numerical analysis of the new surfaces obtained with this construction to be able to classify their regularity: all the results presented throughout the paper have been obtained by implementing codes using Julia language for the surface reconstruction and G+Smo library [23] for obtaining the numerical results. It has also been noticed that the whole construction is computationally efficient, even in presence of complex geometries. Two interesting future research directions are envisaged. First, we would like to advance our constructions by providing explicit basis functions and use them in the isogeometric setting, eg. for analysis of shell structures, cf. [28]. Second, in the spirit of G^1–smooth volumetric splines (cf. [4, 5, 33]), we would like to extend the construction to subdivision solids.

8. Acknowledgments

This paper has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 860843.

References

[1] Barendrecht, P., 2019. Splines for engineers: with selected applications in numerical methods and computer graphics. Thesis fully internal (DIV). Rijksuniversiteit Groningen. [Groningen]. doi:10.33612/diss.102688532. iSBN: 9789403421667.
[2] Barendrecht, P.J., Bartoň, M., Kosinka, J., 2018. Efficient quadrature rules for subdivision surfaces in isogeometric analysis. Computer Methods in Applied Mechanics and Engineering 340, 1–23. doi:https://doi.org/10.1016/j.cma.2018.05.017.
[3] Bercovier, M., Matskewich, T., 2017. Smooth Bezier Surfaces over Arbitrary Quadrilateral Meshes. arXiv:1412.1125 [math] 22. doi:10.1007/978-3-319-63841-6. arXiv: 1412.1125.
[4] Birner, K., Juettler, B., Mantzaflaris, A., 2018. Bases and dimensions of C1-smooth isogeometric splines on volumetric two-patch domains. Graphical Models 99, 46–56. doi:https://doi.org/10.1016/j.gmod.2018.08.001.
[5] Birner, K., Kapl, M., 2019. The space of C1-smooth isogeometric spline functions on trilinearly parameterized volumetric two-patch domains. Computer Aided Geometric Design 70, 16–30. doi:https://doi.org/10.1016/j.cagd.2019.03.002.
[6] Blidia, A., Mourrain, B., Villamizar, N., 2017. G1-smooth splines on quad meshes with 4-split macro-patch elements. Computer Aided Geometric Design 52-53, 106–125. doi:10.1016/j.cagd.2017.03.003.
[7] Bonneau, G.P., Hahmann, S., 2014. Flexible G1 interpolation of quad meshes. Graphical Models 76, 669–681. doi:10.1016/j.gmod.2014.09.001.
[8] Bracco, C., Giannelli, C., Kapl, M., Vázquez, R., 2020. Isogeometric analysis with C1 hierarchical functions on planar two-patch geometries. Computers & Mathematics with Applications 80, 2538–2562. doi:https://doi.org/10.1016/j.camwa.2020.03.018. high-Order Finite Element and Isogeometric Methods 2019.
[9] Buchegger, F., Juettler, B., Mantzaflaris, A., 2016. Adaptively refined multi-patch b-splines with enhanced smoothness. Applied Mathematics and Computation 272, Part 1, 159 – 172. doi:http://dx.doi.org/10.1016/j.amc.2015.06.055.
[10] Cashman, T.J., 2010. NURBS-compatible subdivision surfaces doi:10.2312/8238.
[11] Catmull, E., Clark, J., 1978. Recursively generated B-spline surfaces on arbitrary topological meshes. Computer-Aided Design 10, 350–355. doi:10.1016/0010-4485(78)90110-0.
[12] Cirak, F., Ortiz, M., Schröder, P., 2000. Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. International Journal for Numerical Methods in Engineering 47, 2039–2072. doi:https://doi.org/10.1002/(SICI)1097-0207(20000430)47:12<2039::AID-NME872>3.0.CO;2-1.
[13] Collin, A., Sangalli, G., Takacs, T., 2016. Analysis-suitable G1 multi-patch parametrizations for C1 isogeometric spaces. Computer Aided Geometric Design 47, 93–113. doi:https://doi.org/10.1016/j.cagd.2016.05.009. sI: New Developments Geometry.
[14] Fan, J., Peters, J., 2008. On Smooth Bicubic Surfaces from Quad Meshes, in: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Remagnino, P., Porikli, F., Peters, J., Klosowski, J., Arns, L., Chun, Y.K., Rhyne, T.M., Monroe, L. (Eds.), Advances in Visual Computing, Springer, Berlin, Heidelberg. pp. 87–96. doi:10.1007/978-3-540-89639-5_9.
[15] Hahmann, S., Bonneau, G.P., Caramiaux, B., 2008. Bicubic G1 Interpolation of Irregular Quad Meshes Using a 4-Split, in: Chen, F., Jüttler, B. (Eds.), Advances in Geometric Modeling and Processing, Springer, Berlin, Heidelberg. pp. 17–32. doi:10.1007/978-3-540-79246-8_2.
[16] Hahn, J.M., 1989. Geometric continuous patch complexes. Computer Aided Geometric Design 6, 55–67. doi:10.1016/0167-8396(89)90006-X.
[17] Hughes, T.J., Sangalli, G., Takacs, T., Toshniwal, D., 2021. Smooth multi-patch discretizations in isogeometric analysis, in: Bonito, A., Nochetto, R.H. (Eds.), Geometric Partial Differential Equations - Part II. Elsevier. volume 22 of *Handbook of Numerical Analysis*, pp. 467–543. doi:https://doi.org/10.1016/bs.hna.2020.09.002.
[18] Juettler, B., Mantzaflaris, A., Perl, R., Rumpf, M., 2016. On numerical integration in isogeometric subdivision methods for PDEs on surfaces. Computer Methods in Applied Mechanics and Engineering 302, 131 – 146. doi:https://doi.org/10.1016/j.cma.2016.01.005.
[19] Kapl, M., Sangalli, G., Takacs, T., 2018. Construction of analysis-suitable G1 planar multi-patch parameterizations. Computer-Aided Design 97, 41–55. doi:https://doi.org/10.1016/j.cad.2017.12.002.
[20] Kapl, M., Sangalli, G., Takacs, T., 2019. An isogeometric C1 subspace on unstructured multi-patch planar domains. Computer Aided Geometric Design 69, 55–75. doi:https://doi.org/10.1016/j.cagd.2019.01.002.

[21] Karčiauskas, K., Peters, J., 2017. Improved shape for refinable surfaces with singularly parameterized irregularities. Computer-Aided Design 90, 191–198. URL: https://www.sciencedirect.com/science/article/pii/S0010448517300702, doi:10.1016/j.cad.2017.05.004.
[22] Loop, C., Schaefer, S., 2008. Approximating Catmull-Clark subdivision surfaces with bicubic patches. ACM Transactions on Graphics 27, 8:1–8:11. doi:10.1145/1330511.1330519.
[23] Mantzaflaris, A., 2020. An Overview of Geometry Plus Simulation Modules, in: Mathematical Aspects of Computer and Information Sciences, Springer International Publishing, Cham. pp. 453–456. doi:10.1007/978-3-030-43120-4_35.
[24] Peters, J., . SurfLab Shape Gallery Page.
[25] Peters, J., 1995. Biquartic C1 spline Surfaces over Irregular Meshes. Computer-Aided Design 27, 895–903.
[26] Peters, J., 2000. Patching Catmull-Clark meshes, in: Proceedings of the 27th annual conference on Computer graphics and interactive techniques, ACM Press/Addison-Wesley Publishing Co., USA. pp. 255–258. doi:10.1145/344779.344908.
[27] Peters, J., Fan, J., 2010. On the complexity of smooth spline surfaces from quad meshes. Computer Aided Geometric Design 27, 96–105. doi:10.1016/j.cagd.2009.09.003.
[28] Riffnaller-Schiefer, A., Augsdörfer, U., Fellner, D., 2016. Isogeometric shell analysis with nurbs compatible subdivision surfaces. Appl. Math. Comput. 272, 139–147. doi:10.1016/j.amc.2015.06.113.
[29] Scott, M.A., Simpson, R.N., Evans, J.A., Lipton, S., Bordas, S.P.A., Hughes, T.J.R., Sederberg, T.W., 2013. Isogeometric boundary element analysis using unstructured T-splines. Computer Methods in Applied Mechanics and Engineering 254, 197–221. URL: https://www.sciencedirect.com/science/article/pii/S0045782512003386, doi:10.1016/j.cma.2012.11.001.
[30] Shi, X., Wang, T., Yu, P., 2004. A practical construction of G1 smooth biquintic B-spline surfaces over arbitrary topology. Computer-Aided Design 36, 413–424. doi:10.1016/S0010-4485(03)00111-8.
[31] Toshniwal, D., Speleers, H., Hughes, T.J., 2017. Smooth cubic spline spaces on unstructured quadrilateral meshes with particular emphasis on extraordinary points: Geometric design and isogeometric analysis considerations. Computer Methods in Applied Mechanics and Engineering 327, 411–458. doi:https://doi.org/10.1016/j.cma.2017.06.008. advances in Computational Mechanics and Scientific Computation—the Cutting Edge.
[32] Weinmüller, P., Takacs, T., 2021. Construction of approximate C1 bases for isogeometric analysis on two-patch domains. Computer Methods in Applied Mechanics and Engineering 385, 114017. doi:https://doi.org/10.1016/j.cma.2021.114017.
[33] Xie, J., Xu, J., Dong, Z., Xu, G., Deng, C., Mourrain, B., Zhang, Y.J., 2020. Interpolatory catmull-clark volumetric subdivision over unstructured hexahedral meshes for modeling and simulation applications. Computer Aided Geometric Design 80, 101867. doi:https://doi.org/10.1016/j.cagd.2020.101867.
[34] Zhang, Q., Sabin, M., Cirak, F., 2018. Subdivision surfaces with isogeometric analysis adapted refinement weights. Computer-Aided Design 102, 104–114. doi:https://doi.org/10.1016/j.cad.2018.04.020. proceeding of SPM 2018 Symposium.