Computer Aided Geometric Design 37 (2015) 25–41

Contents lists available at ScienceDirect

Computer Aided Geometric Design

www.elsevier.com/locate/cagd

Watertight conversion of trimmed CAD surfaces to Clough–Tocher splines ☆

Jiří Kosinka [a,*], Thomas J. Cashman [b]

[a] *Computer Laboratory, University of Cambridge, 15 JJ Thomson Avenue, Cambridge CB3 0FD, United Kingdom*
[b] *TranscenData Europe Ltd, 4 Carisbrooke Court, Anderson Road, Buckingway Business Park, Swavesey, Cambridge, CB24 4UQ, United Kingdom*

ARTICLE INFO

Article history:
Received 22 April 2015
Accepted 1 June 2015
Available online 10 June 2015

Keywords:
Clough–Tocher spline
CAD model
Boundary representation
Watertight conversion

ABSTRACT

The boundary representations (B-reps) that are used to represent shape in Computer-Aided Design systems create unavoidable gaps at the face boundaries of a model. Although these inconsistencies can be kept below the scale that is important for visualisation and manufacture, they cause problems for many downstream tasks, making it difficult to use CAD models directly for simulation or advanced geometric analysis, for example. Motivated by this need for watertight models, we address the problem of converting B-rep models to a collection of cubic C^1 Clough–Tocher splines. These splines allow a watertight join between B-rep faces, provide a homogeneous representation of shape, and also support local adaptivity.

We perform a comparative study of the most prominent Clough–Tocher constructions and include some novel variants. Our criteria include visual fairness, invariance to affine reparameterisations, polynomial precision and approximation error. The constructions are tested on both synthetic data and CAD models that have been triangulated. Our results show that no construction is optimal in every scenario, with surface quality depending heavily on the triangulation and parameterisation that are used.

1. Introduction

The common representation of shape available to all Computer-Aided Design systems, and therefore the standard form for exchanging CAD models between systems, is a B-rep (boundary representation); (Stroud, 2006). In this paradigm, a connected wireframe of edges and vertices describes the global topology and sharp features of a model. Faces are bounded by a collection of edges and have their internal geometry provided by a smooth 2-manifold surface: usually one of the natural quadrics or a non-uniform rational B-spline (NURBS) surface (Piegl and Tiller, 1997) for more general freeform shapes. The embedding surface typically extends beyond the bounding edges and the portion that is meaningful is therefore a “trimmed” region of the whole. This representation of shape has been successful in realising the broad range of applications that CAD systems support today, for example in visualisation and manufacturing.

However, when we are interested in the region of space bounded (or excluded) by a shape, for example when performing an engineering simulation (Cottrell et al., 2009; Jaxon and Qian, 2014) or computing a medial axis transform (Bucklow, 2014), our task is often made difficult by the deficiencies of such a B-rep. For example, the intersection curve of two

☆ This paper has been recommended for acceptance by Gerald Farin.
[*] Corresponding author. Tel.: +44 1223 763663.
E-mail address: Jiri.Kosinka@cl.cam.ac.uk (J. Kosinka).

http://dx.doi.org/10.1016/j.cagd.2015.06.001
0167-8396/

NURBS surfaces is not, in general, a NURBS curve, leading to unavoidable gaps in trimmed NURBS models (Skytt and Vuong, 2013). Although the gaps can be made arbitrarily small, the result is still a discontinuous representation of shape. This is a significant barrier for the direct use of CAD models in most types of engineering simulation, which require at least continuity of position. Additionally, it is often desirable in downstream applications (such as slicing for 3D printing) that all surfaces use the same representation, continuity, and degree. This simplifies interrogation considerably, and allows for algorithms that use data-level parallelism (e.g. on a GPU) to improve performance.

There is therefore a need to convert B-reps, and particularly the trimmed surfaces that they comprise, into a form that removes the gaps between adjacent faces and allows for a homogeneous representation of shape. This paper investigates the performance of cubic C^1 Clough–Tocher splines for this purpose. These splines give a way of smoothly interpolating surface samples with gradients and can therefore

- remove gaps between B-rep faces (giving surfaces that are C^1 within a B-rep face or C^0 when stitched together);
- provide a simple, homogeneous representation of freeform shapes;
- allow for local adaptivity.

Cubic C^1 Clough–Tocher splines are described in the next section. Various constructions of this type are summarised in Section 3, including previously unexplored variants. Section 4 addresses the problem of generating input data from trimmed NURBS surfaces that are then sampled and interpolated by a Clough–Tocher spline. Our experiments and a comparison of various constructions are given in Section 5. The paper concludes in Section 6 with a short summary and suggestions for future research.

2. Preliminaries

We start by introducing our notation and some basic concepts including Bézier triangles, the cubic C^1 Clough–Tocher spline space and its reduced variant. More details can be found in the papers by Farin (1986), Lai and Schumaker (2001), Alfeld and Schumaker (2002), and Speleers (2010).

2.1. Bézier triangles

A point P in a triangle $\mathcal{T}(U_0, U_1, U_2)$ with vertices U_0, U_1, U_2 is uniquely given by its barycentric coordinates $\boldsymbol{\tau} = (\tau_0, \tau_1, \tau_2)$ as the convex combination

$$P = \sum_{i=0}^{2} \tau_i U_i; \quad \tau_0 + \tau_1 + \tau_2 = 1,\ \tau_i \geq 0. \tag{1}$$

We assume that $\mathcal{T}$ is non-degenerate. Then the barycentric coordinates can be determined using ratios of signed triangle areas as

$$\tau_i = \frac{\mathcal{A}(U_{i+1}, U_{i+2}, P)}{\mathcal{A}(U_0, U_1, U_2)} \tag{2}$$

with the index i treated modulo 3.

Any polynomial p of total degree at most d defined on $\mathcal{T}$ can be expressed in the Berstein–Bézier form

$$p(\boldsymbol{\tau}) = \sum_{|\mathbf{i}|=d} P_{\mathbf{i}} B^d_{\mathbf{i}}(\boldsymbol{\tau}), \tag{3}$$

where

$$B^d_{\mathbf{i}}(\boldsymbol{\tau}) = \frac{d!}{i!j!k!} \tau_1^i \tau_2^j \tau_3^k \tag{4}$$

with $\mathbf{i} = (i, j, k)$, $|\mathbf{i}| = i + j + k$, and $i, j, k \geq 0$. These polynomials span the linear space denoted by Π_d. In the cubic case, the Bézier ordinates $P_{\mathbf{i}}$, associated with the barycentric coordinates $\mathbf{i}/3$, are shown in Fig. 1, left.

Moving from the above functional setting to the parametric one, a Bézier patch is given by

$$\mathbf{p}(\boldsymbol{\tau}) = \sum_{|\mathbf{i}|=d} \mathbf{P}_{\mathbf{i}} B^d_{\mathbf{i}}(\boldsymbol{\tau}) \tag{5}$$

with control points $\mathbf{P}_{\mathbf{i}} \in \mathbb{R}^3$ forming a triangular control net.

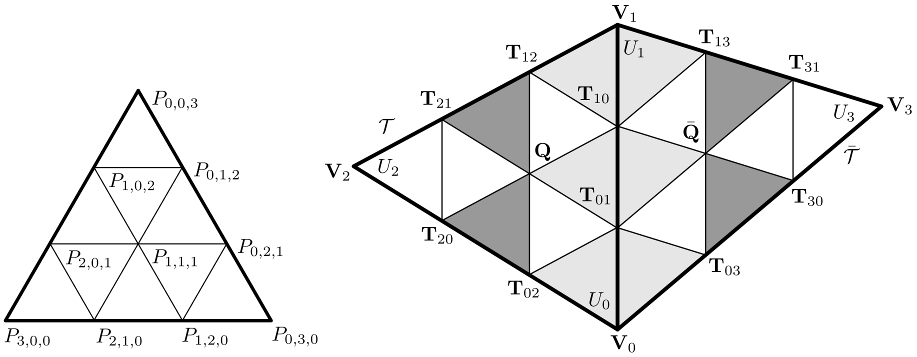

Fig. 1. Left: A cubic Bézier triangle and its associated ordinates. Right: Labelling of control points in two adjacent triangles $\mathcal{T}$ and $\bar{\mathcal{T}}$. The control points $\mathbf{V}_i$ correspond to vertices U_i in parameter space.

2.2. Continuity conditions

With the notation in Fig. 1, right, C^1 continuity conditions between $\mathcal{T}$ and $\bar{\mathcal{T}}$ are, on top of trivial C^0 conditions, given by

$$\begin{aligned} \mathbf{T}_{13} &= \tau_0\mathbf{T}_{10} + \tau_1\mathbf{V}_1 + \tau_2\mathbf{T}_{12}, \\ \bar{\mathbf{Q}} &= \tau_0\mathbf{T}_{01} + \tau_1\mathbf{T}_{10} + \tau_2\mathbf{Q}, \\ \mathbf{T}_{03} &= \tau_0\mathbf{V}_0 + \tau_1\mathbf{T}_{01} + \tau_2\mathbf{T}_{02}, \end{aligned} \tag{6}$$

where (τ_0, τ_1, τ_2) are the barycentric coordinates of U_3 with respect to $\mathcal{T}(U_0, U_1, U_2)$, i.e., $U_3 = \tau_0 U_0 + \tau_1 U_1 + \tau_2 U_2$; see Farin (1985, 1986). In other words, the three quadrilaterals $(\mathbf{V}_1, \mathbf{T}_{12}, \mathbf{T}_{10}, \mathbf{T}_{13})$, $(\mathbf{T}_{10}, \mathbf{Q}, \mathbf{T}_{01}, \bar{\mathbf{Q}})$, and $(\mathbf{T}_{01}, \mathbf{T}_{02}, \mathbf{V}_0, \mathbf{T}_{03})$ (light-shaded in Fig. 1, right) have to be planar and all of the same affine shape as (U_1, U_2, U_0, U_3).

Similarly, conditions for C^2 continuity read

$$\begin{aligned} \tau_0\mathbf{Q} + \tau_1\mathbf{T}_{12} + \tau_2\mathbf{T}_{21} &= \bar{\tau}_0\mathbf{T}_{13} + \bar{\tau}_1\bar{\mathbf{Q}} + \bar{\tau}_2\mathbf{T}_{31}, \\ \tau_0\mathbf{T}_{02} + \tau_1\mathbf{Q} + \tau_2\mathbf{T}_{20} &= \bar{\tau}_0\bar{\mathbf{Q}} + \bar{\tau}_1\mathbf{T}_{03} + \bar{\tau}_2\mathbf{T}_{30}, \end{aligned} \tag{7}$$

where $(\bar{\tau}_0, \bar{\tau}_1, \bar{\tau}_2)$ are the barycentric coordinates of U_2 with respect to $\bar{\mathcal{T}}(U_1, U_0, U_3)$, i.e., $U_2 = \bar{\tau}_0 U_1 + \bar{\tau}_1 U_0 + \bar{\tau}_2 U_3$. The triangles involved are dark-shaded in Fig. 1, right. In the context of cubic splines, these C^2 conditions cannot be, in general, satisfied, as the cubic construction does not provide enough degrees of freedom. Nevertheless, these conditions can be used to minimise C^2 discontinuities between neighbouring triangles, as we shall see in Section 3.3.

2.3. Clough–Tocher spline space

Let $\triangle$ be a conforming triangulation of a domain $\Omega \subset \mathbb{R}^2$ with one or more polygonal boundary loops and let n_v, n_e, and n_t be the number of vertices, edges, and triangles of $\triangle$, respectively. Additionally, let $\mathcal{E}$ denote the set of edges in $\triangle$ and let the vertices U_i of $\triangle$ have Cartesian coordinates (also called parameter values) given by (u_i, v_i).

In the Clough–Tocher construction, each of the triangles in $\triangle$, which we call *macro-triangles*, is partitioned into three *micro-triangles* (Clough and Tocher, 1965); see Fig. 2. For each triangle $\mathcal{T} \in \triangle$, a split point Z is chosen and connected to the three vertices of $\mathcal{T}$ by new edges, giving rise to the Clough–Tocher refinement of $\triangle$, denoted by $\triangle^\star$.

The cubic C^1 spline space defined on $\triangle^\star$ is called the *Clough–Tocher spline space*

$$\mathcal{S}_3^1(\triangle^\star) = \{s \in C^1(\Omega) : s|_{\mathcal{T}^\star} \in \Pi_3, \mathcal{T}^\star \in \triangle^\star\}. \tag{8}$$

Its dimension is equal to $3n_v + n_e$. Note that the dimension of the space $\mathcal{S}_3^1(\triangle)$ over a general triangulation $\triangle$ is still an open question, so the extra freedom granted by a Clough–Tocher refinement is useful to construct C^1 cubic splines in practice. We will call the restriction $s|_{\mathcal{T}}$ the *macro-patch* over $\mathcal{T}$.

The three degrees of freedom corresponding to each vertex in $\triangle$ are typically fixed by assigning the position and gradient at each vertex U_i, $i = 1, \ldots, n_v$. That is, in the parametric setting,

$$\mathbf{s}(U_i) = \mathbf{f}_i \quad \text{and} \quad \nabla\mathbf{s}(U_i) = \nabla\mathbf{f}_i = (\mathbf{f}_i^u, \mathbf{f}_i^v), \tag{9}$$

sampled at the vertices of $\triangle$ from a parametric surface $\mathbf{f}$ defined over Ω. This leaves a degree of freedom left for each edge in $\triangle$. This degree of freedom can be used to force the directional derivative of the spline s along an edge ε to be linear, instead of the general quadratic. More precisely, the reduced Clough–Tocher spline space is defined as (Speleers, 2010)

$$\hat{\mathcal{S}}_3^1(\triangle^\star) = \{s \in \mathcal{S}_3^1(\triangle^\star) : \left.\frac{\partial s}{\partial v_\varepsilon}\right|_\varepsilon \in \Pi_1, \varepsilon \in \mathcal{E}\}, \tag{10}$$

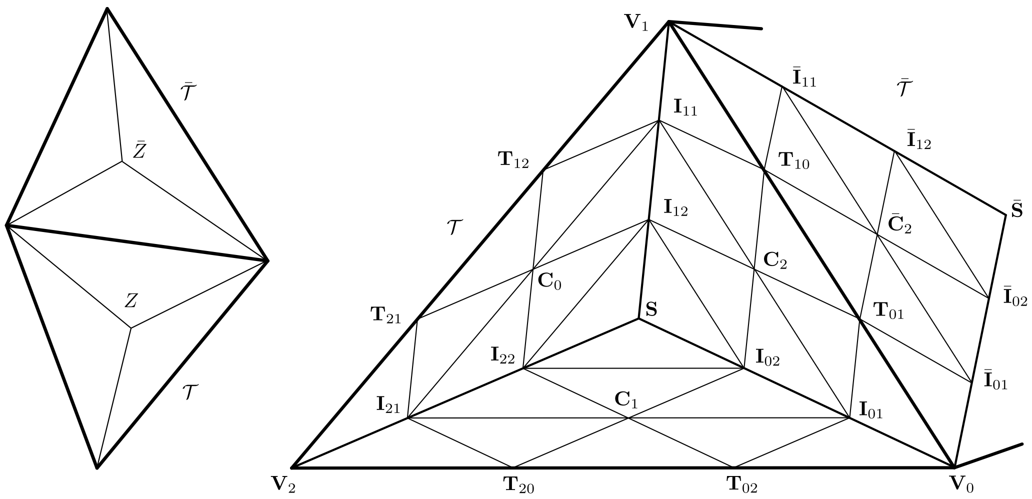

Fig. 2. Left: The Clough–Tocher split performed on two macro-triangles $\mathcal{T}$ and $\bar{\mathcal{T}}$ (thick edges) gives rise to six micro-triangles (thin edges): three per original triangle. Right: Labelling of control points in adjacent micro-triangles.

where v_ε is a (unit) vector not parallel to ε. This reduced space has dimension equal to $3n_v$ and provides a unique solution to the interpolation problem of (9).

A discussion on choosing the split point Z can be found in Schumaker and Speleers (2010). We consider three options for the split point Z: the usual barycentre of $\mathcal{T}$, the incentre of $\mathcal{T}$, and a previously unexplored option, the point in $\mathcal{T}$ that is given by the barycentric coordinates of the incentre of the corresponding triangle in 3D given by $\mathbf{f}_i$. These three options will be referred to by Bary, Inc2, and Inc3, respectively. We discuss the consequences of these options in Section 5.

3. Cubic C^1 Clough–Tocher spline constructions

An excellent survey of various C^1 cubic Clough–Tocher spline constructions was conducted by Kashyap (1996). That survey also included a new construction and a discussion of iterative methods.

We now briefly recall the most prominent C^1 cubic constructions and their novel variants, and, for convenience, provide formulas for computing all the necessary Bézier ordinates (control points).

From the discussion of C^1 continuity conditions (see Section 2.2 and Farin, 1985, 1986), one obtains the following three-step procedure for computing the control points of micro-triangles (see Fig. 2, right) from the input data (9), applied to each $\mathcal{T} \in \triangle$. The triple (τ_0, τ_1, τ_2) provides the barycentric coordinates of the split point Z in $\mathcal{T}(U_0, U_1, U_2)$.

Step 1. Set

$$\mathbf{V}_0 = \mathbf{f}_0, \quad \mathbf{T}_{01} = \mathbf{V}_0 + \frac{1}{3}\nabla\mathbf{f}_i \cdot (U_1 - U_0) \tag{11}$$

and similarly for the remaining control points along the edges of $\mathcal{T}$. Then compute

$$\mathbf{I}_{01} = \tau_0\mathbf{V}_0 + \tau_1\mathbf{T}_{01} + \tau_2\mathbf{T}_{02} \tag{12}$$

and similarly for $\mathbf{I}_{11}$ and $\mathbf{I}_{21}$.

Step 2. Compute $\mathbf{C}_0$, $\mathbf{C}_1$, and $\mathbf{C}_2$ by one of the constructions detailed below.

Step 3. Set

$$\mathbf{I}_{02} = \tau_0\mathbf{I}_{01} + \tau_1\mathbf{C}_2 + \tau_2\mathbf{C}_1 \tag{13}$$

and similarly for $\mathbf{I}_{12}$ and $\mathbf{I}_{22}$. Finally, assign

$$\mathbf{S} = \tau_0\mathbf{I}_{02} + \tau_1\mathbf{I}_{12} + \tau_2\mathbf{I}_{22}. \tag{14}$$

The first step ensures interpolation of the input data. For given split points, the second step fixes the only degree of freedom in the construction, i.e., it computes the central control point of each micro-triangle. The construction is finalised in the third step, which computes all the remaining control points to assure global C^1 continuity.

We now move on to particular constructions and show how to perform Step 2 for each of them in turn. Although triangles with boundary edges require special treatment in most constructions mentioned below, we address boundaries later in Section 3.8.

3.1. CT$_o$: *the Clough–Tocher construction*

The original construction of Clough and Tocher (1965) fixes the extra degree of freedom per edge in $\triangle$ by requiring that the normal derivative at each edge be linear. Thus, this construction yields splines from the reduced space $\hat{\mathcal{S}}_3^1$, see (10), by choosing $v_\varepsilon \perp \varepsilon$. We will refer to this construction as CT$_o$, where the 'o' stands for 'orthogonal'.

A formula for computing $\mathbf{C}_2$ based on a general direction v_ε, derived from the linear derivative requirement of $\hat{\mathcal{S}}_3^1$, is given in Section 2.3 of Speleers (2010):

$$\mathbf{C}_2 = \lambda_0\mathbf{T}_{01} + \lambda_1\mathbf{T}_{10} + \frac{1}{2}\left(\mathbf{I}_{01} + \mathbf{I}_{11} - \lambda_0(\mathbf{V}_0 + \mathbf{T}_{10}) - \lambda_1(\mathbf{V}_1 + \mathbf{T}_{01})\right) \tag{15}$$

with $(\lambda_0, \lambda_1, 0)$ the barycentric coordinates of the projection of Z onto the edge $\varepsilon = U_0U_1$ in the direction of v_ε.

In the special case of $v_\varepsilon \perp \varepsilon$, λ_0 and λ_1 are computed by orthogonal projection as

$$\lambda_1 = \frac{(Z - U_0)\cdot(U_1 - U_0)}{\|U_1 - U_0\|} \quad \text{and} \quad \lambda_0 = 1 - \lambda_1. \tag{16}$$

Clough and Tocher (1965) mention several variants of this construction and note that the original 3-split into micro-triangles was suggested by T.K. Hsieh.

The advantage of this original construction is that it is completely local. More precisely, all control points for the three micro-patches per input triangle $\mathcal{T}$ are computed using only the input data at the vertices of $\mathcal{T}$. This makes it very simple to implement.

On the other hand, the extra degree of freedom per edge is not used in an optimal way. Additionally, using the perpendicular means that the construction is not invariant with respect to affine reparameterisations, not even to non-uniform scaling of the parameter plane. While full invariance in the above sense may seem unnecessary, non-uniform rescaling of parameter space is a common operation in CAD kernels and it is therefore desirable to have this type of invariance. This is elaborated further in Section 5.1.

3.2. CT$_i$: *an invariant modification of* CT$_o$

The idea developed here was mentioned by Farin (1985) and later also by Schumaker and Speleers (2010) and Speleers (2010). Instead of using the perpendicular direction, i.e., $v_\varepsilon \perp \varepsilon$, we set v_ε to be parallel to the line connecting Z and $\bar{Z}$ in the neighbouring triangle $\bar{\mathcal{T}}$.

In this case, λ_0 and λ_1 in (15) need to be set to

$$\lambda_1 = \frac{(Z - U_0)\times(\bar{Z} - Z)}{(U_1 - U_0)\times(\bar{Z} - Z)} \quad \text{and} \quad \lambda_0 = 1 - \lambda_1, \tag{17}$$

where $\times$ denotes the planar cross-product of two vectors. This formula is easily derived by intersecting the line $Z\bar{Z}$ with the edge U_0U_1.

This construction is invariant to affine reparameterisations, hence we denote it by CT$_i$, where the 'i' stands for 'invariant'. On the other hand, it requires the positions of split points from adjacent macro-triangles and thus it is a less local construction than CT$_o$.

3.3. Fa: *Farin's construction*

Farin (1985) describes a construction that has quadratic precision and minimises the C^2 discontinuity between micro-triangles along edges of macro-triangles. We abbreviate this construction by Fa.

First, set $\mathbf{Q}$ of the macro-patch corresponding to $\mathcal{T}$ to guarantee quadratic precision (see Fig. 1, right):

$$\mathbf{Q} = \frac{\mathbf{T}_{01} + \mathbf{T}_{10} + \mathbf{T}_{12} + \mathbf{T}_{21} + \mathbf{T}_{20} + \mathbf{T}_{02}}{4} - \frac{\mathbf{V}_0 + \mathbf{V}_1 + \mathbf{V}_2}{6}. \tag{18}$$

All the other control points of the macro-patch are computed from the input data at vertices of $\mathcal{T}$ via (11). The macro-patch is then subdivided into three micro-patches according to the split point Z, e.g. using the de-Casteljau algorithm (Böhm et al., 1984). This gives initial positions of all micro-triangle control points

$$\begin{aligned}
\mathbf{I}_{01} &= \tau_0\mathbf{V}_0 + \tau_1\mathbf{T}_{01} + \tau_2\mathbf{T}_{02}, \\
\mathbf{C}_2 &= \tau_0\mathbf{T}_{01} + \tau_1\mathbf{T}_{10} + \tau_2\mathbf{Q}, \\
\mathbf{I}_{02} &= \tau_0\mathbf{I}_{01} + \tau_1\mathbf{C}_2 + \tau_2\mathbf{C}_1, \\
\mathbf{S} &= \tau_0\mathbf{I}_{02} + \tau_1\mathbf{I}_{12} + \tau_2\mathbf{I}_{22},
\end{aligned} \tag{19}$$

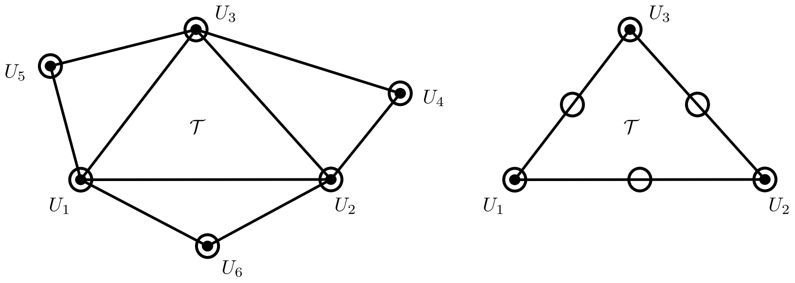

Fig. 3. Left: If the data (position, marked by bullets, and gradients, marked by circles) at the six vertices $U_1, \ldots, U_6$ surrounding $\mathcal{T}$ come from a common cubic, then the macro-patch corresponding to $\mathcal{T}$ constructed by FO or Ka reproduces that cubic. Right: If mid-edge gradients are known, as in MG$_o$ and MG$_i$, cubic reproduction over $\mathcal{T}$ is guaranteed by the input data shown.

and similarly for the remaining points. Note that **S** is actually not needed in the construction because it is later replaced in Step 3.

By formulating a constrained minimisation problem using (7), one obtains, e.g. by the standard Lagrange multipliers method,

$$\begin{aligned}\bar{\mathbf{C}}_2 &= \frac{\tau_0(2\tau_1\mathbf{R}_1 + 2\tau_2\mathbf{R}_2 + a_{12}\mathbf{R}_3 - 2\tau_0(\bar{\tau}_2\mathbf{R}_1 + \bar{\tau}_1\mathbf{R}_2) + a_{11}\mathbf{R}_3}{a_{11} + 2\tau_0 a_{12} + \tau_0^2 a_{22}},\\ \mathbf{C}_2 &= \frac{\bar{\mathbf{C}}_2 - \mathbf{R}_3}{\tau 0},\end{aligned} \tag{20}$$

where

$$\begin{aligned}\mathbf{R}_1 &= \bar{\tau}_0\bar{\mathbf{I}}_{12} + \bar{\tau}_1\bar{\mathbf{I}}_{11} - \tau_0\mathbf{I}_{12} - \tau_2\mathbf{I}_{11},\\ \mathbf{R}_2 &= \bar{\tau}_0\bar{\mathbf{I}}_{02} + \bar{\tau}_2\bar{\mathbf{I}}_{01} - \tau_0\mathbf{I}_{02} - \tau_1\mathbf{I}_{01},\\ \mathbf{R}_3 &= \tau_1\mathbf{T}_{01} + \tau_2\mathbf{T}_{10},\\ a_{11} &= 2(\tau_1^2 + \tau_2^2),\\ a_{12} &= -2(\tau_1\bar{\tau}_2 + \bar{\tau}_1\tau_2),\\ a_{22} &= 2(\bar{\tau}_1^2 + \bar{\tau}_2^2).\end{aligned} \tag{21}$$

The barycentric coordinates involved in (20)–(21) are defined by $\bar{Z} = \tau_0 Z + \tau_1 U_0 + \tau_2 U_1$ and $Z = \bar{\tau}_0\bar{Z} + \bar{\tau}_1 U_1 + \bar{\tau}_2 U_0$. Note that this procedure is done separately for each non-boundary edge using information from edge-adjacent triangles. The other control points, $\mathbf{C}_0$ and $\mathbf{C}_1$, are computed similarly.

3.4. FO: *Foley-Opitz construction*

The method of Foley and Opitz (1992) constructs a hybrid cubic patch with additional control points that allow for cross-boundary derivative control. Mann (1998) described how that construction can be modified to fit into the Clough–Tocher framework. We refer to this construction by FO.

First, repeat the following procedure three times, once for each edge of $\mathcal{T}$: set **Q** in the macro-patch corresponding to $\mathcal{T}$ to guarantee cubic precision across the edge U_0U_1:

$$\begin{aligned}\mathbf{Q} = \Big(&\mathbf{T}_{30} - \tau_1^2\mathbf{V}_0 - 2\tau_1\tau_2\mathbf{T}_{01} - 2\tau_0\tau_1\mathbf{T}_{02} - \tau_2^2\mathbf{T}_{10} - \tau_0^2\mathbf{T}_{20} + \mathbf{T}_{31} - \tau_1^2\mathbf{T}_{01}\\ &- 2\tau_1\tau_2\mathbf{T}_{10} - 2\tau_0\tau_2\mathbf{T}_{12} - \tau_2^2\mathbf{V}_1 - \tau_0^2\mathbf{T}_{21}\Big) / \left(2\tau_0(\tau_1 + \tau_2)\right).\end{aligned} \tag{22}$$

Similarly to Farin's method (Section 3.3), the macro-patch is then subdivided into micro-patches via (19), but only $\mathbf{C}_2$ is actually required. The other two applications result in $\mathbf{C}_0$ and $\mathbf{C}_1$.

As shown in Mann (1998), this construction has cubic precision in the sense that if the positions and gradients at the 6 vertices shown in Fig. 3, left, are sampled from a common cubic, the three micro-triangles corresponding to $\mathcal{T}$ describe patches of the same cubic. Additionally, it is, by construction, invariant to affine reparameterisations.

3.5. Ka: *Kashyap's construction*

Farin's method described in Section 3.3 minimises the C^2 discontinuity between micro-triangles along edges of macro-triangles. To achieve cubic precision, Farin's minimisation can be used to minimise the C^2 discontinuity of macro-patches, which are in turn used to obtain $\mathbf{C}_0$, $\mathbf{C}_1$, and $\mathbf{C}_2$. As in the case of FO, this procedure is applied three times.

First, fit a macro-patch over $\mathcal{T}$ that minimises the C^2 discontinuity across the edge given by U_0U_1. This is done using the same equations (18)–(21) as in Farin's method, with the following modifications:

$$\begin{aligned} \mathbf{R}_1 &= \bar{\tau}_0\bar{\mathbf{T}}_{21} + \bar{\tau}_1\bar{\mathbf{T}}_{12} - \tau_0\mathbf{T}_{21} - \tau_2\mathbf{T}_{12}, \\ \mathbf{R}_2 &= \bar{\tau}_0\bar{\mathbf{T}}_{20} + \bar{\tau}_2\bar{\mathbf{T}}_{02} - \tau_0\mathbf{T}_{20} - \tau_1\mathbf{T}_{02}, \end{aligned} \tag{23}$$

and the barycentric coordinates involved are defined by $U_3 = \tau_0U_2 + \tau_1U_0 + \tau_2U_1$ and $U_2 = \bar{\tau}_0U_3 + \bar{\tau}_1U_1 + \bar{\tau}_2U_0$. Note that they are not, in general, the same as the coordinates used in Farin's method in Section 3.3. Repeating this procedure for the other two edges of $\mathcal{T}$ gives the remaining control points.

This construction, which we denote by Ka, first appeared in Kashyap (1996) and is based on the 18-point interpolant introduced there. It was later rediscovered in the presented form in Mann (1998, 1999). It has cubic precision in the same sense as FO of Section 3.4; see Fig. 3, left.

3.6. MG*: mid-edge gradients*

All the methods discussed so far make use of the data (9) defined on vertices of $\triangle$ only. However, if we want to convert a given surface into a Clough–Tocher spline, we can fix the extra degree of freedom per edge in $\triangle$ by reading extra data off the given surface. Given the role of the control points computed in Step 2, it is natural to try to use them to fit the gradients at mid-points of edges in $\triangle$. However, as only one degree of freedom is available per edge, the mid-edge gradient needs to be projected in a certain direction v_ε not parallel to ε.

There are two obvious options available. One is to set $v_\varepsilon \perp \varepsilon$, leading to a construction denoted by MG$_{\mathrm{o}}$. This was investigated e.g. in Schmidt et al. (2001), Bastian and Schmidt (2001). Similarly to the original Clough–Tocher construction, this ensures complete locality, but does not provide invariance to affine reparameterisations. The other option is to set $v_\varepsilon = Z\bar{Z}$, cf. Section 3.2, leading to a construction MG$_{\mathrm{i}}$ that is invariant to affine reparameterisations.

Turning back to (5) with $d=3$, the gradient of the patch at $(U_0+U_1)/2$ is given by

$$\begin{aligned} \nabla\mathbf{p}\left(\frac{1}{2},\frac{1}{2},0\right) = \frac{3}{4\mathcal{A}(U_0,U_1,U_2)}\big(&(v_1-v_2,u_2-u_1)\mathbf{P}_{3,0,0} + (2v_1-v_0-v_2,u_0-2u_1+u_2)\mathbf{P}_{2,1,0} \\ &+ (v_1+v_2-2v_0,2u_0-u_1-u_2)\mathbf{P}_{1,2,0} + (v_2-v_0,u_0-u_2)\mathbf{P}_{0,3,0} \\ &+ (v_0-v_1,u_1-u_0)(\mathbf{P}_{2,0,1}+2\mathbf{P}_{1,1,1}+\mathbf{P}_{0,2,1})\big) \end{aligned} \tag{24}$$

with $\mathcal{A}(U_0,U_1,U_2)$ again equal to the area of $\mathcal{T}$. The central control point $\mathbf{P}_{1,1,1}$ is then easily computed from the linear equation

$$\nabla\mathbf{p}\left(\frac{1}{2},\frac{1}{2},0\right)\cdot v_\varepsilon = \mathbf{g}_\varepsilon\cdot v_\varepsilon, \tag{25}$$

where $\mathbf{g}_\varepsilon$ is the mid-edge gradient sampled from the input surface. This covers both options mentioned above. $\mathbf{P}_{1,1,1}$, computed three times, once for each edge of $\mathcal{T}$, yields the control points $\mathbf{C}_0$, $\mathbf{C}_1$, and $\mathbf{C}_2$ in Step 2. Input data for cubic reproduction over a macro-triangle are shown in Fig. 3, right. A more general treatment of cross-boundary derivatives for triangular patches can be found in Foley et al. (1993).

3.7. *Iterative methods*

We now turn our attention to the iterative method described in Farin and Kashyap (1992) and Kashyap (1996), which is based on Farin's construction in Section 3.3. Although using Farin's method in Step 2 minimises the C^2 discontinuity between adjacent patches, Step 3 subsequently changes the positions of some of the control points involved in the C^2 conditions (7) to achieve global C^1 continuity. Thus, Step 2 can be performed again, then Step 3, and so on.

This gives an iterative process. It weakens the locality of the construction, but typically produces results of better visual quality; see Section 5. We note that this iterative procedure, one step of which is denoted by IM, can be applied to any of the Clough–Tocher variants described above.

A second procedure, presented in Section 4.2.2 of Kashyap (1996), minimises the C^2 discontinuities within macro-patches. The result is a single cubic patch over each macro-triangle, but with only C^0 continuity between adjacent macro-patches. While it was suggested in Kashyap (1996) that this procedure can be used in combination with the former iterative method, we did not observe any significant improvement in any of our experiments based on this combination. The adjustment of control points when followed by a step of IM is so small that one cannot observe any visual difference (not even using reflection lines or curvature plots). We thus do not include this second iterative procedure in our study in Section 5.

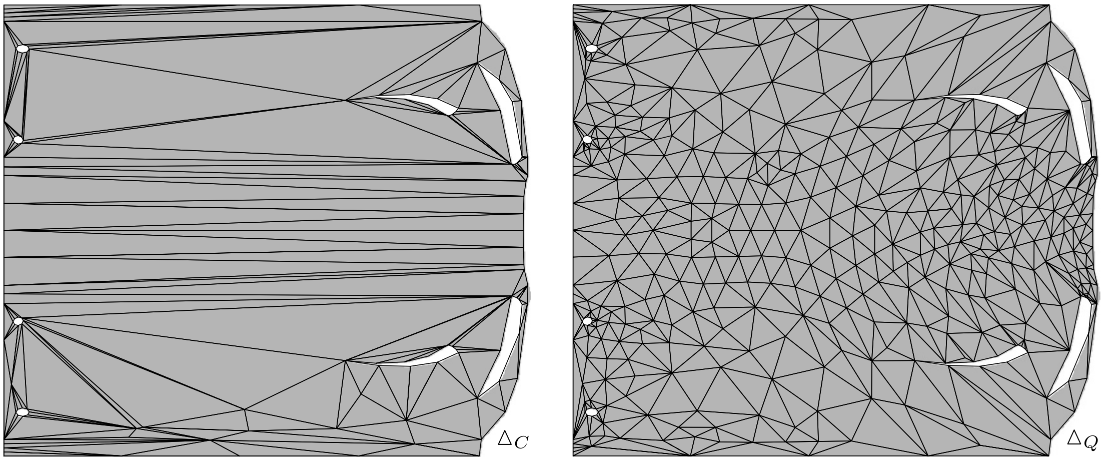

Fig. 4. Triangulations shown in parameter space. Left: Without shape constraints (135 vertices, 160 triangles; CADfix option DelC). Right: With aspect ratio constraints (443 vertices, 776 triangles; CADfix option DelQ). Both triangulations were generated with the same error tolerance. Note that the error and aspect ratio criteria are satisfied in 3D.

3.8. Boundary triangles

So far, we have not considered triangles whose edges are boundary edges of Ω. Note that only triangles with at least one edge on the boundary of Ω are considered as *boundary triangles*.

From the set of constructions detailed above, only $\mathrm{CT_o}$ and $\mathrm{MG_o}$, due to their complete locality per triangle, need no modification near boundaries. All the other constructions cannot be used over boundary triangles in the form they are defined.

If $\mathcal{T}$ is a boundary triangle then its neighbour across its boundary edge ε is not available. In that case, we first consider two options based on (15). One is to use (16), i.e., the perpendicular direction, to compute $\mathbf{C}_2$. The other is to use the edge midpoint ($\lambda_0 = \lambda_1 = 1/2$), which does not preclude invariance to affine reparameterisations.

If mid-edge gradients are known on boundary edges (i.e., not necessarily on all edges), one can use (25) with either $v_\varepsilon \perp \varepsilon$ or v_ε in the direction given by Z, the split point of $\mathcal{T}$, and the boundary edge midpoint. Again, only the second choice may lead to invariance to affine reparameterisations.

While it is likely that the degree of freedom per boundary edge can be put to a better use in some of the constructions, we do not investigate this any further.

4. Delaunay triangulation

Before we can compare the above variants of the original Clough–Tocher construction, we need to generate suitable input data. We therefore turn to the problem of generating triangulations with the necessary gradient information from trimmed surfaces appearing in CAD models. While there are many suitable triangulation algorithms (Piegl and Richard, 1995; Piegl and Tiller, 1998; Kumar et al., 2001), the examples in this paper used the CADfix product (TranscenData, 2015) to import CAD models and generate the required data.

CADfix performs the triangulation in the parameter space of each surface, using an incremental constrained Delaunay triangulation (CDT) algorithm similar to the approach described by Kallmann et al. (2004). For NURBS surfaces, the parameter space is naturally given by the rectangle in $\mathbb{R}^2$ that is the tensor product of the spline domains in each parametric direction. For other surface types, a parameterisation is constructed that ensures that the edges bounding each face project to closed loops in parameter space. The input to the triangulation algorithm also includes a maximum permissible error between the surface and its meshed representation, and a flag that indicates whether there are constraints on the aspect ratio of the constructed triangles; see Fig. 4.

Given this input, the triangulation algorithm starts by polygonising each boundary edge, beginning with a small number of samples and adaptively adding samples in any spans where the current representation exceeds the permissible error. The representation of each edge constructed in this stage is a C^1 cubic spline with derivatives that are also sampled from the edge. Each sample of a boundary loop corresponds to a parameter-space node in $\mathbb{R}^2$, and every pair of adjacent samples has a constrained segment added in the CDT algorithm. The output of this stage is therefore a triangulation of the face which satisfies the error bound on the edges but may be a very poor representation of the internal surface geometry. Note that this edge-first triangulation strategy guarantees that adjacent surfaces share the same boundary vertices and thus makes it possible to produce watertight models.

The second stage of the algorithm iteratively refines the triangulation by examining the error at triangle centroids and triangle edge midpoints, until all triangles have an error that lies below the given bound. For all the examples in this paper,

Table 1
Property comparison. See Section 5.1 for a detailed explanation of the meanings of the rows. In each row, best ranked constructions are highlighted in grey. All criteria apply to all three split point options, with the exception of invariance to affine reparameterisations in the second row. In the third row, data required from edge-adjacent triangles are listed; the index ranges in $\{4, 5, 6\}$; see Fig. 3, left.

	CT$_\mathrm{o}$	CT$_\mathrm{i}$	Fa	FO	Ka	MG$_\mathrm{o}$	MG$_\mathrm{i}$
Polynomial reproduction	2	2	2	3	3	3	3
Invariance (Bary & Inc3 only)	×	✓	✓	✓	✓	×	✓
Data from edge-adjacent triangles	–	U_i	$U_i, \mathbf{f}_i, \nabla\mathbf{f}_i$	$U_i, \mathbf{f}_i, \nabla\mathbf{f}_i$	$U_i, \mathbf{f}_i, \nabla\mathbf{f}_i$	–	U_i
Storage (floats per vertex/edge)	11/0	11/0	11/0	11/0	11/0	11/3	11/3

error was estimated with respect to the CT$_\mathrm{o}$ construction (with Bary), and the same triangulation and input data are fixed for any comparisons between constructions. Note that this error estimate, while being relatively cheap to compute, does not provide an upper bound. An upper bound on the error could be obtained, if needed, by using local convex hulls provided by the Bézier control nets of Clough–Tocher micro-patches; cf. (5).

If required, the constraints on triangle aspect ratio are also satisfied in this second stage by inserting additional samples that are purely to split triangles with a high aspect ratio; see Fig. 4, right. Unfortunately, at the end of this process, there can still be a mismatch between the C^1 cubic splines constructed for each edge and the boundaries of the Clough–Tocher spline surface constructed for each face. An example is shown in Fig. 18. This is a necessary consequence of the geometry of the trimming curves, the image of which is generally not a cubic spline in the spline surface; generically, only straight edges in parameter space map to cubic curves on Clough–Tocher splines.

We can repair this mismatch rather inelegantly, by simply moving the boundary control points constructed for the Clough–Tocher spline to the positions given by the edge cubic spline control points. Following the description given in Section 3, this involves a different value for $\mathbf{T}_{01}$ and $\mathbf{T}_{10}$ (for example) in calculating Step 1. This modification restores C^0 continuity between adjacent Clough–Tocher splines, at the cost of introducing a jump in derivative at the edges of any triangle touching the boundary. In practice, we find that the modifications required by this correction are typically in-plane with the surface, so the spline is usually close to satisfying G^1 continuity conditions even though it is no longer C^1. This is shown in Fig. 18.

5. Comparison and results

We now compare and contrast the constructions introduced in Section 3. Our assessment criteria include invariance to affine reparameterisations, polynomial reproduction, visual quality (reflection lines, Gaussian and mean curvature), and approximation error.

5.1. Quantitative comparison

Table 1 presents a quantitative comparison of the seven Clough–Tocher variants. The best ranked entries for each considered criterion (row) are shaded. None of the constructions performs best across all criteria and only FO and Ka perform best in three out of four of them. Note, however, that different applications may put emphasis on different criteria.

We now address each criterion in more detail.

Polynomial reproduction. Reproduction of polynomials of a certain degree is an important property for nearly any spline construction. Higher polynomial reproduction leads to smaller approximation errors and faster convergence rates under refinement (Lai and Schumaker, 1998). Additionally, cubic precision, instead of quadratic, generally leads to fairer surfaces as we show in the examples below.

Invariance. As already mentioned above, shape invariance with respect to affine reparameterisations guarantees that a rescaling (or any other affine transform) of parameter space does not change the surface. More precisely, let $\mathbf{A}$ be a regular 2×2 matrix and T a translation vector. Then all vertices U_i of $\triangle$ are mapped to $\mathbf{A}U_i + T$ and all gradients $\nabla\mathbf{f}_i$ are transformed to $\bar{\mathbf{A}}\nabla\mathbf{f}_i$, where $\bar{\mathbf{A}}$ is the inverse transpose of $\mathbf{A}$ (this follows from the chain rule). All positions $\mathbf{f}_i$ in 3D remain unchanged. Then, a spline $\mathbf{s}(\mathbf{u})$ is invariant to affine reparameterisations if $\mathbf{s}(U) = \bar{\mathbf{s}}(\mathbf{A}U + T)$ for all values of $U \in \Omega$, where $\bar{\mathbf{s}}$ uses the transformed data. We investigate this property further in the example shown in Fig. 9. Clearly, using Inc2 as split points precludes this invariance. We thus include only Bary and Inc3 in Table 1.

A related concept is that of *affine covariance* (often also called invariance) in 3D. When an affine transform is applied to the input data in 3D, the resulting spline is the same affine image of the one produced by the original input data. It follows from the definitions of the Clough–Tocher constructions that unless incentres from 3D are used, all variants are affinely covariant in the above sense. This property is not as important as the former invariance in our context; our focus is on converting existing models, not their affine images. And although the input NURBS models are affinely covariant in 3D, the triangulation algorithm used to generate our input data is not.

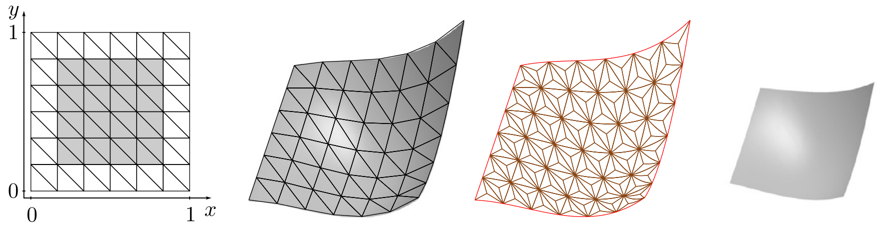

Fig. 5. Farin's cubic given in (26). From left to right: Function values and gradients were sampled at a regular grid of 7 × 7 vertices over the unit square in parameter space; only the highlighted region is reconstructed to avoid boundary influence. A 3D view of the cubic with the triangular grid superimposed. Edges of individual micro-triangles (Ka with BARY). The cubic without the outer-most layer.

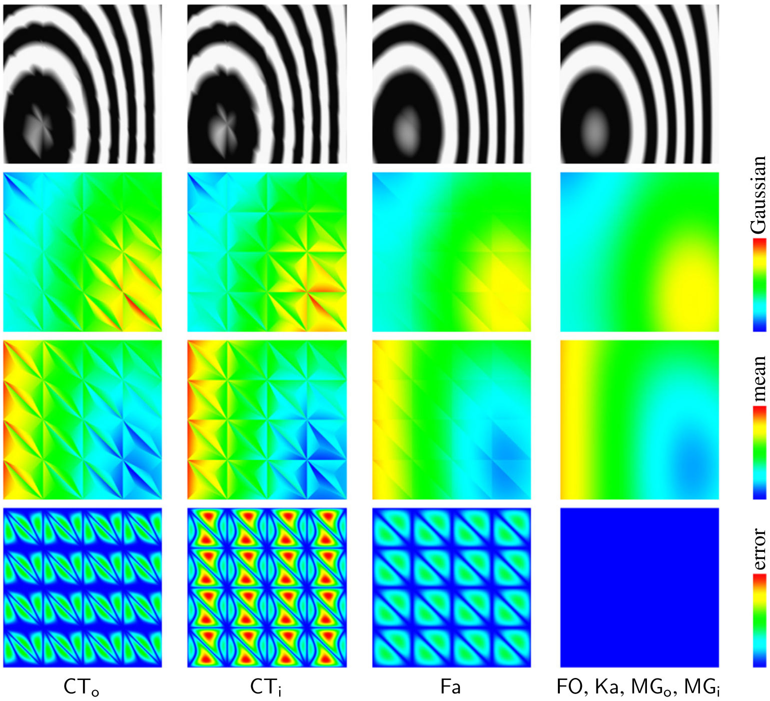

Fig. 6. Clough–Tocher spline approximations of Farin's cubic; see Fig. 5 for setup. Rows from top to bottom: Reflection lines, Gaussian curvature $[-1.938, 4.073]$ and mean curvature $[-2.054, 0.717]$ (the scale is the same across each row). Last row: Error plots on the common scale of $[0, 1.361 \cdot 10^{-4}]$, computed as function value differences. Observe that FO, MG$_o$, MG$_i$, and Ka all give the same result; this follows from the cubic precision of these constructions. Barycentres were used as the split point Z in all cases.

Locality. Complete locality to one triangle simplifies implementation and boundary triangles do not require special treatment. On the other hand, constructions which make use of data from adjacent triangles lead to fairer surfaces, as we demonstrate in our examples below.

Only CT$_o$ and MG$_o$ possess complete locality. More precisely, the construction of the macro-patch over $\mathcal{T}$ is based on data attached to $\mathcal{T}$ only. CT$_i$ and MG$_i$ have the same locality with respect to position and gradients in 3D, but require the split points of edge-adjacent triangles. Finally, all the other constructions need 3D data from adjacent triangles. The data required from edge-adjacent triangles are listed in Table 1.

Storage. Note that the 11 floats per vertex in Table 1 correspond to 3 floats for 3D position plus 6 for gradients plus 2 for parameter values.

Storing 3 extra floats per edge can hardly be considered a serious disadvantage. However, if full gradients, not just their projections in a particular direction, are stored, 11/3 becomes 11/6. An interesting compromise, as already mentioned in Section 3.8, is to store only boundary mid-edge gradients (as opposed to all mid-edge gradients).

Having explored theoretical aspects of Clough–Tocher spline variants, we now proceed to their visual assessment and error comparison on several examples, ranging from simple test cases to full trimmed NURBS models.

The errors presented throughout the paper were, unless specified otherwise, computed as point to mesh distances over dense samples of both the input function (or model) and the Clough–Tocher approximation using the Computational Geometry Algorithms Library (CGAL, 2015). The errors are reported relative to the models' bounding box diagonals.

Table 2
Error table for the Franke function of (27). The maximum error for each spline variant is reported, computed as function value differences. Maximum errors for Ka after 10 steps of IM are shown in the last column.

	$\mathrm{CT_o}$	$\mathrm{CT_i}$	Fa	FO	Ka	$\mathrm{MG_o}$	$\mathrm{MG_i}$	Ka + 10IM
Bary	0.058416	0.059744	0.057606	0.057485	0.057485	0.053633	0.053633	0.056640
Inc2	0.059065	0.060778	0.057447	0.057019	0.057019	0.053632	0.053632	0.056390
Inc3	0.058912	0.060678	0.057426	0.057116	0.057116	0.053632	0.053632	0.056403

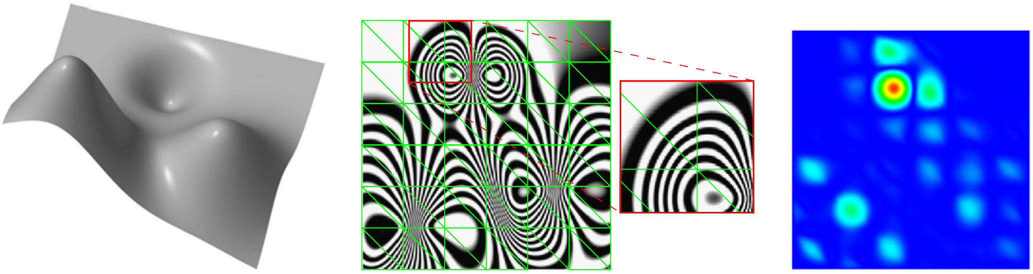

Fig. 7. Franke function (27). From left to right: Shaded plot, reflection lines on the input surface, a close-up, and an error plot for Ka with Bary; data were sampled over the same 7 × 7 grid shown in Fig. 5, left; here superimposed in green. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.2. Academic examples

Our first example is Farin's cubic function (Farin, 1985, Section 6)

$$c(x, y) = (x - 0.3)^3 + x(y - 0.3)^2 - 0.1x, \tag{26}$$

which we used to test polynomial precision of Clough–Tocher spline variants. The general setup is described in Fig. 5. Reflection lines, and Gaussian and mean curvature plots are shown in Fig. 6. The results are in accord with the expected respective quadratic and cubic precision shown in Table 1. While barycentres were used as split points in Fig. 6, we emphasise that the split point does not influence the precision of a particular construction. Additionally, running the iterative algorithm IM on the three constructions with only quadratic precision (first three columns) will force them to converge to the input cubic. Applying IM to any of the cubic precision constructions will not modify the reproduced cubic (which already minimises C^2 discontinuities).

Our second example is one of the Franke bivariate test functions (Franke, 1979):

$$f(x, y) = \frac{3}{4}e^{-\frac{1}{4}(9x-2)^2-\frac{1}{4}(9y-2)^2} + \frac{3}{4}e^{-\frac{1}{49}(9x+1)^2-\frac{1}{10}(9y+1)^2} + \frac{1}{2}e^{-\frac{1}{4}(9x-7)^2-\frac{1}{4}(9y-3)^2} - \frac{1}{5}e^{-(9x-4)^2-(9y-7)^2}. \tag{27}$$

Maximum errors are summarised in Table 2, computed as function value differences. As all error plots over the whole spline look very similar across all constructions, we show only one of them, for Ka with barycentres, in Fig. 7. Constructions that need special treatment over boundary triangles were modified by using edge perpendiculars as described in Section 3.8. Using edge midpoints gave nearly identical results.

Shaded renderings and mean curvature plots on the whole domain, and reflection lines on a subpatch are shown in Fig. 8; all seven constructions are included. The split point has negligible influence on the resulting shape and thus results using Bary only are shown. The influence of split points is discussed further in Section 5.3. Gaussian curvature plots are not shown as they did not provide sufficient visual differences.

Observe that while both $\mathrm{MG_o}$ and $\mathrm{MG_i}$ give the smallest maximum errors (Table 2), they lead to visually poor results (Fig. 8). On the other hand, fairest shapes were achieved by Fa, FO, and Ka. These results can be further improved by applying IM several times.

As our next example, we consider a simple sine wave and data sampled from it using a sparse triangulation. As can be observed in Fig. 9, best results with $\mathrm{CT_o}$ are achieved when the 2D and 3D triangulation 'agree' as much as possible (i.e., when the parameterisation is close to isometric). The example shows that while $\mathrm{CT_o}$ can produce worse results than its invariant modification $\mathrm{CT_i}$ in some extreme situations, proper scaling in parameter space leads to better shapes using the original Clough–Tocher construction $\mathrm{CT_o}$. We obtained similar results for $\mathrm{MG_o}$ when compared with its invariant counterpart $\mathrm{MG_i}$.

One approach to quantifying fairness of Clough–Tocher splines is to consider the maximum magnitude of the differences of the right-hand and left-hand sides in (7), i.e., the maximum magnitude of this C^2 discontinuity measure across edges between micro- and macro-patches. This measure is included in Fig. 9. Note how the C^2 discontinuity values closely correspond to the visual comparison using reflection lines and the significant improvement after affine reparameterisation.

Our next test case is the sphere. We use the rational quartic Bézier triangle representation (Farin et al., 1987) of one of its octants and a tessellation of its equilateral parametric triangle into 4^l macro-triangles, where l denotes the tessellation

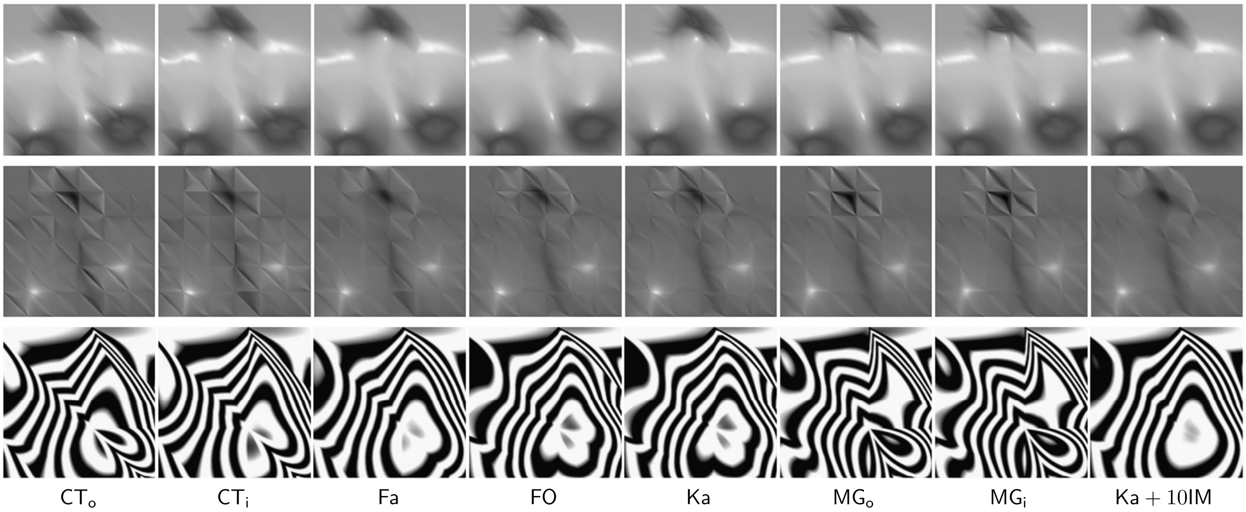

Fig. 8. Clough–Tocher spline approximations of Franke function (27). Top row: Shaded renderings. Middle row: Mean curvature plots, all on the same scale $[-27.48, 39.47]$. In this case, grey-scale (black to white) was used because it better reveals curvature discontinuities. Bottom row: Reflection lines on the close-up part shown in Fig. 7. All results are based on Bary. The rightmost plot shows Ka after 10 steps of the iterative procedure IM.

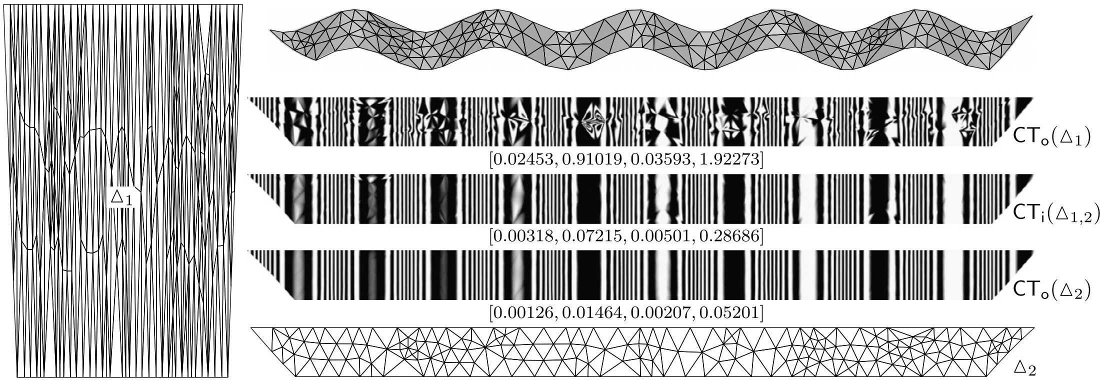

Fig. 9. A simple wave example. The triangulation in 3D (top right) remains fixed, but we consider two parameterisations, $\triangle_1$ and $\triangle_2$, which are related by a non-uniform scaling in parameter space. As expected, CT$_i$ gives identical results on both parameterisations due to its invariance to affine reparameterisations. In contrast, the shape and fairness of CT$_o$ are heavily influenced by the parameterisation; compare CT$_o(\triangle_1)$ and CT$_o(\triangle_2)$. Top views with reflection lines are shown. All three examples were generated using Bary. The four numbers under each reflection line plot are C^2 discontinuities: mean and maximum C^2 discontinuity at macro-edges and the same two measures at micro-edges, in that order.

level. Convergence results for Gaussian and mean curvature, and approximation error are reported in Fig. 10 for levels $l \in [1, \dots, 5]$, i.e., over regular triangulations with 4 to 1024 triangles. The case with $l = 0$, i.e., with only one macro-triangle, is not included, as a single triangle gives no information on the fairness between macro-patches.

A visual comparison of Clough–Tocher approximations of an octant of the unit sphere for $l = 2$, in terms of their fairness and approximation error, is shown in Fig. 11. FO produced indistinguishable results from those generated by Ka. KaG denotes the variant of Ka where mid-edge gradients (Section 3.8) are used at boundary edges (both projection options gave indistinguishable results); the improvement over Ka is significant along boundary edges and it compares favourably even to MG$_o$, which relies on mid-edge gradients at all edges. We observed this behaviour across all tessellation levels; see Fig. 10. Note that the error plots (on logarithmic scale) exhibit the expected convergence rates, i.e., cubic for CT$_o$, FO, and Ka, which possess quadratic precision (in the case of Ka, this is due to boundary effects), and quartic for MG$_o$ and KaG, which have cubic precision over the entire triangulation.

Fa, while superior to CT$_o$, produced worse results than FO, Ka, and KaG, both visually and error-wise, and is therefore not included. Only barycentres were considered in all constructions because other split points gave (nearly) the same results due to symmetry. For the same reason, CT$_i$ and MG$_i$ are not shown as they are indistinguishable from their included counterparts CT$_o$ and MG$_o$, respectively.

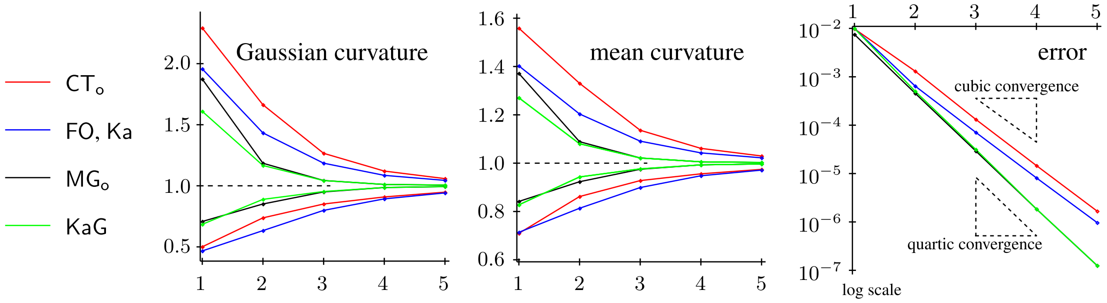

Fig. 10. Octant of a sphere: Convergence of Gaussian (left) and mean curvature (middle), and approximation error (right). All three horizontal axes correspond to the tessellation level l of one octant of the unit sphere; level $l = [1, \dots, 5]$ has 4^l macro-triangles. Bary were used in all cases.

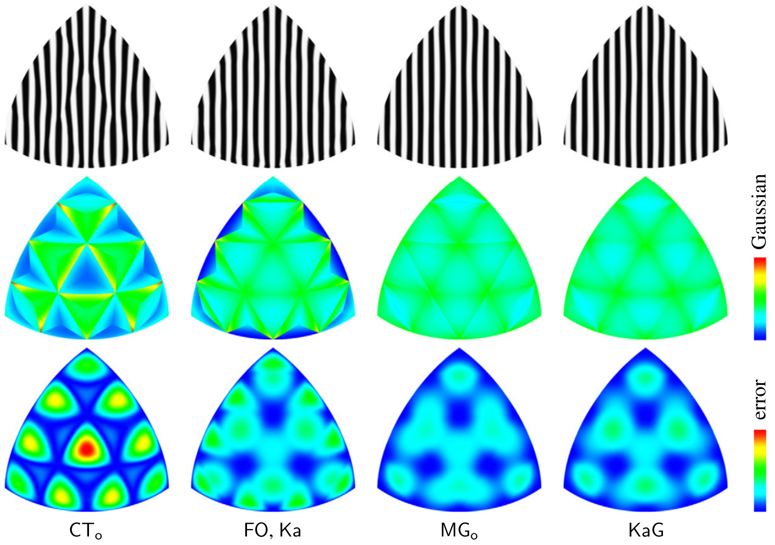

Fig. 11. One octant of the unit sphere with tessellation level $l = 2$. Top row to bottom: reflection lines, Gaussian curvature ranging in [0.738, 1.663], and error in $[0, 2.265 \cdot 10^{-3}]$.

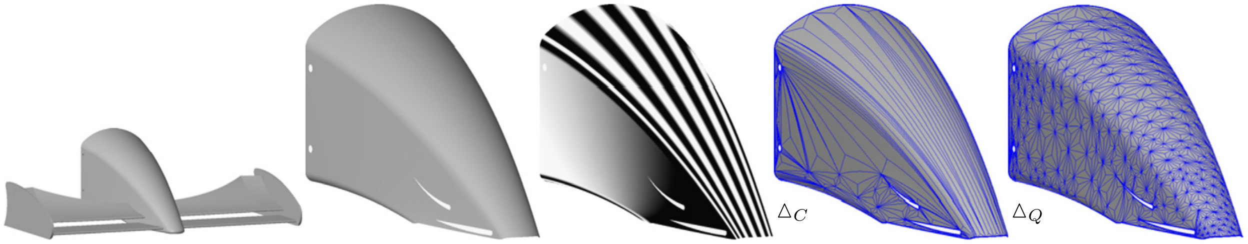

Fig. 12. Car front wing model (GrabCAD, 2015). The full B-rep model is on the left. One trimmed NURBS patch is shown shaded and with reflection lines. Approximation results based on $\triangle_C$ and $\triangle_Q$ (cf. Fig. 4) using Ka with Inc3 are shown on the right.

5.3. CAD models

Our first CAD model example is the car front wing model shown in Fig. 12. We used only one of its trimmed NURBS patches in our tests. We consider two triangulations, $\triangle_C$ and $\triangle_Q$, both of which are shown in parameter space in Fig. 4. The corresponding spline approximations with reflection lines are shown in Fig. 13.

While the triangulation $\triangle_C$ may seem inferior because it contains skinny triangles, it leads to superior results in terms of lateral artifacts when compared to $\triangle_Q$. These artifacts are caused by the well-known 'dinosaur back' effect, which arises in nearly all spline approximations when control meshes are not aligned with features on a model (Farin et al., 2002; Augsdörfer et al., 2011). Additionally, once a triangulation has been fixed, the macro-edges of all macro-patches in 3D are uniquely determined and shared by all Clough–Tocher variants. On the other hand, skinny triangles may give rise to foldovers in the resulting spline; see Fig. 14, top row. However, in our tests it was possible to avoid foldovers by using Inc2

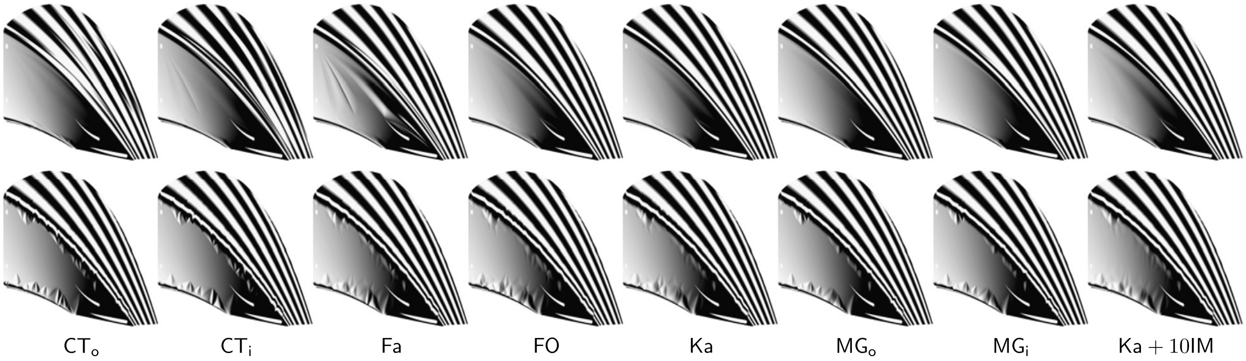

Fig. 13. Spline approximations of the car front wing model shown with reflection lines. Top row: $\triangle_C$. Bottom row: $\triangle_Q$. All results are based on Inc3.

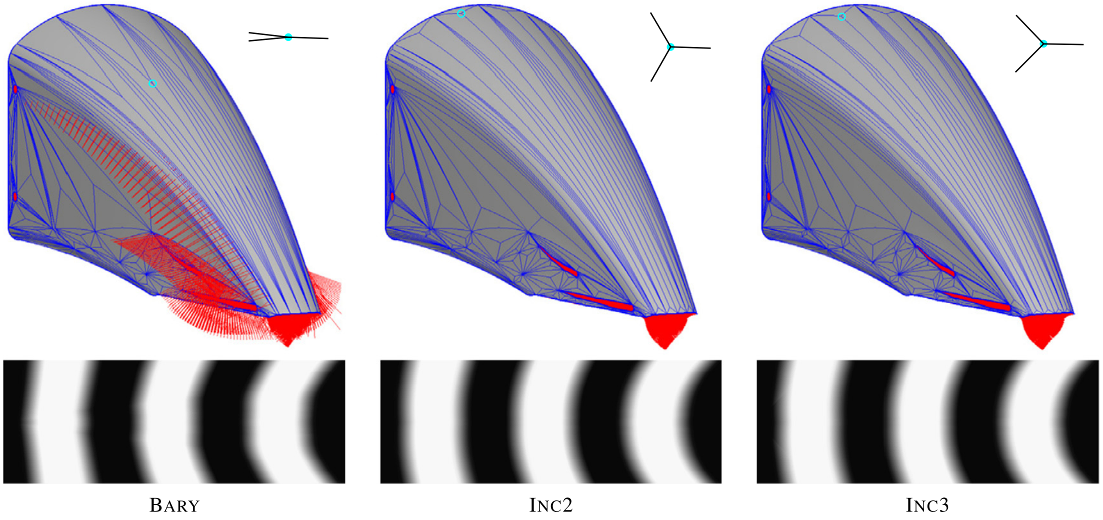

Fig. 14. Split point comparison on the model of Fig. 12, middle. Top row: Macro- and micro-patch edges (blue) and inward normals (red; all of the same magnitude). Observe how the split point influences the shape of the micro-patches within the macro-patches (angles shown in black, split points in cyan) at the top of the model; cf. Fig. 16. Bary leads to fold-overs on the Clough–Tocher spline, indicated by normals pointing outwards. Bottom row: Close-ups of the top of the model with reflections lines. Note that barycentres produce the worst shape behaviour. Ka was used in all examples shown. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

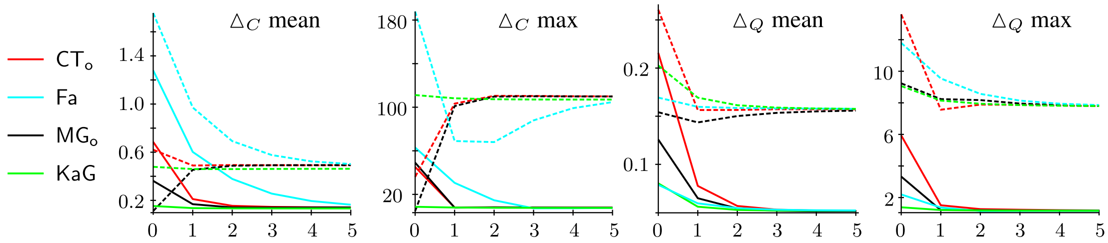

Fig. 15. C^2 discontinuity graphs for the car front wing model. Solid lines represent C^2 discontinuities across macro-edges and dashed lines correspond to C^2 discontinuities across micro-edges. The two graphs on the left are for $\triangle_C$, the two on the right for $\triangle_Q$. The horizontal axes in all four graphs state the number of steps of IM applied. FO with mid-edge gradients at boundary edges produced nearly indistinguishable results from those shown for KaG.

or Inc3. Additionally, Inc2 or Inc3 help to improve the quality of the results over Bary in macro-patches corresponding to skinny triangles (Fig. 14, bottom row).

We gathered C^2 discontinuity data of the constructions and also after several applications of IM. The results are summarised in Fig. 15, including mean and maximal C^2 discontinuities. Observe that while the iterative procedure IM reduces C^2 discontinuities across macro-edges, it is not necessarily the case for micro-edges. Additionally, the maximal C^2 discontinuities across micro-edges are typically greater than those across macro-edges. Errors for all Clough–Tocher spline variants considered in this paper are reported in Table 3 for $\triangle_C$ and $\triangle_Q$.

Table 3
Error table for the car front wing model with $\triangle_C$ and $\triangle_Q$. The maximum error for each spline variant is reported.

$\triangle_C$	CT$_\text{o}$	CT$_\text{i}$	Fa	FO	Ka	MG$_\text{o}$	MG$_\text{i}$	Ka + 10IM
Bary	0.001289	0.000627	0.008061	0.001288	0.001566	0.000435	0.000425	0.002698
Inc2	0.000973	0.000828	0.004077	0.000600	0.000582	0.000437	0.000424	0.000663
Inc3	0.000941	0.000822	0.004551	0.000596	0.000667	0.000438	0.000424	0.000758
$\triangle_Q$	CT$_\text{o}$	CT$_\text{i}$	Fa	FO	Ka	MG$_\text{o}$	MG$_\text{i}$	Ka + 10IM
Bary	0.001414	0.001291	0.001208	0.001346	0.001188	0.001227	0.001222	0.001318
Inc2	0.001330	0.001508	0.001167	0.001532	0.001253	0.001396	0.001396	0.001443
Inc3	0.001369	0.001529	0.001183	0.001281	0.001161	0.001170	0.001198	0.001290

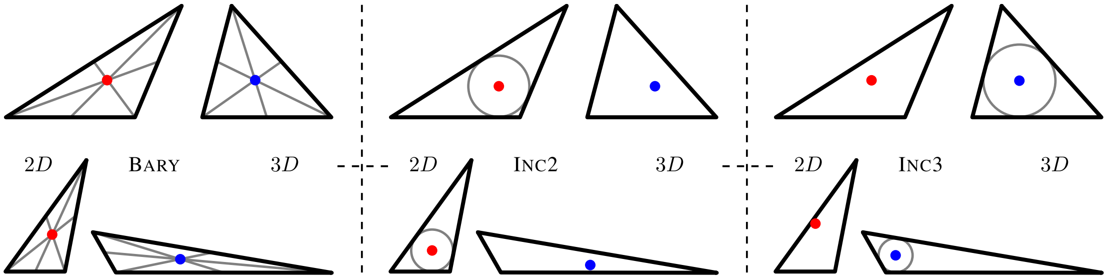

Fig. 16. Split point illustration. Six pairs of triangles, two for each split point type, are shown. In each pair, the left triangle is $\mathcal{T}$ in parameter space given by U_i and the right triangle is the corresponding triangle in 3D given by $\mathbf{f}_i$. Split points Z are shown in red and their corresponding points in 3D using the same barycentric coordinates are shown in blue. In general, the position of the blue point depends on the shape of the macro-patch, but we assume flat 3D triangles for the sake of this illustration. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

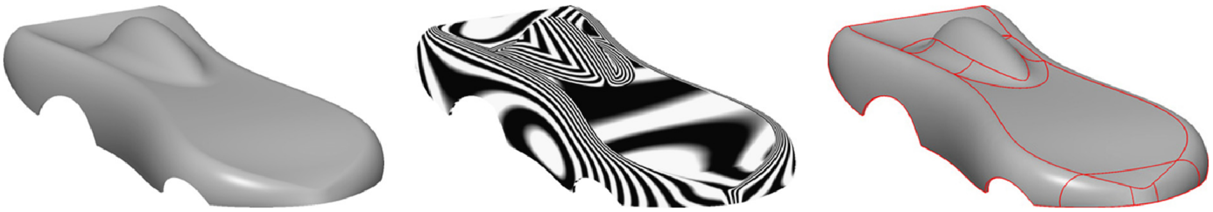

Fig. 17. Car model (GrabCAD, 2015). From left to right: The input model shaded and with reflection lines. The B-rep model consists of 42 NURBS patches, 14 of which have trimming curves that do not follow isoparameter lines. B-rep edges are shown in red. The model is topologically closed (it has no boundary), but it is not geometrically watertight. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

As we have seen in Section 5.2, the split point position has little influence on the resulting shape when the triangulations are close to regular. On the other hand, as shown in Fig. 14, this influence is much higher for highly irregular meshes containing skinny triangles. We explore this in more detail in Fig. 16. The top row shows triangles in 2D and 3D which are of similar shape. All cases can be expected to lead to good splits of the macro-patch into micro-patches. In contrast, the bottom row shows two triangles that are of very different shapes. While the red split point for Inc2 is sensibly placed in 2D, its blue image in 3D lies too close to one of the macro-edges, giving rise to a skinny micro-patch and potential artifacts. Furthermore, Bary may give rise to poorly-shaped micro-patches caused by uneven distribution of angles between inner micro-patch edges at the image of the split point on the spline; see Fig. 14, top.

In summary, Inc3 improves fairness of Clough–Tocher splines in some extreme cases, but Bary is generally the best (and also simplest) choice. Moreover, recall that Inc2 precludes invariance to affine reparameterisations (Section 5.1).

Our second CAD model example is the toy car model shown in Fig. 17. The model is topologically closed but not watertight due to trimming. The triangle layout used for conversion to Clough–Tocher splines is visualised in Fig. 18, left. The middle image shows the result when each input NURBS patch is treated separately. This leads to gaps (as discussed in Section 4). These gaps can be avoided by repositioning some of the boundary vertices in Step 1, giving a watertight model (Fig. 18, right). Observe that the adjustment reduces the continuity of the spline in the immediate neighbourhood of the B-rep edges from C^1 to C^0, but the magnitude of the discontinuity remains relatively low. While not ideal, it is a considerable improvement over the version with gaps in Fig. 18, middle, and provides an analysis-ready model. The effect of the adjustment leading to watertight models composed of several Clough–Tocher splines is further evaluated in Fig. 19.

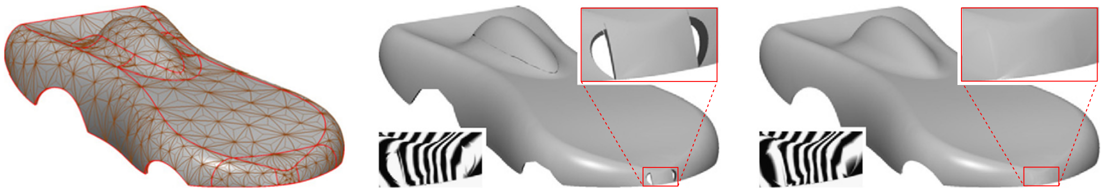

Fig. 18. The car model from Fig. 17 converted into a collection of 42 Clough–Tocher splines using Ka with Bary. Left: The layout of macro-patches. Middle: A conversion result without adjusting control points corresponding to input B-rep edges; note the gaps. Right: A watertight conversion result after adjusting control points to match common cubic splines which approximate B-rep edges; see Section 4.

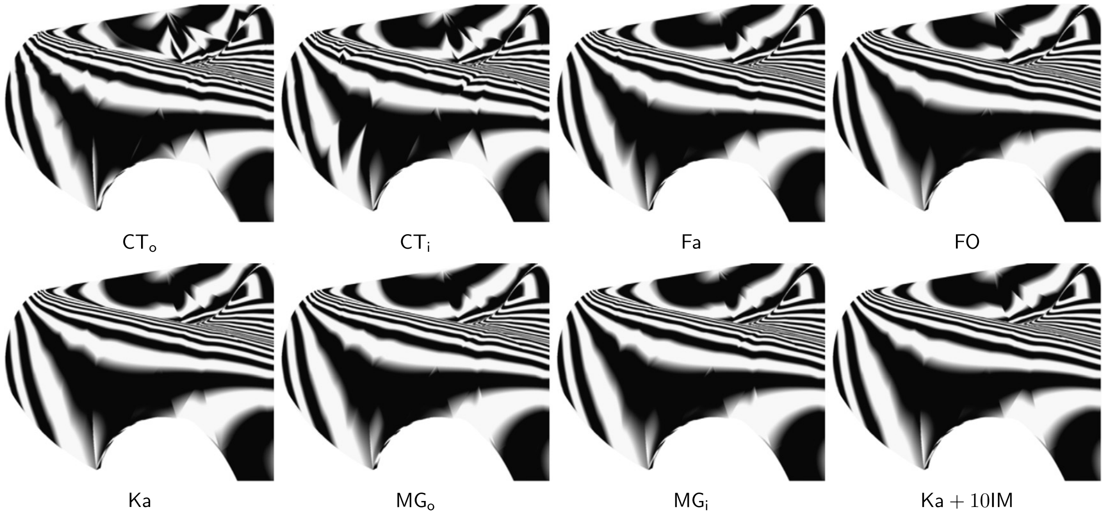

Fig. 19. The car model converted to a collection of 42 splines. The adjustment achieves watertightness at the expense of reducing the continuity across some internal edges from C^1 to C^0; this corresponds to discontinuities in reflection lines. To maintain invariance to affine reparameterisations for all the methods shown (all based on Inc3), boundary triangles used the midpoint modification described in Section 3.8 where necessary.

5.4. Summary

We now present a brief summary of our comparative study. Our main observations are:

- CT$_{\text{o}}$, CT$_{\text{i}}$, and Fa produced worst results in the majority of our experiments, mostly due to the fact that they possess only quadratic reproduction.
- With mid-edge gradients available, MG$_{\text{o}}$ and MG$_{\text{i}}$ resulted in the lowest approximation error, since more information from the converted surface is used compared to constructions that do not rely on these extra gradients. On the other hand, while Ka and FO often give nearly indistinguishable results, they also performed best in most cases in terms of fairness, with Ka slightly better when the adjustment of control points at boundary edges is used.
- The construction KaG which stores mid-edge gradients only at boundary edges is therefore a good compromise between approximation error and fairness.
- Inc3 split points lead to fairer surfaces than Bary in extreme cases such as the skinny triangles shown in Fig. 14. We do not recommend Inc2 split points, as they performed worse than Inc3 in most cases.
- The iterative minimisation of C^2 discontinuity (IM) does improve fairness, but typically only across the macro-edges that appear in the original triangulation, not across micro-edges.
- The parameterisation and triangulation that are used have a significant impact on the resulting shape. While triangles with a low aspect ratio seem desirable for avoiding folded surfaces (Fig. 14), a large number of triangles or poor alignment to model features can lead to much worse surfaces (see Fig. 13). This makes it difficult to design an optimal triangulation strategy.

6. Conclusion and future work

We have performed a comparative study of seven C^1 cubic Clough–Tocher spline constructions, combined with three split point types. An iterative method aimed at improving C^2 discontinuities, which can be applied to any spline variant, was also included.

We compared the constructions on visual fairness, invariance to affine reparameterisations, polynomial precision and approximation error. The constructions were tested on both synthetic data and on triangulated CAD models. Our results show that no construction is optimal across all assessment criteria and that surface quality depends heavily on the triangulation and parameterisation that are used. Nevertheless, if only one construction was to be chosen, we would recommend KaG, the construction of Kashyap (1996) augmented by boundary mid-edge gradients.

Avenues for future research include improving the stitching process of several Clough–Tocher splines which are separately parameterised. We achieved C^0 continuity by adjusting some of the boundary control points, however, it should be possible to join several splines with at least G^1 continuity across their interfaces without compromising their internal C^1 continuity.

Our tests showed strong boundary effects for the majority of constructions. We believe that boundary influence should be investigated in more detail and freedoms at boundary edges put to better use, ideally in conjunction with the boundary adjustment mentioned above.

The triangulation and parameterisation that are used strongly influence the fairness of the resulting approximations. It would be interesting to investigate this influence in more detail. This is also linked to the use of Clough–Tocher splines in isogeometric analysis (Jaxon and Qian, 2014).

Acknowledgements

This research was supported by the Engineering and Physical Sciences Research Council through Grant EP/K503757/1. We thank Chris D. Fellows for his help with some of the input models, Mark Gammon for his guidance, and Malcolm Sabin for proofreading. The first author thanks TranscenData Europe Ltd, the partner company on his EPSRC IAA Knowledge Transfer Fellowship.

References

Alfeld, P., Schumaker, L., 2002. Smooth macro-elements based on Clough–Tocher triangle splits. Numer. Math. 90 (4), 597–616.

Augsdörfer, U., Dodgson, N., Sabin, M., 2011. Artifact analysis on triangular box-splines and subdivision surfaces defined by triangular polyhedra. Comput. Aided Geom. Des. 28 (3), 198–211.

Bastian, M., Schmidt, J.W., 2001. Nonnegative interpolation with Clough–Tocher splines of cubic precision. J. Appl. Math. Mech./Z. Angew. Math. Mech. 81 (S3), 705–706.

Böhm, W., Farin, G., Kahmann, J., 1984. A survey of curve and surface methods in CAGD. Comput. Aided Geom. Des. 1 (1), 1–60.

Bucklow, H., 3D medial object computation using a domain Delaunay triangulation. Presented at the Medial Object Workshop, Cambridge, England, 9–10 October 2014, 2014.

CGAL, 2015. Open source project. CGAL version 4.2, http://www.cgal.org/.

Clough, R.W., Tocher, J.L., 1965. Finite element stiffness matrices for analysis of plates in bending. In: Conference on Matrix Methods in Structural Mechanics. Wright Patterson Air Force Base, Ohio, pp. 515–545.

Cottrell, J., Hughes, T., Bazilevs, Y., 2009. Isogeometric Analysis: Toward Integration of CAD and FEA. John Wiley & Sons.

Farin, G., 1985. A modified Clough–Tocher interpolant. Comput. Aided Geom. Des. 2 (1–3), 19–27.

Farin, G., 1986. Triangular Bernstein–Bézier patches. Comput. Aided Geom. Des. 3 (2), 83–127.

Farin, G., Kashyap, P., 1992. An iterative Clough–Tocher interpolant. Math. Model. Numer. Anal. 26, 201–209.

Farin, G., Piper, B., Worsey, A., 1987. The octant of a sphere as a non-degenerate triangular Bézier patch. Comput. Aided Geom. Des. 4 (4), 329–332.

Farin, G.E., Hoschek, J., Kim, M.-S., 2002. Handbook of Computer Aided Geometric Design. North-Holland/Elsevier, Amsterdam, Boston.

Foley, T., Dayanand, S., Santhanam, R., 1993. Cross boundary derivatives for transfinite triangular patches. In: Farin, G., Noltemeier, H., Hagen, H., Knödel, W. (Eds.), Geometric Modelling. In: Computing, Suppl., vol. 8. Springer, Vienna, pp. 91–100.

Foley, T.A., Opitz, K., 1992. Hybrid cubic Bézier triangle patches. In: Mathematical Methods for Computer Aided Geometric Design II. Academic Press, pp. 275–286.

Franke, R., 1979. A critical comparison of some methods for interpolation of scattered data. Technical report NPS-53-79-003. Naval Postgraduate School, Monterey, California.

GrabCAD, 2015. Open CAD library. https://grabcad.com/library.

ITI TranscenData, 2015. CADfix version 10.0. http://www.transcendata.com/products/cadfix/.

Jaxon, N., Qian, X., 2014. Isogeometric analysis on triangulations. Comput. Aided Des. 46 (0), 45–57.

Kallmann, M., Bieri, H., Thalmann, D., 2004. Fully dynamic constrained Delaunay triangulations. In: Geometric Modeling for Scientific Visualization. Springer, pp. 241–257.

Kashyap, P., 1996. Improving Clough–Tocher interpolants. Comput. Aided Geom. Des. 13 (7), 629–651.

Kumar, G.R., Srinivasan, P., Shastry, K., Prakash, B., 2001. Geometry based triangulation of multiple trimmed NURBS surfaces. Comput. Aided Des. 33 (6), 439–454.

Lai, M.-J., Schumaker, L., 1998. On the approximation power of bivariate splines. Adv. Comput. Math. 9 (3–4), 251–279.

Lai, M.-J., Schumaker, L., 2001. Macro-elements and stable local bases for splines on Clough–Tocher triangulations. Numer. Math. 88 (1), 105–119.

Mann, S., 1998. Cubic precision Clough–Tocher interpolation. Technical report CS-98-15. Computer Science Department, University of Waterloo.

Mann, S., 1999. Cubic precision Clough–Tocher interpolation. Comput. Aided Geom. Des. 16 (2), 85–88.

Piegl, L., Tiller, W., 1997. The NURBS Book, 2nd ed. Springer-Verlag, New York, Inc., New York, NY, USA.

Piegl, L.A., Richard, A.M., 1995. Tessellating trimmed NURBS surfaces. Comput. Aided Des. 27 (1), 16–26.

Piegl, L.A., Tiller, W., 1998. Geometry-based triangulation of trimmed NURBS surfaces. Comput. Aided Des. 30 (1), 11–18.

Schmidt, J., Bastian, M., Mulansky, B., 2001. Nonnegative volume matching by cubic C^1 splines on Clough–Tocher splits. SIAM J. Numer. Anal. 39 (2), 566–586.

Schumaker, L.L., Speleers, H., 2010. Nonnegativity preserving macro-element interpolation of scattered data. Comput. Aided Geom. Des. 27 (3), 245–261.

Skytt, V., Vuong, A.-V., 2013. First extension of the TERRIFIC Isogeometric Toolkit. Technical Report D6.6. The TERRIFIC Consortium.

Speleers, H., 2010. A normalized basis for reduced Clough–Tocher splines. Comput. Aided Geom. Des. 27 (9), 700–712.

Stroud, I., 2006. Boundary Representation Modelling Techniques. Springer-Verlag, London, UK.