On Smooth Bicubic Surfaces from Quad Meshes

Jianhua Fan and Jörg Peters

Dept CISE, University of Florida

Abstract. Determining the least m such that one $m \times m$ bi-cubic macro-patch per quadrilateral offers enough degrees of freedom to construct a smooth surface by local operations regardless of the vertex valences is of fundamental interest; and it is of interest for computer graphics due to the impending ability of GPUs to adaptively evaluate polynomial patches at animation speeds.
We constructively show that $m = 3$ suffices, show that $m = 2$ is unlikely to always allow for a localized construction if each macro-patch is internally parametrically C^1 and that a single patch per quad is incompatible with a localized construction. We do not specify the GPU implementation.

1 Introduction

Quad(rilateral) meshes are used in computer graphics and CAD because they capture symmetries of natural and man-made objects. Smooth surfaces of degree bi-3 can be generated by applying subdivision to the quad mesh [CC78] or, alternatively, by joining a finite number of polynomial pieces [Pet00]. When quads form a checkerboard arrangement, we can interpret 4×4 grids of vertices as B-spline control points of a bi-cubic tensor product patch. Then we call the central quad *ordinary* and are guaranteed that adjacent ordinary quad patches join C^2.

The essential challenge comes from covering *extraordinary* quads, i.e. quads that have one or more vertices of valence $n \neq 4$ as illustrated in Fig. 1,*left.* While this can be addressed by recursive subdivision schemes, in many scenarios, for example GPU acceleration, localized parallel constructions of a finite number of patches are preferable [NYM$^+$08]. Here *localized, parallel* means that each construction step is parallel for all quads or vertices and only needs to access a fixed, small neighborhood of the quad or vertex. Due to the size limitations, this paper does not discuss GPU specifics, but addresses the fundamental lower bound question: how to convert each extraordinary quad into a macro-patch, consisting of $m \times m$ bi-cubic pieces, so as that a general quad mesh is converted into a smooth surface.

Prompted by the impending ability of GPUs to tessellate and adaptively evaluate finitely patched polynomial surface at animation speeds, there have recently been a number of publications close to this problem. Loop and Schaefer[LS08] propose bi-cubic C^0 surfaces with surrogate tangent patches to convey the impression of smoothness via lighting. Myles *et al.* [MYP08] perturb a bi-cubic base patch near non-4-valent vertices by coefficients of a (5,5) patch to obtain a smooth surface. PCCM [Pet00] generates smooth bi-cubic surfaces but requires up to two steps of Catmull-Clark subdivision to separate non-4-valent vertices. This proves that $m = 4$ suffices in principle. But bi-cubic PCCM can have poor shape for certain higher-order saddles (e.g. the 6-valent monkey saddle Fig. 5, row 3) as discussed in [Pet01]. Below we specify an algorithm that constructs smoothly connected 3×3 C^1 macro-patches without this shape problem; and discuss why the approach fails when $m < 3$.

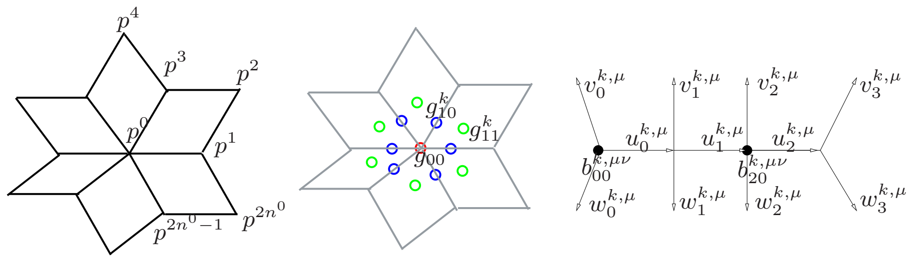

Fig. 1. *left*: extraordinary vertex p^0 with n^0 direct neighbors p^{2k-1}, $k = 1 \dots n^0$. *middle*: limit point g_{00}, tangent points g^k_{10} and 'twist' coefficients g^k_{11}. *right*: BB differences.

2 Notation, and why $m = 1$ need not be considered

We denote the kth bi-cubic Bernstein Bézier (BB) patch, $k = 1 \dots n^0$, surrounding a vertex p^0 of valence n^0 (Fig. 1) by

$$\mathbf{b}^{k,\mu,\nu}(u,v) := \sum_{i=0}^{3}\sum_{j=0}^{3} b^{k,\mu,\nu}_{ij} \binom{3}{i} u^i (1-u)^{3-i} \binom{3}{j} v^i (1-v)^{3-j}. \qquad (1)$$

Here μ, ν indicate a piece of the $m \times m$ macro-patch (see Fig. 2, *left*, for $m = 3$). The BB coefficients (control points) of the tensor-product patch $\mathbf{b}^{k,\mu,\nu}$ are therefore labeled by up to 5 indices when we need to be precise (Fig. 2):

$$b^{k,\mu,\nu}_{ij} \in \mathbb{R}^3, \quad k = 1 \dots n^0, \quad \mu, \nu \in \{0, \dots, m-1\}, \quad i, j \in \{0, 1, 2, 3\}. \qquad (2)$$

For the two macro-patches meeting along the kth boundary curve $\mathbf{b}^{k,\mu 0}(u, 0) = \mathbf{b}^{k-1,0\mu}(0, u)$, $\mu = 0, \dots, m-1$, we want to enforce *unbiased* (logically symmetric) G^1 constraints

$$\partial_2 \mathbf{b}^{k,\mu 0}(u, 0) + \partial_1 \mathbf{b}^{k-1,0\mu}(0, u) = \alpha^k_i(u) \partial_1 \mathbf{b}^{k,\mu 0}(u, 0), \quad i = 0 \dots, m-1, \qquad (3)$$

where each α^k_i is a rational, univariate *scalar function* and ∂_ℓ means differentiation with respect to the ℓth argument. If $\alpha^k_i = 0$, the constraints enforce *(parametric) C^1 continuity.* The polynomial equalities (3) hold for the kth curve exactly when all $m \times n^0$ polynomial coefficients are equal. The coefficients are differences of the BB control points:

$$v_i^{k,\mu} := b_{i1}^{k,\mu 0} - b_{i0}^{k,\mu 0}, \quad w_i^{k,\mu} := b_{1i}^{k-1,0\mu} - b_{0i}^{k-1,0\mu}, \quad u_i^{k,\mu} := b_{i+1,0}^{k,\mu 0} - b_{i0}^{k,\mu 0}.$$

The differences need only have a single subscript since we consider curves (in u) and a simpler superscript since $\nu = 0$. For example, if we choose $\alpha_i^k(u) := \lambda_i^k(1-u) + \lambda_{i+1}^k u$, then (3) formally yields $4m$ equations when $\mu = 0, \ldots, m$:

$$v_0^{k,\mu} + w_0^{k,\mu} = \lambda_\mu^k u_0^{k,\mu} \tag{4}$$
$$3(v_1^{k,\mu} + w_1^{k,\mu}) = 2\lambda_\mu^k u_1^{k,\mu} + \lambda_{\mu+1}^k u_0^{k,\mu} \tag{5}$$
$$3(v_2^{k,\mu} + w_2^{k,\mu}) = \lambda_\mu^k u_2^{k,\mu} + 2\lambda_{\mu+1}^k u_1^{k,\mu} \tag{6}$$
$$v_3^{k,\mu} + w_3^{k,\mu} = \lambda_{\mu+1}^k u_2^{k,\mu}. \tag{7}$$

By definition, $(7)_{\mu=i} = (4)_{\mu=i+1}$, i.e. constraint (7) when substituting $\mu = i$ is identical to constraint (4) for $\mu = i + 1$.

We need not consider $m = 1$, i.e. one bi-cubic patch per quad, since the *vertex-enclosure constraint* [Pet02, p.205] implies, for even $n^0 > 4$ that the normal curvatures and hence the coefficients $b_{20}^{k,00}$ (• Fig. 1,*right*) of the n^0 curves emanating from p^0 are related for $k = 1, \ldots, n^0$: the normal component of their alternating sum $\sum_k (-1)^k b_{20}^{k,00}$ must vanish. Since, for a bi-cubic patch, the control point $b_{20}^{k,00}$ lies in the tangent plane of the kth neighbor vertex (Fig. 1,*right*), the vertex's enclosure constraint constrains the neighboring tangent planes with respect to its tangent plane. Therefore, if we fix the degree of the patches to be bi-cubic and allow only one patch per quad then no localized construction is possible.

For $m > 1$, the coefficients $b_{20}^{k,00}$ no longer lie in the tangent plane of the neighbor; so a local construction may be possible. We next give an explicit construction when $m = 3$.

3 Localized smooth surface construction using a 3 × 3 macro-patch

We factor the algorithm into four localized stages. First, we define the central point g_{00}, the tangents $g_{10} - g_{00}$ and the face coefficients g_{11} as an average (see Fig. 1) of
– the extraordinary vertex p^0 with valence n^0, and
– its 1-ring neighbors $p^1, p^2, \ldots, p^{2n^0}$.
In a second stage, we partition the quad into a 3×3 arrangement (Fig. 2) and establish its boundary; in the third, we determine the cross-boundary derivatives so that pairs of macro-patches join G^1 (Equation (3)) and in the final stage, we determine the interior coefficients. By this construction, a macro-patch joins at least parameterically C^1 with an unpartitioned spline patch (see Fig. 5, row 2, where the second entry displays each polynomial piece in a different color).

1. [Initialization] It is convenient (and shown to be effective to approximate the Catmull-Clark limit surface) to set g_{ij} according to [MYP08]. That is to set g_{00} to the limit of p^0 under Catmull-Clark subdivision (red circle in Fig. 1 *middle*) and place the g^k_{10} (blue circle in Fig. 1 *middle*) on the Catmull-Clark tangent plane:

$$g_{00} = g^k_{00} := \frac{\sum_{l=1}^{n^0} \left(n^0 p_0 + 4p_{2l-1} + p_{2l}\right)}{n^0(n^0+5)}, \quad k = 1 \dots n^0, \tag{8}$$

$$g^k_{10} := g^k_{00} + e_1 c^k_{n^0} + e_2 s^k_{n^0}, \quad e_i := \frac{\sigma_{n^0}}{3(2+\omega_{n^0})} \sum_{j=1}^{n^0} \left(\alpha_i p^{2j-1} + \beta_i p^{2j}\right), \tag{9}$$

$$g^k_{11} := \frac{1}{9}(4p^0 + 2(p^{2k+1} + p^{2k+3}) + p^{2k+2}), \quad k = 1 \dots n^0. \tag{10}$$

where the scalar weights are defined as

$$c^k_{n^0} := \cos\frac{2\pi k}{n^0}, \quad s^k_{n^0} := \sin\frac{2\pi k}{n^0}, \quad c_{n^0} := c^1_{n^0},$$

$$\omega_{n^0} := 16\lambda_{n^0} - 4, \quad \lambda_{n^0} := \frac{1}{16}(c_{n^0} + 5 + \sqrt{(c_{n^0}+9)(c_{n^0}+1)}), \sigma_{n^0} := \begin{cases} 0.53 & \text{if } n^0 = 3, \\ \frac{1}{4\lambda_{n^0}} & \text{if } n^0 > 3, \end{cases}$$

$$\alpha_1 := \omega_{n^0} c^{j-1}_{n^0}, \quad \beta_1 := c^{j-1}_{n^0} + c^j_{n^0}, \quad \alpha_2 := \omega_{n^0} s^{j-1}_{n^0}, \quad \beta_2 := s^{j-1}_{n^0} + s^j_{n^0}.$$

Symmetric construction of the other three corners of the quad yields 4×4 coefficients g_{ij} that can be interpreted as the BB coefficients of one bi-cubic patch $g : [0,1]^2 \to \mathbb{R}^3$ in the form (1).

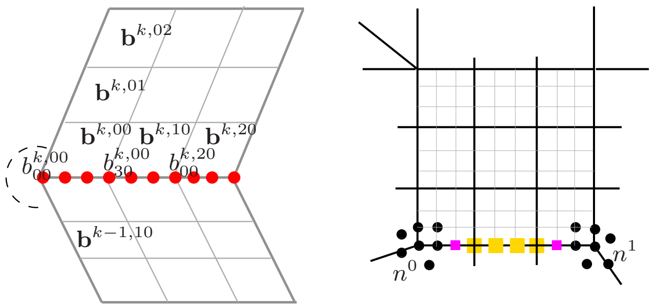

Fig. 2. *left*: **Patches $\mathbf{b}^{k,\mu\nu}$ and coefficients $b^{k,\mu\nu}_{ij}$**. *right*: Coefficients shown as black disks are determined by subdividing the initialization g_{ij} (see (11)), coefficients shown as small squares are determined by (14) and (15), coefficients shown as big yellow squares are determined by C^2 continuity of the boundary curve (16), (17), (18).

2. [domain partition and boundary] We partition the domain into 3×3 pieces (see Fig. 2 *left*) and let the 3×3 macro-patch inherit vertex position and tangents (black disks in Fig. 2 *right*) by subdividing g:

$$b_{00}^{k,00} := g_{00}^k, \quad b_{10}^{k,00} := \frac{2}{3}g_{00}^k + \frac{1}{3}g_{10}^k, \quad b_{11}^{k,00} := \frac{4g_{00}^k + 2g_{10}^k + 2g_{01}^k + g_{11}^k}{9}. \quad (11)$$

Each macro-patch will be parametrically C^1 (and C^2 in the interior). To enforce the G^1 constraints (3) between macro-patches, we need to distinguish two cases when choosing α_i^k depending on whether one of the vertices is regular, i.e. has valence 4. Of course, if both valences are 4 then we simply subdivide g and set $\alpha_i^k = 0$; and if all four corner points have valence 4, no partition is needed in the first place since g is then part of a bi-cubic tensor-product B-spline patch complex and therefore joins C^2 with its spline neighbors and at least C^1 with macro-patches (Fig. 5, row 2, second entry).

2a. [boundary $n^0 \neq 4 \neq n^1$**]** We choose α_i^k as simple as possible, namely linear

$$\text{if } n^0 \neq 4 \neq n^1 : \qquad \alpha_i^k(u) := \lambda_i^k(1-u) + \lambda_{i+1}^k u, \quad (12)$$

$$\lambda_0^k := 2\cos(\frac{2\pi}{n^0}), \quad \lambda_3^k := -2\cos(\frac{2\pi}{n^1}), \quad \lambda_1^k := \frac{2\lambda_0^k + \lambda_3^k}{3}, \lambda_2^k := \frac{\lambda_0^k + 2\lambda_3^k}{3}. \quad (13)$$

The three cubic pieces of the boundary curve have enough free parameters to enforce Equation (5) for $\mu = 0$ and Equation (6) for $\mu = 2$ and (small squares in Fig. 2 *right*) by setting

$$b_{20}^{k,00} := b_{10}^{k,00} + \frac{3(b_{11}^{k,00} + b_{11}^{k-1,00} - 2b_{10}^{k,00}) - \lambda_1^k(b_{10}^{k,00} - b_{00}^{k,00})}{2\lambda_0^k}, \quad (14)$$

$$b_{10}^{k,20} := b_{20}^{k,20} + \frac{\lambda_2^k(b_{30}^{k,20} - b_{20}^{k,20}) - 3(b_{21}^{k,20} + b_{12}^{k-1,02} - 2b_{20}^{k,20})}{2\lambda_3^k} \quad (15)$$

and joining the three curve segments C^2 (cf. large squares in Fig. 2 *right*)

$$\text{for } \mu = 0, 1 : \quad b_{30}^{k,\mu 0} := (b_{20}^{k,\mu 0} + b_{10}^{k,\mu+1,0})/2, \quad (16)$$

$$b_{10}^{k,10} := \frac{4}{3}b_{20}^{k,00} - \frac{1}{3}b_{20}^{k,20} + \frac{2}{3}b_{10}^{k,20} - \frac{2}{3}b_{10}^{k,00}, \quad (17)$$

$$b_{20}^{k,10} := \frac{2}{3}b_{20}^{k,00} - \frac{2}{3}b_{20}^{k,20} + \frac{4}{3}b_{10}^{k,20} - \frac{1}{3}b_{10}^{k,00}. \quad (18)$$

2b. [boundary $n^0 \neq 4 = n^1$**]** If $n^0 \neq 4 = n^1$, Lemma 2 in the Appendix shows that we cannot choose all α_i^k to be linear. (If $\lambda_1^k = 0$ in Lemma 2 then the dependence appears at the next 4-valent crossing $b_{00}^{k,10}$.) We set $\lambda := 2\cos(\frac{2\pi}{n^0})$ and

$$\alpha_0^k(u) := \lambda^k(1-u) + \frac{\lambda^k}{2}u, \quad \alpha_1^k(u) := \frac{\lambda^k}{2}(1-u)^2, \quad \alpha_2^k(u) = 0. \quad (19)$$

Then $b_{20}^{k,00}$ is defined by (5) and $b_{10}^{k,20}$ by subdividing the cubic boundary of g:

$$b_{10}^{k,20} := -\frac{2}{9}b_{00}^{k,00} + \frac{1}{3}b_{10}^{k,00} + \frac{4}{3}b_{20}^{k,20} - \frac{4}{9}b_{30}^{k,20}. \quad (20)$$

From the remaining six G^1 constraints across the macro-patch boundary,

$$3(v_2^{k,0} + w_2^{k,0}) = \lambda^k u_2^{k,0} + \lambda^k u_1^{k,0}, \qquad 3(v_2^{k,1} + w_2^{k,1}) = 0 \tag{21}$$

$$3(v_3^{k,0} + w_3^{k,0}) = \frac{3}{2}\lambda^k(b_{10}^{k,10} - b_{00}^{k,10}), \qquad 3(v_3^{k,1} + w_3^{k,1}) = 0 \tag{22}$$

$$9(v_1^{k,1} + w_1^{k,1}) = \frac{3}{2}\lambda^k(b_{30}^{k,10} - b_{20}^{k,10}), \qquad 9(v_1^{k,2} + w_1^{k,2}) = 0. \tag{23}$$

the two listed as (22) are linked to the remaining four by the requirement that the macro-patches be internally C^1:

$$\text{for } \mu = 0,1: \quad b_{30}^{k,\mu 0} = (b_{20}^{k,\mu 0} + b_{10}^{\mu+1,20})/2 \tag{24}$$

$$v_2^{k,\mu} + v_1^{k,\mu+1} = 2v_3^{k,\mu}, \quad w_2^{k,\mu} + w_1^{k,\mu+1} = 2w_3^{k,\mu}. \tag{25}$$

Thus, by adding 3 times (21) to (23) and subtracting 6 times (22) and observing (25), we eliminate the left hand sides and obtain one constraint purely in the boundary coefficients multiplied by $\lambda^k \neq 0$. A second constraint arises since $\alpha_1^k(u)$ being quadratic implies that the middle segment $\mathbf{b}^{k,10}(u,0)$ is quadratic, i.e. its third derivative is zero:

$$b_{30}^{k,10} - 3b_{20}^{k,10} + 3b_{10}^{k,10} - b_{00}^{k,10} = 0. \tag{26}$$

Both constraints are enforced by setting

$$b_{10}^{k,10} := \frac{41}{25}b_{20}^{k,00} + \frac{4}{25}b_{10}^{k,20} - \frac{4}{5}b_{10}^{k,00}, \quad b_{20}^{k,10} := \frac{36}{25}b_{20}^{k,00} + \frac{9}{25}b_{10}^{k,20} - \frac{4}{5}b_{10}^{k,00}.$$

Together with (24), this fixes the macro-patch boundary (Fig. 2, *right*).

3. [First interior layer, G^1 constraints] Enforcing the remaining four G^1 constraints in terms of the red coefficients in Fig. 3 is straightforward and our symmetric solution is written out below.
3a. $[n^0 \neq 4 \neq n^1]$

$$h_{1\mu} := b_{20}^{k,\mu 0} + \frac{\lambda_\mu^k u_2^{k,\mu} + 2\lambda_{\mu+1}^k u_1^{k,\mu}}{6}, \quad h_{2\mu} := b_{10}^{k,\mu 0} + \frac{2\lambda_\mu^k u_1^{k,\mu} + \lambda_{\mu+1}^k u_0^{k,\mu}}{6}$$

$$\mu = 0,1: \quad b_{21}^{k,\mu 0} := h_{1\mu} + \frac{1}{2}(\tilde{b}_{21}^{k,00} - \tilde{b}_{12}^{k-1,00}), \quad b_{12}^{k-1,0\mu} := h_{1\mu} + \frac{1}{2}(\tilde{b}_{12}^{k-1,00} - \tilde{b}_{21}^{k,00})$$

$$\mu = 1,2: \quad b_{11}^{k,\mu 0} := h_{2\mu} + \frac{1}{2}(\tilde{b}_{11}^{k,10} - \tilde{b}_{11}^{k-1,01}), \quad b_{11}^{k-1,0\mu} := h_{2\mu} + \frac{1}{2}(\tilde{b}_{11}^{k-1,01} - \tilde{b}_{11}^{k,10}).$$

3b.$[n^0 \neq 4 = n^1]$ $h_1 := b_{20}^{k,00} + \dfrac{\lambda_0^k u_2^{k,0} + \lambda_0^k u_1^{k,0}}{6}, \quad h_2 : = b_{10}^{k,10} + \dfrac{\lambda_0^k u_2^{k,1}}{12}$

$$b_{21}^{k,00} := h_1 + \frac{1}{2}(\tilde{b}_{21}^{k,00} - \tilde{b}_{12}^{k-1,00}), \quad b_{12}^{k-1,0j} : = h_1 + \frac{1}{2}(\tilde{b}_{12}^{k-1,00} - \tilde{b}_{21}^{k,00})$$

$$b_{11}^{k,10} := h_2 + \frac{1}{2}(\tilde{b}_{11}^{k,10} - \tilde{b}_{11}^{k-1,01}), \quad b_{11}^{k-1,01} : = h_2 + \frac{1}{2}(\tilde{b}_{11}^{k-1,01} - \tilde{b}_{11}^{k,10})$$

$$b_{21}^{k,10} := b_{20}^{k,10} + \frac{1}{2}(\tilde{b}_{21}^{k,10} - \tilde{b}_{12}^{k-1,01}), \quad b_{12}^{k-1,01} : = b_{20}^{k,10} + \frac{1}{2}(\tilde{b}_{12}^{k-1,01} - \tilde{b}_{21}^{k,10})$$

$$b_{11}^{k,20} := b_{10}^{k,20} + \frac{1}{2}(\tilde{b}_{11}^{k,20} - \tilde{b}_{11}^{k-1,02}), \quad b_{11}^{k-1,02} : = b_{10}^{k,20} + \frac{1}{2}(\tilde{b}_{11}^{k-1,02} - \tilde{b}_{11}^{k,20})$$

where the coefficients $\tilde{b}_{21}^{k,00}$,$\tilde{b}_{12}^{k-1,00}$, $\tilde{b}_{11}^{k,10}$ and $\tilde{b}_{11}^{k-1,01}$ are obtained by subdividing the cubic Hermite interpolate to the transversal derivatives at the endpoints into three parts:

$$\begin{aligned}
\tilde{b}_{21}^{k,00} &:= -\frac{4}{9}b_{01}^{k,00} + \frac{4}{3}b_{11}^{k,00} + \frac{1}{3}b_{21}^{k,20} - \frac{2}{9}b_{31}^{k,20},\\
\tilde{b}_{12}^{k-1,00} &:= -\frac{4}{9}b_{10}^{k-1,00} + \frac{4}{3}b_{11}^{k-1,00} + \frac{1}{3}b_{12}^{k-1,02} - \frac{2}{9}b_{13}^{k-1,02},\\
\tilde{b}_{11}^{k,10} &:= -\frac{20}{27}b_{01}^{k,00} + \frac{4}{3}b_{11}^{k,00} + b_{21}^{k,20} - \frac{16}{27}b_{31}^{k,20},\\
\tilde{b}_{11}^{k-1,01} &:= -\frac{20}{27}b_{10}^{k-1,00} + \frac{4}{3}b_{11}^{k-1,00} + b_{12}^{k-1,02} - \frac{16}{27}b_{13}^{k-1,02}.
\end{aligned}$$

The coefficients $\tilde{b}_{21}^{k,10}$,$\tilde{b}_{12}^{k-1,01}$,$\tilde{b}_{11}^{k,20}$ and $\tilde{b}_{11}^{k-1,02}$ are defined analogously.

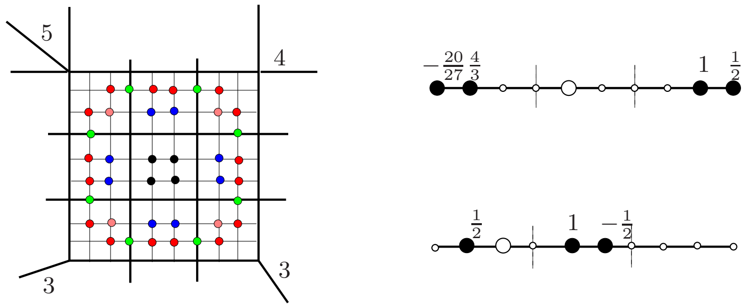

Fig. 3. *left*: Once the red BB-coefficients $b_{21}^{k,00}, b_{11}^{k,10}, b_{21}^{k,10}, b_{11}^{k,20}$ of the first interior layer are set, the green coefficients are C^1 averages of their two red neighbor points. Blue, pink and black coefficients are inner points computed by the rules on the *right*: (*top*) by subdivision and (*bottom*) so that the pieces join C^2: the coefficient indicated by the large ∘ is a linear combination, with weights displayed, of the coefficients shown as •.

4. [macro-patch Interior] At the center (four black disks in Fig. 3, *left*) the coefficients are computed according to 3,*right-top*,

$$b_{11}^{k,11} := \frac{(-\frac{20}{27}b_{01}^{k,01} + \frac{4}{3}b_{11}^{k,01} + b_{21}^{k,21} + \frac{1}{2}b_{31}^{k,21}) + (-\frac{20}{27}b_{10}^{k,10} + \frac{4}{3}b_{11}^{k,10} + b_{12}^{k,12} + \frac{1}{2}b_{13}^{k,12})}{2},$$

and symmetrically for the other three corners. Coefficients marked in blue and pink are defined by the rules of Fig. 3 *right-bottom* and the remaining coefficients on the internal boundaries are the C^1 average of their neighbors, e.g. $b_{10}^{k,11} := (b_{11}^{k,11} + b_{21}^{k,10})/2$ so that the patches join C^1 everywhere and C^2 with the central subpatch.

This completes the local construction of C^1 3×3 macro-patches, one per input quad and so that neighbor macro-patches join G^1. Before we show examples, we discuss why we did not choose $m = 2$.

4 Can $m = 2$ provide a construction?

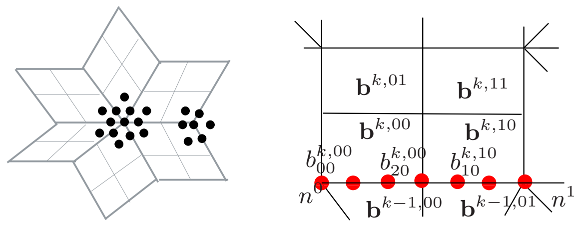

Fig. 4. *left*: Coefficients initialized according to (28). *right*: Indexing.

We show that an analogous construction is not possible for a 2×2 macro-patch. Since the degree of $\partial_2 \mathbf{b}^{k,\mu 0}(u,0)$ and $\partial_1 \mathbf{b}^{k-1,0\mu}(0,u)$ is 3, choosing α_μ^k to be quadratic implies that $\partial_1 \mathbf{b}^{k,\mu 0}(u,0)$ must be linear, i.e. each boundary curve segment is piecewise quadratic. If both n^0 and n^1 are even and not 4, then the vertex enclosure constraint (see Section 2) implies that the shared endpoint of the two quadratic segments is determined independently from both sides – so a local construction is not possible just as in the case $m = 1$. We therefore choose $\alpha_\mu^k(u) := \lambda_\mu^k(1-u) + \lambda_{\mu+1}^k u, \mu = 0, 1$ with the *unbiased* choice (we do not prefer one sector over another)

$$\lambda_0^k := 2\cos\frac{2\pi}{n^0}, \quad \lambda_2^k := -2\cos\frac{2\pi}{n^1}. \tag{27}$$

As in Section 3 (11), we enforce $(4)_{\mu=0}$ and $(7)_{\mu=1}$ of the eight G^1 continuity constraints by initializing position and tangents (black filled circles in Fig. 4 *left*) by subdividing g:

$$b_{00}^{k,00} := g_{00}^k, \quad b_{10}^{k,00} := \frac{g_{00}^k + g_{10}^k}{2}, \quad b_{11}^{k,00} := \frac{g_{00}^k + g_{10}^k + g_{01}^k + g_{11}^k}{4}. \tag{28}$$

Lemma 1. *If each macro-patch is parametrically C^1, α_0^k and α_1^k are linear with λ_0^k and λ_2^k unbiased then the G^1 constraints can only be enforced for all local initialization of $b_{00}^{k,00}$, $b_{10}^{k,00}$, $b_{11}^{k,00}$ if $n^0 = n^1$.*

Proof. Due to the internal C^1 constraints, adding $(6)_{\mu=0}$ and $(5)_{\mu=1}$ and subtracting six times $(7)_{\mu=0}$ yields $3(v_2^{k,0} + w_2^{k,0}) + 3(v_1^{k,1} + w_1^{k,1}) - 6(v_3^{k,0} + w_3^{k,0}) = 0$ and therefore the right hands sides satisfy

$$\lambda_0^k u_2^{k,0} + 2\lambda_1^k u_1^{k,0} + 2\lambda_1^k u_1^{k,1} + \lambda_2^k u_0^{k,1} = 6\lambda_1^k u_2^{k,0}. \tag{29}$$

That is, for an internally C^1 macro-patch, G^1 constraints *across* the macro-patch's boundary imply a constraint exclusively in terms of $u_i^{k,\mu}$, i.e. derivatives *along* the boundary! Since initialization fixes the local position, tangent and twist coefficients at each vertex, $(5)_{\mu=0}$ determines $b_{20}^{k,00}$ and $(6)_{\mu=1}$ determines $b_{10}^{k,10}$; and C^1 continuity determines $b_{30}^{k,00} := (b_{20}^{k,00} + b_{10}^{k,10})/2$. Thus all *vectors* of (29) are fixed and the remaining single free *scalar* λ_1^k cannot always enforce (29). But if $n^0 = n^1$ then $\lambda_0^k = -\lambda_2^k$ and $u_2^{k,0} = u_0^{k,1}$; and $\lambda_1^k = 0$ solves (29).

5 Conclusion

Curvature distribution and highlight lines on the models of Fig. 5 illustrate the geometric soundness of the $m = 3$ macro-patch construction. Choosing α_1^k and hence the middle boundary curve segment to be quadratic, avoids the PCCM shape problem which is due to α_0^k and hence the first segment being quadratic.

Conversely, Section 4 suggests that there is no obvious construction for $m = 2$; whether a more complex Ansatz can yield a localized construction for $m = 2$ remains the subject of research.

References

[CC78] E. Catmull and J. Clark. Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer Aided Design*, 10:350–355, 1978.

[LS08] Charles Loop and Scott Schaefer. Approximating Catmull-Clark subdivision surfaces with bicubic patches. *ACM Trans. Graph.*, 27(1):1–11, 2008.

[MYP08] A. Myles, Y. Yeo, and J. Peters. GPU conversion of quad meshes to smooth surfaces. In D. Manocha et al, editor, *ACM SPM*, pages 321–326, 2008.

[NYM+08] T. Ni, Y. Yeo, A. Myles, V. Goel, and J. Peters. GPU smoothing of quad meshes. In M. Spagnuolo et al, editor, *IEEE SMI*, pages 3–10, 2008.

[Pet00] Jörg Peters. Patching Catmull-Clark meshes. In K. Akeley, editor, *ACM Siggraph*, pages 255–258, 2000.

[Pet01] Jörg Peters. Modifications of PCCM. TR 001, Dept CISE, U Fl, 2001.

[Pet02] J. Peters. Geometric continuity. In *Handbook of Computer Aided Geometric Design*, pages 193–229. Elsevier, 2002.

Appendix: linear parameterization and valence 4

Lemma 2. *If $k = 4$ curves meet at a vertex without singularity and α_i^{k0} is linear with $\lambda_1^k := \ell \cos \frac{2\pi}{n^k}$, for fixed scalar $\ell > 0$ and valence n^k, then the G^1 constraints (3) can only be enforced if $n^k = n^{k+2}$ for $k = 0, 1$.*

Proof. For $k = 1, 2, 3, 4$, $\lambda_0^k = 0$ and therefore $\alpha_i^{k0} := \lambda_1^k u$. Equation $(5)_{\mu=0}$ then simplifies to

$$3(b_{11}^{k,00} + b_{11}^{k-1,00}) = 6b_{10}^{k,00} + \lambda_1^k u_0^{k,0}, k = 1 \dots 4.$$

Since $\sum_{k=1}^{4}(-1)^k(b_{11}^{k,00} + b_{11}^{k-1,00}) = 0$ and $\sum_{k=1}^{4}(-1)^k b_{00}^{k,00} = 0$,

$$0 = \sum_{k=1}^{4}(-1)^k 6(b_{10}^{k,00} - b_{00}^{k,00}) + \lambda_1^k u_0^{k,0} = \sum_{k=1}^{4}(-1)^k(6 + \lambda_1^k)u_0^{k,0} \qquad (30)$$

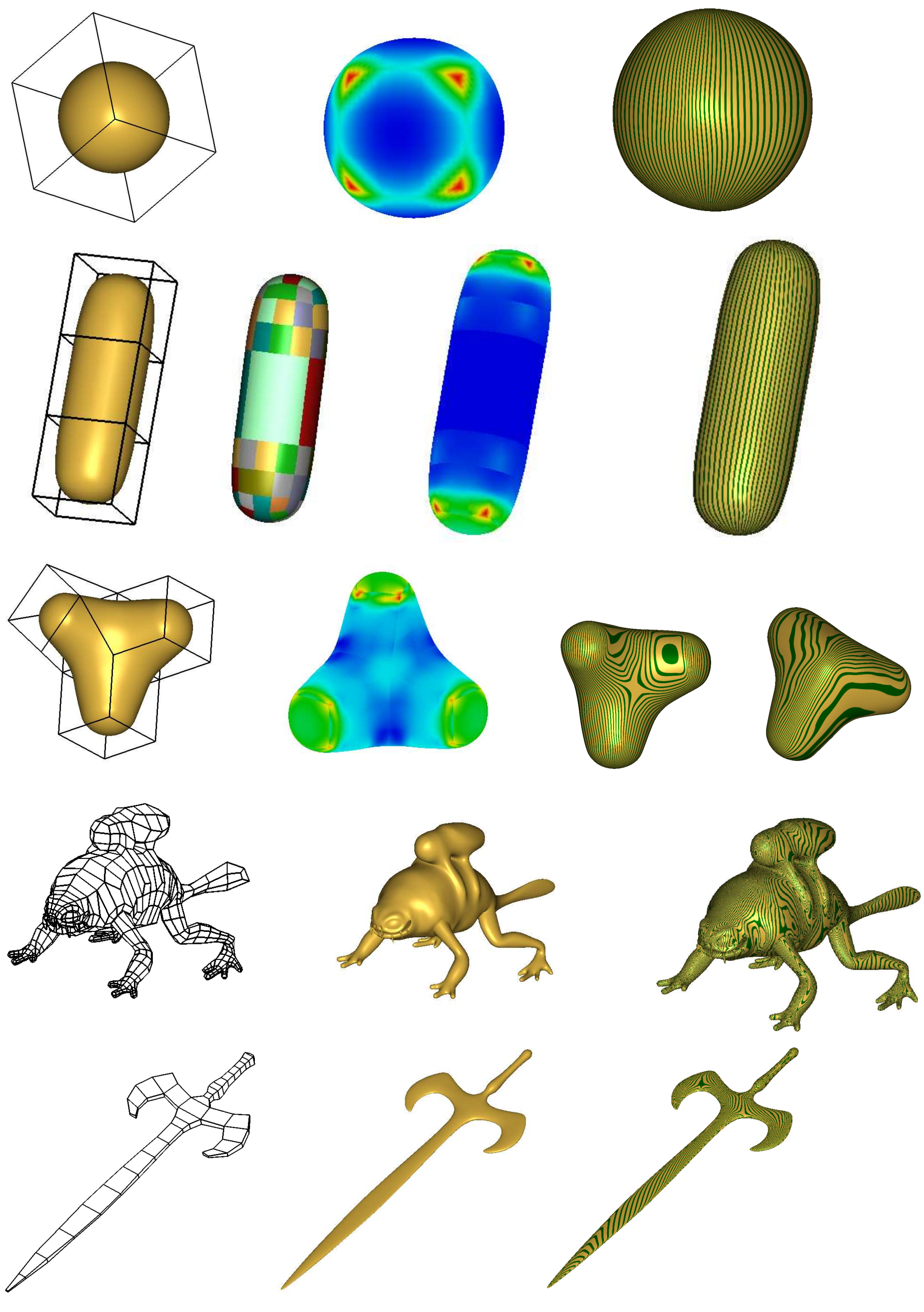

Fig. 5. 3×3 **macro-patch construction.** *left*: Quad mesh and surface; *middle*(top four): Gauss curvature distribution on the surface, *right*: Highlight lines.

implying $\lambda_1^k = \lambda_1^{k+2}$ for $k = 0, 2$ since $u_0^{k,0} = -u_0^{k+2,0}$. Therefore $n^k = n^{k+2}$ must hold.

Acknowledgement This work was supported by the National Science Foundation Grant 0728797.