

Curvature of Approximating Curve Subdivision Schemes

Kęstutis Karčiauskas¹ and Jörg Peters²

¹Vilnius University, ²University of Florida

Abstract. The promise of modeling by subdivision is to have simple rules that avoid cumbersome stitching-together of pieces. However, already in one variable, exactly reproducing a variety of basic shapes, such as conics and spirals, leads to non-stationary rules that are no longer as simple; and combining these pieces within the same curve by one set of rules is challenging. Moreover, basis functions, that allow reading off smoothness and computing curvature, are typically not available. Mimicking subdivision of splines with non-uniform knots allows us to combine the basic shapes. And to analyze non-uniform subdivision in general, the literature proposes interpolating the sequence of subdivision control points by circles. This defines a notion of discrete curvature for interpolatory subdivision. However, we show that this discrete curvature generically yields misleading information for non-interpolatory subdivision and typically does not converge, not even for non-uniform spline subdivision. Analyzing the causes yields three general approaches for solving or at least mitigating the problem: equalizing parameterizations, sampling subsequences and a new skip-interpolating subdivision approach.

Keywords: non-uniform subdivision, non-stationary subdivision, geometric continuity, curvature, splines, shape

1 Introduction

A major selling point of subdivision algorithms has been their conceptual simplicity in smoothly connecting curve or surface regions by refinement with simple rules. A main task of curve modeling in product design is to reproduce segments of a variety of basic shapes, such as conics, spirals and clothoids *exactly*, and to transition smoothly between them. Since the standard uniform, polynomial subdivision algorithms cannot reproduce these basic shapes, a number of non-stationary curve subdivision algorithms have recently been devised to reproduce, in particular, circles and ellipses [15, 20, 3, 6, 18, 4, 7, 2, 19]. However, the introduction of parameter-dependent subdivision means that explicit basis functions for the control points are no longer easily available, removing a reliable technique to compute curvature. Already establishing smoothness or curvature continuity for non-stationary (or similar non-linear schemes [1, 21, 10, 12, 8]) is a challenge as general techniques, such as [5, p18] and [22], do not apply.

Furthermore, even when C^2 smoothness can be proven, this may be meaningless in practice without a reliable technique to compute curvature. Following e.g. Sabin et al. [20, 2], we may therefore attempt to compute a *discrete curvature*, as the reciprocal

of a radius of the circle passing through the point and its two neighbours in the refined polygon. For interpolating curvature continuous schemes this measure must, by definition of a curvature continuous scheme, converge to the proper curvature. Also when refining the control polygon of uniform C^2 splines, the discrete curvature represents the curvature correctly. However, for the standard midpoint subdivision of non-uniform C^2 cubic splines, the measure diverges (cf. Fig. 4). In fact, divergence of the measure is typical as Lemma 1 shows.

This paper therefore discusses three techniques for estimating curvature from control points suitable for approximating subdivision: equalizing parameterization, sampling subsequences and skip-interpolation. To illustrate them, we work with non-uniform subdivision algorithms, a concept that is outlined in Section 2 and made concrete by a simple quadratic subdivision that can reproduce conics, in Section 6. An approach for reproducing more general basic shapes in one framework, is given in [14].

Concretely, for our focus on curvature from control points, we illustrate divergence of the discrete curvature for two non-uniform, polynomial spline subdivision schemes of degree 3, respectively degree 4. We also introduce the concept of skip-interpolation: every second control point is interpolated by the refined polygon. This allows measuring curvature as discrete curvature from the interpolating points while preserving the typically better shape of non-interpolating, approximating subdivision.

Overview. In Section 2, we explain the need for subdivision mimicking splines with non-uniform knots. In Section 3, we review non-uniform subdivision of degree 3 splines. In Section 4 we use this subdivision to analyze discrete curvature, prove divergence and to test strategies for obtaining predictive numbers from discrete curvature: equalizing parameters and subsampling. In Section 5, we derive a degree 4 skip-interpolating subdivision algorithm. In Section 6, we complete the exposition with an example of a non-uniform quadratic subdivision capable of reproducing various conics in one curve.

2 Non-uniform Subdivision

Non-uniform subdivision mimics the subdivision of splines with non-uniform knots. Fig. 1 and 2 illustrate the challenges that motivate non-uniform subdivision. Of the subdivision schemes listed in the introduction, Morin et al.'s scheme [15] is the only one that can reproduce more than one primitive in one curve. But, as Fig. 1 shows, even for this scheme, the underlying, inherently uniform spacing makes perturbations non-local. Fig. 2 (b) shows that the approach of [15] also unable to reproduce a circle on input of unevenly distributed samples. To prove the second claim, we note that for a regular polygon on the unit circle with opening angle α , the approach of [15] produces a circle of radius $\frac{\sin(\alpha)}{\alpha}$. If the designer's spacing of samples for perturbation is to be honored in the control polygon, the control points' distance to the circle center must be scaled so that they can partially reproduce the circle (*red* in Fig. 2(b)). In the transition, however, the reproduction is lost.

In order to reproduce different conics in one framework, splines use non-uniform knot sequences $\{t_i\}$. The rational cubic G^2 constructions in [14] combine curvature

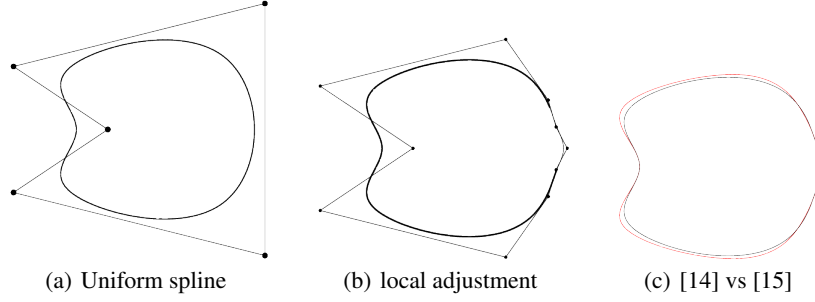


Fig. 1. In design, **local refinement and shape adjustment** naturally requires a switch from (a) uniform to (b) non-uniform spacing. The approach of [14] can preserve segments, shown as *thick segments*, of the original curve while (c) the superimposed *red* curve generated according to [15] deviates everywhere from the original curve, i.e. is less local.

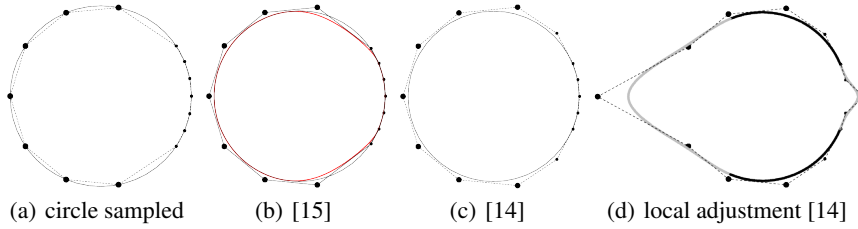


Fig. 2. **Non-uniform spacing to support local adjustments.** (a) Unequally-spaced designer samples on the circle in anticipation of local modification: small disks correspond to an opening angle of $\pi/16$, large ones to $3\pi/16$. (b) Circle in *black*, [15] in *red*. (c) Control net and exact circle generated by [14]. (d) Local modification using [14]: thick segments remain exactly on the circle.

continuity with exact reproduction of different basic shapes by simulating such non-uniformity of the knot sequence by a *non-uniform parameterization*. The basic approach can already be illustrated by a non-uniform quadratic subdivision mimicking rational G^1 splines. The details of such a scheme are given in Section 6. Here we outline the main idea. Let f_i and f_{i+1} be adjacent pieces of a C^1 spline with non-uniform knots, but with their domains re-parameterized to the unit interval $[0, 1]$. Then f_i and f_{i+1} join with geometric continuity (see e.g. [13]):

$$f'_{i+1}(0) = \beta_i f'_i(1), \quad \beta_i := \frac{\Delta_i}{\Delta_{i-1}}, \quad \Delta_i := t_{i+1} - t_i. \quad (1)$$

By refining the control structure of such splines, we arrive at non-uniform subdivision schemes [14] that are capable of combining primitives as shown in Fig. 1 and 2.

3 Non-uniform Subdivision of Cubic C^2 Splines

We now consider subdivision of non-uniform C^2 splines. Since we focus on measuring curvature from control polygons, we may restrict attention to the polynomial scheme,

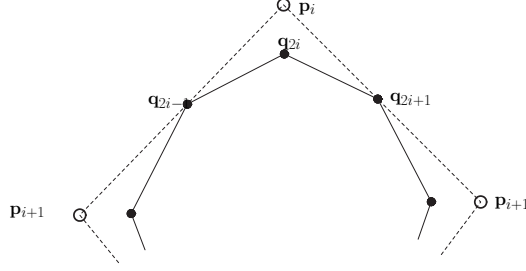


Fig. 3. C^2 cubic control net refinement.

rather than the more complex rational construction of [14] needed for reproducing basic shapes. Many draft shapes can be designed using B-splines with uniform knot sequence. However, subsequent *local* modifications lead to non-uniform splines as illustrated in Fig. 1. Fig. 3 illustrates the subdivision of such a non-uniform cubic B-spline curve. Suppose the cubic C^2 spline has the

control polygon $\{\mathbf{p}_i\}$, knot sequence $\{t_i\}$ and $\beta_i := \frac{\Delta_i}{\Delta_{i-1}}$, $\Delta_i := t_{i+1} - t_i$.

If we set $\bar{t}_{2i} := t_i$ and insert new knots at $\bar{t}_{2i+1} := (1 - e_i)t_i + e_it_{i+1}$, with ratio $0 < e_i < 1$ then the spline's refinement rules yield a new polygon $\{\mathbf{q}_i\}$, and constants,

$$\mathbf{q}_{2i+1} := (1 - \eta_i)\mathbf{p}_i + \eta_i\mathbf{p}_{i+1}, \quad (2)$$

$$\mathbf{q}_{2i} := \mu_i\mathbf{q}_{2i-1} + (1 - \mu_i - \nu_i)\mathbf{p}_i + \nu_i\mathbf{q}_{2i+1} \quad (3)$$

$$\eta_i := \frac{1 + e_i\beta_i}{1 + \beta_i + \beta_i\beta_{i+1}}, \quad \mu_i := \frac{\beta_i(1 - e_i)}{1 + \beta_i}, \quad \nu_i := \frac{e_{i-1}}{1 + \beta_i}, \quad (4)$$

$$\beta_{2i}^{new} := \frac{e_i}{1 - e_{i-1}}\beta_i, \quad \beta_{2i+1}^{new} := \frac{1 - e_i}{e_i}. \quad (5)$$

Here, as in the previous section, we may interpret the terms β_i in (5) as constants of a linear reparameterization.

4 Discrete Curvature from Polygon Sequences

To be able to estimate curvature in the absence of explicit generating functions, we follow [20] in defining the *discrete curvature* to be the inverse radius of the circle interpolating three consecutive control points. The first experiment below demonstrates that discrete curvature does not converge when tracking the control polygon of a non-uniform cubic C^2 spline under midpoint subdivision. A similar failure of discrete curvature as estimator of curvature occurs for subdivision based on a quartic C^2 spline (Fig. 8). The particular setup of non-uniform cubic C^2 spline subdivision is helpful in that we can easily compute the true curvature of the limit curve of the limit curve.

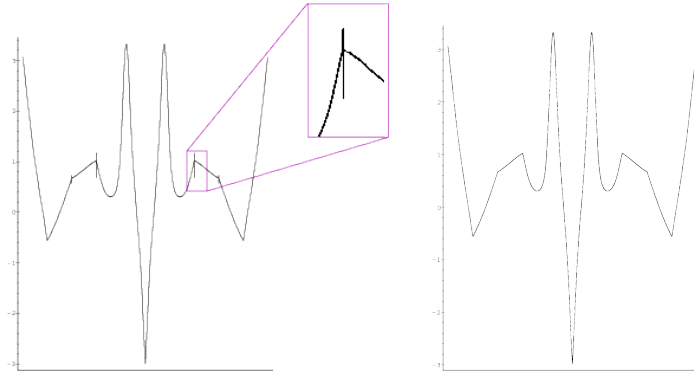


Fig. 4. Discrete curvature of Fig. 1,(b), plotted against the control point index i after $k = 10$ refinement steps of (left) subdivision for non-uniform subdivision (discontinuity enlarged), (right) when applying equalizing subdivision: no visible discontinuity.

Experiment 1 – divergence We locally refine the non-uniform C^2 cubic B-spline of Fig. 1, right, according to Section 3, by inserting knots with ratio $e_i = 1/2$. The discrete curvature after 10 refinement steps is displayed in Fig. 4, left. The spikes hint at a discontinuity at the junction where $\Delta_i := t_{i+1} - t_i$ changes. Indeed, if we pick a

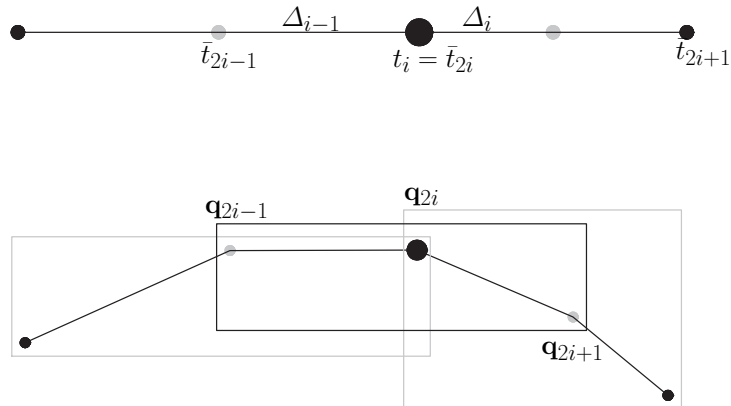


Fig. 5. Calculating the discrete curvature of a cubic C^2 spline from the control polygon.

knot \bar{t}_i , insert neighbors \bar{t}_{2i-1} and \bar{t}_{2i+1} using $e_{i-1} = e_i = 1/2$ so that the ratio $\beta_i = \Delta_i/\Delta_{i-1}$ remains unchanged under refinement, and set $\bar{t}_{2i} := \bar{t}_i$ as in Fig. 5, then there is no convergence. Let κ be the curvature of the spline at \bar{t}_i , denote by $\kappa_{2^k i-1}$ the discrete curvature to the left of \bar{t}_i after k steps, by $\kappa_{2^k i+1}$ the discrete curvature to the right of \bar{t}_i after k steps, and by $\kappa_{2^k i}$ the discrete curvature at \bar{t}_i . Since we have the underlying spline, we can compute the control polygon under midpoint subdivision and

find

$$\lim_{k \rightarrow \infty} \kappa_{2^{k_i-1}} = \frac{6}{\beta_i + 5} \kappa, \quad \lim_{k \rightarrow \infty} \kappa_{2^{k_i}} = \kappa, \quad \lim_{k \rightarrow \infty} \kappa_{2^{k_i+1}} = \frac{6\beta_i}{5\beta_i + 1} \kappa. \quad (6)$$

That is $\kappa_{2^{k_i}}$ converges to κ but its left and right neighbors converge to the same value only if $\beta_i = 1$. In other words, only for uniform B-splines does the refined control polygon provide correct information on the curvature of the limit curve. Fig. 8 shows an even more extreme behavior for the control polygon of a spline of degree 4, defined by (12) in Section 5: the discrete curvature jumps everywhere. Generally, without care, we can therefore not infer the curvature directly from the control net of an approximating subdivision.

Experiment 2 – equalization Since, according to the previous experiment, control polygons of uniform B-splines have useful discrete curvature expressions, we insert new knots \bar{t}_{2i+1} in the spirit of [20]:

$$e_i := \frac{\sqrt{1 + \beta_i}}{\sqrt{1 + \beta_i} + \sqrt{\beta_i} \sqrt{1 + \beta_{i+1}}}.$$

This moving ratio ‘equalizes’, i.e. we get, in the limit, a uniform knot sequence. Indeed, Fig. 4, *right*, shows that the spikes, hence discontinuities, in the discrete curvature disappear. The following lemma formally substantiates this observation.

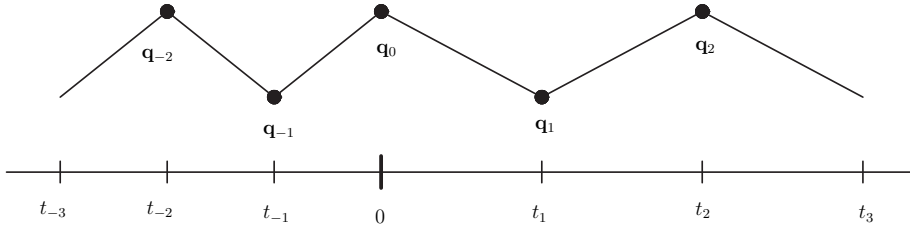


Fig. 6. Calculating the discrete curvature of a cubic G^2 spline.

Lemma 1 (Continuity of the discrete curvature). *The discrete curvature of midpoint subdivision of a cubic G^2 B-spline with non-zero curvature is continuous if and only if its knot sequence is uniform; it is always continuous under equalizing subdivision.*

Proof. Let f be defined over $[-1, 0]$ and g over $[0, \beta]$. Consider subintervals near the origin 0 as illustrated in Fig. 6. For $\epsilon \rightarrow 0$ under subdivision,

$$f : [t_{-3}, t_{-2}], [t_{-2}, t_{-1}], [t_{-1}, 0], \quad (7)$$

$$t_{-3} := -\epsilon(h_{-1} + h_{-2} + h_{-3}), \quad t_{-2} := -\epsilon(h_{-1} + h_{-2}), \quad t_{-1} := -\epsilon h_{-1},$$

$$g : [0, t_1], [t_1, t_2], [t_2, t_3], \quad (8)$$

$$t_1 := \epsilon h_1, \quad t_2 := \epsilon(h_1 + h_2), \quad t_3 := \epsilon(h_1 + h_2 + h_3).$$

For midpoint subdivision $h_{-3} = h_{-2} = h_{-1} = 1$, $h_1 = h_2 = h_3 = \beta$, while for equalizing subdivision, we have h_i depend on ϵ and $\lim_{\epsilon \rightarrow 0} h_i(\epsilon) = 1$. We compute the curvature κ of the spline at the origin 0; and we compute the discrete curvatures κ_i , $i = -1, 0, 1$, from triples $(\mathbf{q}_{i-1}, \mathbf{q}_i, \mathbf{q}_{i+1})$ of control points of the refined spline. Then with $h_i^\infty := \lim_{\epsilon \rightarrow 0} h_i(\epsilon)$ and $\rho_i := \lim_{\epsilon \rightarrow 0} \kappa_i / \kappa$,

$$\begin{aligned} \rho_{-1} &= \frac{3(h_{-2}^\infty + h_{-1}^\infty)}{h_{-3}^\infty + 2h_{-2}^\infty + 2h_{-1}^\infty + h_1^\infty}, \quad \rho_0 = \frac{3(h_{-1}^\infty + h_1^\infty)}{h_{-2}^\infty + 2h_{-1}^\infty + 2h_1^\infty + h_2^\infty}, \\ \rho_1 &= \frac{3(h_1^\infty + h_2^\infty)}{h_{-1}^\infty + 2h_1^\infty + 2h_2^\infty + h_3^\infty}. \end{aligned} \quad (9)$$

For midpoint subdivisions this yields as in (6)

$$\rho_{-1} = \frac{6}{5 + \beta}, \quad \rho_0 = 1, \quad \rho_1 = \frac{6\beta}{1 + 5\beta}, \quad (10)$$

while for equalizing subdivisions all $\rho_i = 1$ and the claim follows.

Note that the proof of Lemma 1 is based on the explicit knowledge of the underlying B-spline curve. A similar proof for a general non-stationary or non-linear subdivision scheme would be tricky.

Experiment 3 – subsequences Since equalization leads to complicated subdivision rules, we try another approach. We insert new knots midway as in Experiment 1, but determine the discrete curvature from subsequences. If we choose every 2³rd control point, the result for non-uniform C^2 subdivision looks gratifyingly like Fig. 4, *right*. However, proving convergence in a general setting is a subtle affair. By contrast, for the next approach it is straightforward.

5 Skip-interpolating Subdivision

In order to obtain a subdivision scheme with easily measurable curvature, we generalize splines of degree 4, but such that every second control point stays fixed. Then the discrete curvature of the interpolating subsequence represents the limit curvature as it would for an interpolating subdivision algorithm.

First, we review subdivision of quartic C^2 splines. If all β_i equal 1 and the domain intervals of the Bézier quartics are split at their center then the refinement rules for obtaining new control points $[\bar{\mathbf{q}}, \tilde{\mathbf{q}}]$ from $[\bar{\mathbf{p}}, \tilde{\mathbf{p}}]$ are (cf. Fig. 7(a))

$$\begin{aligned} \bar{\mathbf{q}}_{2i} &:= \frac{3}{16}\tilde{\mathbf{p}}_{i-1} + \frac{5}{8}\bar{\mathbf{p}}_i + \frac{3}{16}\tilde{\mathbf{p}}_i, & \bar{\mathbf{q}}_{2i+1} &:= \frac{1}{8}\bar{\mathbf{p}}_i + \frac{3}{4}\tilde{\mathbf{p}}_i + \frac{1}{8}\bar{\mathbf{p}}_{i+1}, \\ \tilde{\mathbf{q}}_{2i} &:= \frac{1}{16}\tilde{\mathbf{p}}_{i-1} + \frac{3}{8}\bar{\mathbf{p}}_i + \frac{9}{16}\tilde{\mathbf{p}}_i, & \tilde{\mathbf{q}}_{2i+1} &:= \frac{9}{16}\tilde{\mathbf{p}}_i + \frac{3}{8}\bar{\mathbf{p}}_{i+1} + \frac{1}{16}\tilde{\mathbf{p}}_{i+1}. \end{aligned} \quad (11)$$

In Fig. 8, the control points $\bar{\mathbf{p}}_i$ (black disks) and $\tilde{\mathbf{p}}_i$ (gray disks) are equally distributed on the unit circle. The discrete curvature of the refined polygons in Fig. 8(c,d) shows

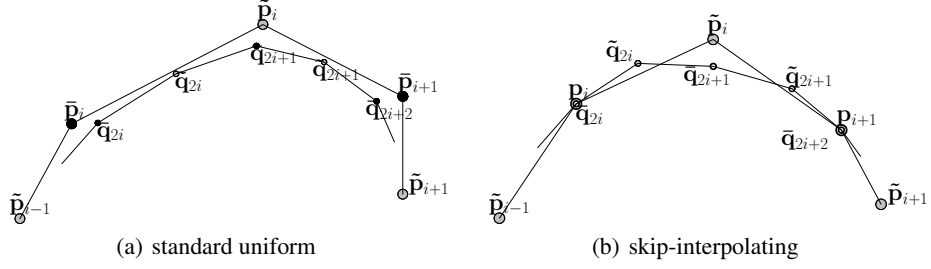


Fig. 7. Standard subdivision $[\bar{\mathbf{p}}, \bar{\mathbf{p}}] \rightarrow [\bar{\mathbf{q}}, \bar{\mathbf{q}}]$ and skip-interpolating subdivision $[\mathbf{p}, \bar{\mathbf{p}}] \rightarrow [\mathbf{q}, \bar{\mathbf{q}}]$.

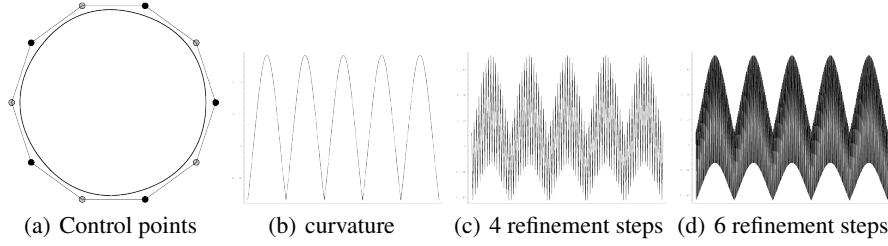


Fig. 8. (a) **Quartic C^2 spline**, (b) the spline's exact curvature plotted against the parameter on the abscissa and (c,d) the discrete curvature at $i/2^k$ during the k th refinement. The discrete curvature jumps, oscillating densely about the true curvature.

that there is no sense in tracing their densely oscillating discrete curvature to estimate the curvature of the limit C^2 spline.

Next, we consider the conversion of the B-spline control points $\bar{\mathbf{p}}_i$ of the polynomial spline to its Bernstein-Bézier coefficients $\mathbf{b}_{i,j}$ (see e.g. [11, 17] for the definitions of the B-spline form and the Bernstein-Bézier (BB) form). With the constants β_i representing the ratios of the non-uniform lengths of adjacent knot intervals,

$$\begin{aligned} \mathbf{b}_{i-1,3} &:= \frac{\beta_i \bar{\mathbf{p}}_{i-1} + \bar{\mathbf{p}}_i}{\beta_i + 1}, & \mathbf{b}_{i1} &:= \frac{\beta_i \bar{\mathbf{p}}_i + \bar{\mathbf{p}}_{i+1}}{\beta_i + 1}, \\ \mathbf{b}_{i,2} &:= \bar{\mathbf{p}}_i, & \mathbf{b}_{i-1,4} = \mathbf{b}_{i0} &:= \frac{\beta_i \mathbf{b}_{i-1,3} + \mathbf{b}_{i1}}{\beta_i + 1}. \end{aligned} \quad (12)$$

In particular, for $\beta_i = 1$, we get the familiar formulas $\mathbf{b}_{i-1,3} := \frac{1}{2} \bar{\mathbf{p}}_{i-1} + \frac{1}{2} \bar{\mathbf{p}}_i$, $\mathbf{b}_{i1} := \frac{1}{2} \bar{\mathbf{p}}_i + \frac{1}{2} \bar{\mathbf{p}}_{i+1}$, $\mathbf{b}_{i-1,4} = \mathbf{b}_{i0} := \frac{1}{2} \mathbf{b}_{i-1,3} + \frac{1}{2} \mathbf{b}_{i1}$.

To arrive at skip-interpolation, we define $\mathbf{p}_i := \mathbf{b}_{i,0}$ and observe in (12) and Fig. 9 that the relation between the sequence of point triples $\bar{\mathbf{p}}_{i-1}, \bar{\mathbf{p}}_i, \bar{\mathbf{p}}_{i+1}$ and $\bar{\mathbf{p}}_{i-1}, \mathbf{p}_i, \bar{\mathbf{p}}_{i+1}$ is linear. Therefore, we can equally well express a subdivision with the structure of (11) in terms of points $\bar{\mathbf{p}}_i, \mathbf{p}_i$ as shown in Fig. 7(b). The corresponding subdivision rules for

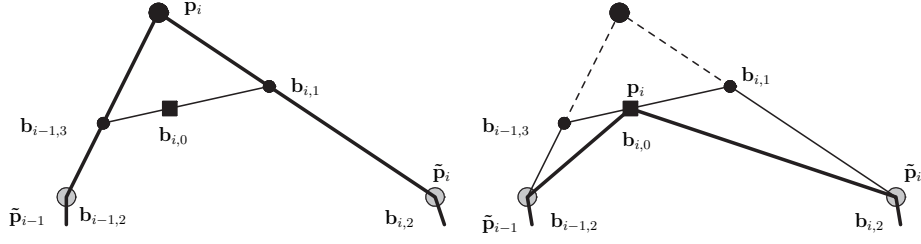


Fig. 9. Control points $\tilde{\mathbf{p}}_i$, $\tilde{\mathbf{p}}_i$, \mathbf{p}_i and BB-coefficients $\mathbf{b}_{i,k}$, $k = 0 \dots 4$. (left) standard C^2 B-spline to BB-form conversion (12) (right) Skip-interpolating conversion with points $\mathbf{p}_i = \mathbf{b}_{i,0}$ on the resulting curve.

deriving new points $[\mathbf{q}, \tilde{\mathbf{q}}]$ from $[\mathbf{p}, \tilde{\mathbf{p}}]$, generalized to account for varying β_i , are

$$\begin{aligned} \mathbf{q}_{2i} &:= \mathbf{p}_i, \\ \mathbf{q}_{2i+1} &:= a_0 \tilde{\mathbf{p}}_{i-1} + a_1 \mathbf{p}_i + a_2 \tilde{\mathbf{p}}_i + a_3 \mathbf{p}_{i+1} + a_4 \tilde{\mathbf{p}}_{i+1}, \\ \tilde{\mathbf{q}}_{2i} &:= e_0 \tilde{\mathbf{p}}_{i-1} + e_1 \mathbf{p}_i + e_2 \tilde{\mathbf{p}}_i, \\ \tilde{\mathbf{q}}_{2i+1} &:= \ell_0 \tilde{\mathbf{p}}_i + \ell_1 \mathbf{p}_{i+1} + \ell_2 \tilde{\mathbf{p}}_{i+1}. \end{aligned} \quad (13)$$

where

$$\begin{aligned} a_0 &:= -\frac{1}{8} \frac{\beta_i^2}{\beta_i + 1}, & a_1 &:= \frac{1}{8} \beta_i + \frac{3}{16}, & a_2 &:= 1 - a_0 - a_1 - a_3 - a_4, \\ e_0 &:= -\frac{1}{4} \frac{\beta_i^2}{\beta_i + 1}, & e_1 &:= \frac{1}{2} + \frac{1}{4} \beta_i, & e_2 &:= 1 - e_0 - e_1, \end{aligned}$$

and $[a_3, a_4, \ell_2, \ell_1, \ell_0]$ are obtained from $[a_1, a_0, e_0, e_1, e_2]$ by replacing $\beta_i \rightarrow 1/\beta_{i+1}$. This subdivision is called skip-interpolating, since every second point \mathbf{q}_{2j} of the control polygon ends up on the limit curve. The curve inherits curvature continuity in the limit from the underlying C^2 spline.

We note that in both (11) and (13), the discrete curvature of the combined control nets is meaningless: already the subpolygons, $\tilde{\mathbf{p}}_i$ or $\tilde{\mathbf{p}}_i$, yield wildly oscillating plots. But the subpolygon based on \mathbf{p}_i shows no spikes, as predicted.

6 Non-uniform Subdivision Based on a Rational Quadratic G^1 Curve Construction

As promised in Section 2, we present a non-uniform subdivision scheme based on a G^1 curve construction. This construction is useful in its own right and its derivation is similar but simpler than that in [14].

Given a control polygon \mathbf{p} and weights ω_i at the control points as in Fig. 10, *middle*, we derive the BB-control-points of a rational quadratic G^1 curve with BB-pieces f_i Fig. 10, *left*, such that numerator and denominator are in BB-form (Bernstein-Bézier

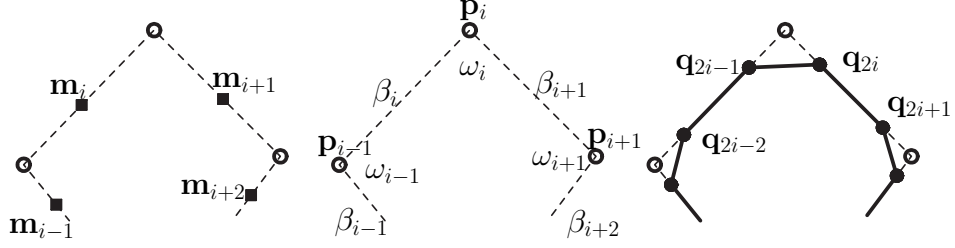


Fig. 10. Construction of Non-uniform Rational Quadratic Subdivision. (*left*) BB-control polygons with end points \mathbf{m}_i . (*middle*) Affine control polygon \mathbf{p} with weights ω and the G^1 constant β_i associated with edge $\mathbf{p}_{i-1}\mathbf{p}_i$. (*right*) Once-refined control polygon.

form):

$$f_i : u \mapsto \frac{\sum_{k=0}^2 w_k \mathbf{b}_k B_k^2(u)}{\sum_{k=0}^2 w_k B_k^2(u)} \quad B_k^n := \binom{n}{k} (1-u)^{n-k} u^k. \quad (14)$$

In terms of β_i , the BB-control-points of a rational quadratic f_i are

$$\begin{aligned} \mathbf{m}_i &:= (1 - \nu_i)\mathbf{p}_{i-1} + \nu_i\mathbf{p}_i, & \nu_i &:= \frac{\omega_i}{\beta_i\omega_{i-1} + \omega_i}, \\ \mathbf{b}_{i,0} &:= \mathbf{m}_i, \quad \mathbf{b}_{i,1} := \mathbf{p}_i, \quad \mathbf{b}_{i,2} := \mathbf{m}_{i+1}, & w_{i,0} = w_{i,2} &:= 1, \quad w_{i,1} := \omega_i. \end{aligned} \quad (15)$$

We now associate the weights ω_i with the coefficients $\mathbf{b}_{i,1}$ and the constants β_i with \mathbf{m}_i and hence with edges $\mathbf{p}_{i-1}\mathbf{p}_i$ (see Fig. 10, *middle*).

To subdivide, we split each quadratic curve segment by de Casteljau's algorithm at its center $u = 1/2$. Then we re-normalize each piece's rational weights so that the first and last are both 1:

$$\begin{aligned} \bar{w}_{i,k} &:= \frac{w_{i,k}}{w_{i,0}}, \quad k = 0, 1, 2; \\ w_{i,k}^{sym} &:= \bar{w}_{i,k} h_i^k, \quad k = 0, 1, 2; \quad h_i := \frac{1}{\sqrt{\bar{w}_{i,2}}}. \end{aligned} \quad (16)$$

Then

$$\beta_i^{sym} := h_{i-1} h_i \beta_i.$$

Subdivision generates the new control points \mathbf{q}_{2i-1} , \mathbf{q}_{2i} of the two subquadratics, a symmetrized weight ω_j per point and a symmetrized constant β_j per edge (cf. Fig. 10, *right*) by the following weight-dependent, hence non-stationary, and constant-dependent, hence non-uniform, subdivision algorithm.

Algorithm. [Non-uniform Rational Quadratic Subdivision]

Input: Control polygon \mathbf{p} , weights ω and constants β (see Fig. 10, *middle*).

Output: Control polygon \mathbf{q} , new weights ω and constants β (see Fig. 10, *right*).

The explicit refinement rules are

$$\mathbf{q}_{2i-1} := a_0^i \mathbf{p}_{i-1} + a_1^i \mathbf{p}_i, \quad \mathbf{q}_{2i} := b_0^i \mathbf{p}_i + b_1^i \mathbf{p}_{i+1}, \quad (17)$$

$$a_0^i := \frac{\omega_{i-1} \beta_i}{(\omega_{i-1} \beta_i + \omega_i)(1 + \omega_i)}, \quad b_1^i := \frac{\omega_{i+1}}{(\omega_i \beta_{i+1} + \omega_{i+1})(1 + \omega_i)}, \quad (18)$$

$$a_1^i := 1 - a_0^i, \quad b_0^i := 1 - b_1^i. \quad (19)$$

For the next refinement step, set $\dot{\omega}_i := \sqrt{(1 + \omega_i)/2}$ and redefine

$$\mathbf{p}_k := \mathbf{q}_k, \quad \omega_{2i-1} = \omega_{2i} := \dot{\omega}_i, \quad \beta_{2i-1} := \frac{\dot{\omega}_{i-1}}{\dot{\omega}_i} \beta_i, \quad \beta_{2i} := 1.$$

We call the scheme ‘non-uniform’ to emphasize its dependence on the constants β_i .

The smoothness of the subdivision algorithm is immediate since it corresponds to subdividing an underlying G^1 rational spline. If all weights ω_i are equal and all $\beta_i = 1$ in the algorithm then for $a_0^i = b_1^i = c := \frac{1}{2(1+\omega)}$,

$$\mathbf{q}_{2i-1} := c \mathbf{p}_{i-1} + (1 - c) \mathbf{p}_i, \quad \mathbf{q}_{2i} := (1 - c) \mathbf{p}_i + c \mathbf{p}_{i+1}, \quad \omega \leftarrow \sqrt{\frac{1 + \omega}{2}}, \quad (20)$$

i.e. the subdivision simplifies to a known uniform non-stationary C^1 circle-reproducing subdivision [9].

Indeed, the non-uniform subdivision can reproduce a number of conics in one framework. Fig. 11(a) illustrates the construction of a circle from an asymmetric circumscribed control polygon. We need only to set

$$w_i := \cos \frac{\alpha_i}{2}, \quad \beta_i := \frac{\sin \frac{\alpha_i}{2}}{\sin \frac{\alpha_{i-1}}{2}}, \quad (21)$$

where the α_i are the opening angles between consecutive points on the circle that are interpolated by the circumscribed control polygon. Fig. 11 (b) and (c) make the point that the subdivision can reproduce the ‘uniform’ non-stationary subdivision from [9] but additionally vary shape by varying β_i . The uniform subdivision, even though non-stationary, can only reproduce one primitive at a time, here an ellipse, making designs such as Fig. 1(b) cumbersome. By contrast, Non-uniform Rational Quadratic Subdivision adapts to two or more different prescribed conics by replicating the pieces as rational quadratic splines and converting them to control polygons of the subdivision algorithm. Fig. 11(d) shows (dotted) the circle as in (a), now reproduced from a control polygon that is a circumscribed triangle. For the solid-drawn variant, the weights of the top two control points are increased to locally yield hyperbolic pieces and the β_i has been adjusted to keep the bottom segment exactly on the circle. The curve in (e) corresponds to uniform $\beta_i = 1$.

7 Discussion

Our original goal was to address a shortcoming of recent subdivision algorithms for practical design: none locally reproduces several basic shapes within the same curve by

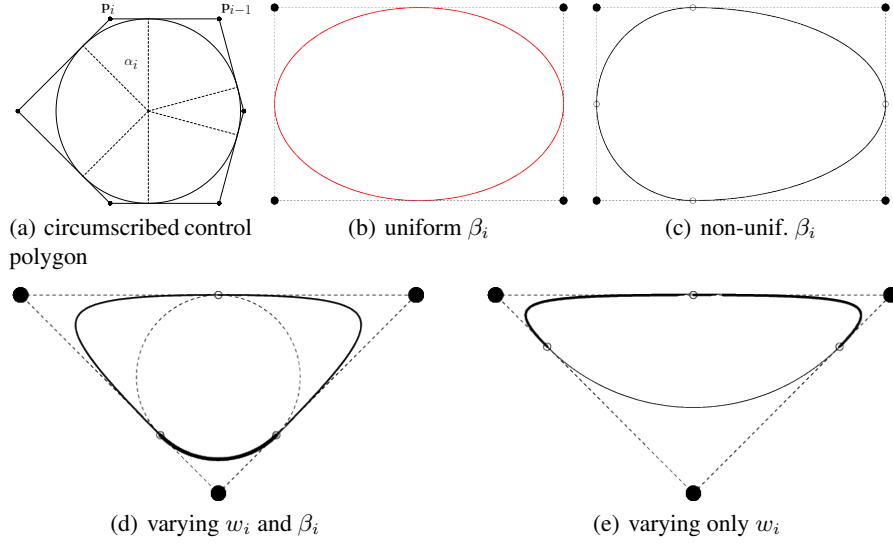


Fig. 11. Non-uniform Rational Quadratic Subdivision. (a) Circle-circumscribed control polygon with unequal opening angles α_i yields a circle (cf. (21)). (b) The result of setting $\beta_i = 1$ and $w_i := \cos \frac{\pi}{4}$ is identical to [9]. (c) The result of unequal β_i for $w_i := \cos \frac{\pi}{4}$. (d) Circle from a triangle (both *dotted*) and an alternative shape where the circle piece is preserved by varying the β_i corresponding to the bottom; the remainder is replaced by two hyperbolic pieces, abutting at the hollow circle marker, obtained by increasing the top two weights w_i . (e) Same as (d) but with all β_i identical.

one algorithm. In addressing this challenge algorithmically by non-uniform subdivision following the approach of [14], we noticed a second, related challenge, already present in non-uniform subdivision of polynomial splines: While control polygons often work well for extracting first-order information about curves, the experiments in Section 4 and Lemma 1 show that curvature of an approximating non-stationary subdivision is not easily gleaned from control polygons. In retrospect, this should not surprise since control nets without associated generating functions do not allow for a mathematical analysis of the resulting limit shape [16, Introduction].

Among the options that we explored in order to nevertheless generate useful curvature information in a practical way from control polygons, equalization leads to complicated subdivision rules; and selecting subsequences requires additional careful analysis. Skip-interpolation, on the other hand, is a simple technique to be able to read off curvature while still preserving the typically better shape of non-interpolating, approximating subdivision.

Acknowledgments. Work supported in part by NSF Grant CCF-0728797.

References

1. Aspert, N., Ebrahimi, T., Vandergheynst, P.: Non-linear subdivision using local spherical coordinates. *Computer Aided Geometric Design* 20(3), 165–187 (2003)
2. Augsdörfer, U.H., Dodgson, N.A., Sabin, M.A.: Variations on the four-point subdivision scheme. *Computer Aided Geometric Design* 27(1), 78–95 (2010)
3. Beccari, C., Casciola, G., Romani, L.: A non-stationary uniform tension controlled interpolating 4-point scheme reproducing conics. *Computer Aided Geometric Design* 24(1), 1–9 (2007)
4. Beccari, C., Casciola, G., Romani, L.: Shape controlled interpolatory ternary subdivision. *Applied Mathematics and Computation* 215(3), 916–927 (2009)
5. Cavaretta, A.S., Dahmen, W., Micchelli, C.A.: Stationary subdivision. *Mem. Amer. Math. Soc.* 93(453), vi+186 (1991)
6. Chalmovianský, P., Jüttler, B.: A non-linear circle-preserving subdivision scheme. *Adv. Comput. Math* 27(4), 375–400 (2007)
7. Conti, C., Romani, L.: Affine combination of B-spline subdivision masks and its non-stationary counterparts. *BIT Numerical Mathematics* 50(2), 269–299 (Jun 2010)
8. Deng, C., Wang, G.: Incenter subdivision scheme for curve interpolation. *Computer Aided Geometric Design* 27(1), 48 – 59 (2010)
9. Dyn, N., Levin, D.: Subdivision schemes in geometric modelling. *Acta Numerica* pp. 73–144 (2002)
10. Dyn, N., Floater, M.S., Hormann, K.: Four-point curve subdivision based on iterated chordal and centripetal parameterizations. *Computer Aided Geometric Design* 26(3), 279–286 (2009)
11. Farin, G.: *Curves and Surfaces for Computer-Aided Geometric Design*. Academic Press (1988)
12. Grohs, P.: Approximation order from stability for nonlinear subdivision schemes. *Journal of Approximation Theory* 162(5), 1085–1094 (2010)
13. Hohmeyer, M.E., Barsky, B.A.: Rational continuity: Parametric, geometric, and frenet frame continuity of rational curves. *ACM Transactions on Graphics* 8(4), 335–359 (Oct 1989)
14. Karčiauskas, K., Peters, J.: Rational G^2 splines. *Graphical Models* pp. 1–23 (in revision)
15. Morin, G., Warren, J.D., Weimer, H.: A subdivision scheme for surfaces of revolution. *Computer Aided Geometric Design* 18(5), 483–502 (2001)
16. Peters, J., Reif, U.: *Subdivision Surfaces, Geometry and Computing*, vol. 3. Springer-Verlag, New York (2008)
17. Prautzsch, H., Boehm, W., Paluszny, M.: *Bézier and B-spline techniques*. Springer Verlag (2002)
18. Romani, L.: From approximating subdivision schemes for exponential splines to high-performance interpolating algorithms. *Journal of Computational and Applied Mathematics* 224(1), 383 – 396 (2009)
19. Romani, L.: A circle-preserving C^2 hermite interpolatory subdivision scheme with tension control. *Computer Aided Geometric Design* 27(1), 36–47 (2010)
20. Sabin, M., Dodgson, N.: A circle-preserving variant of the four-point subdivision scheme. In: *Mathematical Methods for Curves and Surfaces: Troms 2004, Modern Methods in Mathematics*. pp. 275–286. Nashboro Press (2005)
21. Schaefer, S., Vouga, E., Goldman, R.: Nonlinear subdivision through nonlinear averaging. *Comput. Aided Geom. Des.* 25(3), 162–180 (2008)
22. Schaefer, S., Warren, J.: Exact evaluation of non-polynomial subdivision schemes at rational parameter values. In: *Pacific Graphics*. pp. 321–330 (2007)