Journal of Computational and Applied Mathematics 235 (2011) 4754–4769

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Polynomial-based non-uniform interpolatory subdivision with features control

Carolina Beccari [a], Giulio Casciola [a], Lucia Romani [b,*]

[a] Department of Mathematics, University of Bologna, P.zza di Porta San Donato 5, 40127 Bologna, Italy

[b] Department of Mathematics and Applications, University of Milano-Bicocca, Via R. Cozzi 53, 20125 Milano, Italy

ARTICLE INFO

Article history:
Received 25 December 2009
Received in revised form 3 August 2010

Keywords:
Subdivision
Interpolation
Non-uniform parameterization
Features control
Arbitrary point insertion

ABSTRACT

Starting from a well-known construction of polynomial-based interpolatory 4-point schemes, in this paper we present an original affine combination of quadratic polynomial samples that leads to a non-uniform 4-point scheme with edge parameters. This blending-type formulation is then further generalized to provide a powerful subdivision algorithm that combines the fairing curve of a non-uniform refinement with the advantages of a shape-controlled interpolation method and an arbitrary point insertion rule. The result is a non-uniform interpolatory 4-point scheme that is unique in combining a number of distinctive properties. In fact it generates visually-pleasing limit curves where special features ranging from cusps and flat edges to point/edge tension effects may be included without creating undesired undulations. Moreover such a scheme is capable of inserting new points at any positions of existing intervals, so that the most convenient parameter values may be chosen as well as the intervals for insertion.

Such a fully flexible curve scheme is a fundamental step towards the construction of high-quality interpolatory subdivision surfaces with features control.

1. Introduction

Given a mesh of points, linear subdivision is a recursive process by which, at each step, new vertices are inserted as linear combinations of the old ones. If the set of points computed at each refinement is retained at all the successive ones, the scheme is said to be interpolatory since the given vertices will be also part of the limit shape. Interpolatory subdivision is thus considered of great interest for applications because of its intuitive link with the starting mesh and in recent years interpolatory subdivision curves and surfaces have become an important alternative to their parametric counterpart.

This paper originates from our final intention of designing an interpolatory subdivision scheme for quadrilateral meshes with arbitrary topology and lays the foundations of this future work. To this aim, we are mainly concerned with the definition of refinement rules that fulfill three properties that any smooth interpolating scheme should possess to be of practical use in modeling or reconstruction applications: (i) non-uniform parameterization, (ii) accurate features control and (iii) the possibility of inserting points at arbitrary locations. The first of these requirements ensures a good quality of the limit surface; the second one provides a greater design flexibility by allowing the generation of a number of special shape effects, including cusps and creases; the last one is essential for tuning the scheme around extraordinary vertices.

As concerns the class of approximating subdivision schemes, there already exist several proposals in this direction, both in the univariate and in the bivariate contexts [1–4]. The leading idea supporting these kinds of proposals consists in a

* Corresponding author.
E-mail addresses: beccari@dm.unibo.it (C. Beccari), casciola@dm.unibo.it (G. Casciola), lucia.romani@unimib.it (L. Romani).

0377-0427/$ – see front matter
doi:10.1016/j.cam.2010.09.014

natural generalization of non-uniform spline theory. In particular, the features control derives from the splines capability of arbitrarily moving the knots in the knot-partition, up to include multiple knots, and the arbitrary point insertion trivially generalizes the standard midpoint knot-insertion. Differently, as concerns the family of interpolatory schemes, since there does not exist a compactly-supported refinable basis behind the derivation of this kind of refinement rules, features control must be handled separately from non-uniform parameterization. Moreover, if compared to the approximating theory, also the insertion of a new point corresponds to a quite different idea. In fact, whenever, at a certain refinement level, we insert a new point at a location different from the midpoint of an edge, we modify the shape of the limit curve/surface. Hence, in any situation, it is important to ensure that the overall quality of the interpolant is preserved.

In this framework, the present work is concerned with the definition of non-uniform, univariate, interpolatory subdivision schemes including both features control as well as the capability of inserting points at arbitrary locations. While for many years interpolatory subdivision schemes that appeared in the literature were limited to the uniform case [5,6], only recently an increasing number of papers dealing with non-uniform interpolatory refinements were presented [7,8] and the theoretical tools concerning their analysis were proposed [9,10]. The trend in these works is in exploiting the benefits of a properly chosen parameterization to reduce the undesired undulations that naturally appear when interpolating unevenly spaced data through the original Dubuc–Deslauriers's 4-point scheme. However, so far, none of the available proposals has concentrated on defining a variant of such a scheme that, besides incorporating the advantages of a non-uniform parameterization, is capable of generating visually-pleasing limit curves where special shape effects, like cusps and flat edges, can be included when desired.

The key idea behind the present work naturally emerges if we rewrite the existing polynomial-based 4-point refinements in the form of a parameter-depending blending between the two quadratic polynomials interpolating the two consecutive subsequences of three points having a pair of central vertices in common. Thanks to this formulation it is possible to work out the refinement rules of a non-uniform 4-point scheme with edge parameters possessing several interesting properties. First of all it comes out that such a scheme is the unique interpolatory 4-point scheme capable of producing a piecewise polynomial curve passing through the initial vertices, even if they are not its samples. In particular, piecewise quadratic polynomials that join C^0 continuously at the given points can be automatically produced by a suitable setting of the edge parameters, and piecewise C^1 quadratic polynomials can be further obtained after a simple preprocessing step. Moreover, by opportunely handling the edge parameters, it is also possible to include in the limit curve special shape effects ranging from cusps and flat edges to point/edge tension effects.

As we will show, by generalizing the cited construction, it is also possible to include in the scheme the capability of inserting new points at arbitrary locations, thus improving the visual quality of the limit curve, especially where the points generated at each step turn out to be irregularly distributed. We also emphasize that the possibility of an arbitrary point insertion is considered a fundamental step towards the generalization of a curve subdivision scheme to the bivariate case [1,2]. In fact, it allows us to create a locally uniform configuration of points around a selected vertex, which may be of crucial importance for tuning the surface in the vicinity of extraordinary points. While this issue will be analyzed in a forthcoming paper, in this work we lay the foundations for the generalization of our interpolatory curve scheme with edge parameters to surfaces on quadrilateral meshes. Although in the literature one may find proposals of approximating subdivision schemes with features control [3,4], to our knowledge there are no interpolatory schemes providing intuitive edge parameters for producing more flexible and various limit surfaces.

The remainder of the paper is organized as follows. Section 2 deals with non-uniform interpolatory 4-point subdivision schemes. In particular, Section 2.1 contains the main basic notions and the needed background, while Section 2.2 presents a blending-type formulation of well-known polynomial-based interpolatory 4-point schemes, ranging from the pioneers Dubuc–Deslauriers [5,6] to the non-uniform 4-point scheme in [9,8], and introduces the proposal of a novel non-uniform 4-point scheme with edge parameters. Section 3 focuses on the latter and discusses all its characterizing properties in detail. Section 4 addresses the attention towards the special features that can be achieved by the new scheme and provides some examples of practical use. Successively, based on the proposed blending-type formulation, Section 5 introduces a generalization of the previously presented refinement rules enriched by the flexibility of arbitrary point insertion, which makes our algorithm an eligible candidate for many applications. Finally, Section 6 presents some illustrations of surfaces with features control, obtained by generalizing the proposed curve scheme to regular quadrilateral meshes. Concluding remarks and a brief summary of the main contributions of the paper can be found in Section 7.

Many figures included in this work present some details that cannot be appreciated on a hard copy of the manuscript. We thus invite the reader to refer to the electronic version of the paper and zoom in to enlarge the salient details as necessary.

2. Non-uniform interpolatory 4-point subdivision schemes

2.1. Main definitions and related background

A *subdivision scheme* is an iterative algorithm aimed at producing curves or surfaces from given discrete data by refining these on denser and denser grids. In the univariate functional case, starting with some initial points $\mathbf{p} = \{p_i \in \mathbb{R} : i \in \mathbb{Z}\}$ attached to a sequence of parameter values x_i in ascending order, we set $\mathbf{p}^0 \equiv \mathbf{p}, \mathbf{x}^0 \equiv \mathbf{x}$ and then for all $k \geq 0$ we iteratively compute a sequence $\mathbf{p}^{k+1}$ by repeated application of the refinement rules

$$p_i^{k+1} = \sum_{j\in\mathbb{Z}} m_{i-2j}^k p_j^k, \quad i \in \mathbb{Z}, \tag{1}$$

defined in terms of the k-level coefficients $\{m_i^k \in \mathbb{R},\ i \in \mathbb{Z}\}$ forming the so called *k-level subdivision mask* $\mathbf{m}^k$. Notice that, in general, the k-level coefficients m_i^k depend on the parameterization $\mathbf{x}^k$ of the k-level polyline of vertices p_i^k. Additionally, in practice, any mask $\mathbf{m}^k$ has finite support S for all $k \geq 0$ and thus the k-level refinement rules (1) can be viewed as multiplication of a local subdivision matrix M^k whose columns are given by the masks $\mathbf{m}^k$, times a column vector with elements p_i^k where $i \in S$.

By the subsequent application of the subdivision operators M^k, $k \geq 0$, we see that the subdivision process generates denser and denser sequences of data so that a notion of convergence can be established by taking into account the piecewise linear function $\mathcal{L}^k$ that interpolates the values p_i^k in correspondence to the parameters x_i^k, namely $\mathcal{L}^k(x_i^k) = p_i^k$, $\mathcal{L}^{(k)}|_{[x_i^k, x_{i+1}^k]} \in \Pi_1, i \in \mathbb{Z},\ k \geq 0$, where Π_1 is the space of linear polynomials. If the sequence $\{\mathcal{L}^k,\ k \geq 0\}$ converges, then we denote its limit by $\mathcal{F}_{\mathbf{p}} := \lim_{k\to\infty} \mathcal{L}^k$ and say that $\mathcal{F}_{\mathbf{p}}$ is the *limit function* of the subdivision scheme based on the rule (1) for the initial data $\mathbf{p}$.

An equivalent description of convergence investigates the existence of the so called *basic limit function* as the limit of the subdivision scheme when applied to the initial data $(x_h, \delta_{h,i})$, $h \in \mathbb{Z}$. If this is convergent with limit $\mathcal{F}_{\boldsymbol{\delta}_i}$, then we have $\mathcal{F}_{\mathbf{p}}(t) = \sum_{i\in\mathbb{Z}} \mathcal{F}_{\boldsymbol{\delta}_i}(t)\, p_i$, $t \in \mathbb{R}$, for any initial data sequence $\mathbf{p}$.

In the *stationary* case, namely when the coefficients m_i^k do not vary with the refinement level k, then $M^k \equiv M$ for all $k \geq 0$ and many properties of the basic limit function can be deduced from the eigenstructure of M. For a more detailed description of the fundamental notions at the base of subdivision theory, we refer the reader to [11].

An interesting class of subdivision schemes is that of the so called *interpolatory* schemes. These refine the sequence $\mathbf{p}$ while keeping the original data in the sense that for all $k \geq 0$ it holds $p_{2i}^{k+1} = p_i^k,\ i \in \mathbb{Z}$. Their refinement mask $\mathbf{m}^k$ is of a special type since it has the even index subsequence $(\mathbf{m}^k)_{even} := \{m_{2i}^k,\ i \in \mathbb{Z}\}$ that satisfies $(\mathbf{m}^k)_{even} = \boldsymbol{\delta}_0$. Whenever they converge, the associated limit functions are cardinal interpolants to the given data, i.e. $\mathcal{F}_{\mathbf{p}}(x_i) = p_i,\ i \in \mathbb{Z}$ and their basic limit function $\mathcal{F}_{\boldsymbol{\delta}_i}$ is a cardinal interpolant to the sequence $\boldsymbol{\delta}_i$, i.e. $\mathcal{F}_{\boldsymbol{\delta}_i}(x_h) = \delta_{h,i}$, $h \in \mathbb{Z}$.

When the odd index subsequence of the interpolatory mask $(\mathbf{m}^k)_{odd} := \{m_{2i+1}^k,\ i \in \mathbb{Z}\}$ contains exactly 4 coefficients, the interpolatory scheme is termed 4-*point* and if these coefficients change with the location of the newly inserted point the scheme is said to be *non-uniform*. For the sake of simplicity and clarity, hereinafter we will denote this sequence of 4 coefficients by $c_{0,i}^k,\ c_{1,i}^k,\ c_{2,i}^k,\ c_{3,i}^k$. This notation allows us to write the refinement rules of an interpolatory non-uniform 4-point scheme in the general form

$$\begin{aligned} p_{2i}^{k+1} &= p_i^k \\ p_{2i+1}^{k+1} &= c_{0,i}^k p_{i-1}^k + c_{1,i}^k p_i^k + c_{2,i}^k p_{i+1}^k + c_{3,i}^k p_{i+2}^k. \end{aligned}$$

2.2. A blending-type formulation of polynomial based interpolatory 4-point schemes

In the last two decades interpolatory subdivision schemes have gained great popularity and proposals of 4-point schemes with an insertion rule p_{2i+1}^{k+1} obtained by evaluating locally fitted polynomials at a certain parameter value, have appeared as advantageous alternatives to the well-established parametric interpolating models. These schemes, also called polynomial based interpolatory 4-point schemes, can be presented in many different guises. For example, it is known that the insertion rule of the Dubuc–Deslauriers 4-point interpolatory scheme [5,6] comes from fitting a local cubic polynomial to four successive equispaced data points and evaluating this at the center of the interval. Since this is a peculiarity that univocally identifies this proposal, in the following we will refer to such a scheme as the *uniform polynomial based* interpolatory 4-point scheme.

It is well known (see for example [12]) that the insertion rule of such a scheme can also be derived by following an alternative approach based on quadratic interpolation, which is actually a special case of Neville–Aitken's algorithm [13]. More precisely, denoting by $p_{i-1}, p_i, p_{i+1}, p_{i+2}$ a quadruple of points attached to the integer grid, if we determine the quadratic polynomials $B_{i-1}(x)$ and $B_i(x)$ interpolating the triples p_{i-1}, p_i, p_{i+1} and p_i, p_{i+1}, p_{i+2} respectively, and then compute the average between the center points of the corresponding curve segments that are confined between p_i and p_{i+1}, we can equivalently get the rule of the Dubuc–Deslauriers 4-point scheme.

The described construction can be straightforwardly extended to a sequence of four arbitrarily spaced points p_{i-1}, p_i, p_{i+1}, p_{i+2} with corresponding parameter values x_{i-1}, x_i, x_{i+1}, x_{i+2}, by generalizing the average between the center points of the two quadratic segments to an affine combination with coefficients $1-\gamma, \gamma$, where $\gamma = \frac{1}{2}\frac{x_{i+1}-2x_{i-1}+x_i}{x_{i+2}-x_{i-1}}$. As a result, we get the stencil of the non-uniform 4-point scheme in [9,10]. Because of its analogy with the Dubuc–Deslauriers 4-point scheme, this last proposal will be called the *non-uniform polynomial based* interpolatory 4-point scheme.

The above construction procedure implies that, whenever the starting points are uniformly sampled from a cubic polynomial, the uniform polynomial based interpolatory 4-point scheme reproduces that polynomial. Analogously the non-uniform polynomial based interpolatory 4-point scheme reproduces the sample polynomial, when applied both to the sample points and the corresponding parameters.

The objective of this section is to show that, by losing one degree of polynomial reproduction in a polynomial based 4-point scheme (namely settling for reproducing quadratics instead of cubics), we can generate a family of subdivision schemes whose refinement rules incorporate a free parameter that can be intuitively modified to get limit curves with features control.

In the remainder of the paper we will adopt the following notation. Let $\mathbf{p} = \{p_i\}$ be the initial polyline and $\mathbf{x} = \{x_i\}$ the associated parameterization. We will indicate by $B_i(x)$ the quadratic polynomial (written in the Bernstein basis) interpolating the triple p_i, p_{i+1}, p_{i+2} at the corresponding parameter values x_i, x_{i+1}, x_{i+2}, namely

$$B_i(x) = p_i(1-\xi)^2 + \frac{p_{i+1}(x_{i+2}-x_i)^2 - p_i(x_{i+2}-x_{i+1})^2 - p_{i+2}(x_{i+1}-x_i)^2}{2(x_{i+1}-x_i)(x_{i+2}-x_{i+1})}2\xi(1-\xi) + p_{i+2}\xi^2$$
$$\text{with } \xi = \frac{x-x_i}{x_{i+2}-x_i}, \tag{2}$$

and we will denote by $\bar{x}_i := \frac{x_i+x_{i+1}}{2}$ the midpoint of the i-th parameter interval.

Given the four subsequent initial points p_{i+h} and parameters x_{i+h} for $h = -1, 0, 1, 2$ we consider an affine combination of the polynomial values $B_{i-1}(\bar{x}_i)$ and $B_i(\bar{x}_i)$ of the form

$$C(\lambda_i) = (1-\omega(\lambda_i))B_{i-1}(\bar{x}_i) + \omega(\lambda_i)B_i(\bar{x}_i). \tag{3}$$

We will call the value $\lambda_i \in [0, 1]$, associated with the i-th edge $\overline{p_i p_{i+1}}$, the *edge parameter*. Moreover we require that the blending function $\omega(\lambda)$ satisfies

$$\begin{aligned}
&\text{(a) } \omega : [0, 1] \to [0, 1];\\
&\text{(b) } \omega(0) = 1 \text{ and } \omega(1) = 0;\\
&\text{(c) } \omega \text{ is monotonically decreasing;}\\
&\text{(d) } \omega \text{ is regular (at least } C^1\text{);}\\
&\text{(e) } \omega \text{ is an odd function with respect to the point } \left(\frac{1}{2}, \frac{1}{2}\right).
\end{aligned} \tag{4}$$

To our aims, conditions (b) and (c) above are equivalent to

(bb) $\omega(0) = 0$ and $\omega(1) = 1$;
(cc) ω is monotonically increasing;

since the expression in (3) will give us the same values in correspondence to $1 - \lambda_i$.

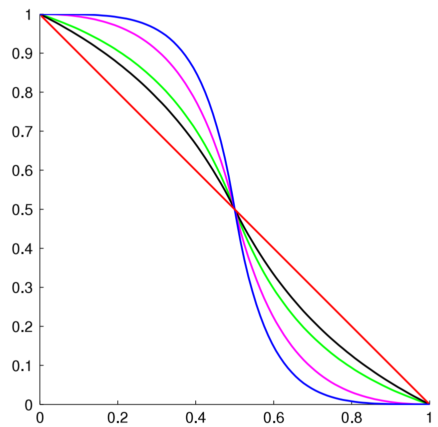

Fig. 1. Different choices of functions $\omega(\lambda)$ satisfying the requirements (4).

Fig. 1 shows different choices of functions $\omega(\lambda)$ satisfying (4), whose derivation will be discussed in Section 5. Notice that Eq. (3) describes a whole family of affine combinations, where each member of the family is identified by a different function $\omega(\lambda)$ that fulfills the above list of requirements.

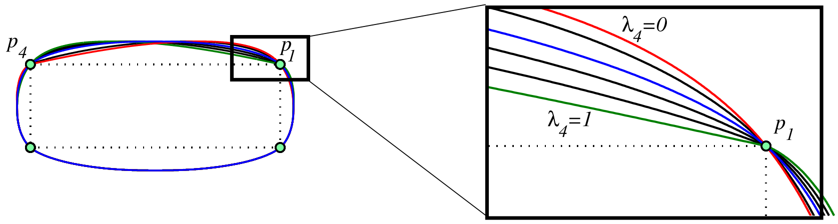

Fig. 2. Limit curves obtained by setting the edge parameter λ_4 to the values: 1 (green curve), $\frac{5}{6}$, $\frac{2}{3}$, $\frac{1}{2}$ (blue curve), $\frac{1}{6}$, 0 (red curve). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

In this way, assuming $p^0_{i+h} := p_{i+h}, x^0_{i+h} := x_{i+h}$ for $h = -1, 0, 1, 2$ and $\lambda^0_i := \omega(\lambda_i)$, for all $k \geq 0$ we can derive the k-level refinement rules of an interpolatory 4-point scheme as

$$\begin{aligned} p^{k+1}_{2i} &= p^k_i \\ p^{k+1}_{2i+1} &= C(\lambda^k_i) = c^k_{0,i}p^k_{i-1} + c^k_{1,i}p^k_i + c^k_{2,i}p^k_{i+1} + c^k_{3,i}p^k_{i+2} \end{aligned} \tag{5}$$

where the coefficients $c^k_{j,i}, j = 0, 1, 2, 3$ turn out to possess the following expressions

$$\begin{aligned} c^k_{0,i} &:= c_0(\lambda^k_i, d^k_{i-1}, d^k_i, d^k_{i+1}) = -\frac{(1-\lambda^k_i)(d^k_i)^2}{4d^k_{i-1}(d^k_{i-1}+d^k_i)} \\ c^k_{1,i} &:= c_1(\lambda^k_i, d^k_{i-1}, d^k_i, d^k_{i+1}) = \frac{(1-\lambda^k_i)(d^k_i)^2 - \lambda^k_i d^k_i(d^k_{i-1}+d^k_{i+1}) + d^k_i d^k_{i+1}}{4d^k_{i-1}(d^k_i+d^k_{i+1})} + \frac{1}{2} \\ c^k_{2,i} &:= c_2(\lambda^k_i, d^k_{i-1}, d^k_i, d^k_{i+1}) = \frac{\lambda^k_i(d^k_i)^2 - (1-\lambda^k_i)d^k_i(d^k_{i-1}+d^k_{i+1}) + d^k_{i-1}d^k_i}{4d^k_{i+1}(d^k_{i-1}+d^k_i)} + \frac{1}{2} \\ c^k_{3,i} &:= c_3(\lambda^k_i, d^k_{i-1}, d^k_i, d^k_{i+1}) = -\frac{\lambda^k_i(d^k_i)^2}{4d^k_{i+1}(d^k_i+d^k_{i+1})}. \end{aligned} \tag{6}$$

Assuming $d^0_j := x^0_{j+1} - x^0_j$ for $j = i-1, i, i+1$, the parameters $d^k_{i-1}, d^k_i, d^k_{i+1}$ represent three consecutive knot intervals related to the k-level refinement and λ^k_i is the parameter of the edge $\overline{p^k_i p^k_{i+1}}$.

In general, if not fixed independently of the refinement level, at the k-th iteration both the values of the knot-intervals d^k_i and the edge parameters λ^k_i should be computed according to a suitable method. Of course, this choice will influence the linearity and stationarity of the scheme as well as the properties of the limit function.

In order to guarantee that (6) identify a linear and stationary subdivision process, we will use an updating strategy to deduce the k-level knot-intervals and edge parameters from those at level 0. In particular, in the sequel for all $k \geq 0$ the knot-intervals will be updated according to the formula

$$d^{k+1}_{2i} = d^{k+1}_{2i+1} = \frac{d^k_i}{2} \tag{7}$$

and the edge parameters will be defined as

$$\begin{cases} \lambda^{k+1}_{2i} = \dfrac{1}{2} \\ \lambda^{k+1}_{2i+1} = \lambda^k_i \end{cases} \quad \text{if } \lambda^k_i < \frac{1}{2}, \tag{8}$$

and

$$\begin{cases} \lambda^{k+1}_{2i} = \lambda^k_i \\ \lambda^{k+1}_{2i+1} = \dfrac{1}{2} \end{cases} \quad \text{if } \lambda^k_i \geq \frac{1}{2}. \tag{9}$$

Through the refinement process, the behavior of the parameters λ^k_i can be understood as follows. At subdivision level $k = 0$ the edge parameter $\lambda^0_i := \omega(\lambda_i)$ is assigned to the edge $\overline{p_i\, p_{i+1}}$. Then, at the successive step $k = 1$ the edge $\overline{p_i\, p_{i+1}}$ is split into the two edges $\overline{p^1_{2i}\, p^1_{2i+1}}, \overline{p^1_{2i+1}\, p^1_{2i+2}}$ and, according to (8)–(9), λ^1_{2i} will inherit the edge parameter value λ^0_i if $\lambda^0_i \geq \frac{1}{2}$, otherwise the edge parameter λ^0_i will be assigned to λ^1_{2i+1}. In this way the chosen updating rule provides a linear subdivision process.

Fig. 2 shows the limit curves of the scheme corresponding to $\omega(\lambda) = 1 - \lambda$ for different values of λ_4 (edge parameters λ_i on the other edges are set to the value $\frac{1}{2}$). We observe that, if $\lambda_4 > 1/2$, in the vicinity of vertex p_1 the limit curve is more ‘spiky’, otherwise it is more ‘rounded’.

Remark 1. If the parameters x_i are uniform and $\lambda_i = \frac{1}{2} \forall i$, since by the definition of the scheme any $\omega(\lambda)$ satisfies condition (e) in (4), $\lambda_i^0 = \frac{1}{2} \forall i$; then relations (7)–(9) imply that $d_i^k = d$ and $\lambda_i^k = \frac{1}{2}$ for all $k > 0$ and the coefficients in (6) coincide with the Dubuc–Deslauries 4-point scheme [5,6].

3. Properties of the polynomial-based interpolatory 4-point scheme with edge parameters

In this section we analyze the analytic properties of limit curves generated by the family of polynomial-based schemes given by Eqs. (5)–(6), concerning support width, polynomial precision, approximation order and smoothness.

Proposition 1 (*Support Width*)**.** *Let $\mathcal{F}_{\delta_i}$ the basic limit function of the non-uniform 4-point scheme centered at x_i, namely the limit function of the rules (5)–(6) applied to the data $(x_h, \delta_{h,i})$, $h \in \mathbb{Z}$. Then $\mathcal{F}_{\delta_i}$ has support $S = [x_{i-3},\ x_{i+3}]$.*

Proof. Since the mask of the scheme is of finite support, by definition, the basic limit function $\mathcal{F}_{\delta_i}$ has compact support and its width is

$$S = \left[x_{i-2} - \sum_{k=1}^{+\infty} \frac{x_{i-2} - x_{i-3}}{2^k},\ x_{i+2} + \sum_{k=1}^{+\infty} \frac{x_{i+3} - x_{i+2}}{2^k} \right] = [x_{i-3},\ x_{i+3}]. \quad \square$$

Proposition 2 (*Polynomial Precision in the Non-Uniform Setting*)**.** *For any initial set of edge parameters $\{\lambda_i\}_{i\in\mathbb{Z}}$ the non-uniform 4-point scheme (5)–(6) reproduces the set Π_2 of polynomials up to degree 2 whenever applied to any sequence of arbitrarily spaced samples.*

Proof. Let p_{i+h}^k, $h = -1, 0, 1, 2$ the samples of a quadratic polynomial $\mathcal{P}(x)$ at the parameters x_{i+h}. Since the scheme (5) is interpolatory, we should just verify that the point p_{2i+1}^{k+1} belongs to $\mathcal{P}(x)$. To this aim recall that p_{2i+1}^{k+1} is computed by applying Eq. (3) to the i-th interval $[x_i, x_{i+1}]$, where $B_{i-1}(x)$ and $B_i(x)$ are interpolating quadratic polynomials given by (2), and thus $B_{i-1}(x) = B_i(x) = \mathcal{P}(x)$. Hence, by applying formula (3), we trivially get $p_{2i+1}^{k+1} = \mathcal{P}(\bar{x}_i)$. $\square$

Proposition 3 (*Polynomial Precision in the Uniform Setting*)**.** *The non-uniform 4-point scheme (5)–(6) reproduces the set Π_3 of polynomials up to degree 3 whenever applied to evenly-spaced samples, provided that $\lambda_i = \frac{1}{2}$, $\forall i$.*

Proof. This result follows from the fact that, in the uniform setting, the refinement rules (5)–(6) reduce to Dubuc's 4-point scheme (see Remark 1). $\square$

As it is well known, under certain conditions, the exactness of a non-uniform subdivision scheme for polynomials up to degree m is necessary and sufficient for achieving an approximation order $m + 1$ for any function which is smooth enough (see e.g. [14,15]). Thus the following result holds.

Corollary 1 (*Approximation Order*)**.** *The non-uniform 4-point scheme with coefficients in (6) has approximation order 3.*

Proposition 4. *The non-uniform 4-point scheme with edge parameters $\lambda_{i-1} = 0$ and $\lambda_i = 1$ generates a limit curve that between the knot values x_{i-1}, x_{i+1} coincides with the quadratic polynomial $B_{i-1}(x)$ interpolating the points (x_{i-1}, p_{i-1}), (x_i, p_i), (x_{i+1}, p_{i+1}).*

Proof. We start by observing that, at refinement step $k = 1$, the two points inserted on the intervals $[x_{i-1}, x_i]$ and $[x_i, x_{i+1}]$ both belong to $B_{i-1}(x)$.

To this aim, let us first analyze the effect of setting the edge parameter $\lambda_{i-1} = 0$ on the initial edge between x_{i-1} and x_i. Since for any function $\omega(\lambda)$ that satisfies condition (b) in (4) we have $\omega(\lambda_{i-1}) = \omega(0) = 1$, it follows that $p_{2i-1}^1 = C(\lambda_{i-1}) = B_{i-1}(\bar{x}_{i-1})$ and thus the point inserted along the considered edge belongs to the quadratic polynomial $B_{i-1}(x)$. In a similar way, it can also be proven that the new point p_{2i+1}^1, inserted on $[x_i, x_{i+1}]$, belongs to $B_{i-1}(x)$.

Suppose now that, after $k > 1$ iterations of the scheme, all the points inserted between x_{i-1} and x_{i+1} belong to $B_{i-1}(x)$. Then we will prove that all the new points inserted at the successive refinement level $k + 1$ still belong to $B_{i-1}(x)$.

Let us first focus on the initial span $[x_{i-1}, x_i]$. After k steps, the considered edge has been split into 2^k new edges. Recalling the updating relation (9), it turns out that the first of these edges — i.e. the edge containing the initial vertex $x_{i-1} = x_{2^k(i-1)}^k$ — has an edge parameter $\lambda_{2^k(i-1)}^k = 1$, while for the others $\lambda_i^k = \frac{1}{2}$. For the first edge, we have $\lambda_{2^k(i-1)}^k$ and thus, by repeating the procedure above, it can be easily seen that the newly inserted point belongs to $B_{i-1}(x)$. For all the other $2^k - 1$ edges, since the insertion formula (5) involves only points of the polynomial $B_{i-1}(x)$, the thesis is straightforward (see Proposition 2).

Analogously, if we now move to consider the edge $[x_i, x_{i+1}]$, the updating relation (8) implies that only the edge containing the point $x_{i+1} = x_{2^k(i+1)}^k$ has edge parameter $\lambda_{2^k(i+1)-1}^k = 0$ and the same arguments as above hold.

To conclude, for all $k > 0$, all the new points inserted between x_{i-1} and x_{i+1} belong to $B_{i-1}(x)$ and thus, in the considered interval, the limit curve reproduces the entire polynomial $B_{i-1}(x)$. $\square$

We conclude this section by analyzing the smoothness properties of the proposed non-uniform 4-point subdivision scheme.

To this aim we start by observing that, after a few rounds of subdivision, the knot intervals in the neighborhood of any initial point x_i assume a piecewise-uniform configuration of the kind $\ldots,\ 1,\ 1,\ 1,\ \alpha,\ \alpha,\ \alpha, \ldots$ where $\alpha > 0$ (see Fig. 3).

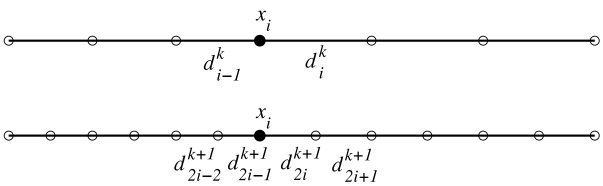

Fig. 3. Knot-interval configuration in the neighborhood of the initial knot x_i after k and $k+1$ iterations of the non-uniform 4-point scheme.

Thus the parameters 1 and α identify two adjacent uniform regions, whose junction point is represented by the vertex x_i. By relations (8)–(9), in the uniform regions the edge parameters assume everywhere the common value $\lambda_i^k = 1/2$, except possibly at those edges containing x_i. As a consequence, away from the junction point x_i, the non-uniform 4-point scheme brings back to the uniform 4-point scheme [6], which is known to be C^1. Thus, we only need to analyze the smoothness of the scheme in the regions surrounding the junction points.

To this purpose, we will rely on a generalization of the analysis in [10], concerning binary refinements defined over non-uniform knot sequences that are halved at each step. Differently from [10], in the neighborhood of the junction points, we need to take into account both the local knot intervals d_i^k-s and the edge parameters λ_i^k-s. However, when using the parameter updating strategy in (7), the scheme is still stationary, namely the same refinement matrix $M^k \equiv M$ for all $k \geq 0$ is applied at each iteration around the point x_i. In particular, it can be easily proven that, for each eigenvalue ℓ_i of M with eigenvector $\mathbf{r}_i$, the basis function $\mathcal{F}_{\mathbf{r}_i}$ of the scheme satisfies

$$\ell_i \mathcal{F}_{\mathbf{r}_i}(x) = \mathcal{F}_{\mathbf{r}_i}\left(\frac{x}{2}\right). \tag{10}$$

Therefore, the following result holds true, analogously to Theorem 7 in [10].

Proposition 5. *Let $Mx = \ell x$, with $|\ell| < \frac{1}{2^k}$. If $\mathcal{F}_{\mathbf{r}_i}(x)$ is C^k-continuous everywhere except at $x = 0$, then $\mathcal{F}_{\mathbf{r}_i}$ is C^k-continuous everywhere.*

A consequence of this proposition is that, if the scheme satisfies relation (10) and the two leading eigenvectors reproduce the constant and linear functions — which is true by construction of the scheme — the conditions

$$\ell_0 = 1, \qquad \ell_1 = \frac{1}{2}, \qquad |\ell_i| < \frac{1}{2}, \quad \forall i \geq 2 \tag{11}$$

are sufficient to guarantee C^1-smoothness of the scheme. Thus, we can prove the following result.

Proposition 6 (*Smoothness Order*)**.** *The non-uniform 4-point scheme generates C^1-continuous limit curves for any choice of initial knots $\{x_i\}_{i\in\mathbb{Z}}$ and edge parameters $\{\lambda_i\}_{i\in\mathbb{Z}}$ such that two subsequent initial edges do not assume at the same time the values $\lambda_{i-1} = 1$ and $\lambda_i = 0$. In this case the limit curve will be only C^0 at the point p_i.*

Proof. Without loss of generality, we will assume the junction point of the two regular knot sequences obtained after $k > 2$ subdivision steps to be $x_i = 0$. From the above discussion, the non-uniform 4-point scheme is C^1-continuous everywhere except at the point $x_i = 0$. Thus we only need to analyze the eigenproperties of the local subdivision matrix in the neighborhood of $x_i = 0$. For the sake of generality, we will consider a local subdivision matrix M of the most general form (namely involving a different edge parameter λ_i^k for each stencil), even if, when the updating rules are the ones in (8)–(9), the matrix assumes a simplified structure. In particular,

$$M = \begin{bmatrix}
c_0(\lambda_{i-2}^k,1,1,1) & c_1(\lambda_{i-2}^k,1,1,1) & c_2(\lambda_{i-2}^k,1,1,1) & c_3(\lambda_{i-2}^k,1,1,1) & 0 & 0 & 0\\
0 & 0 & 1 & 0 & 0 & 0 & 0\\
0 & c_0(\lambda_{i-1}^k,1,1,\alpha,) & c_1(\lambda_{i-1}^k,1,1,\alpha) & c_2(\lambda_{i-1}^k,1,1,\alpha) & c_3(\lambda_{i-1}^k,1,1,\alpha) & 0 & 0\\
0 & 0 & 0 & 1 & 0 & 0 & 0\\
0 & 0 & c_0(\lambda_i^k,1,\alpha,\alpha) & c_1(\lambda_i^k,1,\alpha,\alpha) & c_2(\lambda_i^k,1,\alpha,\alpha) & c_3(\lambda_i^k,1,\alpha,\alpha) & 0\\
0 & 0 & 0 & 0 & 1 & 0 & 0\\
0 & 0 & 0 & c_0(\lambda_{i+1}^k,\alpha,\alpha,\alpha) & c_1(\lambda_{i+1}^k,\alpha,\alpha,\alpha) & c_2(\lambda_{i+1}^k,\alpha,\alpha,\alpha) & c_3(\lambda_{i+1}^k,\alpha,\alpha,\alpha)
\end{bmatrix}$$

and thus, by substituting formulas (6) above

$$M = \begin{bmatrix}
\frac{\lambda_{i-2}^k-1}{8} & \frac{3(2-\lambda_{i-2}^k)}{8} & \frac{3(1+\lambda_{i-2}^k)}{8} & -\frac{\lambda_{i-2}^k}{8} & 0 & 0 & 0\\
0 & 0 & 1 & 0 & 0 & 0 & 0\\
0 & \frac{\lambda_{i-1}^k-1}{8} & \frac{3(1+\alpha)-\lambda_{i-1}^k(2+\alpha)}{4(1+\alpha)} & \frac{3\alpha+\lambda_{i-1}^k(2+\alpha)}{8\alpha} & -\frac{\lambda_{i-1}^k}{4\alpha(1+\alpha)} & 0 & 0\\
0 & 0 & 0 & 1 & 0 & 0 & 0\\
0 & 0 & -\frac{\alpha^2(1-\lambda_i^k)}{4(1+\alpha)} & \frac{2(2+\alpha)-\lambda_i^k(1+2\alpha)}{8} & \frac{2+\alpha+\lambda_i^k(1+2\alpha)}{4(1+\alpha)} & -\frac{\lambda_i^k}{8} & 0\\
0 & 0 & 0 & 0 & 1 & 0 & 0\\
0 & 0 & 0 & \frac{\lambda_{i+1}^k-1}{8} & \frac{3(2-\lambda_{i+1}^k)}{8} & \frac{3(1+\lambda_{i+1}^k)}{8} & -\frac{\lambda_{i+1}^k}{8}
\end{bmatrix}. \tag{12}$$

Using the symbolic computation program Mathematica, it can be easily verified that, for all possible edge parameters configurations—except for the case $\lambda_{i-1} = 1$ and $\lambda_i = 0$—the eigenvalues of M satisfy the necessary C^1 conditions (11). Conversely, the local subdivision matrix M in (12) with $\lambda_{i-1} = 1$ and $\lambda_i = 0$, generates the eigenvalues $\ell_0 = 1, \ell_1 = \ell_2 = \frac{1}{2}$, $|\ell_i| < \frac{1}{2} \forall i \geq 3$. In this case the eigenvalues ℓ_1 and ℓ_2 have two linearly independent eigenvectors,

$$\mathbf{r}_1 = \left[\, 0,\ 0,\ 0,\ 0,\ \frac{1}{3},\ \frac{2}{3},\ 1 \,\right], \qquad \mathbf{r}_2 = [\, 3,\ 2,\ 1,\ 0,\ 0,\ 0,\ 0 \,]$$

causing the scheme to be C^0 at the junction point $x_i = 0$. $\square$

4. Features control and special behaviors

One of the main contributions of this paper consists in introducing a new formulation for the construction of non-uniform 4-point schemes with edge parameters that allows us to provide an efficient method for generating flexible and various shapes passing through a given set of points.

It is important to observe that, even though with respect to the polynomial based 4-point schemes we have reduced the polynomial reproduction degree by one, we do not lose quality in the limit curves. In fact, as we have experienced in [7], the underlying non-uniform parameterization is sufficient to guarantee a satisfactory approximation to the initial data. On the other hand, relaxing the constraints on the polynomial reproduction degree as described, allows us to incorporate the edge parameters λ_i-s in the coefficients of the scheme. As we will show in this section, the edge parameters can be properly set to include special features in the limit curve, extending the applicability of this interpolation method to many practical contexts. In particular, with the term *features* we indicate a number of different curve shape effects and special behaviors that can be classified into the following groups:

- generation of polynomial curves and piecewise polynomials;
- tension effects focused on prescribed vertices or edges;
- C^0 effects (cusps);
- subsequent flat edges, i.e. degenerating to one or more line segments;
- automatic handling of open curves.

While in the class of approximating subdivision there exist schemes managing these special shape effects [3,4], to our knowledge this is the unique interpolatory scheme with such capabilities.

In this section, we describe in detail how the edge parameters λ_i in formulas (6) should be handled, in order to specify prescribed features in the limit curves. The proposed examples refer to the subdivision scheme with $\omega(\lambda) = 1 - \lambda$.

4.1. Piecewise continuous quadratic interpolation

Usually, when we say that a uniform or non-uniform interpolating scheme reproduces polynomials, we mean that the refinement process converges in a certain span to a polynomial when a sufficient number of starting points in that span lie on it. This is also the case e.g. of the schemes [9] or [8], that, in this sense, reproduce cubic polynomials.

Differently, the scheme (5)–(6) is capable of producing a piecewise continuous quadratic curve even if the starting points are not sampled from a quadratic. This is a straightforward consequence of Proposition 4. In fact, if on any number of initial edges we set the parameters λ_i alternating the values 0 and 1 on successive edges, the corresponding limit curve will consist of a sequence of quadratic polynomials, each one corresponding to two initial edges, with a continuous join between them. The two following special behaviors depend on this property and can be achieved by just adding to the refinement algorithm a simple preprocessing step as described.

4.1.1. Piecewise smooth quadratic interpolation

Starting from any arbitrary initial data, it is possible to obtain a limit curve formed of quadratic polynomial pieces joined C^1-continuously. To this aim it is sufficient to properly add some extra points to the initial polyline before starting the refinement process.

For each initial edge $\overline{p_i p_{i+1}}$ the additional points are specified as follows. In correspondence with each vertex p_i we first compute the tangent T_i of the quadratic polynomial $B_{i-1}(x)$ interpolating $p_{i-1},\ p_i,\ p_{i+1}$ at the parameter values $x_{i-1},\ x_i,\ x_{i+1}$. Then, in front of each edge $\overline{p_i p_{i+1}}$, we extend the initial sequence of vertices with three new points v_{2i-1}, s_i and v_{2i} by evaluating the C^1-joined quadratic Bézier curves

$$q_1(u), \quad u \in [x_i, \bar{x}_i], \qquad q_2(u), \quad u \in [\bar{x}_i, x_{i+1}]$$

respectively with control points

$$Q_0^1 = p_i, \qquad Q_1^1 = p_i + \frac{1}{4} T_i (x_{i+1} - x_i), \qquad Q_2^1 = \frac{1}{2}(Q_1^1 + Q_1^2)$$

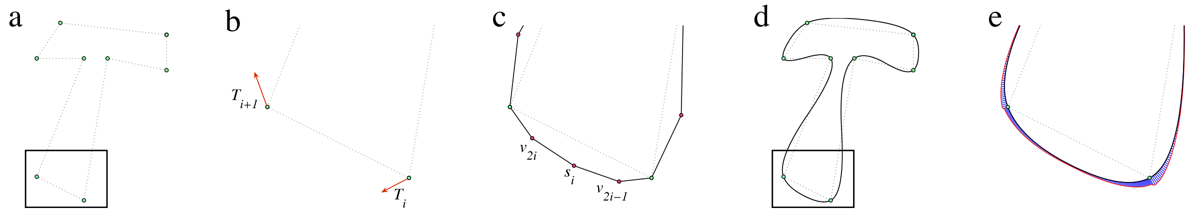

Fig. 4. Generation of an interpolatory limit curve made of C^1-joined quadratic pieces: (a) starting polyline; (b) tangents at the initial points; (c) starting polyline enriched by vertices v_i and s_i; (d) limit curve; (e) curvature comb.

and

$$Q_0^2 = Q_2^1, \qquad Q_1^2 = p_{i+1} - \frac{1}{4}T_{i+1}(x_{i+1} - x_i), \qquad Q_2^2 = p_{i+1}.$$

In particular we set

$$v_{2i-1} = q_1\left(\frac{3}{4}x_i + \frac{1}{4}x_{i+1}\right),$$
$$s_i = Q_2^1,$$
$$v_{2i} = q_2\left(\frac{1}{4}x_i + \frac{3}{4}x_{i+1}\right),$$

so that v_{2i-1} and v_{2i} correspond to the value of $q_1(u)$ and $q_2(u)$ at the midpoints of the respective intervals of definition and s_i is the junction point between q_1 and q_2.

By refining the starting polyline $\{p_i\}_{i\in\mathbb{Z}}$ enriched by the so computed vertices v_i and s_i through the non-uniform 4-point scheme with edge parameters $\lambda_{i-1} = 0$ and $\lambda_i = 1$, we generate a limit curve that is made of C^1-joined quadratic pieces with endpoints at the vertices p_i and s_i (see Fig. 4).

Remark 2. The limit curve so obtained coincides with the non-uniform local interpolatory quadratic spline in [16]. As a consequence, in this particular setting, the proposed method is the only interpolatory scheme capable of generating a C^1 piecewise limit curve whose analytic representation is known.

4.1.2. Flat edges

As a special case of the property illustrated in 4.1, we also have that, when three points p_{i-1}, p_i, p_{i+1} are collinear, the scheme is capable of reproducing the linear segment passing through them, provided that the initial parameters are specified as explained in Section 4.1 (see e.g. Fig. 5, center).

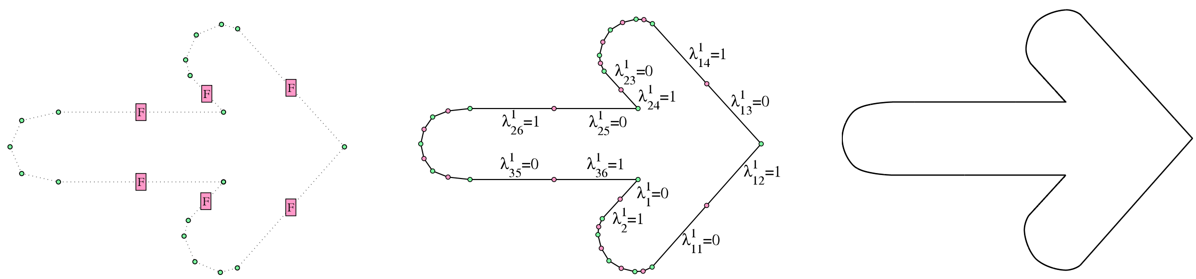

Fig. 5. Left: initial polyline; flat edges are marked with the letter F. Center: parameters configuration on the polyline after the first refinement step; green circles correspond to the initial points, while pink circles represent the inserted points. Right: limit curve. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

If the starting polyline does not contain three collinear points, it is still possible to generate a flat edge just by forcing, at the first refinement iteration, that the points inserted on the selected flat edges be placed at their midpoints, so as to create three collinear points in the refined polyline. Obviously, in this case, also the edge parameters should be properly reassigned after the first refinement, as illustrated in Fig. 5.

We now emphasize two remarkable behaviors of the scheme in the presence of flat edges. First, in case two consecutive flat edges occur in the starting polyline, by Proposition 6 the limit curve will be C^0 at the joint. The results in Proposition 6 also imply that, if the flat edge is isolated, the limit curve will be C^1 continuous at the joint. Moreover, even when it is smoothly connected with a curvilinear part, the flat region is incorporated in the limit curve without creating undesired artifacts in correspondence of its end points (see Fig. 5, right). This is not the case of any other interpolatory scheme, being either uniform or non-uniform.

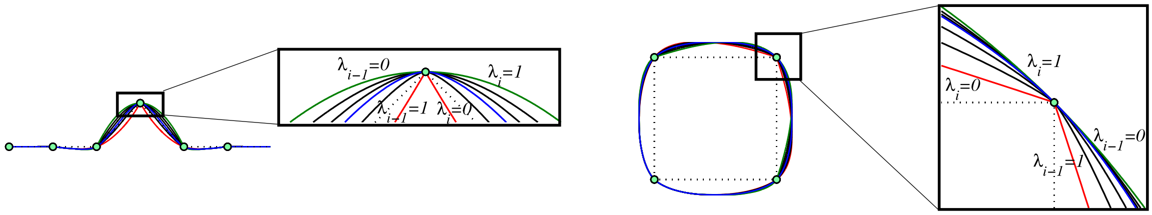

Fig. 6. Point tension effects obtained by setting the couple of edge parameters $(\lambda_{i-1}, \lambda_i)$ to the values: $(1, 0)$ (red curve), $(\frac{5}{6}, \frac{1}{6})$, $(\frac{2}{3}, \frac{1}{3})$, $(\frac{1}{2}, \frac{1}{2})$ (blue curve), $(\frac{1}{3}, \frac{2}{3})$, $(\frac{1}{6}, \frac{5}{6})$, $(0, 1)$ (green curve). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4.2. Cusps and point tension effects

In Proposition 6 it was proven that, when two non collinear flat edges meet at a vertex p_i, it is possible to obtain in the limit curve a C^0 behavior at one initial vertex p_i by setting the edge parameters of the neighboring edges to the values $\lambda_{i-1} = 1$ and $\lambda_i = 0$. This property has a straightforward geometric interpretation, due to the structure of the limit curve in the neighborhood of the vertex p_i. In fact, due to this setting of edge parameters, a portion of the limit curve to the left of p_i coincides with the quadratic polynomial $B_{i-2}(x)$ interpolating the triple p_{i-2}, p_{i-1}, p_i, while to the right it reproduces the polynomial $B_i(x)$, that interpolates p_i, p_{i+1}, p_{i+2}. The vertex p_i is thus the junction point between the two quadratic polynomials, that, except for ad hoc constructions (see Section 4.1.1), meet only C^0 continuously.

It is important to observe that a C^0 point in the limit curve is just the 'limit' of a tension effect focused on the corresponding initial vertex p_i, which can be obtained by gradually increasing the edge parameter λ_{i-1} up to the value 1 and at the same time decreasing the parameter λ_i down to 0, as illustrated in Fig. 6.

4.3. End-point rules

When the initial set of points represents an open polyline it is not possible to refine the first or the last edges by using the 4-point refinement rules introduced in Section 2.2, since the refinement equations require a well-defined 2-neighborhood in the vicinity of the boundary points. Usually, this problem is overcome by specifying an ad hoc insertion rule for the points on the boundary edges, which very often relies on their linear extrapolation. Conversely, the constructive approach at the base of our 4-point scheme naturally lends itself to refine boundary edges without using any auxiliary point. In fact, to refine the first edge $\overline{p_0p_1}$, it is sufficient to set the edge parameter $\lambda_0 = 0$. In this way, by formula (3), it trivially follows that the point p_1^1 is simply obtained as

$$p_1^1 = B_0(\bar{x}_0) = \frac{d_0 + 2d_1}{4(d_0 + d_1)} p_0 + \frac{d_0 + 2d_1}{4d_1} p_1 - \frac{(d_0)^2}{4d_1(d_0 + d_1)} p_2.$$

Analogously, associating the edge parameter $\lambda_{N-1} = 1$ to the last edge $\overline{p_{N-1}p_N}$, formula (3) provides

$$p_{2N-1}^1 = B_{N-2}(\bar{x}_{N-1}) = -\frac{(d_{N-1})^2}{4d_{N-2}(d_{N-2} + d_{N-1})} p_{N-2} + \frac{2d_{N-2} + d_{N-1}}{4d_{N-2}} p_{N-1} + \frac{2d_{N-2} + d_{N-1}}{4(d_{N-2} + d_{N-1})} p_N.$$

4.4. Tension control

In this subsection we show that the non-uniform 4-point scheme introduced in Section 2.2 has the property of tension control, i.e. the possibility of intuitively controlling the tension of the limit curve segment corresponding to a certain initial edge. Excluding some proposals involving non-stationary schemes (see e.g. [17,18]), we are not aware of any other interpolatory 4-point scheme with the behavior described in the following.

It is easy to observe that, if the knot interval d_i associated with the edge $\overline{p_ip_{i+1}}$ is not automatically set according to a centripetal or chordal parameterization, but is assigned by the user, then it assumes the role of a *tension parameter*. More precisely, the smaller it is the tighter is the portion of the limit curve confined between the vertices p_i, p_{i+1} (see Fig. 7). However, an undesired side effect is that changing the parameterization on a single edge affects also the neighboring curve segments (Fig. 7 (left)). Exploiting the unique properties of the proposed schemes, we can confine this effect to the edge of interest, by just combining the change in the parameterization with a proper setting of edge parameters. Fig. 7 (center) shows the combined effect of changing the parameter d_2 and opportunely modifying the parameters λ_1 and λ_3 on the two adjacent edges. In the figure in the center, the initial edge parameters λ_1 and λ_3 are set to the values 1 and 0, respectively, in order to confine the modification of the limit curve only to the edge $\overline{p_2p_3}$. Moreover, the figure on the right illustrates the effect of changing the parameterization of an edge in correspondence with different configurations of the edge parameters on the adjacent edges.

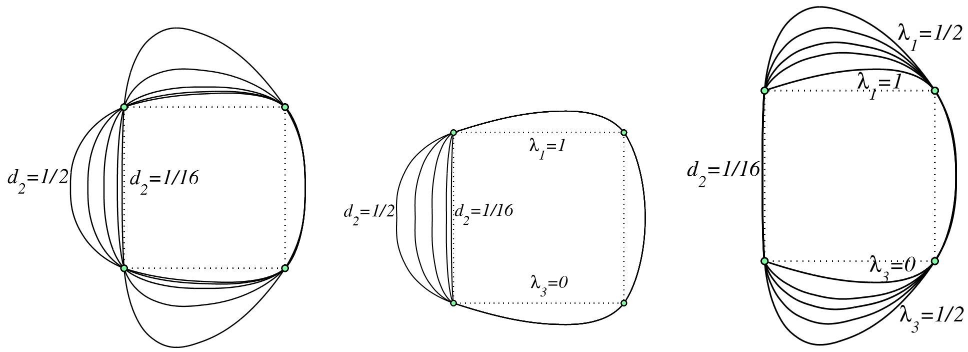

Fig. 7. Left: the effect of setting the parameter d_2 to the values $\frac{1}{2}$, $\frac{3}{8}$, $\frac{1}{4}$ (value of centripetal parameterization), $\frac{1}{8}$, $\frac{1}{16}$. Center: combined effect of changing the parameter d_2 as before and at the same time setting the edge parameters $\lambda_1 = 1$ and $\lambda_3 = 0$. Right: combined effect of setting the parameter $d_2 = \frac{1}{16}$ and at the same time the couple of edge parameters (λ_1, λ_3) to $(1, 0)$, $(\frac{3}{4}, \frac{1}{4})$, $(\frac{2}{3}, \frac{1}{3})$, $(\frac{7}{12}, \frac{5}{12})$, $(\frac{1}{2}, \frac{1}{2})$.

4.5. Application examples

The fully flexible refinement rules introduced in Section 2.2 make our interpolatory scheme an eligible candidate for many applications. An interesting example consists in proposing this novel interpolatory method to represent curves where one switches frequently between round shapes and flat shapes that are stitched together at sharp corners, exactly like it happens in the description of the outline curves of digital fonts. In particular, due to its capability of including flat edges and cusps, the proposed subdivision algorithm provides a mathematical description of vector outlines that turns out to be very simple and efficient.

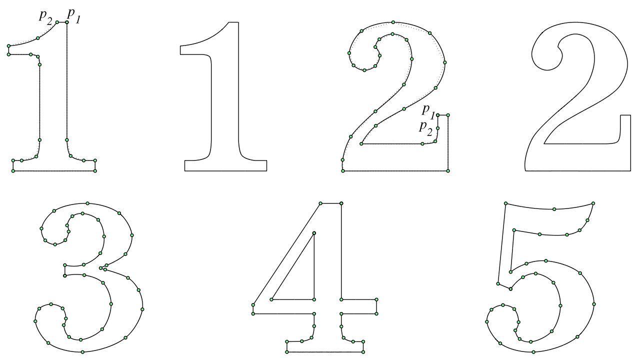

Fig. 8. Application of the proposed scheme to the description of digital fonts.

In the examples shown in Fig. 8 knot intervals are computed according to the centripetal parameterization of the initial polyline. As it is apparent, the limit curve consists in an alternation of flat edges, smoothly connected with free-form curvilinear pieces. In particular, the straight line segments of the limit curve correspond to edges of the initial polyline that are tagged as flat before starting the subdivision process.

In the initial polyline, assume that the first edge has endpoints p_1 and p_2 and that the following edges are numbered in subsequent order. The outline representing the number "1" has been generated as follows: we have set all the edge parameters to the value $\frac{1}{2}$ except on edges 2 and 3, where the edge parameters are 0 and 1 respectively in such a way that we have two cusp features in correspondence of vertices p_2 and p_4 and the segment of the limit curve between these vertices is a quadratic polynomial. Moreover we have treated as flat the edges 1, 4, 5, 8, 11, 12, 13, 14, 15, 18 (i.e. the refinement and edge parameters updating of these edges is performed as explained in Section 4.1.2).

To generate the outline of number "2" we have treated as flat the edges 1, 4, 27, 28, 29 and in addition we have generated two cusps by setting the edge parameters to 0, 1, 0, 1 on edges 5, 17, 18, 26 respectively. The other number outlines in Fig. 8 have been obtained with a similar parameter setting strategy.

5. Subdivision rules for features control and arbitrary point insertion

In this section, we investigate a generalization of the family of schemes presented in Section 2.2, that allows for the insertion of new points at arbitrary locations, independent of the underlying parameterization. As already mentioned, at a general subdivision level k, the 4-point scheme related to the construction (3) limits the insertion of a new point at the position $\bar{x}_i^k$, corresponding to the midpoint of the interval $[x_i^k, x_{i+1}^k]$. Differently, we will see in the following that new points may be placed at any position of existing intervals, so that the most convenient parameter values may be chosen as well as the intervals for insertion. We remark that, since the scheme is interpolatory, the limit curve depends on the location of the point of insertion.

To this aim, if instead of combining the two values of the polynomials $B_{i-1}(x)$ and $B_i(x)$ at $x = \bar{x}_i$ like in (3), we let them vary inside the interval $[x_i, x_{i+1}]$, we can write a blending formula of the kind

$$C(x) = (1 - \omega(x))B_{i-1}(x) + \omega(x)B_i(x), \quad x \in [x_i, x_{i+1}] \tag{13}$$

where, once reparameterized in the interval $[0, 1]$, $\omega(x)$ is a blending function as assumed in Section 2.2. As it will become clear by the end of this section, at this stage it is convenient to consider a function $\omega(x)$ satisfying conditions (a), (bb), (cc), (d) and (e). Observe that, in the equation above, instead of controlling the generation of special features like it happened in (3), the parameter x determines the location of the new point $p_{2i+1} = C(x)$ in the interval $[x_i, x_{i+1}]$.

Remark 3. In Section 2.2 we have recalled Daubechies et al. proposal [9] as an example of polynomial based non-uniform midpoint scheme. Such a scheme was called by the authors as the *semi-regular* 4*-point scheme*, as opposed to its more general version termed the *irregular* 4*-point scheme*, that allows for the insertion of a new point at any arbitrary location $x \in [x_i, x_{i+1}]$. We remark that the formulation of the latter can also be represented by Eq. (13), but in this case the function $\omega(x)$ is

$$\omega(x) = \frac{x - x_{i-1}}{x_{i+2} - x_{i-1}} \tag{14}$$

and does not satisfy condition (bb).

Our purpose is now to generalize (13) to design a two-parameter function $C(\lambda, x)$, which possesses both a parameter λ to manipulate features, and another parameter x corresponding to the location of the newly inserted point. Thus, in the following, we will provide a construction procedure for the blending function $\omega(\lambda, x)$ that, once blended with the two polynomials $B_{i-1}(x)$ and $B_i(x)$, fulfills this requirement. The key idea of this construction is to consider a two-piece blending function defined as follows. Let us first introduce the couple of functions

$$\bar{L}_1(x) = \frac{x - x_i}{\bar{x}_i - x_i} \quad \text{and} \quad \bar{L}_2(x) = \frac{x_{i+1} - x}{x_{i+1} - \bar{x}_i}, \quad x \in [x_i, x_{i+1}],$$

and consider the two-piece C^1 blending function $\omega : [x_i, x_{i+1}] \to [0, 1]$, defined through the formula

$$\omega(x) = \begin{cases} \dfrac{1}{2}\,\dfrac{\bar{L}_1^m(x)}{\bar{L}_2^n(x)} & \text{if } x \in [x_i, \bar{x}_i) \\[2ex] 1 - \dfrac{1}{2}\,\dfrac{\bar{L}_2^m(x)}{\bar{L}_1^n(x)} & \text{if } x \in [\bar{x}_i, x_{i+1}], \end{cases} \tag{15}$$

for any integers $m > 0$, $n \geq 0$, indicating the m-th and n-th powers of the considered functions.

We can now think of (15) as of $\omega(\frac{1}{2}, x)$ and afterwards generalize this one to a function $\omega(\lambda_i, x)$ for $\lambda_i \neq \frac{1}{2}$, such that, for any fixed value of $\lambda_i \in [0, 1]$, the following hold

(a′) $\omega : [x_i, x_{i+1}] \to [0, 1]$;
(b′) $\omega(\lambda_i, x_i) = 0$ and $\omega(\lambda_i, x_{i+1}) = 1$;
(c′) ω is monotonically increasing;
(d′) ω is regular (at least C^1) at t_i,

where $t_i = x_i + \lambda_i(x_{i+1} - x_i)$. We can now extend the definition of $\bar{L}_1(x)$ and $\bar{L}_2(x)$ on two intervals proportional to λ_i and $1 - \lambda_i$ respectively, as

$$L_1(\lambda_i, x) = \frac{x - x_i}{t_i - x_i}, \qquad L_2(\lambda_i, x) = \frac{x_{i+1} - x}{x_{i+1} - t_i} \tag{16}$$

and define the function $\omega(\lambda_i, x)$ by formula (15), where the expressions $\bar{L}_1(x)$ and $\bar{L}_2(x)$ are substituted by the functions $L_1(\lambda_i, x)$ and $L_2(\lambda_i, x)$ in (16), namely

$$\omega(\lambda_i, x) = \begin{cases} c\,\dfrac{L_1^m(\lambda_i, x)}{L_2^n(\lambda_i, x)} & \text{if } x \in [x_i, t_i) \\[2ex] 1 - (1 - c)\,\dfrac{L_2^m(\lambda_i, x)}{L_1^n(\lambda_i, x)} & \text{if } x \in [t_i, x_{i+1}], \end{cases} \tag{17}$$

where, to fulfill condition (d′), the coefficient c is given by

$$c = \frac{(1-\lambda_i)\,n + \lambda_i m}{n+m}$$

and m, n are two integers as above. Now, to allow the insertion of a new point at any arbitrary position x between x_i and x_{i+1}, we consider the following blending formula depending both on the parameter $\lambda_i \in [0, 1]$ and on an arbitrary $x \in [x_i, x_{i+1}]$

$$C(\lambda_i, x) = (1 - \omega(\lambda_i, x))B_{i-1}(x) + \omega(\lambda_i, x)B_i(x), \quad x \in [x_i, x_{i+1}]. \tag{18}$$

As a result, for any integers $m > 0$ and $n \geq 0$, $C(\lambda_i, x)$ is a two-piece function, C^1 continuous on the interval $[x_i, x_{i+1}]$, interpolating the values $B_{i-1}(x_i)$, $B'_{i-1}(x_i)$ at x_i and $B_i(x_{i+1})$, $B'_i(x_{i+1})$ at x_{i+1}. Moreover, for the forthcoming choices of m and n, $C(\lambda_i, x)$ identifies the following particular cases:

- for $m = 1$ and $n = 0$, $\omega(\lambda_i, x) = x$ and $C(\lambda_i, x)$ coincides with the C^1 Catmull–Rom spline [19];
- for $n = 1$, $C(\lambda_i, x)$ is the C^1 join of two degree-$(m + 1)$ polynomials; in particular, when $m = 1$, $C(\lambda_i, x)$ coincides with the two-piece quadratic polynomial proposed in [20,16] and, when $m = 2$, $C(\lambda_i, x)$ is a two-piece cubic polynomial that further interpolates the values $B''_{i-1}(x_i)$ at x_i and $B''_i(x_{i+1})$ at x_{i+1};
- for $n \neq 1$, $C(\lambda_i, x)$ is purely rational.

In the case where $x = \bar{x}_i$ (corresponding to the new point p_{2i+1} inserted so that its parameter x_{2i+1} is the midpoint between x_i and x_{i+1}), the blending function $\omega(\lambda_i, x)$ in (17) reduces to a function of λ, that we will indicate by $\omega(\lambda, \bar{x}_i)$, where

$$\omega(\lambda, \bar{x}_i) = \begin{cases} 1 - (1-c)\,\dfrac{L_2^m(\lambda, \bar{x}_i)}{L_1^n(\lambda, \bar{x}_i)} & \text{if } \lambda \in \left[0, \dfrac{1}{2}\right], \\[2ex] c\,\dfrac{L_1^m(\lambda, \bar{x}_i)}{L_2^n(\lambda, \bar{x}_i)} & \text{if } \lambda \in \left(\dfrac{1}{2}, 1\right], \end{cases}$$

and

$$L_1(\lambda, \bar{x}_i) = \frac{1}{2\lambda}, \qquad L_2(\lambda, \bar{x}_i) = \frac{1}{2(1-\lambda)}.$$

For any $m, n > 0$, the definition of $\omega(\lambda, x)$ ensures that $\omega(\lambda, \bar{x}_i)$ satisfies the conditions (4), in such a way that the corresponding insertion rule belongs to the family of schemes with coefficients in (6). Conversely, when $m > 0$ and $n = 0$, the corresponding function $\omega(\lambda, \bar{x}_i)$ does not fulfill the requirements (4), and thus this setting does not generalize any midpoint insertion scheme of the family (6).

Fig. 1 shows different functions $\omega(\lambda, \bar{x}_i)$, obtained in correspondence to $m = n = 1$ (black), $m = n = 2$ (magenta), $m = n = 3$ (blue), $m = 2,\ n = 1$ (green). The generalized rules derived from $\omega(\lambda, x)$, instead, handle non-uniform knot vectors, edge parameters and are not restricted to midpoint knot insertion.

Remark 4. When $m = 0$ and $n \neq 0$, the blending function $\omega(\lambda, x)$ does not vanish at x_i and x_{i+1}, and thus $C(\lambda_i, x)$ interpolates only the values of $B_{i-1}(x)$ and $B_i(x)$ at x_i and x_{i+1}, respectively. In this case $\omega(\lambda, \bar{x}_i)$ is a piecewise polynomial blending function analogous to (14) but, differently from (14), it also contains the edge parameter λ.

We observe now that, if the rules for arbitrary point insertion are applied only to a limited number of refinement steps, the smoothness properties of the limit curve are determined by the corresponding midpoint scheme. Thus, as proved in Section 3, the described method will generate C^1 limit curves for any arbitrary choice of the edge parameters except in the neighborhood of those initial vertices p_i that separate subsequent edges with parameters $\lambda_{i-1} = 1$ and $\lambda_i = 0$.

Given an initial polyline, the corresponding subdivision algorithm can be outlined as follows. First, a suitable blending function $\omega(\lambda, x)$ needs to be selected and the corresponding arbitrary point insertion scheme is derived. Also the midpoint scheme corresponding to $\omega(\lambda, \bar{x}_i)$ is computed. For a limited number of steps $k < \bar{k}$ the arbitrary point insertion refinement algorithm is applied. Successively, for any $k \geq \bar{k}$ the k-th polyline is refined through the corresponding midpoint scheme.

Indeed, any $\omega(\lambda, x)$ of the kind (17) can be conveniently assumed as a blending function in (18). However, to illustrate the above procedure, let us consider the scheme with $\omega(\lambda, x)$ where $m = n = 1$, namely

$$\omega(\lambda, x) = \begin{cases} \dfrac{1}{2}\,\dfrac{(1-\lambda)(x-x_i)}{\lambda(x_{i+1}-x)} & \text{if } \dfrac{x-x_i}{x_{i+1}-x_i} < \lambda \\[2ex] 1 - \dfrac{1}{2}\,\dfrac{\lambda(x_{i+1}-x)}{(1-\lambda)(x-x_i)} & \text{if } \dfrac{x-x_i}{x_{i+1}-x_i} \geq \lambda. \end{cases} \tag{19}$$

Now, if we set $\xi = \frac{x-x_i}{x_{i+1}-x_i}$, for any arbitrary x the scheme derived from (18) can be rewritten as a linear combination of the four points $p_{i-1}, p_i, p_{i+1}, p_{i+2}$ with coefficients

$$\begin{aligned}
c_{0,i} &:= c_0(\lambda_i, d_{i-1}, d_i, d_{i+1}, x) = \frac{d_i^2 \xi(\lambda_i(\xi-2)+\xi)}{2\lambda_i d_{i-1}(d_i+d_{i-1})}\\
c_{1,i} &:= c_1(\lambda_i, d_{i-1}, d_i, d_{i+1}, x)\\
&= -\frac{\xi^2 d_i(\lambda_i(d_i-d_{i-1}+d_{i+1})+d_i+d_{i-1}+d_{i+1})+2\xi\lambda_i(d_{i-1}-d_i)(d_i+d_{i+1})-\lambda_i 2d_{i-1}(d_i+d_{i+1})}{2\lambda_i d_{i-1}(d_i+d_{i+1})}\\
c_{2,i} &:= c_2(\lambda_i, d_{i-1}, d_i, d_{i+1}, x) = -\frac{\xi(\xi d_i(\lambda_i(d_i+d_{i-1}-d_{i+1})-d_i-d_{i-1}-d_{i+1}))-2\lambda_i d_{i-1}d_{i+1}}{2\lambda_i d_{i+1}(d_i+d_{i-1})}\\
c_{3,i} &:= c_3(\lambda_i, d_{i-1}, d_i, d_{i+1}, x) = \frac{d_i^2(\lambda_i-1)\xi^2}{2\lambda_i d_{i+1}(d_i+d_{i+1})}
\end{aligned} \tag{20}$$

if $\lambda_i \geq \xi$ and

$$\begin{aligned}
c_{0,i} &:= c_0(\lambda_i, d_{i-1}, d_i, d_{i+1}, x) = \frac{d_i^2 \lambda_i(\xi-1)^2}{2(\lambda_i-1)d_{i-1}(d_i+d_{i-1})}\\
c_{1,i} &:= c_1(\lambda_i, d_{i-1}, d_i, d_{i+1}, x)\\
&= -\frac{(\xi-1)(\xi d_i(\lambda_i(d_i-d_{i-1}+d_{i+1})+2d_{i-1})-2d_{i-1}(d_i+d_{i+1})-\lambda_i((d_i-d_{i-1})(d_i+d_{i+1})-d_{i-1}d_{i+1}))}{2(\lambda_i-1)d_{i-1}(d_i+d_{i+1})}\\
c_{2,i} &:= c_2(\lambda_i, d_{i-1}, d_i, d_{i+1}, x)\\
&= -\frac{\xi^2 d_i(\lambda_i(d_i+d_{i-1}-d_{i+1})-2(d_{i-1}+d_i))+2\xi((d_{i-1}+d_i)(d_i+d_{i+1})-\lambda_i d_{i-1}d_{i+1})-\lambda_i d_i(d_i+d_{i-1}+d_{i+1})}{2(\lambda_i-1)d_{i+1}(d_i+d_{i-1})}\\
c_{3,i} &:= c_3(\lambda_i, d_{i-1}, d_i, d_{i+1}, x) = \frac{d_i^2(\xi-1)((\lambda_i-1)(\xi+1)-(\xi-1))}{2(\lambda_i-1)d_{i+1}(d_i+d_{i+1})}
\end{aligned} \tag{21}$$

if $\lambda_i < \xi$, where $\lambda_i^0 := \omega(\lambda_i, x)$. For the ease of notation, in the above expressions we have dropped the superscript index k, as it is obvious that we refer to the generic k-th iteration.

Note that the set of coefficients $c_{0,i}, c_{1,i}, c_{2,i}, c_{3,i}$ in (21) can be obtained by applying the transformation

$$\lambda_i \to 1-\lambda_i, \qquad \xi \to 1-\xi, \qquad d_{i-1} \to d_{i+1}, \qquad d_{i+1} \to d_{i-1}$$

to the coefficients $c_{3,i}, c_{2,i}, c_{1,i}, c_{0,i}$ in (20), and that the function $\omega(\lambda, \bar{x}_i)$ related to (19) is given by

$$\omega(\lambda) = \begin{cases} \dfrac{2-3\lambda}{2(1-\lambda)} & \text{if } \lambda \in \left[0, \dfrac{1}{2}\right) \\ \dfrac{1-\lambda}{2\lambda} & \text{if } \lambda \in \left[\dfrac{1}{2}, 1\right]. \end{cases} \tag{22}$$

In particular, we remark that, in the case of midpoint insertion, the refinement rules corresponding to (22) coincide with the proposal in [7].

In the remainder of this section we describe by an example one of the possible applications of the discussed arbitrary point insertion method.

We preliminarily observe that the possibility of inserting new points at arbitrary locations can be used to make the scheme uniform around a specified vertex.

Fig. 9. Local configuration of parameters to make the scheme uniform around the point x_i at levels $k = 0$ (left) and $k = 1$ (right).

Focusing on the point x_i in Fig. 9, the procedure can be outlined by a simple algorithm as follows. We first determine the shortest interval containing x_i, in this case $[x_i, x_{i+1}]$, and its parameter d_i. After the refinement, the considered interval will be split into two subintervals, both having parameters $\frac{1}{2}d_i$. We now want to split the longest interval, i.e. $[x_{i-1}, x_i]$, into two subintervals so that the one still containing x_i has parameter $\frac{1}{2}d_i$. To this aim we compute the location x that splits the interval $[x_{i-1}, x_i]$ into two subintervals, proportionally to the values $d_{i-1} - \frac{d_i}{2}$ and $\frac{d_i}{2}$, namely

$$x = \frac{d_i}{2d_{i-1}}x_{i-1} + \left(1 - \frac{d_i}{2d_{i-1}}\right)x_i$$

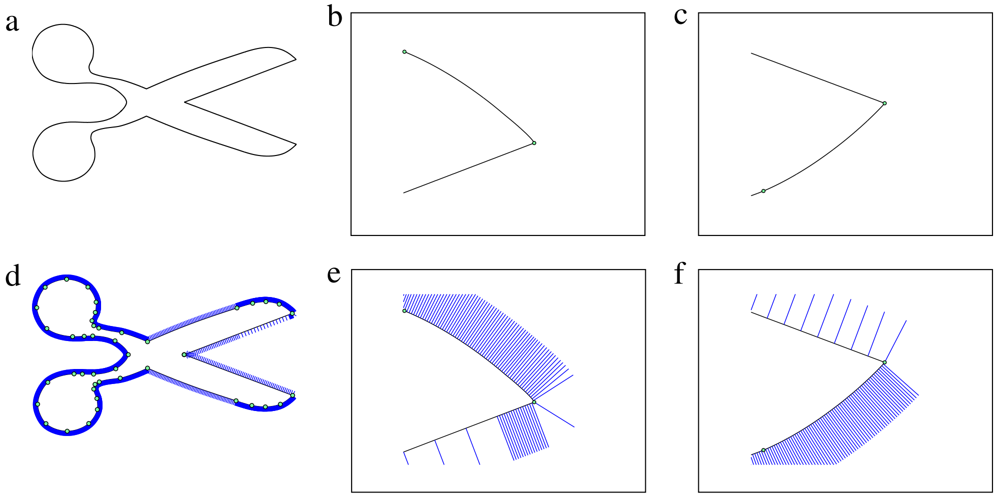

Fig. 10. Comparisons between the effects of selective point insertion ((b), (e)) and midpoint insertion ((c), (f)).

and, after inserting a new point at x, we assign the parameters to the new edges as illustrated in Fig. 9. After one more subdivision step, due to the updating relation (7), the scheme will become uniform around x_i and will also be uniform from this iteration onward (see Remark 1).

Fig. 10 illustrates how, despite arbitrary point insertion, the features of the limit curve are preserved. This example was generated using Eqs. (20)–(21) for arbitrary point insertion and (22) for midpoint refinement. Let us focus the attention on the two symmetric vertices, corresponding to the tip of the scissors (sub-figures (b)–(e) and (c)–(f)). The edge parameters allowed us to obtain a flat edge, smoothly (C^1) connected with a curved segment. Taking into account the orientation of the curve, the parameters' configuration is the same for the two considered vertices; in particular, for the vertex in (b) we have set $\lambda_i = 1$ (on the flat edge) and $\lambda_{i+1} = 0.3$ on the subsequent edge, so as to precisely model the tip. Analogously, for the vertex in (c), $\lambda_i = 0.7$ and $\lambda_{i+1} = 1$ (on the flat edge). Through proper point insertion, the vertex in (b) has become uniform in a few steps; conversely, the vertex in (c) has been refined from the beginning with standard midpoint insertion. To better visualize the location of points, Figures (e) and (f) show the comb of the normals to the vertices of the polyline after six refinement iterations. This example emphasizes also the fact that selective point insertion can be used to modify the parameterization at a critical zone of the polyline, improving the visual quality of the limit curve. In fact, as proven in Proposition 5, the limit curve is C^1 continuous at the considered vertices. However, in the neighborhood of the vertex in (c), the bad parameterization deriving from the remarkable difference in length between the initial edges erroneously suggests that the curve be only C^0 at the vertex. Conversely, this undesired visual effect is not noticeable around the vertex in (b).

6. Generalization to subdivision surfaces on quadrilateral meshes

The work done in the univariate case maps readily into the definition of tensor product subdivision surfaces with features, defined on quadrilateral meshes. As suggested in [21], the possibility of handling surface features is fundamental to implement subdivision schemes of practical use in applications. The considered example (Fig. 11) illustrates the inclusion of creases, cusps and flat faces in a regular torus model. All these shape effects have been obtained through the tensor product of the proposed non-uniform 4-point rules with edge parameters, where the parameters λ_i are defined on each edge of the starting mesh and the handling of edge parameters described in Section 4 has been properly generalized.

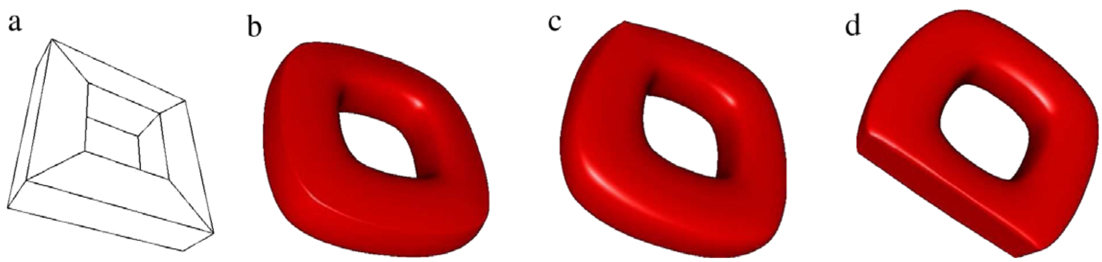

Fig. 11. The effect of setting the edge parameters λ_i in order to generate creases (b), cusps (c), flat faces (d).

7. Conclusions and ongoing research

We have introduced a blending-type formulation of polynomial-based interpolatory 4-point schemes and a related novel class of 4-point refinement rules that turn out to be very attractive in practical applications. In fact, besides possessing the capability of generating visually-pleasing limit curves, they include edge parameters for handling boundaries, selectively reducing continuity, and integrating features like cusps and flat edges when desired. Additionally, such a family of subdivision schemes is the only one capable of producing a piecewise polynomial curve passing through the initial vertices, even if they are not its samples. In particular, piecewise quadratic polynomials that join C^0 continuously at the given points can be automatically produced by a suitable setting of the edge parameters, and piecewise C^1 quadratic polynomials can be further obtained after the application of a preprocessing step aimed at inserting properly chosen vertices in the initial polyline. Finally, the proposed schemes can be used to refine a given control polygon in a selective way, since they allow for arbitrarily specifying the locations of the inserted points. To the authors knowledge, existing interpolatory schemes do not present at the same time all the aforementioned properties.

Furthermore, the proposed algorithm provides the univariate foundations for a novel non-uniform interpolatory surface subdivision scheme with features control. We have presented a first generalization to tensor product of the univariate 4-point scheme with edge parameters, tailored for regular quadrilateral control meshes. There is in fact no point in making ad hoc rules for extraordinary vertices without having a firm foundation to build on. A further extension of this work to quadrilateral meshes with arbitrary topology will be a topic for future research.

Acknowledgements

This research was supported by the University of Bologna and the University of Milano-Bicocca, Italy.

The authors thank the anonymous referees for their useful suggestions.

References

[1] T. Cashman, U.H. Augsdorfer, N.A. Dodgson, M.A. Sabin, NURBS with extraordinary points: high-degree, non-uniform, rational subdivision schemes, in: Proceedings of ACM SIGGRAPH, ACM Trans. Graph. (TOG) 28 (3) (2009).
[2] T. Cashman, N.A. Dodgson, M.A. Sabin, Selective knot insertion for symmetric, non-uniform refine and smooth B-spline subdivision, Comput. Aided Geom. Design 26 (4) (2009) 472–479.
[3] K. Müller, L. Reusche, D. Fellner, Extended subdivision surfaces: building a bridge between NURBS and Catmull-Clark surfaces, ACM Trans. Graph. 25 (2) (2006) 268–292.
[4] T.W. Sederberg, J. Zheng, D. Sewell, M. Sabin, Non-uniform recursive subdivision surfaces, in: Proceedings of the 25th Conference on Computer Graphics and Interactive Techniques, 1998, pp. 387–394.
[5] G. Deslauriers, S. Dubuc, Symmetric iterative interpolation processes, Constr. Approx. 5 (1989) 49–68.
[6] S. Dubuc, Interpolation through an iterative scheme, J. Math. Anal. Appl. 114 (1986) 185–204.
[7] C. Beccari, G. Casciola, L. Romani, Non-uniform interpolatory curve subdivision with edge parameters built-upon compactly supported cardinal splines. (submitted for publication).
[8] N. Dyn, M. Floater, K. Hormann, Four-point curve subdivision based on iterated chordal and centripetal parameterizations, Comput. Aided Geom. Design 26 (3) (2009) 279–286.
[9] I. Daubechies, I. Guskov, W. Sweldens, Regularity of irregular subdivision, Constr. Approx. 15 (3) (1999) 381–426.
[10] J. Warren, Binary subdivision schemes for functions over irregular knot sequences, in: M. Dæhlen, T. Lyche, L.L. Schumaker (Eds.), Mathematical Methods in CAGD III, Vanderbilt University Press, Nashville (TN), 1995, pp. 543–562.
[11] J. Warren, H. Weimer, Subdivision Methods for Geometric Design: a Constructive Approach, Morgan Kaufmann, San Francisco, 2002.
[12] G. Li, W. Ma, A new interpolatory subdivision for quadrilateral meshes, Comput. Graph. Forum 24 (1) (2005) 3–16.
[13] G. Farin, Curves and Surfaces for Computer-Aided Geometric Design: A Practical Guide, 4th edition, Academic Press, San Diego, 1997, pp. 81–83.
[14] A. Levin, Polynomial generation and quasi-interpolation in stationary non-uniform subdivision, Comput. Aided Geom. Design 20 (1) (2003) 41–60.
[15] A. Levin, The importance of polynomial reproduction in piecewise uniform subdivision, in: R. Martin, H. Bez, M.A. Sabin (Eds.), Mathematics of Surfaces XI (Proceedings of 11th IMA International Conference), Springer, 2005, pp. 272–307.
[16] W. Dahmen, T.N.T. Goodman, C.A. Micchelli, Compactly supported fundamental functions for spline interpolation, Numer. Math. 52 (1988) 639–664.
[17] C. Beccari, G. Casciola, L. Romani, A non-stationary uniform tension controlled interpolating 4-point scheme reproducing conics, Comput. Aided Geom. Design 24 (2007) 1–9.
[18] C. Beccari, G. Casciola, L. Romani, Shape controlled interpolatory ternary subdivision, Appl. Math. Comput. 215 (2009) 916–927.
[19] E. Catmull, R. Rom, A class of local interpolating splines, in: R.E. Barnhill, R.F. Riesenfeld (Eds.), Computer Aided Geometric Design, Academic Press, New York, 1974, pp. 317–326.
[20] C.K. Chui, J.M. De Villiers, Applications of optimally local interpolation to interpolatory approximants and compactly supported wavelets, Math. Comp. 65 (213) (1996) 99–114.
[21] H. Hoppe, Surface reconstruction from unorganized points, Thesis, University of Washington, 1994.