Some applications of new spline spaces
in computer aided geometric design

PAOLO COSTANTINI – CARLA MANNI

ABSTRACT: *Aim of this paper is to describe how the so called Variable Degree Polynomial Spaces can be used for the construction of C^3 spatial curves, approximating or interpolating a given set of data. Their main advantages rely in the easy control on their shape, provided by the variable degrees, and in the low computational cost, comparable with that of standard quintic splines.*

1 – Introduction

Geometric continuous curves and surfaces based on polynomial or rational splines constitute the main tool of Computer Aided Geometric Design because of their simplicity and because of the easy and intuitive control on their shape provided by the so-called shape parameters. However, in some CAD/CAM applications, as, for instance, in the description of the motion of a milling machine, the physical meaning of the parameter is not negligible and a certain order of analytic continuity is often required; therefore new tools which encompass the new and the old requests would be highly desirable.

Aim of this paper is to describe the properties and some applications of new *quintic-like* spline spaces (called Variable Degree Polynomial Spaces, VDPS for short) which permit the construction of C^3 polynomial (or rational) curves and surfaces with the same simplicity, computational cost and ease of shape control as the classical quintics. Indeed, these spaces are isomorphic to the spaces of C^3

quintic splines and possess a control polygon (called *pseudo Bézier control net*) with all the usual geometric properties. Therefore, all the geometric construction that are used in CAGD can be repeated. Additionally, the degrees play the role of tension parameters, since their large values force the curve to have a piecewise linear appearance.

This paper is divided in five sections. In the next one the structure of VDPS will be briefly recalled and in Section 3 we will describe a simplification of the geometric construction for C^4 quintic splines, which is suitable for our purposes. Section 4 is devoted to show applications of VDPS in the interpolation and approximation of ordered spatial data. In the last section are reported some concluding remarks and open problems.

It is worthwhile to say that this paper has a structure very similar to [5]; in that paper an analogous geometric construction, also derived from C^4 quintic splines, is used to produce C^2 quintic splines with a third order *Frenet continuity* ($C^2 - FC^3$) – that is curvature and torsion continuous – splines. The advantages and disadvantages of [5] and of the present paper are, roughly speaking, symmetric and with an equivalent comprehensive effect: here, we have an higher continuity order (C^3 implies FC^3) at the price of the more complex space structure induced by the degrees; there, a lack in the continuity with the advantage of low degree splines, which so far constitute the standard *mathematical engines* of CAD/CAM environments.

2 – The spline space

In this section we want to briefly introduce the main properties of the C^3 *quintic-like* VDPS, referring for details to [4] and [7]. Let $\{u_0, u_1, \dots, u_m\}$ be an ordered knot sequence, let h_i, $i = 0, \dots, m-1$, be the knot spacing, and let

$$\mathbf{k} = \{k_i; \ i = 0, 1, \dots, m\},$$

with $k_i \geq 5$ be a given sequence of integers. For each interval $[u_i, u_{i+1}]$ we consider the six dimensional polynomial space:

$$VP_{k_i, k_{i+1}} := span\{(1-v), v, (1-v)^{k_i}, v(1-v)^{k_i-1}, v^{k_{i+1}-1}(1-v), v^{k_{i+1}}\}$$

with $v = (u - u_i)/h_i$, called *quintic-like variable degree polynomial space*. Denoting by $\mathbb{P}_n$ the space of algebraic polynomials of degree less than or equal to n, we remark that $VP_{k_i, k_{i+1}}$ is isomorphic to $\mathbb{P}_5$ and, in particular, $VP_{5,5} = \mathbb{P}_5$. Moreover, as it is shown in [4], $VP_{k_i, k_{i+1}}$ admits a *pseudo Bernstein-Bézier basis*

$$\{\mathcal{B}_0, \mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4, \mathcal{B}_5\}$$

that is a basis with the usual properties – positivity, partition of unity – as the Bernstein-Bézier basis for $\mathbb{P}_5$. Therefore, for any $b \in VP_{k_i,k_{i+1}}$, we have

$$b = \sum_{j=0}^{5} b_{i,j}\mathcal{B}_{i,j}, \qquad \mathcal{B}_{i,j}(u) = \mathcal{B}_j((u - u_i)/h_i),$$

where the $b_{i,j}$ are called *pseudo Bézier ordinates* (the suffix i refers to the interval $[x_i, x_{i+1}]$) and play the same role as the usual control points for quintic polynomials. In particular, setting

$$\xi_{i,0} := u_i, \;\; \xi_{i,1} := u_i + \frac{h_i}{k_i}, \;\; \xi_{i,2} := u_i + 2\frac{h_i}{k_i},$$

$$\xi_{i,3} := u_{i+1} - 2\frac{h_i}{k_{i+1}}, \;\; \xi_{i,4} := u_{i+1} - \frac{h_i}{k_{i+1}}, \;\; \xi_{i,5} := u_{i+1}.$$

we have

$$u = \sum_{j=0}^{5} \xi_{i,j}\mathcal{B}_{i,j}(u).$$

For details we refer to [4].

Now, let us consider

$$VS_{\mathbf{k}} := \{s \in C^3[u_0, u_m] \text{ s.t. } s|_{[u_i, u_{i+1}]} \in VP_{k_i,k_{i+1}}\}$$

the space of quintic-like VDPS.

In [7] it is shown that $VS_{\mathbf{k}}$ admits a basis,

$$\{N_{2i}, N_{2i+1}; \; i = -1, 0, \dots, m\}$$

defined, as usual, on an extended knot sequence

$$u_{-2} < u_{-1} < u_0 < u_1 < \cdots < u_m < u_{m+1} < u_{m+2},$$

having the classical properties of the B-spline basis.

The tension effect achieved with large degree values clearly appears from the plots of the B-spline basis functions shown in Figure 1; indeed, if the degrees tend simultaneously to infinity, the B-splines tend to the normalized piecewise linear B-splines.
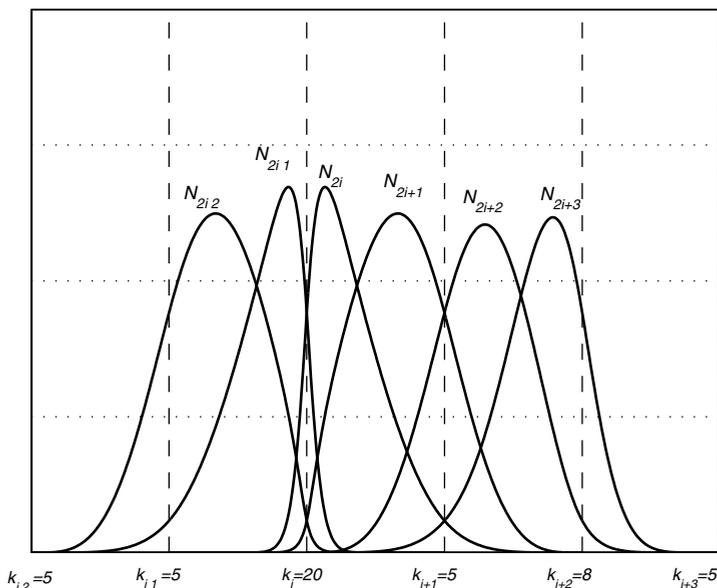
Fig. 1: Some B-spline basis functions.

3 – Geometric construction

We consider in this section variable degree spline curves in $\mathbb{R}^3$, that is

$$\mathbf{VS_k} = \{\mathbf{s} \ \text{ s.t. } \mathbf{s} : [u_0, u_m] \to \mathbb{R}^3, \ \text{ has components in } VS_\mathbf{k}\} \ .$$

For explaining our ideas, we start with standard C^4, quintic spline curves which can be seen as a particular case of VDPS with $k_0 = k_1 = \ldots = k_m = 5$. If we denote with $\{\tilde{N}_i, \ i = -2, -1, \ldots, m+2\}$ the sequence of normalized quintic B-splines of class C^4 and take a sequence of coefficients (often referred to as *de Boor control points*) $\{\mathbf{D}_i, \ i = -2, -1, \ldots, m+2\}$, a C^4 quintic spline curve can be expressed as

$$\mathbf{s} = \sum_{i=-2}^{m+2} \mathbf{D}_i \tilde{N}_i, \quad \mathbf{D}_i \in \mathbb{R}^3.$$

Obviously, $\mathbf{s}_i := \mathbf{s}|_{[u_i.u_{i+1}]}$ can be expressed in the Bernstein-Bézier form

$$\mathbf{s}_i = \sum_{n=0}^{5} \mathbf{b}_{i,n} \mathcal{B}_{i,n} \ ,$$

the coefficients $b_{i,j}$ are called *Bézier control points*.

One of the most attractive features of spline curves is their *geometric construction*, that is the possibility of constructing the Bézier control points $\mathbf{b}_{i,n}$ directly from the de Boor control points $\mathbf{D}_j$ via a *corner-cutting* process. This geometric construction can be schematically divided in two main steps (explained in Figures 2.a and 2.b). In the first one (see Figure 2.a) the polygonal legs $\mathbf{D}_i\mathbf{D}_{i+1}$, $i = -2, -1, \dots, m + 1$ are divided in three segments proportional to $h_{i-2} + h_{i-1}$, h_i and $h_{i+1} + h_{i+2}$ and the additional points $\mathbf{F}_i^+$, $\mathbf{F}_{i+1}^-$ are inserted; then for $i = -1, 0, \dots, m + 1$ the segment $\mathbf{F}_i^-\mathbf{F}_i^+$ is divided in three subsegments proportional to h_{i-2}, $h_{i-1} + h_i$, h_{i+1} and the points $\mathbf{p}_i$, $\mathbf{r}_i$ are placed on it. In the second step (see Figure 2.b) the segments $\mathbf{r}_i\mathbf{p}_{i+1}$, $i = -1, 0, \dots, m + 1$ are subdivided with proportionality h_{i-1}, h_i, h_{i+1} and the Bézier control points $\mathbf{b}_{i,2}$, $\mathbf{b}_{i,3}$ are inserted; then, for $i = -1, 0, \dots, m + 1$ the point $\mathbf{q}_i$ is inserted in $\mathbf{p}_i\mathbf{r}_i$ with proportionality h_{i-1} and h_i; finally, the same factors are used to insert the control points $\mathbf{b}_{i-1,4}$, $\mathbf{b}_{i,1}$ in the segments $\mathbf{b}_{i-1,3}\mathbf{q}_i$, $\mathbf{q}_i\mathbf{b}_{i,2}$ and $\mathbf{b}_{i-1,5} = \mathbf{b}_{i,0}$ in the segment $\mathbf{b}_{i-1,4}\mathbf{b}_{i,1}$ respectively. Note that this procedure (which is mathematically proved using the subdivision scheme given by the De Casteljau algorithm) automatically constructs C^4 curves. We refer to to [13], [8], [19] for the formal details.

Now let us consider curves in the more general space $\mathbf{VS_k}$. A spline curve $\mathbf{s} \in \mathbf{VS_k}$ can be expressed as

$$\mathbf{s} = \mathbf{r}_{-1}N_{-2} + \sum_{i=0}^{m}(\mathbf{p}_iN_{2i-1} + \mathbf{r}_iN_{2i}) + \mathbf{p}_{m+1}N_{2m+1} \ .$$
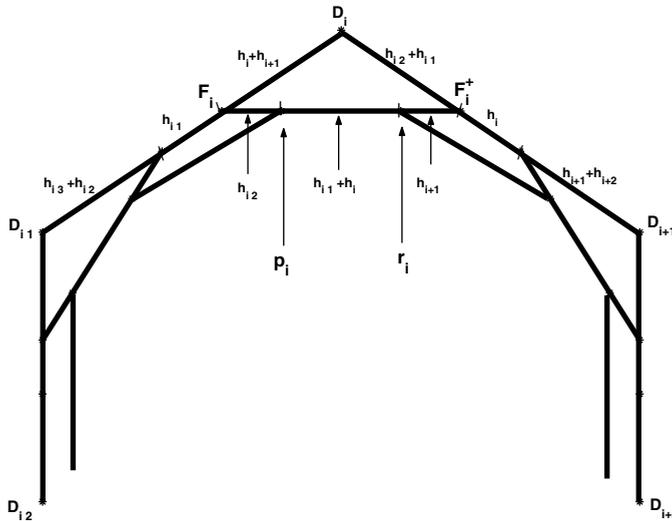
Fig. 2a: Geometric construction of C^4 quintic splines. First step.
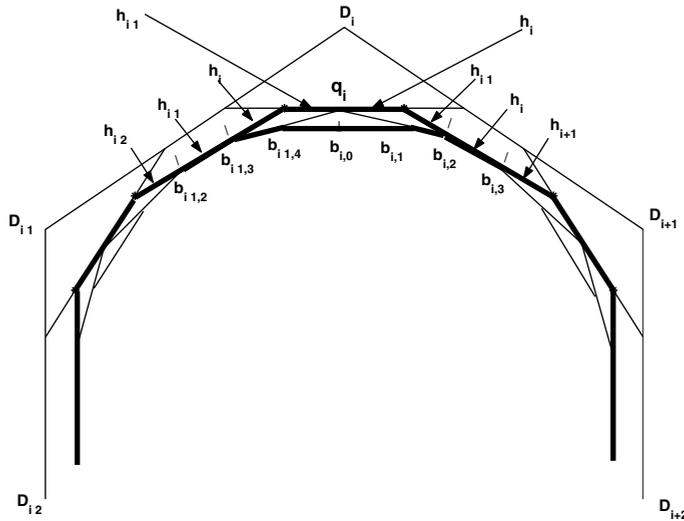
Fig. 2b: Geometric construction of C^4 quintic splines. Second step.

Obviously, $\mathbf{s}_i := \mathbf{s}|_{[u_i.u_{i+1}]}$ can be expressed in the Bernstein-Bézier form

$$\mathbf{s}_i = \sum_{n=0}^{5} \mathbf{b}_{i,n} \mathcal{B}_{i,n} \ ,$$

the coefficients $b_{i,j}$ are called *pseudo Bézier control points.*

In [4] it is shown that a geometric construction similar to that one illustrated in Figure 2.b holds also for the general case. More specifically it is possible to construct a C^3 curve belonging to $\mathbf{VS_k}$ starting from a control polygon connecting the control points $\mathbf{p}_i$, $\mathbf{r}_i$ as specified in Figure 3. The remarkable fact of this construction is that, for a large value of the degree k_i, both the points $\mathbf{p}_i$, $\mathbf{r}_i$ and the pseudo Bézier control points $\mathbf{b}_{i-1,3}, \ldots, \mathbf{b}_{i,2}$ are attracted by the central point $\mathbf{q}_i$. In other words, the degrees play the role of *tension parameters* and the shape of the curve can be easily modified to reach a piecewise linear appearance; in practice we have the same shape control as for the *geometric continuous* ([13]) splines with the advantage of maintaining the analytical continuity. It is worthwhile to recall that the computational cost does not depend on the degrees and is approximately the same as the quintic one. See [4] for details.

However, in this construction we have two control points associated to each knot and, instead of being an advantage, this flexibility implies the additional difficulty of choosing the slope of the segment $\mathbf{p}_i\mathbf{r}_i$.

The idea of this paper is very simple: to consider the points $\mathbf{r}_i$, $\mathbf{p}_i$ as obtained from the first corner-cutting step embedding the construction of C^3 VDPS of
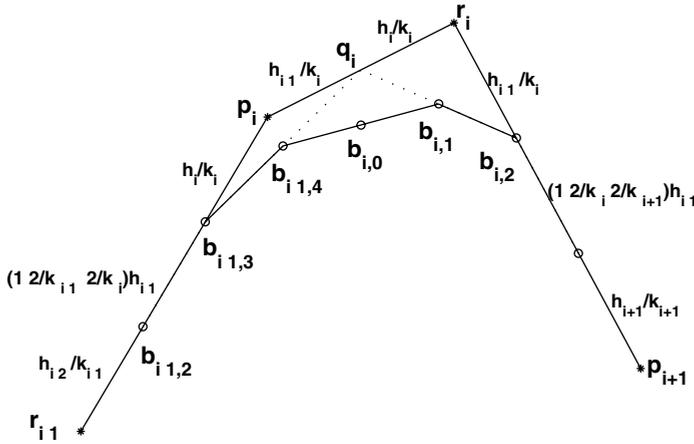
Fig. 3: Geometric construction of C^3 VDPS splines.

Figure 3 in the quintic C^4 scheme of Figures. 4. Thus, only *one* control point is associated to each knot. Of course in this way we are dealing with a subspace of $\mathbf{VS_k}$. The simplified geometric construction of the elements of this subspace is a consistent advantage both for their use in interpolation/approximation of spatial data and in free form design. We refer to [4], [7] for a comparison. More
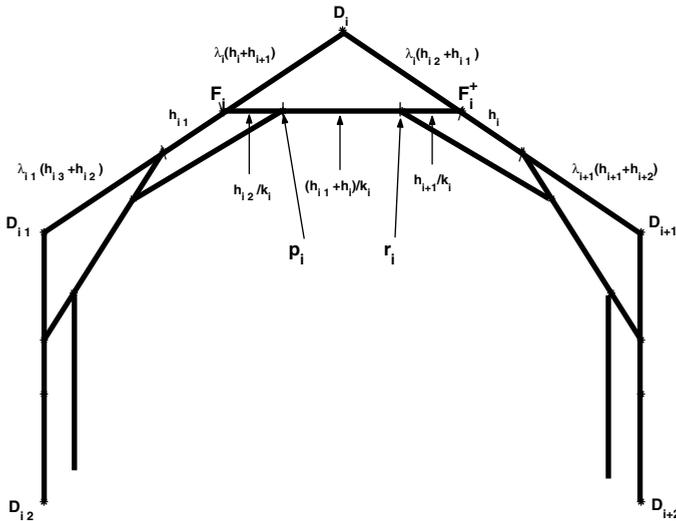
Fig. 4a: Geometric construction of C^3 VDPS. First step.
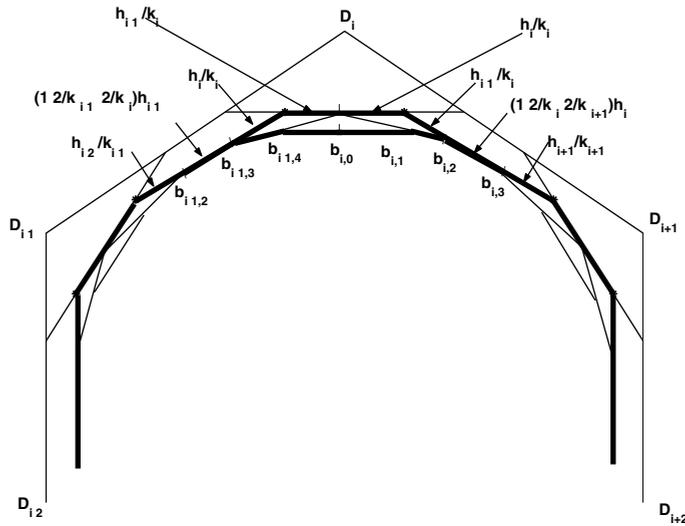
Fig. 4b: Geometric construction of C^3 VDPS. Second step.

specifically, the second step of the corner-cutting remains unchanged (see Figures. 3 and 4.b) while in the first step we introduce at each knot a new shape parameter, λ_i (see Figures. 2.a and 4.a). Obviously, we have again quintic C^4 splines for the choice $\lambda_i = 1$, $k_i = 5$, all i.

Since the polygonal legs $\mathbf{D}_i\mathbf{D}_{i+1}$ are divided in three segments proportional to $\lambda_i(h_{i-2} + h_{i-1})$, h_i and $\lambda_{i+1}(h_{i+1} + h_{i+2})$ it is clear that the points $\mathbf{F}_i^-$ and $\mathbf{F}_i^+$ are attracted by $\mathbf{D}_i$ for small values of λ_i. Therefore, the combined effect of small λ_i and large k_i produces a tension effect on the final curve. See Figure 5 for a graphical example, where we have chosen $\lambda_i = 1/k_i$ (obviously this is just one among the possible choices: λ_i and k_i can be chosen independently).

4 – Applications

The researches described in this paper have been mainly motivated by the necessity of constructing interpolating (for CAD/CAM applications) or approximating (for some reverse engineering applications) curves capable of maintaining the geometric characteristics (discrete curvature and discrete torsion [20]) of the data set.

4.1 – Interpolation of spatial data

We start with a brief description of the interpolation problem, referring to [1], [2], [3], [4], [10], [11], [12], [15], [16], [17] for related papers.
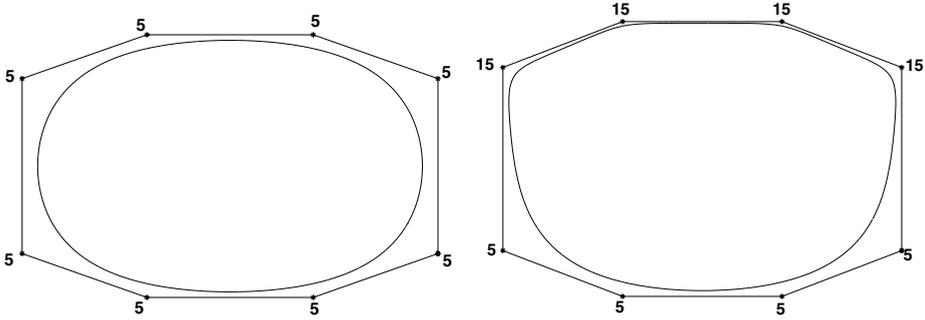
Fig. 5: Left: an example of C^4 quintic spline curve. Right: an example of C^3 VDPS curve. The numbers indicate the degree k_i associated to each de Boor control point

Let
$$\mathbf{I}_i \in \mathbb{R}^3, \qquad i = 0, \dots, m,$$

be the interpolation points with $\mathbf{I}_i \neq \mathbf{I}_{i+1}$. For a given matrix M let $|M| := \det(M)$. Define, for all admissible indices,

$$\mathbf{L}_i = \mathbf{I}_{i+1} - \mathbf{I}_i \ , \hspace{5cm} i = 0, \dots, m-1,$$

$$\mathbf{B}_i := \begin{cases} \dfrac{\mathbf{L}_{i-1} \times \mathbf{L}_i}{\|\mathbf{L}_{i-1}\| \ \|\mathbf{L}_i\|}, & \text{if } \|\mathbf{L}_{i-1}\| \ \|\mathbf{L}_i\| > 0, \\[2mm] 0, & \text{elsewhere,} \end{cases} \hspace{1cm} i = 1, \dots, m-1,$$

$$\Delta_i := \begin{cases} \dfrac{|\mathbf{L}_{i-1} \ \mathbf{L}_i \ \mathbf{L}_{i+1}|}{\|\mathbf{L}_{i-1} \times \mathbf{L}_i\| \ \|\mathbf{L}_i \times \mathbf{L}_{i+1}\|}, & \text{if } \|\mathbf{L}_{i-1} \times \mathbf{L}_i\| \|\mathbf{L}_i \times \mathbf{L}_{i+1}\| > 0, \\[2mm] 0, & \text{elsewhere,} \end{cases} \hspace{0.4cm} i = 1, \dots, m-2,$$

The vectors $\mathbf{B}_i$ are the *discrete binormals* and the scalars Δ_i are the *discrete torsion* of the data ([20]).

Given a spline curve $\mathbf{s} = \mathbf{s}(u)$, we consider the corresponding *curvature vector* $\mathbf{K}(u)$ and the *torsion* $\tau(u)$:

$$\mathbf{K}(u) := \frac{\mathbf{s}'(u) \times \mathbf{s}''(u)}{\|\mathbf{s}'(u)\|^3}, \hspace{1.5cm} \text{if } \mathbf{s}'(u) \neq \mathbf{0} \ ,$$

$$\tau(u) := \frac{|\mathbf{s}'(u) \ \mathbf{s}''(u) \ \mathbf{s}'''(u)|}{\|\mathbf{s}'(u) \times \mathbf{s}''(u)\|^2}, \hspace{1cm} \text{if } \mathbf{s}'(u) \times \mathbf{s}''(u) \neq \mathbf{0},$$

and following the usual definitions (see, e.g., [1], [11], [7]), we formally define the *shape-constraints* (for a geometric interpretation the reader is referred, for example, to [1]). Let us denote with $\mathbf{I}$ the polygonal line connecting the data points $\{\mathbf{I}_0, \dots, \mathbf{I}_m\}$.

DEFINITION 1. Let $\mathbf{s}(u)$ be a spline curve defined for $u \in [u_0, u_m]$ and let ϵ_1, ϵ_2 two real, positive tolerances. We say that $\mathbf{s}(u)$ is $\mathbf{I}$-shape preserving if the following criteria are satisfied:

(i) *Weak collinearity criteria*

If $\|\mathbf{B}_i\| \le \epsilon_1$ and $\mathbf{L}_{i-1} \cdot \mathbf{L}_i > 0$ then $\left\| \dfrac{\mathbf{s}'(u)}{\|\mathbf{s}'(u)\|} \times \dfrac{\mathbf{L}_j}{\|\mathbf{L}_j\|} \right\| \le \epsilon_2,$

$j = i - 1, i$, in each arbitrary but fixed closed subinterval of (u_{i-1}, u_{i+1}) where $\|\mathbf{s}'(u)\| \ne 0$.

(ii) *Convexity criteria*

(ii.1) If $\|\mathbf{B}_i\| \ne 0$, then $\mathbf{K}(u_i) \cdot \mathbf{B}_i > 0$.

(ii.2) If $\mathbf{B}_i \cdot \mathbf{B}_{i+1} > 0$, then $\mathbf{K}(u) \cdot \mathbf{B}_j > 0, j = i, i+1$,
$u \in (u_i, u_{i+1})$.

(iii) *Weak coplanarity criteria*

If $|\Delta_i| \le \epsilon_1$ then $\left\| \dfrac{\mathbf{s}'(u) \times \mathbf{s}''(u)}{\|\mathbf{s}'(u) \times \mathbf{s}''(u)\|} \times \mathbf{B}_i \right\| \le \epsilon_2$, $u \in [u_i, u_{i+1}]$, if $\|\mathbf{K}(u)\| \ne 0$.

(iv) *Torsion criteria*

(iv.1) If $\Delta_{i-1}\Delta_i > 0$ then $\tau(u_i)\Delta_j > 0, j = i - 1, i$.

(iv.2) If $\Delta_i \ne 0$ then $\tau(u)\Delta_i > 0$, n each arbitrary but fixed closed subinterval of (u_i, u_{i+1}).

Our goal is to construct an $\mathbf{I}$-shape preserving interpolating spline, that is

$$\mathbf{s} \in \mathbf{VS_k} \text{ such that } \mathbf{s}(u_i) = \mathbf{I}_i \ , \ i = 0, \dots, m$$

which satisfies conditions (i)-(iv) of Definition 1.

To uniquely solve the problem, four conditions must be added. Following [5], in the case of **closed curves**, we assume that $\mathbf{I}_0 = \mathbf{I}_m$ and we impose

$$\mathbf{D}_{-2} = \mathbf{D}_{m-2} \ , \ \mathbf{D}_{-1} = \mathbf{D}_{m-1} \ ; \mathbf{D}_{m+1} = \mathbf{D}_1 \ , \ \mathbf{D}_{m+2} = \mathbf{D}_2 \ .$$

The problem is more complex in the case of **open curves**; there are indeed many possible solutions which strictly depend on the applications and on the user desires. Very simple equations are

$$\mathbf{D}_{-2} = \mathbf{D}_{-1} = \mathbf{D}_0 \ ; \mathbf{D}_{m+2} = \mathbf{D}_{m+1} = \mathbf{D}_m \ ,$$

which imply that the shape of the curve at the end points is not influenced by extraneous conditions, but only depend on the first interpolation points, or

$$\mathbf{D}_{-1} = \mathbf{D}_0 + \alpha_0(\mathbf{D}_1 - \mathbf{D}_0) \ ,$$
$$\mathbf{D}_{-2} = \mathbf{D}_0 + \alpha_0(\mathbf{D}_1 - \mathbf{D}_0) + \beta_0(\mathbf{D}_2 - \mathbf{D}_1) \ ;$$
$$\mathbf{D}_{m+1} = \mathbf{D}_m + \alpha_m(\mathbf{D}_m - \mathbf{D}_{m-1}) \ ,$$
$$\mathbf{D}_{m+2} = \mathbf{D}_m + \alpha_m(\mathbf{D}_m - \mathbf{D}_{m-1}) + \beta_m(\mathbf{D}_{m-1} - \mathbf{D}_{m-2}) \ ,$$

which produces natural-like end conditions, that is vanishing curvature and torsion at end points.

The theoretical aspects are grounded on the results of [4] and are completely similar to those of [5]; we limit therefore to sketch the main points, avoiding useless duplications.

Let us consider the augmented set of control points $\{\mathbf{D}_{-2}, \dots, \mathbf{D}_{m+2}\}$ (given by proper end conditions). The pseudo Bézier control points $\mathbf{b}_{0,0}, \mathbf{b}_{1,0}, \dots,$ $\mathbf{b}_{m-1,0}, \mathbf{b}_{m,0} := \mathbf{b}_{m,5}$, play a particular role, since $\mathbf{b}_{i,0} = \mathbf{s}(u_i)$, $i = 0, \dots, m$ and therefore the interpolation conditions can be rewritten as $\mathbf{b}_{i,0} = \mathbf{I}_i$, $i = 0, \dots, m$. Now, if we use the corner cutting process described in Figures 4 for their computations, we have linear equations of the form

$$\mathbf{b}_{i,0} = \sum_{j=i-2}^{i+2} \alpha_{i,j} \mathbf{D}_j \ .$$

The coefficients $\alpha_{i,j}$ can be computed using the Maple instructions reported in the appendix; the explicit expressions are extremely cumbersome and are not reported here for reasons of space. Let us denote with A the matrix obtained augmenting the collocation matrix $(\alpha_{i,j})_{0 \leq i \leq m, -2 \leq j \leq m+2}$ with the four rows given by one of the boundary conditions. We have the following result.

THEOREM 1. *For λ_i sufficiently small and k_i sufficiently large, the matrix A is strictly diagonally dominant.*

The proof can be obtained running the Maple program of the appendix. Obviously we have the following corollary.

COROLLARY 1. *For λ_i sufficiently small and k_i sufficiently large, there exists one and only one interpolating spline $\mathbf{s} \in \mathbf{VS_k}$*

We observe that, as in other interpolation problems with geometric continuous curves, we have not a formal proof for the existence of a solution for all λ_i, k_i. Additional results could be obtained, for specified boundary conditions, using a geometric analysis of the null space similar to those presented in [2] and [4]. We remark, however, that in the huge amount of numerical experiments singular matrices have never occurred. Therefore we safely conjecture that A is non-singular for any choice of λ_i, k_i, also supported by the consideration that the geometric structure of the corner cutting described in Figures. 4 – which produces the matrix elements – is not dependent on λ_i and k_i.

Using the same arguments of [5] we have the following asymptotic result.

THEOREM 2.   *Let* $\lambda_i \to 0$ *and* $k_i \to \infty$. *Then* $\mathbf{b}_{\nu,\mu} \to \mathbf{D}_i$, *for* $\nu = i-1, \mu = 3, 4, 5$ *and* $\nu = i, \mu = 0, 1, 2$.

The above theorem says that, for proper values of the shape parameters λ_i, k_i, the pseudo Bézier control net has the same shape of the *pseudo De Boor control net*, that is the polygonal line connecting the $\mathbf{D}_i$. This can be restated saying that both the pseudo Bézier and the pseudo De Boor control nets tend to the polygonal interpolating the data points. Taking a well-known result of [9] and repeating the same considerations of [4] we claim the following proposition.

PROPOSITION 1.   *For* λ_i *sufficiently small and* k_i *sufficiently large the shape induced by the curvature and the torsion of* $\mathbf{s}$ *is the same as the shape induced by the discrete curvature and torsion of the pseudo Bézier control polygon.*

Summarizing we have the following result.

THEOREM 3.   *It is possible to find sequences* $\lambda_0, \dots, \lambda_m$ *and* $k_0, \dots, k_m$ *such that the interpolating spline curve* $\mathbf{s}$ *is* $\mathbf{I}$*-shape preserving.*

Figure 6. (left) shows the plot of the C^4 quintic curve interpolating the so called *chair data*, [15]. For emphasizing the shape effect we have used the uniform, instead of the centripetal arc-length, parameterization. The choice $k_1 = k_{11} = 27$ ($\mathbf{I}_0$ is the highest point) reduces the unwanted inflections, as shown in Figure 6 right.

4.2 – Approximation of spatial data

Now let us turn to shape preserving approximation of spatial data. Despite its practical importance, this argument has received much less attention. For planar data the only papers seems to be [14] and [18] and, for the spatial case, [6], [7] and, partially, [5].

Let $\{(t_j, \mathbf{I}_j), \ j = 0, \dots, N\}$, with $\mathbf{I}_j \in \mathbb{R}^3$ be a set of data points. The first problem we must solve is the definition of the *shape of the data*. Again, for reason
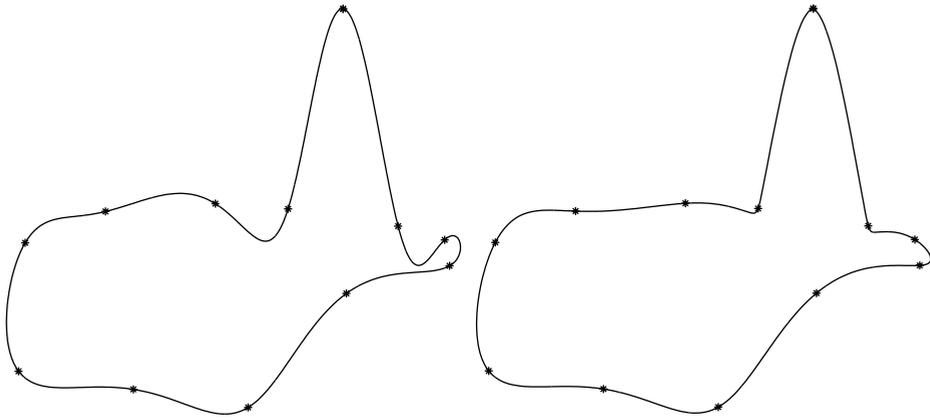
Fig. 6: Chair data. Left: C^4 quintic interpolating spline curve. Right: C^3 interpolating VDPS curve.

of space, we refer to [6] for details. The basic idea goes as follows. We extract from the data parameters a sequence of *significant* knots $\{u_0, u_1, \ldots, u_m\}$ with $u_0 = t_0$, $u_m = t_N$ and we define the space

$$\mathcal{L} := \{\ell \in C[u_0, u_m] \text{ s.t. } \ell|_{[u_i, u_{i+1}]} \text{ has components in } \mathbb{P}_1\}.$$

We then take $\psi \in \mathcal{L}$, the best least squares approximation to data and we simply use the discrete curvature and torsion of ψ for defining the shape of the data; see Figure 7 for a planar example taken from [14].
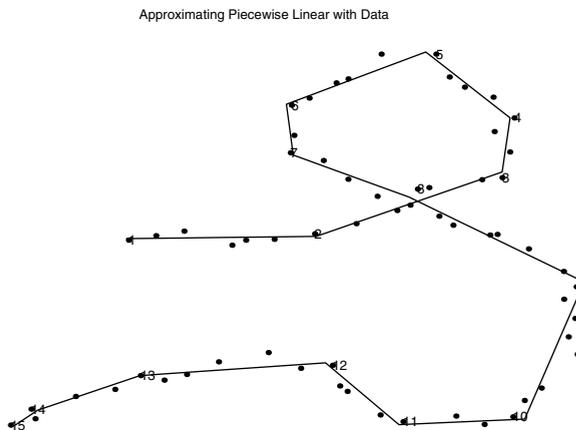
Fig. 7: The shape of a data set.

Our goal is to compute the spline curve of best approximation, that is $\mathbf{s}^*$ such that

$$|\mathbf{s}^* - \mathbf{I}| \leq |\mathbf{s} - \mathbf{I}|, \quad \forall\, \mathbf{s} \in \mathbf{VS_k} \ ,$$

where $|\mathbf{v}|^2 := \sum_{j=0}^{N} \|\mathbf{v}(t_j)\|^2$. Since we are mainly interested in CAD applications, our splines will be constrained to satisfy the boundary conditions of the previous section. However, following the same ideas of [6] and [7], we will not use constrained least square techniques but we simply perform an unconstrained minimization in the subspace

$$\mathbf{V\Sigma}_k := \{\mathbf{s} \in \mathbf{VS_k} \text{ such that } \mathbf{s} \text{ satisfies boundary conditions}\} \ ,$$

seeking for $\boldsymbol{\sigma}^* \in \mathbf{V\Sigma}_k$ such that

$$|\boldsymbol{\sigma}^* - \mathbf{I}| \leq |\boldsymbol{\sigma} - \mathbf{I}|, \quad \forall\, \boldsymbol{\sigma} \in \mathbf{V\Sigma}_k \ .$$

Again, the theoretical aspects are grounded on the results of [7] and are similar to those of [5].

Obviously, Proposition 1 and Theorems 1 and 2 still hold; since for $\lambda_i \to 0$ and $k_i \to \infty$, all i, the space $\mathbf{V\Sigma}_k$ approaches $\boldsymbol{\mathcal{L}}$, we state the following result.

THEOREM 4.   *Let $\lambda_i \to 0$, $k_i \to \infty$ for $i = 0, 1, \dots, m$. Then $\boldsymbol{\sigma}^* \to \boldsymbol{\psi}$.*

Note that the above theorem implies that also the pseudo Bézier and de Boor control polygons tend to $\boldsymbol{\psi}$; therefore we have the following result

COROLLARY 2.   *It is possible to find sequences $\lambda_0, \dots, \lambda_m$ and $k_0, \dots, k_m$ such that the approximating spline curve $\boldsymbol{\sigma}^*$ is $\boldsymbol{\psi}$-shape preserving.*

The main drawback of the above result is that it is *global* in nature; if only some shape parameters tend to the limit values the space $\mathbf{V\Sigma}_k$ does not tend to $\boldsymbol{\mathcal{L}}$ and the asymptotic shape preserving properties vanish. The consequence is that all the segments of the curve are simultaneously stretched and the curve can assume an unpleasant appearance. In [7] is presented a solution which uses a *weighted* approximation, which can be here adopted. The basic idea is that we accept a compromise, obtaining the convergence at the price of a reduction in the approximation power. In order to force the spline to locally approach $\boldsymbol{\psi}$ when a local increase is applied, we work with an extension of the approximation problem. Let $w = \{w_0, \dots, w_m\}$ be a sequence of positive weights. In the following we use the notation $\boldsymbol{\sigma} = \boldsymbol{\sigma}_{k,w}$ and use $\boldsymbol{\theta}_{k,w}$ and $\boldsymbol{\ell}_{k,w} = \boldsymbol{\ell}(\boldsymbol{\sigma}_{k,w})$ to denote, respectively, the piecewise linear curves interpolating the control points $\{\mathbf{D}_0, \dots, \mathbf{D}_m\}$ and the spline at the knots $\{\boldsymbol{\sigma}(u_0), \dots, \boldsymbol{\sigma}(u_m)\} = \{\mathbf{b}_{0,0}, \dots, \mathbf{b}_{m,0}\}$. The basic idea is to push $\boldsymbol{\sigma}_{k,w}^*(u_i)$ towards $\boldsymbol{\psi}(u_i)$, by inserting $\boldsymbol{\psi}(u_0), \dots, \boldsymbol{\psi}(u_m)$ as

weighted points in the approximation problem. The approximation points become:

$$\{(t_j, \mathbf{I}_j) : j = 0, \dots, N\} \cup \{(u_i, \boldsymbol{\psi}(u_i)) : i = 0, \dots, m\},$$

and we find $\boldsymbol{\sigma}_{k,w}^* \in \mathbf{V}\boldsymbol{\Sigma}_k$ which is the best weighted least squares approximation, that is which minimizes the following functional

$$\sum_{j=0}^{N} ||\boldsymbol{\sigma}_{k,w}(t_j) - \mathbf{I}_j||^2 + \sum_{i=0}^{m} w_i ||\boldsymbol{\sigma}_{k,w}(u_i) - \boldsymbol{\psi}(u_i)||^2, \qquad \text{s.t.} \quad \boldsymbol{\sigma}_{k,w} \in \mathbf{V}\boldsymbol{\Sigma}_k.$$

We can use the values of the weights w_i to control behavior of the curve at u_i; in particular if $w_i = 0, i = 0, \dots, m$ we have the old approximation problem, and

$$\lim_{w_i \to \infty} \boldsymbol{\sigma}_{k,w}^*(u_i) \to \boldsymbol{\psi}(u_i).$$

To be more precise, we use the weights for imposing that $\boldsymbol{\ell}_{k,w}$ has the same shape of $\boldsymbol{\psi}$ and the degrees for stretching the curve, that is for imposing that $\boldsymbol{\sigma}_{k,w}^*$ has the same shape of $\boldsymbol{\ell}_{k,w}$. Obviously, the larger are the weights, the more the approximation to the *true data* $\mathbf{I}_0, \dots, \mathbf{I}_N$ deteriorates, and we want to keep the weights as small as possible. We refer to [7] for details on the algorithm.

We limit ourselves to Figure 8 for a graphical comparison. In Figure 8 (left) are reported the data set (random perturbations of equally spaced points over an helix), $\boldsymbol{\psi}$ and $\boldsymbol{\sigma}_k^*$ obtained using the global scheme; in Figure 8 (right) have been depicted similar plots for the local scheme.
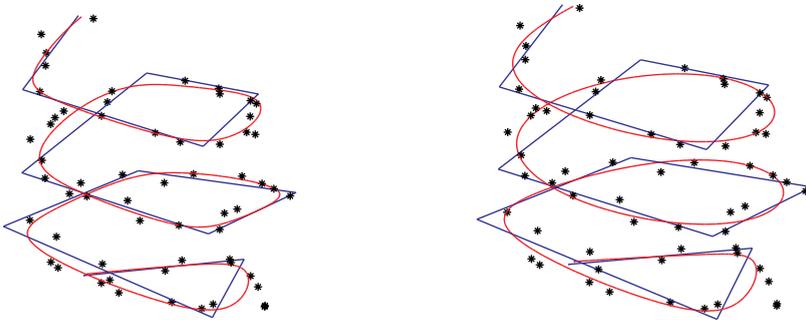
Fig. 8: Left: the global approximant. Right: the local approximant.

5 – Closure

We have presented a new class of C^3 functions which can be used for solving some important problems of CAGD. Their main advantage relies in the simple

geometric construction which, in turn, permits an easy description of the shape constraints and an easy choice of the optimal shape parameters.

It is worthwhile to remind that many CAD/CAM systems are based on standard NURBS with low degrees. The structure of Figures. 4 can be adapted to produce FC^3 (*Frenet-frame continuous* [13]) quintic curve; the corresponding results are reported in [5]. However, it is important to point out that, even if $C^2 - FC^3$ is a reasonable smoothness property both from the mechanical point of view (the motion of a point on the curve has a continuous acceleration) and from the geometric point of view (the tangent and curvature vectors and the torsion are continuous), the C^3 continuity is sometimes required. For instance, if the physical meaning of the parameter is time and the spline curve is used to control the motion of a robot, a smooth (C^1) acceleration will preserve the engines from harsh stresses.

We conclude the paper observing that the tensor-product extension seems straightforward; the non-obvious problems are to extract the information on the shape of the data, especially in the approximation case, and to assign to the data a parameterization suitable for our purposes. The corresponding researches are under study.

– **APPENDIX** – **Maple instructions**

Explicit expression of the points obtained in the corner cutting process described in Figures. 4; check of the corresponding limits; explicit computation of the coefficients of the i-th row of the interpolation matrix M and check of its asymptotic diagonal dominance.

For notational simplicity we have set: `delta[i]:=1/k[i]` .

```
> Fp[i-2]:=((h[i-2]+lambda[i-1]*(h[i-1]+h[i]))*D[i-2]
> +lambda[i-2]*(h[i-4]+h[i-3])*D[i-1])/(lambda[i-2]*(h[i-4]+
> h[i-3])+h[i-2]+lambda[i-1]*(h[i-1]+h[i]));


> Fm[i-1]:=((lambda[i-2]*(h[i-4]+h[i-3])+h[i-2])*D[i-1]+
> lambda[i-1]*(h[i-1]+h[i])*D[i-2])/(lambda[i-2]*(h[i-4]+h[i-3])+
> h[i-2]+lambda[i-1]*(h[i-1]+h[i]));
> Fp[i-1]:=((h[i-1]+lambda[i]*(h[i]+h[i+1]))*D[i-1]+
> lambda[i-1]*(h[i-3]+h[i-2])*D[i])/(lambda[i-1]*(h[i-3]+h[i-2])+
> h[i-1]+lambda[i]*(h[i]+h[i+1]));


> Fm[i]:=((lambda[i-1]*(h[i-3]+h[i-2])+h[i-1])*D[i]+
> lambda[i]*(h[i]+h[i+1])*D[i-1])/(lambda[i-1]*(h[i-3]+h[i-2])+
> h[i-1]+lambda[i]*(h[i]+h[i+1]));
> Fp[i]:=((h[i]+lambda[i+1]*(h[i+1]+h[i+2]))*D[i]+
```

```
> lambda[i]*(h[i-2]+h[i-1])*D[i+1])/(lambda[i]*(h[i-2]+h[i-1])+
> h[i]+lambda[i+1]*(h[i+1]+h[i+2]));

> Fm[i+1]:=((lambda[i]*(h[i-2]+h[i-1])+h[i])*D[i+1]+
> lambda[i+1]*(h[i+1]+h[i+2])*D[i])/(lambda[i]*(h[i-2]+h[i-1])+
> h[i]+lambda[i+1]*(h[i+1]+h[i+2]));
> Fp[i+1]:=((h[i+1]+lambda[i+2]*(h[i+2]+h[i+3]))*D[i+1]+
> lambda[i+1]*(h[i-1]+h[i])*D[i+2])/(lambda[i+1]*(h[i-1]+h[i])+
> h[i+1]+lambda[i+2]*(h[i+2]+h[i+3]));

> Fm[i+2]:=((lambda[i+1]*(h[i-1]+h[i])+h[i+1])*D[i+2]+
> lambda[i+2]*(h[i+2]+h[i+3])*D[i+1])/(lambda[i+1]*(h[i-1]+h[i])+
> h[i+1]+lambda[i+2]*(h[i+2]+h[i+3]));

> p[i-1]:=collect(((h[i-2]+h[i-1]+h[i])*Fm[i-1]+h[i-3]*Fp[i-1])/
> (h[i-3]+h[i-2]+h[i-1]+h[i]),[D[i-2],D[i-1],D[i]]);
> r[i-1]:=collect((h[i-3]*Fm[i-1]+(h[i-2]+h[i-1]+h[i])*Fp[i-1])/
> (h[i-3]+h[i-2]+h[i-1]+h[i]),[D[i-2],D[i-1],D[i]]);

> p[i]:=collect(((h[i-1]+h[i]+h[i+1])*Fm[i]+h[i-2]*Fp[i])/
> (h[i-2]+h[i-1]+h[i]+h[i+1]),[D[i-1],D[i],D[i+1]]);
> limit(p[i],lambda[i]=0);
> r[i]:=collect((h[i-2]*Fm[i]+(h[i-1]+h[i]+h[i+1])*Fp[i])/
> (h[i-2]+h[i-1]+h[i]+h[i+1]),[D[i-1],D[i],D[i+1]]);
> limit(r[i],lambda[i]=0);

> p[i+1]:=collect(((h[i]+h[i+1]+h[i+2])*Fm[i+1]+h[i-1]*Fp[i+1])/
> (h[i-1]+h[i]+h[i+1]+h[i+2]),[D[i],D[i+1],D[i+2]]);
> r[i+1]:=collect((h[i-1]*Fm[i+1]+(h[i]+h[i+1]+h[i+2])*Fp[i+1])/
> (h[i-1]+h[i]+h[i+1]+h[i+2]),[D[i],D[i+1],D[i+2]]);

> b[i-1,3]:=collect((delta[i]*h[i]*r[i-1]+
> (delta[i-1]*h[i-2]+(1-2*delta[i-1]-2*delta[i])*h[i-1])*p[i])/
> (delta[i-1]*h[i-2]+(1-2*delta[i-1]-2*delta[i])*h[i-1]+
> delta[i]*h[i]),
> [D[i-2],D[i-1],D[i],D[i+1]]);

> simplify(limit(b[i-1,3],lambda[i]=0,delta[i]=0));

> b[i,2]:=collect(((((1-2*delta[i]-2*delta[i+1])*h[i]+
> delta[i+1]*h[i+1])*r[i]+
> delta[i]*h[i-1]*p[i+1])/
> (delta[i]*h[i-1]+(1-2*delta[i]-2*delta[i+1])*h[i]+
```

```
> delta[i+1]*h[i+1]),
> [D[i-1],D[i],D[i+1],D[i+2]]);
> simplify(limit(b[i,2],lambda[i]=0,delta[i]=0));

> q[i]:=collect((h[i]*p[i]+h[i-1]*r[i])/(h[i-1]+h[i]),
> [D[i-2],D[i-1],D[i],D[i+1],D[i+2]]);
> simplify(limit(q[i],lambda[i]=0,delta[i]=0));

> b[i-1,4]:=collect((h[i]*b[i-1,3]+h[i-1]*q[i])/
> (h[i-1]+h[i]),[D[i-2],D[i-1],D[i],D[i+1],D[i+2]]);
> simplify(limit(b[i-1,4],lambda[i]=0,delta[i]=0));

> b[i,1]:=collect((h[i]*q[i]+h[i-1]*b[i,2])/
> (h[i-1]+h[i]),[D[i-2],D[i-1],D[i],D[i+1],D[i+2]]);
> simplify(limit(b[i,1],lambda[i]=0,delta[i]=0));

> b[i,0]:=collect((h[i]*b[i-1,4]+h[i-1]*b[i,1])/
> (h[i-1]+h[i]),[D[i-2],D[i-1],D[i],D[i+1],D[i+2]]);

> row:=coeffs(b[i,0],D[i-2],D[i-1],D[i],D[i+1],D[i+2]);
> limit(row[1],lambda[i]=0,delta[i]=0);
> limit(row[2],lambda[i]=0,delta[i]=0);
> simplify(limit(row[3],lambda[i]=0,delta[i]=0));
> limit(row[4],lambda[i]=0,delta[i]=0);
> limit(row[5],lambda[i]=0,delta[i]=0);
```

REFERENCES

[1] S. ASATURYAN – P. COSTANTINI – C. MANNI: *Local shape- preserving interpolation by space curves*, IMA J. Numer. Anal., **21** (2001), 301-325.

[2] P. COSTANTINI P.: *Curve and surface construction using variable degree polynomial splines*, Computer Aided Geometric Design, **17** (2000), 419-446.

[3] P. COSTANTINI – T.N.T. GOODMAN – C. MANNI: *Constructing C^3 shape preserving interpolating space curves*, Adv. Comput. Math., **14** (2001), 103-127.

[4] P. COSTANTINI – C. MANNI: *Shape–preserving C^3 interpolation: the curve case*, Adv. Comput. Math., **18** (2003), 41-63.

[5] P. COSTANTINI – C. MANNI: *Geometric construction of spline curves with tension properties*, Computer Aided Geometric Design, **20** (2003), 579-599.

[6] P. COSTANTINI – F. PELOSI: *Shape–preserving approximation by space curves*, Num. Alg., **27** (2001), 219-316.

[7] P. COSTANTINI – F. PELOSI: *Shape–preserving approximation of spatial data*, Adv. Comput. Math., **20** (2004), 25-51.

[8] M. Eck – D. Lasser: *B-spline-Bézier representation of geometric spline curves: quartics and quintics*, Computers Math. Applic., **23** (1992), 23-39.

[9] T.N.T. Goodman: *Total positivity and the shape of curves*, in *Total Positivity and its Applications*, M. Gasca and C.A. Micchelli (eds.), Kluwer, Dordrecht, 1996, pp. 157-186.

[10] T.N.T. Goodman – B.H. Ong: *Shape preserving interpolation by space curves*, Computer Aided Geomatric Design, **15** (1997), 1-17.

[11] T.N.T. Goodman – B.H. Ong: *Shape preserving interpolation by G^2 curves in three dimensions*, in *Curves and Surfaces with Applications in CAGD*, A. Le Méhauté, C. Rabut and L.L. Schumaker (eds.) Vanderbilt University Press, 1997, pp. 151-158.

[12] T.N.T. Goodman – B.H. Ong – M.L. Sampoli: *Automatic interpolation by fair, shape preserving, G^2 space curves*, Computer Aided Design, **30** (1998), 813-822.

[13] J. Hoschek – D. Lasser: *Fundamentals of Computer Aided Geometric Design*, A.K. Peters Ldt, Wellesley, Massachusetts, 1993.

[14] B. Jüttler: *Shape preserving least-squares approximation by polynomial parametric spline curves*, Computer Aided Geometric Design, **14** (1997), 731-747.

[15] P.D. Kaklis – M.T. Karavelas: *Shape preserving interpolation in $\mathbb{R}^3$*, IMA J. Numer. Anal., **17** (1997), 373-419.

[16] M.I. Karavelas – P.D. Kaklis: *Spatial shape preserving interpolation using ν–splines*, Numer. Alg., **23** (2000), 217-250.

[17] V.P. Kong – B.H. Ong: *Shape preserving interpolation using Frenet frame continuous curve of order 3*, preprint, 2001.

[18] R. Morandi – D. Scaramelli – A. Sestini: *A geometric approach for knot selection in convexity-preserving spline approximation*, in *Curve and Surface Design: Saint-Malo 1999*, Pierre-Jean Laurent, Paul Sablonniere and L.L. Schumaker (eds.), Vanderbilt University Press, Nashville, TN, 2000, pp. 287-296.

[19] P. Sablonnière: *Spline and Bézier polygons associated with a polynomial spline curve*, Computer Aided Design, **10** (1978), 257-261.

[20] R. Sauer R.: *Differenzengeometrie*, Springer Verlag, Berlin, 1970.

INDIRIZZO DEGLI AUTORI:

Paolo Costantini – Dipartimento di Scienze Matematiche ed Informatiche "R. Magari" – Pian dei Mantellini 44, 53100 Siena, Italy
email: costantini@unisi.it

Carla Manni – Dipartimento di Matematica - Università di Roma "Tor Vergata" – Via della Ricerca Scientifica, 00133 Roma, Italy
email: manni@mat.uniroma2.it