From PS–splines to NURPS

Joris Windmolders, Paul Dierckx

Report TW 297, october 1999

Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

From PS–splines to NURPS

Joris Windmolders, Paul Dierckx

Report TW 297, october 1999

Department of Computer Science, K.U.Leuven

Abstract

A normalized B–spline representation for Powell–Sabin (PS) spline surfaces is extended to piecewise rational surfaces (NURPS). We investigate the adaption of existing algorithms operating on B–splines to this more general case, the influence of weights and their geometrical interpretation, the possibility to represent planar sections, and the conversion from rational Bézier to NURPS surfaces.

Keywords : Powell–Sabin splines, NURPS, Shape parameters, control triangles

From PS–splines to NURPS

Paul Dierckx
Joris Windmolders

§1. Basic Concepts

1.1. PS–splines

Let $\Omega \subset \mathbb{R}^2$ be a simply connected subset with polygonal boundary $\delta\Omega$. Let Δ be a conforming triangulation of Ω having n vertices V_i with coordinates (u_i, v_i), $i = 1, \dots, n$, and denote with Δ^* a Powell–Sabin (PS) refinement of Δ (see, e.g.,[3]), where each triangle $\rho \in \Delta$ is divided up into 6 subtriangles. A Powell–Sabin (PS) spline is a piecewise quadratic polynomial with C^1 continuity on Ω. Dierckx [1] shows how to calculate a normalized B–spline basis for PS–splines:

Definition 1. *A PS–spline surface has a normalized B–spline representation:*

$$s(u,v) = \sum_{i=1}^{n} \sum_{j=1}^{3} \mathbf{c_{i,j}} B_i^j(u,v), \ (u,v) \in \Omega \tag{1}$$

where $\mathbf{c_{i,j}} = (c_{i,j}^x, c_{i,j}^y, c_{i,j}^z)$ *are the B–spline control points and* $B_i^j(u,v)$ *are the normalized B–splines.*

This representation shares a number of properties with tensor–product B–splines, making it a powerfull tool for representing surfaces in CAGD. We summarize the most important properties here, for details we refer to the original paper [1].

Property 1. $\{B_i^j(u,v)\}_{i=1,\dots,n}^{j=1,2,3}$ *is a convex partition of unity:*

$$\begin{cases} B_i^j(u,v) \geq 0, \ (u,v) \in \Omega \\ \sum_{i=1}^{n} \sum_{j=1}^{3} B_i^j(u,v) \equiv 1, \ (u,v) \in \Omega \end{cases}$$

Furthermore, $B_{i,j}(u,v)$ is nonzero only on triangles $\rho \in \Delta$ having V_i as a vertex:

Property 2.

$$B_i^j(V_l) = \frac{\partial B_i^j(V_l)}{\partial u} = \frac{\partial B_i^j(V_l)}{\partial v} = 0, \ l \neq i \tag{2}$$

The local control, affine invariance and convex hull properties follow immediately. Linear functions can be represented exactly; more particular we will further refer to the representations

$$u = \sum_{i=1}^{n} \sum_{j=1}^{3} U_{i,j} B_i^j(u,v), \quad v = \sum_{i=1}^{n} \sum_{j=1}^{3} V_{i,j} B_i^j(u,v)$$

Definition 2. *The PS–triangles* $t_l(Q_{l,1}, Q_{l,2}, Q_{l,3})$, $l = 1, \dots, n$ *in the domain plain have as vertices the B–spline ordinates* $Q_{l,j}(U_{l,j}, V_{l,j})$, $j = 1, 2, 3$.

Submitted to the proceedings
of the Saint-Malo conference
"Curves and Surfaces 1999"

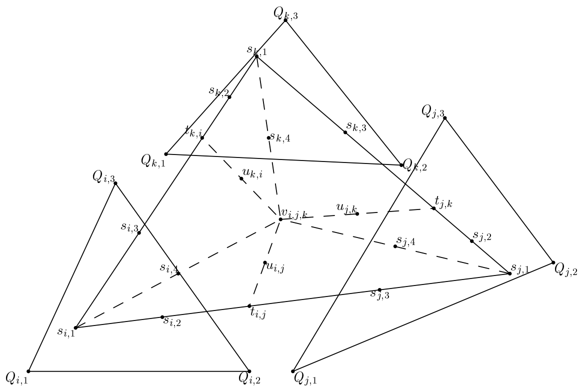

Fig. 1. Domain triangle.

Consider a domain triangle $\rho_{i,j,k}(V_i, V_j, V_k) \in \Delta$ with its PS–refinement (see figure 1). Denote the Bézier ordinates as $s_{v,l}$, $v = i, j, k, l = 1, 2, 3, 4$; $t_{l,m}, u_{l,m}$, $(l, m) \in \{(i, j), (j, k), (k, i)\}$ and $v_{i,j,k}$. They can be written as unique barycentric combinations of the B–spline ordinates:

$$s_{v,l} = \alpha_{v,l}\ Q_{v,1} + \beta_{v,l}\ Q_{v,2} + \gamma_{v,l}\ Q_{v,3} \tag{3}$$

$$t_{l,m} = \delta_{l,m}\ s_{l,2} + \epsilon_{l,m}\ s_{m,3} \tag{4}$$

$$u_{l,m} = \delta_{l,m}\ s_{l,4} + \epsilon_{l,m}\ s_{m,4} \tag{5}$$

$$v_{i,j,k} = \lambda_{i,j,k}\ s_{i,4} + \mu_{i,j,k}\ s_{j,4} + \nu_{i,j,k}\ s_{k,4} \tag{6}$$

For a given PS–refinement Δ^*, the position of the Bézier ordinates is fixed. This is not the case for the B–spline ordinates. The following lemma however states that there is a restriction on the B–spline ordinates.

Lemma 1. *In order for the basis functions* $\left\{B_i^j(u,v)\right\}_{i=1,\ldots,n}^{j=1,2,3}$ *to constitute a convex partition of unity on* Ω*, it is required that for each vertex* V_i*,* $i = 1, \ldots, n$ *the PS–triangle* $t_i(Q_{i,1}, Q_{i,2}, Q_{i,3})$ *contains the Powell–Sabin points, i.e. the Bézier ordinates* $s_{i,l}, l = 1, 2, 3, 4$ *of any domain triangle having* V_i *as one of its vertices.*

There is a one–one connection between the barycentric coordinates of the Powell–Sabin points at vertex V_i with respect to t_i and the value of the basis functions $B_i^j(u, v)$, $j = 1, 2, 3$ and its derivatives at V_i, e.g.

$$B_i^1(V_i) = \alpha_{i,1},\ B_i^2(V_i) = \beta_{i,1},\ B_i^3(V_i) = \gamma_{i,1} \tag{7}$$

Given a PS–spline surface (1), the corresponding Bézier net can be calculated efficiently by using convex barycentric combinations of the B–spline control points only:

Property 3. *Applying equations (3)–(6) where the ordinates are replaced by control points, yields the corresponding Bézier net of the surface.*

Finally, via the concept of control triangles, the B–spline control points give us valuable insight in the shape of the surface:

Definition 3. *The control triangles are defined as* $T_l(\mathbf{c_{l,1}}, \mathbf{c_{l,2}}, \mathbf{c_{l,3}})$.

Property 4. *Each control triangle* $T_l(\mathbf{c_{l,1}}, \mathbf{c_{l,2}}, \mathbf{c_{l,3}})$ *is tangent to the PS–surface at* $s(V_l)$.

1.2. NURPS

The Normalized B–spline theory for PS–surfaces can now be extended to a rational scheme just like tensor product B–splines are extended to NURBS. Referring to figure 1, we use the boldface notation for the Bézier points, e.g. $\mathbf{s_{v,l}}$. Points in homogeneous space get a h–superscript, e.g. $\mathbf{s^h_{v,l}}$. Its components are $\left(s_{v,l}^{h,x}, s_{v,l}^{h,y}, s_{v,l}^{h,z}, s_{v,l}^{w}\right)$.

Definition 4. *A Non Uniform Rational Powell–Sabin (NURPS) spline surface has the form*

$$s(u,v) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{3} \mathbf{c_{i,j}} w_{i,j} B_i^j(u,v)}{\sum_{i=1}^{n}\sum_{j=1}^{3} w_{i,j} B_i^j(u,v)}, \ (u,v) \in \Omega \tag{8}$$

where $\mathbf{c_{i,j}} = (c_{i,j}^x, c_{i,j}^y, c_{i,j}^z)$ *are the B–spline control points. We impose that* $w_{i,j} > 0$ *in order for* $s(u,v)$ *to be defined anywhere on* Ω.

If $w_{i,j} = 1, \ i = 1, \ldots, n, \ j = 1, 2, 3$ (8) reduces to (1). The following properties are readily verified:

Property 5.

$$s(u,v) = \sum_{i=1}^{n}\sum_{j=1}^{3} \mathbf{c_{i,j}} \phi_i^j(u,v) \tag{9}$$

where

$$\phi_i^j(u,v) = \frac{w_{i,j} B_i^j(u,v)}{\sum_{i=1}^{n}\sum_{j=1}^{3} w_{i,j} B_i^j(u,v)} \tag{10}$$

and

$$\begin{cases} \phi_i^j(u,v) \geq 0, \ (u,v) \in \Omega \\ \sum_{i=1}^{n}\sum_{j=1}^{3} \phi_i^j(u,v) \equiv 1, \ (u,v) \in \Omega \end{cases}$$

Furthermore, $\phi_{i,j}(u,v)$ *is nonzero only on triangles* $\rho \in \Delta$ *having* V_i *as a vertex.*

This again implies the local control, affine invariance and convex hull properties.

Property 6. *A NURPS representation (8) is the 3D–projection in Eucledian space of a 4D PS–spline in homogeneous space:*

$$s(u,v) = \sum_{i=1}^{n}\sum_{j=1}^{3} \mathbf{c_{i,j}^h} B_i^j(u,v) \tag{11}$$

$$\mathbf{c_{i,j}^h} = (w_{i,j} c_{i,j}^x, w_{i,j} c_{i,j}^y, w_{i,j} c_{i,j}^z, w_{i,j}) \tag{12}$$

§2. Evaluation and subdivision

The evaluation of $s(u,v)$ is performed in two steps:

- First, the corresponding rational piecewise Bézier representation is calculated.
- Then, the rational de Casteljau–algorithm calculates a point on this rational piecewise quadratic Bézier surface. This section shows how to perform the first step in a numerically stable way, for the second step we refer to Farin [2].

2.1. In homogeneous space

Formulae (3)–(6) can be applied directly in homogeneous space, e.g.

$$\begin{aligned} \mathbf{s_{v,l}^h} &= \alpha_{v,l} \ \mathbf{c_{v,1}^h} + \beta_{v,l} \ \mathbf{c_{v,2}^h} + \gamma_{v,l} \ \mathbf{c_{v,3}^h} \\ &= \left(s_{v,l}^{h,x}, s_{v,l}^{h,y}, s_{v,l}^{h,z}, s_{v,l}^{w} \right) \end{aligned} \tag{13}$$

$$\begin{aligned} \mathbf{v_{i,j,k}^h} &= \lambda_{i,j,k} \ \mathbf{s_{i,4}^h} + \mu_{i,j,k} \ \mathbf{s_{j,4}^h} + \nu_{i,j,k} \ \mathbf{s_{k,4}^h} \\ &= \left(v_{i,j,k}^{h,x}, v_{i,j,k}^{h,y}, v_{i,j,k}^{h,z}, v_{i,j,k}^{w} \right) \end{aligned} \tag{14}$$

Projection back to Eucledian space yields

$$\mathbf{s_{v,l}} = \left(\frac{s_{v,l}^{h,x}}{s_{v,l}^{w}}, \frac{s_{v,l}^{h,y}}{s_{v,l}^{w}}, \frac{s_{v,l}^{h,z}}{s_{v,l}^{w}} \right) \quad \mathbf{v_{i,j,k}} = \left(\frac{v_{i,j,k}^{h,x}}{v_{i,j,k}^{w}}, \frac{v_{i,j,k}^{h,y}}{v_{i,j,k}^{w}}, \frac{v_{i,j,k}^{h,z}}{v_{i,j,k}^{w}} \right)$$

This algorithm has a serious drawback: if the weights vary greatly in magnitude, the coordinates $s_{v,l}^{h,r}, v_{i,j,k}^{h,r}, \ r = x, y, z$ are *blown away*; the calculations don't operate in the convex hull of the control net anymore and numerical stability is endangered.

2.2. A rational algorithm

The idea behind the rational de Casteljau–algorithm from Farin [2] allows to improve numerical stability by rearranging the calculations, avoiding to work in homogeneous space:

$$s^w_{v,l} = \alpha_{v,l}\ w_{v,1} + \beta_{v,l}\ w_{v,2} + \gamma_{v,l}\ w_{v,3} \tag{15}$$

Set

$$\tilde{\alpha}_{v,l} = \frac{\alpha_{v,l}\ w_{v,1}}{s^w_{v,l}} \geq 0,\ \tilde{\beta}_{v,l} = \frac{\beta_{v,l}\ w_{v,2}}{s^w_{v,l}} \geq 0,\ \tilde{\gamma}_{v,l} = \frac{\gamma_{v,l}\ w_{v,3}}{s^w_{v,l}} \geq 0 \tag{16}$$

then

$$\mathbf{s_{v,l}} = \tilde{\alpha}_{v,l}\ \mathbf{c_{v,1}} + \tilde{\beta}_{v,l}\ \mathbf{c_{v,2}} + \tilde{\gamma}_{v,l}\ \mathbf{c_{v,3}} \tag{17}$$

with

$$\tilde{\alpha}_{v,l} + \tilde{\beta}_{v,l} + \tilde{\gamma}_{v,l} = 1 \tag{18}$$

The point $\mathbf{s_{v,l}}$ is a convex barycentric combination of $\mathbf{c_{v,1}}$, $\mathbf{c_{v,2}}$ and $\mathbf{c_{v,3}}$ so numerical stability is guaranteed. Likewise, we find:

$$t^w_{l,m} = \delta_{l,m}\ s^w_{l,2} + \epsilon_{l,m}\ s^w_{m,3} \tag{19}$$

$$u^w_{l,m} = \delta_{l,m}\ s^w_{l,4} + \epsilon_{l,m}\ s^w_{m,4} \tag{20}$$

$$v^w_{i,j,k} = \lambda_{i,j,k}\ s^w_{i,4} + \mu_{i,j,k}\ s^w_{j,4} + \nu_{i,j,k}\ s^w_{k,4} \tag{21}$$

$$\tilde{\delta}_{l,m} = \frac{\delta_{l,m}\ s^w_{l,2}}{t^w_{l,m}}, \quad \tilde{\epsilon}_{l,m} = \frac{\epsilon_{l,m}\ s^w_{m,3}}{t^w_{l,m}} \tag{22}$$

$$\tilde{\delta}^*_{l,m} = \frac{\delta_{l,m}\ s^w_{l,4}}{u^w_{l,m}}, \quad \tilde{\epsilon}^*_{l,m} = \frac{\epsilon_{l,m}\ s^w_{m,4}}{u^w_{l,m}} \tag{23}$$

$$\tilde{\lambda}_{i,j,k} = \frac{\lambda_{i,j,k}\ s^w_{i,4}}{v^w_{i,j,k}}, \quad \tilde{\mu}_{i,j,k} = \frac{\mu_{i,j,k}\ s^w_{j,4}}{v^w_{i,j,k}}, \quad \tilde{\nu}_{i,j,k} = \frac{\nu_{i,j,k}\ s^w_{k,4}}{v^w_{i,j,k}} \tag{24}$$

where

$$\tilde{\delta}_{l,m} + \tilde{\epsilon}_{l,m} = \tilde{\delta}^*_{l,m} + \tilde{\epsilon}^*_{l,m} = \tilde{\lambda}_{i,j,k} + \tilde{\mu}_{i,j,k} + \tilde{\nu}_{i,j,k} = 1$$

and finally

$$\mathbf{t_{l,m}} = \tilde{\delta}_{l,m}\ \mathbf{s_{l,2}} + \tilde{\epsilon}_{l,m}\ \mathbf{s_{m,3}} \tag{25}$$

$$\mathbf{u_{l,m}} = \tilde{\delta}^*_{l,m}\ \mathbf{s_{l,4}} + \tilde{\epsilon}^*_{l,m}\ \mathbf{s_{m,4}} \tag{26}$$

$$\mathbf{v_{i,j,k}} = \tilde{\lambda}_{i,j,k}\ \mathbf{s_{i,4}} + \tilde{\mu}_{i,j,k}\ \mathbf{s_{j,4}} + \tilde{\nu}_{i,j,k}\ \mathbf{s_{k,4}} \tag{27}$$

All formulae are convex barycentric combinations operating in the convex hull of the B–spline control net. After having computed the rational Bézier representation, Farins rational de Casteljau algorithm can be used to evaluate the surface at any point $(u, v) \in \Omega$.

2.3. Subdivision on uniform triangulations

The evaluation and subdivision of spline curves and surfaces are closely related problems. For the particular case of a uniform triangulation Δ, a subdivision scheme for PS–surfaces has been derived [4]. As an application it was shown how a wireframe of the surface can be calculated in an efficient and numerically stable way. This scheme can easily be extended to NURPS on uniform triangulations again using Farins technique from the previous section. The details are omitted here.

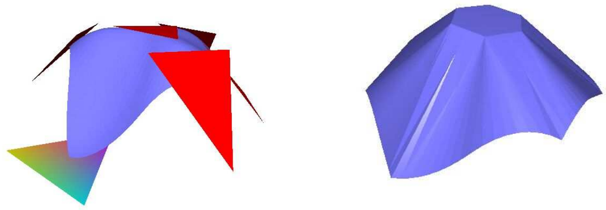

Fig. 2. NURPS surface and its control planes; Local planar effects.

§3. Controlplanes

Recall that the NURPS representation inherits the convex hull, affine invariance and local control property from the normalized B–spline representation. This section adds the *tangent property* to the inheritance list and shows how the rational representation allows for more flexibilty when designing surfaces.

3.1. Tangent property

Referring to the locality of the B–splines (2), it is easy to verify that the evaluation of $s(u,v)$ and its derivatives at vertex V_i yield:

$$s(V_i) = \tilde{\alpha}_{i,1}\ \mathbf{c_{i,1}} + \tilde{\beta}_{i,1}\ \mathbf{c_{i,2}} + \tilde{\gamma}_{i,1}\ \mathbf{c_{i,3}} \tag{28}$$

$$\frac{\partial s(V_i)}{\partial u} = e_{i,1}\ \mathbf{c_{i,1}} + e_{i,2}\ \mathbf{c_{i,2}} + e_{i,3}\ \mathbf{c_{i,3}} \tag{29}$$

$$\frac{\partial s(V_i)}{\partial v} = d_{i,1}\ \mathbf{c_{i,1}} + d_{i,2}\ \mathbf{c_{i,2}} + d_{i,3}\ \mathbf{c_{i,3}} \tag{30}$$

for some

$$e_{i,1} + e_{i,2} + e_{i,3} = d_{i,1} + d_{i,2} + d_{i,3} = 0$$

It follows that the control triangle at V_i is tangent to the surface at $s(V_i)$: any point $\mathbf{p}$ in the tangent plane is a barycentric combination of the control points $\mathbf{c_{i,1}}, \mathbf{c_{i,2}}, \mathbf{c_{i,3}}$:

$$\mathbf{p} = s(V_i) + a\ \frac{\partial s(V_i)}{\partial u} + b\ \frac{\partial s(V_i)}{\partial v},\ \ a, b \in \mathbb{R}$$

This is illustrated in figure 2 (left).

3.2. Shape Parameters

Farin [2] introduces the concept of shape parameters with respect to rational Bézier curves. A geometric handle allows the designer to influence the shape of the curve in a predictable way, rather then requiring the input of numbers for the weights. In the same work, it is stated that this property does not carry over to rational Bézier surfaces on triangles. Shape parameters however can be defined for NURPS.

Recall that $(\tilde{\alpha}_{v,1}, \tilde{\beta}_{v,1}, \tilde{\gamma}_{v,1})$ are the barycentric coordinates of $\mathbf{s_{v,1}}$ with respect to control triangle $T_v(\mathbf{c_{v,1}}, \mathbf{c_{v,2}}, \mathbf{c_{v,3}})$. From (16) it follows that $\mathbf{s_{v,1}}$ can be moved within T_v to a new location $(\tilde{\alpha}'_{v,1}, \tilde{\beta}'_{v,1}, \tilde{\gamma}'_{v,1})$, while keeping its weight $s^w_{v,1}$ constant: the corresponding PS–weights are found immediately as

$$w_{v,1} = \frac{\tilde{\alpha}'_{v,1} s^w_{v,1}}{\alpha_{v,1}}\quad w_{v,2} = \frac{\tilde{\beta}'_{v,1} s^w_{v,1}}{\beta_{v,1}}\quad w_{v,3} = \frac{\tilde{\gamma}'_{v,1} s^w_{v,1}}{\gamma_{v,1}} \tag{31}$$

This shows how $(\tilde{\alpha}_{v,1}, \tilde{\beta}_{v,1}, \tilde{\gamma}_{v,1})$ can be used as *shape parameters*.

3.3. Planar sections

Definition 5. *Let* $[\mathbf{t_1}, \mathbf{t_2}, \ldots, \mathbf{t_n}]$ *denote the convex hull of the 3D points* $\mathbf{t_1}, \mathbf{t_2}, \ldots, \mathbf{t_n}$.

Definition 6. *Let* $S(A)$ *denote the image of a subset* $A \subset \Omega$ *under (8).*

Definition 7. *Let* $\tau(a, b, c)$ *denote the Bézier subtriangle in the domain plane with vertices* a, b *and* c.

If the control triangles of adjacent vertices V_i, V_j, V_k are choosen to be equiplanar, the surface section $S(\rho_{i,j,k}(V_i, V_j, V_k))$ will be in the same plane, as a consequence of the convex hull property. Using the weights in the NURPS representation however, it is possible to achieve more local planar effects.

The rational evaluation algorithm from section 2.2 reveals that, for $w_{i,1} = w_{i,2} = w_{i,3} = w \to \infty$, $i \in (1, \ldots, n)$ and referring to figure 1 the following holds on domain triangle $\rho_{i,j,k}(V_i, V_j, V_k)$:

$$\frac{\delta_{i,j}}{\delta_{i,j} + \frac{\epsilon_{i,j} s^w_{j,3}}{w}} \mathbf{s_{i,2}} + \frac{\frac{\epsilon_{i,j} s^w_{j,3}}{w}}{\delta_{i,j} + \frac{\epsilon_{i,j} s^w_{j,3}}{w}} \mathbf{s_{j,3}} \tag{32}$$

so

$$\lim_{w \to \infty} \mathbf{t_{i,j}} = \mathbf{s_{i,2}} \tag{33}$$

Likewise, for the other Bézier points of $\tau(s_{i,1}, t_{i,j}, v_{i,j,k})$ we find:

$$\mathbf{s_{i,l}} = \alpha_{i,l}\ \mathbf{c_{i,1}} + \beta_{i,l}\ \mathbf{c_{i,2}} + \gamma_{i,l}\ \mathbf{c_{i,3}},\ \ l = 1, 2, 4 \tag{34}$$

$$\lim_{w \to \infty} \mathbf{u_{i,j}} = \lim_{w \to \infty} \mathbf{v_{i,j,k}} = \mathbf{s_{i,4}} \tag{35}$$

Consequently

$$S(\tau(s_{i,1}, t_{i,j}, v_{i,j,k})) = [\mathbf{s_{i,1}}, \mathbf{s_{i,2}}, \mathbf{s_{i,4}}]$$

Similar reasoning on the other Bézier subtriangles shows that

$$\begin{aligned} S(\tau(s_{i,1}, v_{i,j,k}, t_{k,i})) &= [\mathbf{s_{i,1}}, \mathbf{s_{i,4}}, \mathbf{s_{i,3}}] \\ S(\tau(t_{i,j}, s_{j,1}, v_{i,j,k})) &= [\mathbf{s_{i,2}}, \mathbf{s_{i,4}}] \\ S(\tau(t_{k,i}, v_{i,j,k}, s_{k,1})) &= [\mathbf{s_{i,3}}, \mathbf{s_{i,4}}] \\ S(\tau(v_{i,j,k}, s_{j,1}, t_{j,k})) &= [\mathbf{s_{i,4}}] \\ S(\tau(v_{i,j,k}, t_{j,k}, s_{k,1})) &= [\mathbf{s_{i,4}}] \end{aligned}$$

and therefore

$$S(\rho_{i,j,k}(V_i, V_j, V_k)) = [\mathbf{s_{i,1}}, \mathbf{s_{i,2}}, \mathbf{s_{i,4}}, \mathbf{s_{i,3}}] \subset [\mathbf{c_{i,1}}, \mathbf{c_{i,2}}, \mathbf{c_{i,3}}]$$

The latter image is a planar surface section. Figure 2 (right) shows some NURPS surface with very large weights at a vertex.

§4. Rational Bézier to NURPS conversion

Be given a rational quadratic Bézier surface on one domain triangle (see figure 3) $\rho(s_{i,1}, s_{j,1}, s_{k,1})$

$$b(u, v) = \sum_{i_1+i_2+i_3=2} \mathbf{b}^h_{i_1,i_2,i_3} B^2_{i_1,i_2,i_3}(t_1, t_2, t_3) \tag{36}$$

where $i_1, i_2, i_3 > 0, (u, v) \in \rho$ and (t_1, t_2, t_3) are the barycentric coordinates of (u, v) with respect to ρ. In this section it is shown how a NURPS representation

$$s(u, v) = \sum_{l=i,j,k} \sum_{m=1}^{3} \mathbf{c}^h_{l,m} B^m_l(u, v) \tag{37}$$

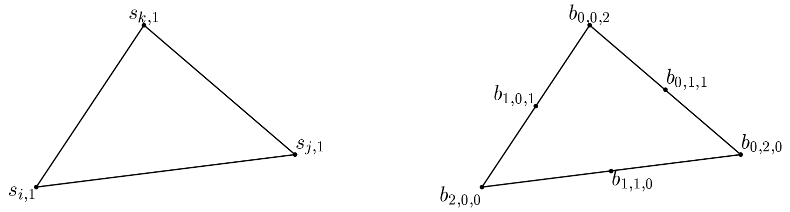

Fig. 3. Bézier triangle.

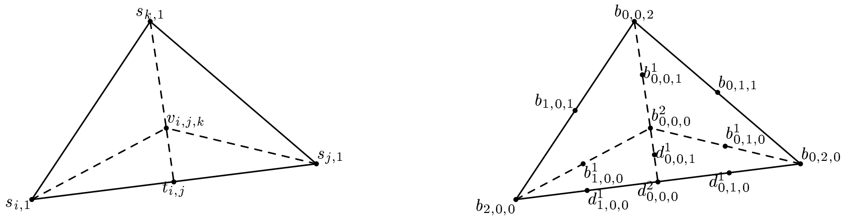

Fig. 4. Subdivision at $v_{i,j,k}$ and $t_{i,j}$.

of the given surface, for a specific choice of the PS–triangles, is immediately obtained. To simplify the notation, the surfaces are considered in homogeneous space.

Lemma 2. *Given a triangle $t\left(V_i, V_j, V_k\right)$ with barycenter z. If W_l denotes the midpoint of the side opposite to V_l, then $\left(z+V_l\right)/2$ is the barycenter of the triangle $t\left(V_l, W_m, W_n\right), l, m, n \in \{i,j,k\}, l \neq m \neq n$.*

The construction of the NURPS representation relies on the *de Casteljau-algorithm* for Bézier triangles (see, e.g., [2]). Subdivision at the barycenter of ρ and at the midpoint of edge $s_{i,1}s_{j,1}$ (see figure 4) yields the new Bézier points:

$$\mathbf{b}^{1,h}_{1,0,0} = \frac{1}{3}\left(\mathbf{b}^h_{2,0,0} + \mathbf{b}^h_{1,1,0} + \mathbf{b}^h_{1,0,1}\right) \tag{38}$$

$$\mathbf{b}^{1,h}_{0,1,0} = \frac{1}{3}\left(\mathbf{b}^h_{1,1,0} + \mathbf{b}^h_{0,2,0} + \mathbf{b}^h_{0,1,1}\right) \tag{39}$$

$$\mathbf{b}^{1,h}_{0,0,1} = \frac{1}{3}\left(\mathbf{b}^h_{1,0,1} + \mathbf{b}^h_{0,1,1} + \mathbf{b}^h_{0,0,2}\right) \tag{40}$$

$$\mathbf{b}^{2}_{0,0,0} = \frac{1}{3}\left(\mathbf{b}^{1,h}_{1,0,0} + \mathbf{b}^{1,h}_{0,1,0} + \mathbf{b}^{1,h}_{0,0,1}\right) \tag{41}$$

$$\mathbf{d}^{1,h}_{1,0,0} = \frac{1}{2}(\mathbf{b}^h_{2,0,0} + \mathbf{b}^h_{1,1,0}) \tag{42}$$

$$\mathbf{d}^{1,h}_{0,1,0} = \frac{1}{2}(\mathbf{b}^h_{1,1,0} + \mathbf{b}^h_{0,2,0}) \tag{43}$$

$$\mathbf{d}^{1,h}_{0,0,1} = \frac{1}{2}(\mathbf{b}^{1,h}_{1,0,0} + \mathbf{b}^{1,h}_{0,1,0}) \tag{44}$$

$$\mathbf{d}^{2,h}_{0,0,0} = \frac{1}{2}(\mathbf{d}^{1,h}_{1,0,0} + \mathbf{d}^{1,h}_{0,1,0}) \tag{45}$$

After subdivision of the two remaining edges, the 6 subtriangles thus obtained constitute a PS–refinement of ρ, say, with interior point $v_{i,j,k}$ and edge points $t_{i,j}, t_{j,k}, t_{k,i}$ (see figure 5, left). A NURPS representation of the given surface on this PS–refinement is easily obtained (see figure 5, right): set

$$Q_{v,1} = s_{v,1}, \;\; v = i,j,k$$
$$Q_{i,2} = Q_{j,3} = t_{i,j}; \;\; Q_{j,2} = Q_{k,3} = t_{j,k}; \;\; Q_{k,2} = Q_{i,3} = t_{k,i}$$

and

$$\mathbf{c}^h_{i,1} = \mathbf{b}^h_{2,0,0}, \;\; \mathbf{c}^h_{i,2} = \mathbf{b}^h_{1,1,0}, \;\; \mathbf{c}^h_{i,3} = \mathbf{b}^h_{1,0,1}$$
$$\mathbf{c}^h_{j,1} = b^h_{0,2,0}, \;\; \mathbf{c}^h_{j,2} = \mathbf{b}^h_{0,1,1}, \;\; \mathbf{c}^h_{j,3} = \mathbf{b}^h_{1,1,0}$$
$$\mathbf{c}^h_{k,1} = \mathbf{b}^h_{0,0,2}, \;\; \mathbf{c}^h_{k,2} = \mathbf{b}^h_{1,0,1}, \;\; \mathbf{c}^h_{k,3} = \mathbf{b}^h_{0,1,1}$$

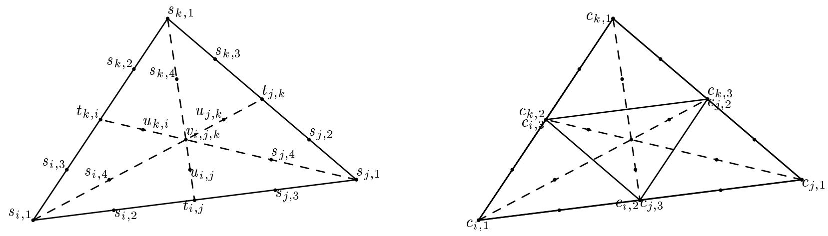

Fig. 5. PS–refinement.

then by lemma 2 it follows that the PS–points $s_{v,l}$, $l = 1, 2, 3, 4$ are inside the PS–triangle t_v, $v = i, j, k$. Now, since for this case, in formula (3) $\alpha_{v,4} = \beta_{v,4} = \gamma_{v,4} = \frac{1}{3}$, $v = i, j, k$, formula (13) with $l = 4$, $v = i, j, k$ and formulae (38)–(40) are exactly the same. Furthermore, since in formula (6) $\lambda_{i,j,k} = \mu_{i,j,k} = \nu_{i,j,k} = \frac{1}{3}$, formulae (14) and (41) are exactly the same. Likewise, since in (3)–(6)

$$(\alpha_{v,2}, \beta_{v,2}, \gamma_{v,2}) = (\frac{1}{2}, \frac{1}{2}, 0)$$
$$(\alpha_{v,3}, \beta_{v,3}, \gamma_{v,3}) = (\frac{1}{2}, 0, \frac{1}{2})$$
$$(\delta_{l,m}, \epsilon_{l,m}) = (\frac{1}{2}, \frac{1}{2})$$

for $v = i, j, k$ and $(l, m) \in \{(i, j), (j, k), (k, i)\}$ similar reasoning shows that calculating the corresponding Bézier net of (37) exactly yields the Bézier net of (36) after the proposed subdivisions. Hence, $b(u, v) \equiv s(u, v)$ on ρ.

References

1. Dierckx, P., On calculating normalized Powell–Sabin B–splines, Comput. Aided Geom. Design **15** (1997), 61–78.
2. Farin, G., *Curves And Surfaces For Computer Aided Geometric Design: A Practical Guide*, 4th edition, Academic Press, Boston, 1997.
3. Powell, M. J. D. and Sabin, M. A., Piecewise quadratic approximations on triangles, ACM Trans. Math. Software **3** (1977), 316–325.
4. Windmolders, J., Dierckx, P., Subdivision of uniform Powell–Sabin splines, Comput. Aided Geom. Design **16** (1999), 301–315.

Paul Dierckx and Joris Windmolders
Celestijnenlaan 200A
3001 Heverlee
Belgium
paul.dierckx@cs.kuleuven.ac.be
joris.windmolders@cs.kuleuven.ac.be