

Skinning of Circles and Spheres by Geometric Optimization in Minkowski Space

Bernhard Blaschitz

Waagner-Biro Stahlbau AG
Leonard Bernstein Strasse 10, 1220 Vienna, Austria
email: bernhard.blaschitz@waagner-biro.at

Abstract. Assuming a discrete set of circles \mathbf{p}_i in the plane, a real envelope is looked for. The new approach of this work is reformulating the original task as a constrained optimization in the point set model. The quadratic objective function minimizes the Euclidean distance between the cyclographic images of circles \mathbf{p}_i and a cubic B-Spline \mathbf{b} by observing the footpoint problem, which brings a better fit, but results in a non-linear problem. The reality of the envelope results in a quadratic, but non-convex constraint, which can be linearized. This linearization is discussed in detail, as its formulation is central to this work.

The ideas discussed for circles are also generalized for spheres; in the 1-parameter case that leads to a new method for interpolation points in $\mathbb{R}^{3,1}$ by curves, which translates to interpolation of spheres by canal surfaces.

Approximating 2-parameter sets of points by surfaces in $\mathbb{R}^{3,1}$ gives rise to general envelope surfaces of 2-parameter families of spheres, that have not been studied before in this generality. For this, a calculus was reinvented and applied, that classifies 2-planes in $\mathbb{R}^{3,1}$ according to their steepness.

Key words: Minkowski space, numerical optimization, curve fitting, surface fitting, Laguerre geometry, cyclography

MSC 2010: 51B20, 68U07, 51N30, 65K10

1. Introduction

Circles in \mathbb{R}^2 can be represented by points in Minkowski space $\mathbb{R}^{2,1}$. In order to find envelopes for the circles, we have to approximate the corresponding points by curves $\mathbf{c}(u)$ in $\mathbb{R}^{2,1}$ (see Figure 1). This would be a standard quadratic approximation problem if reality of envelopes would be ignored. But the envelopes are real if and only if $\mathbf{c}(u)$ has no pseudo-Euclidean tangents, which is a non-convex quadratic constraint of the curve fitting.

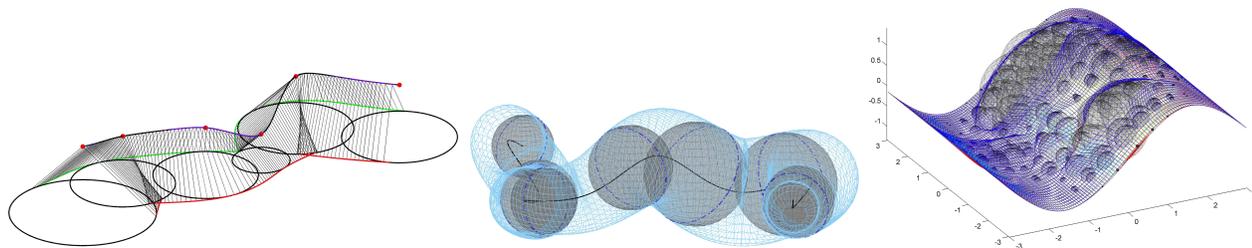


Figure 1: *Left:* Given a set of circles in \mathbb{R}^2 , its envelope (curves in *red* and *green*) can be found as the inverse cyclographic image $\zeta^{-1}(\mathbf{c}(u))$ of an interpolating curve $\mathbf{c}(u)$ (*purple*) in Minkowski space $\mathbb{R}^{2,1}$ through points Q_k (*red dots*) which are the ζ -images of the given circles. The envelopes are real if and only if $\mathbf{c}(u)$ has no pseudo-Euclidean tangents, which is guaranteed through a constraint of the curve fitting.

Middle: One dimension higher, the input is a set of spheres (*gray*) for which we find the optimal envelope (*light blue* surface).

Right: Analogously, given a 2-parameter set of spheres (*gray*) what is the optimal surfaces enveloping them? These questions are answered through a non-convex constrained optimization which ensures reality of the envelopes.

1.1. Minkowski space

We only show some aspects of Minkowski space and Laguerre geometry; a more complete introduction to this classical topic can be found in [4].

Pseudo-Euclidean inner product

Minkowski space $\mathbb{R}^{2,1}$ is equipped with an indefinite inner product

$$\langle \mathbf{a}, \mathbf{b} \rangle_L = a_1 b_1 + a_2 b_2 - a_3 b_3 \quad (1)$$

and thus for a vector $\mathbf{a} \in \mathbb{R}^{2,1}$ the product $\langle \mathbf{a}, \mathbf{a} \rangle_L$ can be negative/zero/positive; the vector is then called *pseudo-Euclidean/isotropic/Euclidean* (see Figure 2, *left*). The (convex) set $\{(x_1, x_2, x_3) \in \mathbb{R}^2 \times \mathbb{R} : x_3^2 \geq x_1^2 + x_2^2\}$ is called *Lorentz cone* Γ .

Quadratic constraint

Due to this geometric constraint, the tangents of these curves have to enclose an angle less than or equal to $\frac{\pi}{4}$ with the plane $x_3 = 0$. One can reformulate this constraint by saying that the hodograph (derivative curve) has to stay outside the Lorentz cone Γ ; it follows that this constraint is quadratic and non-convex.

1.2. Previous work

The problem of finding envelopes of circles has been dealt with before, but not in this generality as we will see in a quick literature overview.

POTTMANN and PETERNELL [4] consider curves \mathbf{c} in $\mathbb{R}^{3,1}$ and interpret them as canal surfaces. If \mathbf{c} is a NURBS curve, its (*convex hull, variation diminishing*) properties translate into certain properties for the rational canal surfaces. Only curves with Euclidean tangents are considered, i.e., the question of reality of envelopes is not touched. The paper gives a

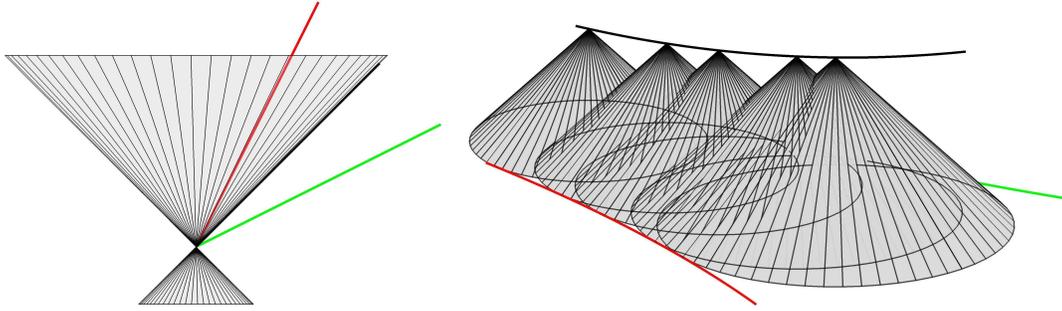


Figure 2: *Left:* The scalar product $\langle \mathbf{a}, \mathbf{a} \rangle_L$ is negative/zero/positive if and only if the vector \mathbf{a} is *pseudo-Euclidean/isotropic/Euclidean* (colors *red / black / green*); the Lorentz cone is shown in *gray*.

Right: The cyclographic preimage of a curve $\mathbf{c}(u)$ in Minkowski space $\mathbb{R}^{2,1}$ is the envelope of oriented circles. If the tangent direction $\dot{\mathbf{c}}(u)$ encloses an angle with the plane $x_3 = 0$ that is more than/equal/less than $\frac{\pi}{4}$; we call the tangent line *pseudo-Euclidean/isotropic/Euclidean*, consequently $\zeta^{-1}(\mathbf{c}(u))$ is either imaginary or it consists of one or two real curves.

full description of Dupin cyclides as images of intersections of three cones in $\mathbb{R}^{3,1}$. They also define *pseudo-Euclidean* (‘pe’ in brief) *circular splines* and *pe biarcs*.

SLABAUGH et al. have published [6] about envelopes of 1-parameter families of circles and [7] as a direct extension to 1-parameter families of spheres.

Their 2D method is to look at the two envelope curves separately and choose a point of contact S_i for every circle, which automatically gives a tangent direction T_i . Given the circles, these points of contact can be coded as an angle α_i . For two consecutive circles, a cubic Hermite interpolation is computed. This interpolation is then set up as an unconstrained quadratic optimization in the unknown α_i with minimization of arc length and curvature as objective function. In 3D the points of contact are substituted by circles of contact, the rest extends straight forward.

The drawback of this method is that the curves are only C^1 and the knot vector of the piecewise cubic spline is determined by the rules of Hermite interpolation, thus the shape is not very flexible. We also assume that our algorithm converges faster to a specified input, even though a formal proof of convergence is still missing (see Section 4).

KUNKLI and HOFFMANN [3] do a G^1 interpolation of circles via the circle of Apollonius: for three consecutive circles S_1, S_2, S_3 two other touching circles A_1, A_2 are constructed and the points of contact with A_1 are used for one envelope and the ones with A_2 for the other. Then, for two consecutive circles, tangents at the points of contact are computed and a Hermite interpolation is performed.

Their extension for spheres is to take the plane ϵ through the centers of three consecutive spheres S_i , thus get three circles and once again apply Apollonius’ method. Through two touching points a plane orthogonal to ϵ intersects S_i in a circle, which is used for another Hermite interpolation.

Such a Hermite interpolation is only a local construction and success is heavily dependent on the input. As we will see in Section 2, the admissible set of circles of the present algorithm is more restricted than the definition given in [3].

2. Envelopes of circles

Finding an envelope of an ordered set of circles in \mathbb{R}^2 can be translated by cyclography into the following task: Given a set of points Q_k in $\mathbb{R}^{2,1}$, find a curve $\mathbf{c}(u)$ that interpolates these. The inverse cyclographic mapping maps $\mathbf{c}(u)$ to the two branches of the circles' envelope (see Figure 2, right).

A first look at the objective function

For given points Q_k , we want to

$$\text{minimize } \sum_{k=0}^{n-1} |Q_k - \mathbf{c}(u)|^2, \quad (2)$$

which means solving a least squares problem for unknown control points of a cubic B-spline curve $\mathbf{c}(u)$. In fact, this approximation uses a footpoint, i.e., a point $\mathbf{c}(u_k)$ which changes in an iterative sub-routine, thus making (2) non-linear.

Note that the norm in Eq. (2) is the Euclidean norm, because the Minkowski norm of Eq. (1) can become zero even for isotropic vectors, which is impractical for this curve fitting.

2.1. Introducing the constraint

The main challenge is that not every curve \mathbf{c} through points Q_k maps to a real envelope. As we have seen in Section 1 this is only satisfied if

$$c_1'^2 + c_2'^2 \geq c_3'^2, \quad (3)$$

where $\frac{\partial \mathbf{c}}{\partial u} = \mathbf{c}' = (c_1', c_2', c_3') \in \mathbb{R}^{2,1}$ is called *hodograph*. Wherever (3) is strictly $>$, the envelopes do not coincide. The condition (3) is equivalent to saying

$\Leftrightarrow \langle \mathbf{c}', \mathbf{c}' \rangle_L \geq 0$ by using the Lorentz inner product of Eq. (1), i.e., the derivative vectors are *Euclidean* everywhere.

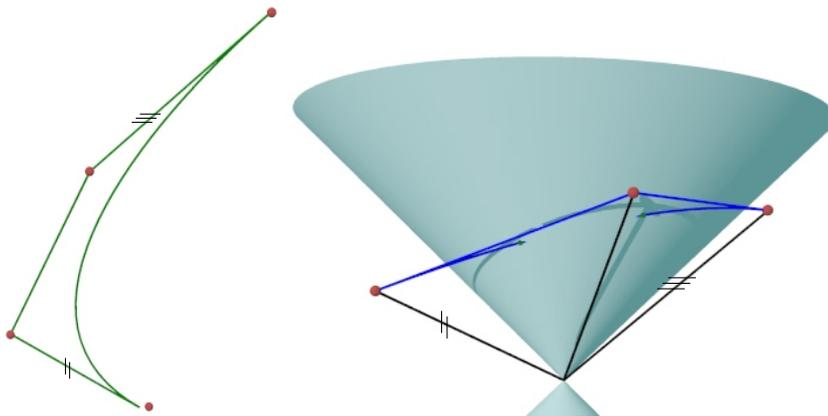


Figure 3: *Left*: a cubic Bézier curve (*green*) and its control polygon, *Right*: the hodograph (*blue*) is the curve of the first derivatives, which is a quadratic Bézier curve in the legs (*black*), i.e., the difference vectors. They originate at the origin of the Lorentz cone Γ (*turquoise*).

\iff the angle between \mathbf{c}' and the xy -plane is $\leq \frac{\pi}{4}$,

\iff that the hodograph has to stay outside the Lorentz cone Γ (see Figure 3).

This quadratic constraint is in fact non-convex as we will see shortly, thus standard optimization algorithms can not be applied. We will present a possibility to linearize it in Section 2.2.

Quadratic objective in matrix notation

The objective function Eq. (2) can be written as $(\mathbf{c}^T - \mathbf{q}^T) \cdot (\mathbf{c} - \mathbf{q}) \rightarrow \min$, thus

$$f(\mathbf{b}) = \mathbf{b}^T \cdot \mathbf{G} \cdot \mathbf{b} - 2 \cdot \mathbf{b}^T \cdot \mathbf{e} + \mathbf{q}^T \cdot \mathbf{q} \quad (4)$$

where $\mathbf{G} = \text{blkdiag}(\mathbf{N}^T \mathbf{N}, \mathbf{N}^T \mathbf{N}, \mathbf{N}^T \mathbf{N})$ is a symmetric block-diagonal matrix, and \mathbf{e} equals the product $\mathbf{e} = \text{blkdiag}(\mathbf{N}^T, \mathbf{N}^T, \mathbf{N}^T) \cdot \mathbf{q}$. $f(\mathbf{b})$ is also called *numerical convergence rate*.

Solution via a linear system

To solve (4) iteratively, one writes $\mathbf{b}_{i+1} = \mathbf{b}_i + \Delta \mathbf{b}_i$ and arrives at an optimum for

$$\nabla_{\mathbf{b}} f = \mathbf{G} \cdot (\mathbf{b}_i + \Delta \mathbf{b}_i) - \mathbf{e} = 0 \iff \mathbf{G} \cdot \Delta \mathbf{b}_i = \mathbf{e} - \mathbf{G} \cdot \mathbf{b}_i,$$

which is a linear system in $\Delta \mathbf{b}_i$ for fixed parameters u_k . We will assume to have found an optimum, whenever the absolute value of the change in error measure $\|\nabla f\|$ is smaller than some threshold.

2.2. Non-convex quadratic constraint

The curve $\mathbf{c} \in \mathbb{R}^{2,1}$ shall be constrained to have only Euclidean tangents, or equivalently, its cyclographic preimage shall be a real envelope to circles $\zeta^{-1}(Q_k)$. The derivative curve or *hodograph* is given as

$$[c'_1 | c'_2 | c'_3] = \mathbf{N}' \cdot [b_1 | b_2 | b_3],$$

with the same control points $\bar{\mathbf{b}} = [b_1 | b_2 | b_3]$ as the curve itself, and \mathbf{N}' denoting the collocation matrix of the derivatives of the basis functions. Then the constraint (3) can be reformulated as

$$b_1^T \mathbf{N}'^T \mathbf{N}' b_1 + b_2^T \mathbf{N}'^T \mathbf{N}' b_2 - b_3^T \mathbf{N}'^T \mathbf{N}' b_3 \geq 0 \iff \mathbf{b}^T \cdot \mathbf{A} \cdot \mathbf{b} \geq 0 \quad (5)$$

with $\mathbf{A} = \text{blkdiag}(\mathbf{N}'^T \mathbf{N}', \mathbf{N}'^T \mathbf{N}', -\mathbf{N}'^T \mathbf{N}')$. Note that the collocation matrix of the curve's derivative \mathbf{N}' does not need to be given at the same parameter values u_k , and the u_k do not need to change at every iteration.

Nevertheless, matrix \mathbf{A} is indefinite by construction, so methods like 'Quadratically constrained quadratic program' (QCQP) cannot be applied. Note that since \mathbf{A} stems from evaluating basis functions at finitely many points, this condition is necessary, but not sufficient to satisfy the constraints.

2.3. Optimization procedure

Now that we have defined the matrices representing both objective function and the constraint in Eqs. (4) and (5) respectively, we can restate the original problem:

$$\text{minimize } \mathbf{b}^T \cdot \mathbf{G} \cdot \mathbf{b} - 2 \cdot \mathbf{b}^T \cdot \mathbf{e} + d \quad (6a)$$

$$\text{subject to } \mathbf{b}^T \cdot \mathbf{A} \cdot \mathbf{b} \geq 0, \quad (6b)$$

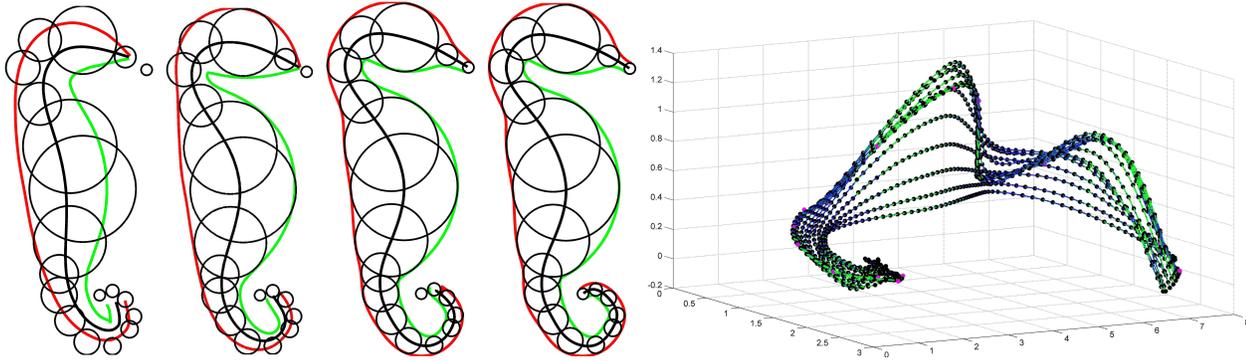


Figure 4: An example of envelopes using the Interior Point algorithm at different stages, *From left to right*: fifth iteration after initialization with a flat curve, after ten iterations, after 15 iterations, final result after 30 iterations.

Right: curves in $\mathbb{R}^{2,1}$ for the first 15 iterations. Note that the position of circles was inspired by [3] for easy comparison of methods and to prove that the present algorithm can also handle this situation.

where \mathbf{G} is positive definite and \mathbf{A} is indefinite (both by construction).

Due to the indefiniteness of the constraint, (6) is a non-convex problem. If we take reparametrizations into account, which are necessary in order to include footpoints, the objective function is nonlinear.

Outline of the optimization procedure

To overcome these challenges, we will follow this procedure:

- (1) Start with a good initial position \mathbf{b}_0 for the curve's control polygon.
- (2) Choose footpoints $\mathbf{c}_i(u_k)$.
- (3) Compute a linearization \mathbf{D}_i of the quadratic constraint depending on the current control points \mathbf{b}_i .
- (4) Minimize the distance from \mathbf{c}_i to Q_k while staying feasible w.r.t. \mathbf{D}_i and update the control points.

Repeat (2)–(4) until an optimum is reached.

We will make sure that the error introduced in (3) stays small by iteratively adapting the linearization in (2) and (3). Therefore, step (4) turns the non-convex, quadratically constraint problem (6) into a “Quadratic Program”. This important linearization will be carried out in the coming section.

2.4. Linearizing the quadratic constraint

In this section we will present a local linearization and proceed in three steps:

- Show in Section 2.4 that bounding the samples of the hodograph away from the Lorentz cone can locally be accomplished through Γ 's tangent planes.
- Present a routine that computes the corresponding tangent plane for each sample in Section 2.4 with the help of a projection
- Give a matrix formulation of the linearized constraint that only depends on the control points of the last iterate in Section 2.4.

Local linearization of Γ

As was said in the introduction, the quadratic constraint is equivalent to the hodograph \mathbf{c}' staying outside the Lorentz cone Γ .

Note that the space $\mathbb{R}^{2,1} \setminus \Gamma$ is not a vector space; a linear combination of two Euclidean vectors is not necessarily Euclidean. So rather than giving a condition on the *control polygon* of the hodograph, we look at *sampling points* $\mathbf{c}'(u_k)$.

The linearization for condition “ $\mathbf{c}'(u_k)$ must stay outside Γ ” will be “ $\mathbf{c}'(u_k)$ must stay on the positive side of a tangent plane of Γ ”. The former can be written as $c_1'^2 + c_2'^2 - c_3'^2 \geq 0$, the latter as $\bar{p}_1 c_1' + \bar{p}_2 c_2' - \bar{p}_3 c_3' \geq 0$ for $\bar{\mathbf{p}} = (\bar{p}_1, \bar{p}_2, \bar{p}_3)$ on the cone.

This connection comes naturally when looking at the equation of the border of the Lorentz cone $\partial\Gamma$, which is given by

$$\partial\Gamma: x^2 + y^2 - z^2 = 0 \quad (7)$$

thus the normal vector of its (isotropic) tangential plane ϵ is

$$\nabla(x^2 + y^2 - z^2) = (2x, 2y, -2z) = (x, y, -z).$$

Therefore, given a point $\bar{\mathbf{p}}$ on the cone it is contained in the plane

$$\epsilon: \bar{p}_1 x + \bar{p}_2 y - \bar{p}_3 z = 0. \quad (8)$$

For this tangent plane, the normal is pointing away from Γ and thus the linear form $\langle \bar{\mathbf{p}}, \mathbf{x} \rangle_L$ derived from Eq. (8) is positive for \mathbf{x} close to $\bar{\mathbf{p}}$ and outside Γ . This linear form is also a good local approximation of the quadratic form $\langle \mathbf{x}, \mathbf{x} \rangle_L$ for a suitable $\bar{\mathbf{p}}$.

Projection orthogonal to Γ

We have seen in the last section that a local linearization of Γ depends on the choice of a suitable point $\bar{\mathbf{p}} = (\bar{p}_1, \bar{p}_2, \bar{p}_3)$ on the cone for a point $\mathbf{p} = (p_1, p_2, p_3)$, such that $\langle \bar{\mathbf{p}}, \mathbf{p} \rangle_L$ is positive/negative/zero for a Euclidean/pseudo-Euclidean/isotropic point.

Keep in mind that these “points” are actually *derivative vectors* and hence belong to the tangent space of $\mathbb{R}^{2,1}$. A suitable choice is

$$\bar{\mathbf{p}} = \mathbf{p} - \lambda \cdot \frac{\mathbf{n}_\epsilon}{\|\mathbf{n}_\epsilon\|}, \quad \lambda = \frac{p_1^2 + p_2^2 \mp p_3 \sqrt{p_1^2 + p_2^2}}{\sqrt{2(p_1^2 + p_2^2)}} \quad (9)$$

for the smaller of the two λ 's.

2.5. Matrix formulation of the linearization

We are now ready to write the linearization of the quadratic constraint (5) in a matrix formulation that can be used in an optimization algorithm.

We have seen in Section 2.4 that for a derivative vector $\mathbf{d}_k = \mathbf{c}'(u_k) = \mathbf{N}'(u_k) \cdot \bar{\mathbf{b}}$ the quadratic constraint $\langle \mathbf{d}_k, \mathbf{d}_k \rangle_L \geq 0$ can be linearized as

$$\begin{aligned} \langle \bar{\mathbf{d}}_k, \mathbf{d}_k \rangle_L &= [\bar{d}_{k1}, \bar{d}_{k2}, \bar{d}_{k3}] \operatorname{diag}(1, 1, -1) \begin{bmatrix} d_{k1} \\ d_{k2} \\ d_{k3} \end{bmatrix} \\ &= [\bar{d}_{k1} \cdot \mathbf{N}'(u_k), \bar{d}_{k2} \cdot \mathbf{N}'(u_k), -\bar{d}_{k3} \cdot \mathbf{N}'(u_k)] \cdot \bar{\mathbf{b}} \geq 0, \end{aligned}$$

with $\bar{\mathbf{d}}_k$ the projection of \mathbf{d}_k onto Γ as described in Eq. (9), $\mathbf{N}'(u_k)$ one row of the collocation matrix of the hodograph \mathbf{c}' and \mathbf{b} the $(3m \times 1)$ -vector of the m control points of \mathbf{c} .

If we want to carry out this operation for all \mathbf{d}_k with a single matrix multiplication, we have to write the $\bar{\mathbf{d}}_k$ coordinate-wise in diagonal matrices and multiply these with the $(n \times m)$ -matrix \mathbf{N}' :

$$\mathbf{D} := [\text{diag}(\bar{d}_{11} \dots \bar{d}_{1n})\mathbf{N}', \text{diag}(\bar{d}_{21}, \dots, \bar{d}_{2n})\mathbf{N}', \dots, \text{diag}(\bar{d}_{31}, \dots, \bar{d}_{3n})\mathbf{N}']$$

and get the matrix (=linear form) linearizing the quadratic form of the constraint

$$\mathbf{D} \cdot \mathbf{b} = [\langle \bar{\mathbf{d}}_1, \mathbf{d}_1 \rangle_L, \dots, \langle \bar{\mathbf{d}}_n, \mathbf{d}_n \rangle_L]^T. \quad (10)$$

Remember that an entry $\langle \bar{\mathbf{d}}_k, \mathbf{d}_k \rangle_L$ in (10) is actually the signed distance for a point $\mathbf{c}'(u_k)$ to a certain (closest) tangent plane of Γ . Also keep in mind that even though \mathbf{N}' might remain unchanged throughout the iteration, \mathbf{D}_i depends on the control points \mathbf{b}_{i-1} of the last iterate, because $\mathbf{c}'_i = \mathbf{N}' \cdot \mathbf{b}_{i-1}$.

2.6. Initial position

We would like to have a much bigger number m of control points \mathbf{b} than the number n of points $Q_k \in \mathbb{R}^{2,1}$ representing the circles in order to have more flexibility in optimization. Trying to solve this directly would lead to a rank-deficient linear system whose solution \mathbf{b} can, of course, not be trusted.

To circumvent this problem, we introduce an auxiliary cubic B-spline curve \mathbf{c}_{aux} , interpolating the input Q_k with a minimal knot vector and ignoring the steepness constraint (see Figure 5, right).

We take $m_1 > m + 4$ equally spaced auxiliary parameter values v_i , a knot vector of length $m + 4$ and compute thus the collocation matrix \mathbf{N}_{aux} of rank m . The z -coordinates of the control points \mathbf{b}_{aux} of \mathbf{c}_{aux} are then moved half the way to the mean value of the z -coordinates of the Q_k , called $mean(Q_{k,z})$. Should \mathbf{c}_{aux} still have pseudo-Euclidean tangents, repeat this scaling; sample the final Euclidean curve $R_i = \mathbf{c}_{aux}(v_i)$.

The initial control polygon \mathbf{b} for the optimization procedure is the least squares fit of the over-determined system $\mathbf{N}_{aux} \cdot \mathbf{b} = R_i$. It is Euclidean by construction and the number of control points m is a user-specified number.

Note that due to the repeated scaling of the z -coordinates of \mathbf{c}_{aux} , the B-spline defined by \mathbf{b} can be quite far from the Q_k (only in z -direction); in the worst case, it is approximating the top view projection of the Q_k in the plane $z = mean(Q_{k,z})$.

3. Envelopes of spheres

The direct extension of the theory of Section 2 to the 4-dimensional $\mathbb{R}^{3,1}$ would be curves in $\mathbb{R}^{3,1}$, which represent envelopes of 1-parameter families of spheres, i.e., canal surfaces. Basically, the theory stays the same as in Section 2; we refer the interested reader to [1] to see which formulas have to be adapted to the higher dimension.

The main contribution of the section lies with surfaces in $\mathbb{R}^{3,1}$, whose inverse cyclographic image are envelopes of 2-parameter families of spheres. The constraint of having real envelopes means that the tangent planes of these surfaces have to stay Euclidean. In fact, we are talking of 2-planes in 4-space and Section 3.1 introduces a calculus that allows to classify such planes through a bilinear form.

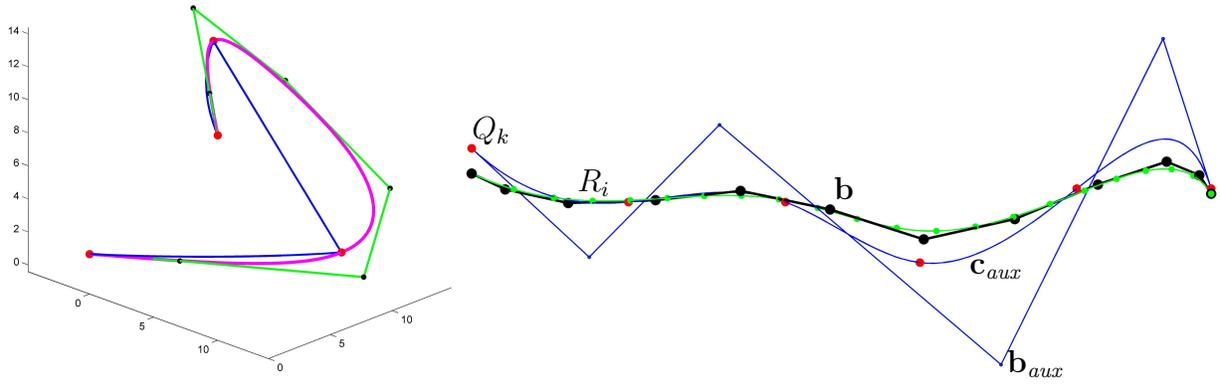


Figure 5: *Left:* The original points in $\mathbb{R}^{2,1}$ in *red* and their connecting lines in *blue* — note that the blue line in the middle has an angle of $\frac{\pi}{4}$ with the xy -plane by construction. The final curve (*purple*) avoids steep tangents, as can be seen by the final control polygon in *green*.

Right: Procedure to find the initial position of the control polygon: Compute auxiliary curve \mathbf{c}_{aux} (*blue curve*) with control polygon \mathbf{b}_{aux} (*blue lines*) through input points Q_k (*red dots*). Move this control polygon toward the mean of the z -coordinates of the Q_k until no steep tangents appear, then sample points R_i (*green dots*). A finer control polygon \mathbf{b} (*black*) for approximating those is found via a linear system.

3.1. Wedge product in 4-space

We want to define the wedge product $\mathbf{x} \wedge \mathbf{y}$ of two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^4$ (or $\mathbb{R}^{3,1}$; the differences between those two exterior algebras solely depend on the choice of basis), which is a vector in the $\binom{4}{2} = 6$ -dimensional vector space $\wedge^2 \mathbb{R}^4$.

Let $(\mathbf{e}_0, \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ be a basis of \mathbb{R}^4 , then a basis of $\wedge^2 \mathbb{R}^4$ is given as $(\mathbf{e}_0 \wedge \mathbf{e}_1, \mathbf{e}_0 \wedge \mathbf{e}_2, \mathbf{e}_0 \wedge \mathbf{e}_3, \mathbf{e}_2 \wedge \mathbf{e}_3, \mathbf{e}_3 \wedge \mathbf{e}_1, \mathbf{e}_1 \wedge \mathbf{e}_2)$. This linear space is the set of all 2-spaces (=planes through the origin) in 4-space.

Thus for two vectors $\mathbf{x} = (x_0, x_1, x_2, x_3)$, $\mathbf{y} = (y_0, y_1, y_2, y_3)$ the coefficients of the wedge product $\mathbf{x} \wedge \mathbf{y}$ are given by the numbers $l_{ij} = x_i y_j - x_j y_i$ for $i, j = 0, \dots, 3$, $i \neq j$, and the coordinates of the plane through the origin spanned by $\mathbf{x} \wedge \mathbf{y} \in \wedge^2 \mathbb{R}^4$ are $L = (l_{01}, l_{02}, l_{03}, l_{23}, l_{31}, l_{12})$. Note that a vector L satisfies the Plücker identity

$$\Omega_q(L) = l_{01}l_{23} + l_{02}l_{31} + l_{03}l_{12} = 0, \quad (11)$$

if and only if it represents a 2-plane in \mathbb{R}^4 (see [5] and the use of this formalism in *line geometry*). We heavily rely on the fact that (projective) lines in projective three-space \mathcal{P}^3 are isomorphic to 2-planes through the origin in \mathbb{R}^4 .

The bilinear form $\Phi_{3,1}$

We want to relate the inner product and the wedge product in $\wedge^2 \mathbb{R}^4$.

Lemma 1. *Let the Plücker coordinates of a plane $\mathbf{u} \wedge \mathbf{v}$ be l_{ij} for $\mathbf{u}, \mathbf{v} \in \mathbb{R}^{3,1}$ and those of $\mathbf{x} \wedge \mathbf{y}$ be m_{ij} . Then the following holds*

$$\det \begin{pmatrix} \langle \mathbf{u}, \mathbf{x} \rangle_L & \langle \mathbf{v}, \mathbf{x} \rangle_L \\ \langle \mathbf{u}, \mathbf{y} \rangle_L & \langle \mathbf{v}, \mathbf{y} \rangle_L \end{pmatrix} = l_{01}m_{01} + l_{02}m_{02} + l_{12}m_{12} - l_{03}m_{03} - l_{13}m_{13} - l_{23}m_{23}$$

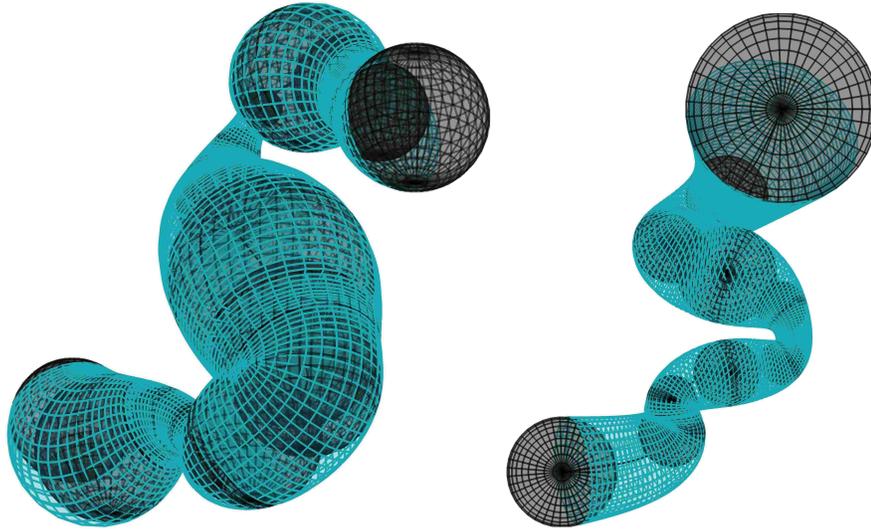


Figure 6: Two examples of envelopes of spheres using the Interior Point algorithm generalized to $\mathbb{R}^{3,1}$, e.g. canal surfaces. The original spheres are shown in *black*, the resulting envelope in *turquoise*. The curve approximation and the steepness constraint carry over directly from $\mathbb{R}^{2,1}$ to $\mathbb{R}^{3,1}$.

We define a bilinear form $\Phi_{3,1}: \bigwedge^2 \mathbb{R}^{3,1} \times \bigwedge^2 \mathbb{R}^{3,1} \rightarrow \mathbb{R}$ by setting

$$\Phi_{3,1}(\mathbf{u} \wedge \mathbf{v}, \mathbf{x} \wedge \mathbf{y}) := l_{01}m_{01} + l_{02}m_{02} + l_{12}m_{12} - l_{03}m_{03} - l_{13}m_{13} - l_{23}m_{23},$$

where l_{ij} are the Plücker coordinates of a plane $\mathbf{u} \wedge \mathbf{v}$ and m_{ij} those of $\mathbf{x} \wedge \mathbf{y}$ as in Lemma 1.

Let $\mathbf{u}, \mathbf{v} \in \mathbb{R}^{3,1}$ be two vectors. Then we get the important relation

$$\Phi_{3,1}(\mathbf{u} \wedge \mathbf{v}, \mathbf{u} \wedge \mathbf{v}) = \langle \mathbf{u}, \mathbf{u} \rangle_L \cdot \langle \mathbf{v}, \mathbf{v} \rangle_L - \langle \mathbf{u}, \mathbf{v} \rangle_L^2.$$

This is analogous to the lower dimensional case, which serves as a classification of 1-planes in $\mathbb{R}^{2,1}$ (see [1]).

3.2. Classification of 2-planes in Minkowski space

In this section we present a classification of 2-planes in $\mathbb{R}^{3,1}$ that fits consistently to the lower dimensional case, and extend a proposition on the classification of 1-planes in Minkowski space $\mathbb{R}^{2,1}$.

Lemma 2. *The bilinear form $\Phi_{3,1}$ allows for classification of 2-planes in $\mathbb{R}^{3,1}$: The plane $\mathbf{u} \wedge \mathbf{v}$ is*

1. *Euclidean* $\iff \Phi_{3,1}(\mathbf{u} \wedge \mathbf{v}, \mathbf{u} \wedge \mathbf{v}) > 0 \iff l_{01}^2 + l_{02}^2 + l_{12}^2 > l_{03}^2 + l_{13}^2 + l_{23}^2,$
2. *isotropic* $\iff \Phi_{3,1}(\mathbf{u} \wedge \mathbf{v}, \mathbf{u} \wedge \mathbf{v}) = 0 \iff l_{01}^2 + l_{02}^2 + l_{12}^2 = l_{03}^2 + l_{13}^2 + l_{23}^2,$
3. *pseudo-Euclidean* $\iff \Phi_{3,1}(\mathbf{u} \wedge \mathbf{v}, \mathbf{u} \wedge \mathbf{v}) < 0 \iff l_{01}^2 + l_{02}^2 + l_{12}^2 < l_{03}^2 + l_{13}^2 + l_{23}^2$

Note that this lemma first appeared in [2], and was reinvented in [1], where an alternative proof was found. We will use this classification of 2-planes in Section 3.3 to ensure that a tangent plane to a surface in $\mathbb{R}^{3,1}$ is Euclidean in an optimization routine.

3.3. Surface optimization

We want to turn our attention to surfaces in $\mathbb{R}^{3,1}$, e.g., the optimization of envelopes for 2-parameter families of spheres. The optimization problem thus becomes

$$\text{minimize } \sum_{i=1}^k |Q_i - \mathbf{f}(u, v)|^2 \quad (12a)$$

$$\text{subject to } \zeta^{-1}(\mathbf{f}(u, v)) \text{ is real,} \quad (12b)$$

and the distance in the objective function (12a) is the *Euclidean* distance, rather than the distance based on the *Minkowski norm* of Eq. (1). In the latter case we would run into all sorts of problems with isotropic directions, which we want to avoid altogether.

The constraint (12b) means that the inverse cyclographic image of the surface parametrization in \mathbb{R}^3 stays real, or, equivalently, that the tangent planes of $\mathbf{f}(u, v) \subset \mathbb{R}^{3,1}$ stay Euclidean. The formulation of this constraint is more involved than in the curve case, and we will explain it in several steps in Section 3.4.

Computation and parametrization

We have introduced a formalism in Section 3.2 that allows us to check whether a 2-plane in $\mathbb{R}^{3,1}$ is pseudo-Euclidean, isotropic or Euclidean; here we want to apply it to tangent planes of a surface. We must therefore choose a parametrization $\mathbf{f}(u, v)$ and for practical reasons we restrict our attention to tensor product B-spline surfaces.

Handling the constraint — three levels of rigidity

For three different classes of surfaces we present three levels of rigidity for the steepness constraint:

1. The most general class of surfaces we will look at are *bicubic tensor product B-spline surfaces* $\mathbf{f}(u, v)$, i.e., the parameter lines in u - and v -direction are all cubic B-spline curves. On these, we use a sampling of the surface and can only guarantee that the tangent planes are Euclidean at each sampling point, which means that the envelope of a 2-parameter family of spheres $\zeta^{-1}(\mathbf{f}(u, v))$ is real at a discrete number of points. This general approach, which is the foundation of all three, will be explained in Section 3.4.
2. If we consider tensor product B-splines surfaces of bidegree $(3, 1)$, i.e., strips of *ruled surfaces*, one direction of parameter lines, say u -lines, will be cubic B-splines and the other direction linear B-splines, i.e., polylines. We will sample the cubic u -parameter lines and given that they are Euclidean (on a sampling), the tangent planes along the rulings (= the v -parameter lines) can be guaranteed to be Euclidean — without the use of a sampling.
3. If both directions are linear, the single patches of the tensor product surface are *hyperbolic paraboloids* in $\mathbb{R}^{3,1}$. For this class of surfaces, we can even guarantee Euclidean surfaces without the need of a sampling.

For brevity, we will only describe the most general class here and refer to [1] for the other two.

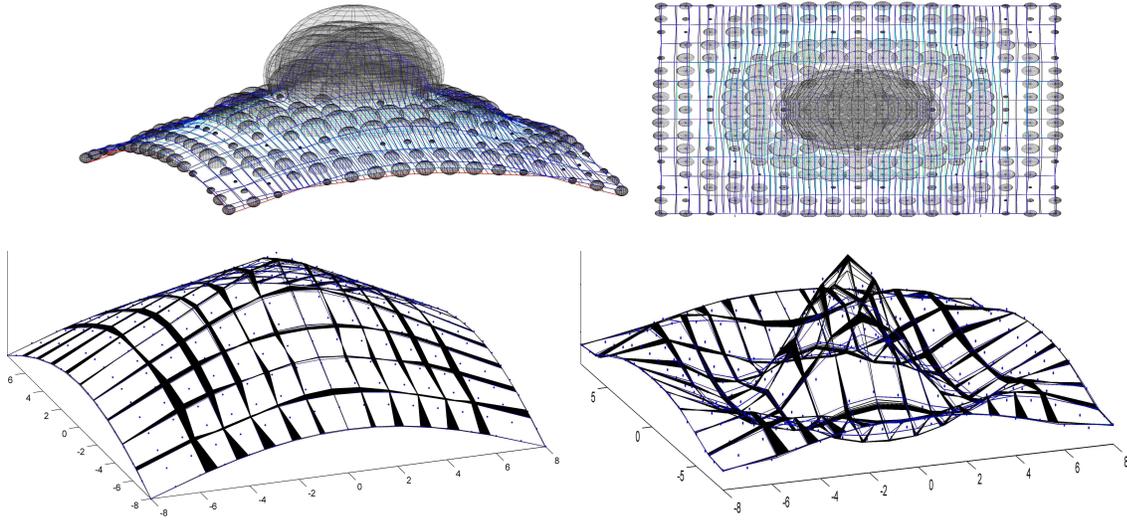


Figure 7: The surface $f(u, v) = (u, v, 3 - \frac{r}{2}, 3 \frac{\sin(r)}{r})$:

Top row, Left: an axonometric view of the hyperplanar section $x_3 = 0$, *Right:* an axonometric view of the hyperplanar section $x_2 = 0$; surface evaluated on a grid, points with pseudo-Euclidean (mixed) partial derivatives in *red*;

Second row: the cyclographic preimage of an initial value for a bicubic B-spline surface approximation; *Left:* values in the hyperplanar section $x_3 = 0$, *Right:* in hyperplanar section $x_2 = 0$ from start (*blue*) and intermediates to finish (*black*).

3.4. Constraints for surface optimization

Analogously to Section 2.4, we have to give a linearized version of the quadratic constraint (12b), which in the surface case is actually slightly more involved. The constraint should be, that for a surface $\mathbf{f}(u, v)$ its tangent planes stay Euclidean everywhere. We will sample the parameter lines to check the steepness of the partial derivatives.

We describe this constraint in the following steps:

1. We ensure Euclidean tangents in the direction of the parameter lines, i.e., the constraint shall be fulfilled for a sampling of the partial derivatives $\mathbf{f}_u(u_i, v_j)$ and $\mathbf{f}_v(u_i, v_j)$.
2. If both $\mathbf{f}_u(u_i, v_j)$ and $\mathbf{f}_v(u_i, v_j)$ are Euclidean, we test if the tangent plane at $\mathbf{f}(u_i, v_j)$ is Euclidean by computing $\Phi_{3,1}(\mathbf{f}_u \wedge \mathbf{f}_v, \mathbf{f}_u \wedge \mathbf{f}_v)(u_i, v_j)$. We give a matrix formulation for constraining the tangent plane of a surface in $\mathbb{R}^{3,1}$ to be Euclidean.

For the special cases of strips of ruled surfaces and patches of hyperbolic paraboloids, further steps are necessary (see [1]).

The constraint on parameter lines

We assume $\mathbf{f}(u, v)$ is parametrized as a bicubic tensor product B-spline surface, hence the parameter lines in u and v -direction are cubic B-spline curves. Therefore, constraining them to have only Euclidean tangents is exactly constraint (12b), i.e., the constraint for a curve in $\mathbb{R}^{3,1}$ to have only Euclidean tangents.

Then, for k sample points on the surface $\mathbf{f}(u, v)$, the $(k \times 4)$ -matrix of partial derivatives w.r.t. u is given as $\mathbf{f}_u = \mathbf{N}_{du} \cdot \mathbf{b}$ and we define $\bar{\mathbf{f}}_u$ to be the projection — again a $(k \times 4)$ -matrix

— of \mathbf{f}_u onto Γ as defined in Section 2.4.

Then $\mathbf{D}_u \cdot [\mathbf{b}_x^T | \mathbf{b}_y^T | \mathbf{b}_z^T | \mathbf{b}_w^T]^T = \langle \mathbf{f}_u, \bar{\mathbf{f}}_u \rangle_L$ is a vector, whose k th entry is less than zero if and only if $\mathbf{f}_u(u_k, v_k)$ is pseudo-Euclidean.

The matrix \mathbf{D}_u thus linearizes the quadratic constraint that the angle between the partial derivatives w.r.t. u , $\mathbf{f}_u(u_k, v_k)$, and the hyperplane $x_4 = 0$ is less than $\frac{\pi}{4}$ for a sampling of all parameter lines in u -direction.

The constraint for a discrete sampling of the surface

Knowing that the partial derivatives $\mathbf{f}_u(u_i, v_j)$ and $\mathbf{f}_v(u_i, v_j)$ are Euclidean does not necessarily mean that the tangent plane at $\mathbf{f}(u_i, v_j)$ is Euclidean.

We thus employ the calculus summarized in Lemma 2: Compute $\Phi_{3,1}(\mathbf{f}_u \wedge \mathbf{f}_v, \mathbf{f}_u \wedge \mathbf{f}_v)(u_i, v_j)$; if it is ≤ 0 , the tangent plane at $\mathbf{f}_v(u_i, v_j)$ is isotropic or pseudo-Euclidean, even though the dual variables of the optimization are positive by assumption. We hence change their sign and thus activate this constraint.

Algorithm for constraining the tangent planes of a surface

Let us summarize the ideas of this section as an algorithm:

1. For a sampling of a B-spline surface $\mathbf{f}(u_i, v_j)$, compute the matrices \mathbf{D}_u and \mathbf{D}_v of the linearized constraint.
2. For index pairs (\bar{i}, \bar{j}) for which the slack variables > 0 , compute the bilinear form $\Phi_{3,1}(\mathbf{f}_u \wedge \mathbf{f}_v, \mathbf{f}_u \wedge \mathbf{f}_v)(u_{\bar{i}}, v_{\bar{j}})$.
3. For those (\bar{i}, \bar{j}) for which $\Phi_{3,1} \leq 0$, change the sign of the slack variable to be minus.

With the simple trick in 3., which is at the heart of the primal-dual interior point algorithm used to solve these problems, we have activated the constraint with the help of $\Phi_{3,1}$, even though both partial derivatives are Euclidean. This way, isotropic or pseudo-Euclidean tangent planes enter the optimization.

4. Conclusions and future work

This paper summarizes the author's PhD thesis [1], which finds envelopes for circles in the plane and for 1- and 2-parameter families of spheres in 3-space via a nonlinear quadratic optimization in Minkowski space. Methods for handling a non-convex constraint are employed and results are presented through examples.

For future work we point out, that a formal proof of convergence for a given set of circles or spheres of the presented algorithm is still missing.

Futhermore, we have tried to avoid self-intersections of the envelope curves $\zeta(p(t))$ by adding a penalty term to the objective function, whenever the curve of regression of the torsal surface of constant slope through $p(t)$ intersects the xy -plane. In our implementation, this penalty term was too restrictive, i.e., the envelope never got close enough to the input circles. One can also imagine constellations of circles, for which self-intersections are necessary in order to achieve a good fit.

Acknowledgments

I want to thank Martin PETERNELL who was the supervisor of my PhD thesis [1], of which this paper is a condensed version, for his advice and feedback. Thanks also to the reviewers of the Journal of Geometry and Graphics for their helpful comments as well as the Managing Editor Professor STACHEL for his careful copy-editing.

References

- [1] B. BLASCHITZ: *Geometric Optimization in Minkowski Space*. PhD Thesis, Vienna University of Technology, 2014.
- [2] J. KOSINKA, B. JÜTTLER: *MOS surfaces: Medial surface transforms with rational domain boundaries*. In *Mathematics of Surfaces XII*, Springer, 2007, pp. 245–262.
- [3] R. KUNKLI, M. HOFFMANN: *Skimming of circles and spheres*. *Comput. Aided Geom. Design* **27**(8), 611–621 (2010).
- [4] H. POTTMANN, M. PETERNELL: *Applications of Laguerre geometry in CAGD*. *Comput. Aided Geom. Design* **15**(2), 165–186 (1998).
- [5] H. POTTMANN, J. WALLNER: *Computational Line Geometry*. Springer Verlag, Berlin 2001.
- [6] G. SLABAUGH, G. UNAL, T. FANG, J. ROSSIGNAC, B. WHITED: *Variational skinning of an ordered set of discrete 2D balls*. *Proceedings 5th International Conference on Advances in Geometric Modeling and Processing*, Springer-Verlag, 2008, pp. 450–461.
- [7] G. SLABAUGH, B. WHITED, J. ROSSIGNAC, T. FANG, G. UNAL: *3D ball skinning using PDEs for generation of smooth tubular surfaces*. *Computer-Aided Design* **42**(1), 18–26 (2010).

Received August 6, 2014; final form October 17, 2014