

Optimal Synthesis of Overconstrained 6R Linkages by Curve Evolution

Tudor-Dan Rad and Hans-Peter Schröcker

*University of Innsbruck, Unit Geometry and CAD, Technikerstraße 13, 6020
Innsbruck, Austria, e-mail: {tudor-dan.rad}{hans-peter.schroecker}@uibk.ac.at*

Abstract. The paper presents an optimal synthesis of overconstrained linkages, based on the factorization of rational curves (representing one parametric motions) contained in Study's quadric. The group of Euclidean displacements is embedded in a affine space where a metric between motions based on the homogeneous mass distribution of the end effector is used to evolve the curves such that they are fitted to a set of target poses. The metric will measure the distance (in Euclidean sense) between the two resulting vectors of the feature points displaced by the two motions. The evolution is driven by the normal velocity of the curve projected in the direction of the target points. In the end we present an example for the optimal synthesis of an overconstrained 6R linkage by choosing a set of target poses and explaining in steps how this approach is implemented.

Key words: optimal mechanism synthesis, curve evolution, kinematic mapping, motion factorization, overconstrained linkage

1 Introduction

A linkage is a mechanism which generates a complex motion. The synthesis of a linkage means determining its geometric structure such that it generates a predetermined motion or trajectory and satisfies some structural restrictions. Fulfilling the previous requirements puts a lot of limitations on the linkage. This gave rise to optimal synthesis which aims at approximating these requirements. Some of the optimization techniques used in optimal synthesis of linkages are: interior-point methods [13], Gauss constraint methods [11], genetic algorithms [3] and evolution [12].

The paper also takes an evolutionary approach to synthesis. The novelty consists in using the factorization of motion polynomials [4] synthesis process. We demonstrate that it is particularly well-suited for evolution techniques because it allows to construct (overconstrained) linkages directly from a given approximated rational motion.

The factorization of motion polynomials is a process that generates a linkage which performs a one parametric motion (the functions of the joint angles share the same parameter). This motion must be defined by a rational curve in the kinematic

image space. The motion curve is constructed by starting from a set of target points in this space that resemble the poses needed to be achieved by the linkage. Using curve evolution methods [1], the initial motion curve will converge and approximate the specified poses.

For the evolution process to work for such curves the group of Euclidean displacements $SE(3)$ is embedded in 12-dimensional affine space \mathbb{R}^{12} . A *metric* between two motions is used by equipping the end effector of the linkage with a homogeneous mass distribution or a set of “feature points” whose barycenter is the tool center point (TCP) [8]. The metric will measure the distance (in Euclidean sense) between the two resulting vectors of the feature points displaced by the two motions.

The paper is structured as follows Section 2 explains the Euclidean metric in the affine space \mathbb{R}^{12} and offers a quick glimpse into motion factorization and overconstrained linkage construction, Section 3 presents the evolutionary design of the motion curve. Section 4 follows up with an example and in Section 5 some conclusions are drawn.

2 Preliminaries

The group of special Euclidean displacements $SE(3)$ represents rigid body displacements and is used to map a point \mathbf{p} to a new position \mathbf{p}' in Euclidean three-dimensional space:

$$\gamma: \mathbb{R}^3 \rightarrow \mathbb{R}^3, \quad \mathbf{p}' = \mathbf{A}\mathbf{p} + \mathbf{a}. \quad (1)$$

The matrix \mathbf{A} is a 3×3 special orthogonal matrix representing an element of the rotation group $SO(3)$ and the vector \mathbf{a} is a translation vector. Because displacements incorporate multiple distance concepts defining a metric between them can be problematic. In the past the concept was addressed for example by [10] but due to the nature of motion design used in this paper we have chosen the method proven in motion design by [6]. This approach embeds $SE(3)$ in a 12-dimensional affine space by mapping the entries of \mathbf{A} and \mathbf{a} to a 12-dimensional vector. In an object oriented metric the gripper is given by a set of “feature points” \mathbf{fp}_i and the Euclidean metric is defined by the inner product $\langle \alpha, \beta \rangle := \sum_i \langle \alpha(\mathbf{fp}_i), \beta(\mathbf{fp}_i) \rangle$ for any $\alpha, \beta \in SE(3)$. The corresponding squared distance is $\|\alpha - \beta\|^2 = \langle \alpha - \beta, \alpha - \beta \rangle$. It is well-known [8] that this metric only depends on the barycenter and the inertia tensor of the set of feature points and is capable of representing more general mass distributions in a computationally simple way.

Motion factorization is a method developed by Hegedüs, Schicho and Schröcker in [4] and can be used to synthesize linkages with one degree of freedom joints whose end link motion is defined by a rational curve on Study’s quadric. By combining multiple factorizations overconstrained linkages can be constructed as was demonstrated in [4].

For further understanding of the synthesis process a quick introduction to the kinematic image space and Study’s quadric is necessary. Study’s kinematic mapping

maps the group $SE(3)$ to a quadric in seven dimensional projective space \mathbb{P}^7 with the equation $x_0y_0 + x_1y_1 + x_2y_2 + x_3y_3 = 0$ called Study's quadric and denoted by \mathcal{S} . A more detailed explanation is given by Husty and Schröcker in [9]. The points of \mathbb{P}^7 are represented by the skew ring of dual quaternions \mathbb{DH} , denoted as $q = x_0 + x_1\mathbf{i} + x_2\mathbf{j} + x_3\mathbf{k} + \varepsilon(y_0 + y_1\mathbf{i} + y_2\mathbf{j} + y_3\mathbf{k})$ with the multiplication properties:

$$\varepsilon^2 = 0, \quad \mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1, \quad \varepsilon\mathbf{i} = \mathbf{i}\varepsilon, \quad \varepsilon\mathbf{j} = \mathbf{j}\varepsilon, \quad \varepsilon\mathbf{k} = \mathbf{k}\varepsilon.$$

The conjugate of a dual quaternion is given by replacing \mathbf{i} , \mathbf{j} and \mathbf{k} with $-\mathbf{i}$, $-\mathbf{j}$, and $-\mathbf{k}$, respectively. A dual quaternion on \mathcal{S} is characterized by $q\bar{q} \in \mathbb{R}$.

The motion factorization algorithm of [4] starts with a rational curve of degree n on \mathcal{S} given by the polynomial $P(t) = c_0t^n + c_1t^{n-1} + \dots + c_n$ where $c_\ell \in \mathbb{DH}$ and $P\bar{P} \in \mathbb{R}[t]$. Generically (only generic cases are relevant for evolution based synthesis), it can be factored as $P(t) = (t - h_1) \cdot \dots \cdot (t - h_n)$. The linear factors $t - h_\ell$ are computed by polynomial division over the dual quaternions using the quadratic irreducible factors M_i of $P\bar{P} = M_1M_2 \cdot \dots \cdot M_n$ one at a time in the following manner: By polynomial division, polynomials P_{n-1}, R are attained with $P_n = P_{n-1}M_i + R$ and $R = r_1t + r_2$. In [4] it was proven that the unique dual quaternion zero $h_n = -r_1^{-1}r_2$ of R gives the rightmost factor $t - h_n$ in a possible factorization of P_n . To obtain the remaining linear factors, another quadratic factor M_k is chosen and the process is repeated with P_{n-1} instead of P_n .

Each of the n linear factors represent a revolute displacement around an axis and by consecutive multiplication to the right they form a linkage whose leftmost factor is the fixed joint and the rightmost factor is the distal joint. There are, in general, $n!$ different possibilities for the selection order of the M_ℓ 's. This leads to the synthesis of $n!$ different open chains that perform the same motion. As it was shown in [4] an overconstrained linkage can be constructed by combining multiple kinematic chains to form a closed structure.

3 Curve evolution on Study's quadric

Curve evolution is a widely used procedure in image processing and design and of late is also used in motion generation [7]. Our evolutionary approach is based on curve fitting to a set of data points driven by the normal velocity of the curve in the direction of the target points [2]. By mapping the desired poses to \mathcal{S} is obtained the set of target points TP_m which need to be approximated by a rational curve C also contained in \mathcal{S} , that is, satisfying $C\bar{C} \in \mathbb{R}[t]$. The validity of this condition is ensured throughout the evolution process by writing C in factorized form $C = (t - h_1) \cdot \dots \cdot (t - h_n)$ where each linear factor represents a rotation about an axis in space. The linear factors are defined in (2) where the Plücker coordinates of the revolute axes are $(\mathbf{d}_i, \mathbf{m}_i)$:

$$t - h_i = \frac{t - h_{0i} - \mathbf{d}_i - \varepsilon\mathbf{m}_i}{\|\mathbf{d}_i\|}. \quad (2)$$

By fluctuating the shape parameters Sp_1, \dots, Sp_k (coefficients of C) in time a family of curves C_k is obtained such that the target points TP_1, \dots, TP_m are optimally approximated. The moving velocity of a curve point $C(t_j)$ is given by the amount of change in time of the shape parameters Sp_k :

$$v_{Cj} = \sum_{l=1}^k \frac{\partial C(t_j)}{\partial Sp_l} \dot{Sp}_l \quad (3)$$

We are interested in moving the points on $C(t)$ which are closest to the target points. These points are computed as the foot-normals between the TP_m and $C(t)$ using the Euclidean structure given by the inner product defined in Section 2:

$$\langle TP_m - C(t), C'(t) \rangle = 0 \rightsquigarrow \{t_{m1}, \dots, t_{ml}\}, \quad (4)$$

$$t_m = \arg \min(\|TP_m - C(t_{mi})\|^2 : i \in \{1, \dots, l\}). \quad (5)$$

Note that the involved computations essentially boils down to finding the zeros of a univariate polynomials because the motion is given by a polynomial C . This is one of the advantages inherent to our approach.

The foot-points $FP_m = C(t_m)$ are computed using relations (4) and (5) and so the ideal velocity vector $\hat{\mathbf{d}}$ of the foot-points should be $TP_m - FP_m$. Comparing coefficients of both vectors in an orthonormal basis (with respect to the given scalar product) and using (3) results in an overconstrained system of linear equations for Sp_l that can be solved in least square sense. The new shape parameters of the curve are computed as: $Sp_l = Sp_l + \lambda \dot{Sp}_l$ where λ is a scaling parameter used such that the curve doesn't overshoot. In time as the distance between the curve $C(t)$ and the target points TP_m decreases, the system will converge to a local minimum.

4 Numeric Example

For the example, a set of 11 target poses is chosen as shown in Table 1. Without loss of generality, the first pose is the identity. A cubic curve $C(t) = (t - \mathbf{x})(t - \mathbf{y})(t - \mathbf{z})$ is chosen to approximate the target poses. We limit ourselves to polynomials of degree three because the end goal of the example is to construct a 6R overconstrained linkage. More explicitly, the linear factor $t - \mathbf{x}$ is of the shape

$$\frac{t - x_0 + x_1 \mathbf{i} + x_2 \mathbf{j} + x_3 \mathbf{k} - \varepsilon((x_2 x_7 - x_3 x_6) \mathbf{i} + (x_3 x_5 - x_1 x_7) \mathbf{j} + (x_1 x_6 - x_2 x_5) \mathbf{k})}{\sqrt{x_1^2 + x_2^2 + x_3^2}}, \quad (6)$$

and similar for $t - \mathbf{y}$ and $t - \mathbf{z}$, resulting in a total of 21 shape parameters. The special shape of (6) is crucial. It ensures validity of the Study condition for each factor and hence also for $C(t)$ throughout the whole evolution process. The linear factors are normalized to avoid numeric fluctuation of $C(t)$ without any geometric change.

Table 1 Target Poses

TP	Study Parameters
1	1
2	$\mathbf{i}(-\frac{337}{7})\boldsymbol{\varepsilon} - \frac{79}{968} + \mathbf{j}(\frac{22}{379})\boldsymbol{\varepsilon} + \frac{3}{5066} + \mathbf{k}(\frac{67}{161})\boldsymbol{\varepsilon} + \frac{55}{509} - (\frac{108}{53})\boldsymbol{\varepsilon} + \frac{79}{41}$
3	$\mathbf{i}(-\frac{2533}{19})\boldsymbol{\varepsilon} - \frac{73}{328} + \mathbf{j}(\frac{92}{105})\boldsymbol{\varepsilon} + \frac{8}{3103} + \mathbf{k}(\frac{46}{89})\boldsymbol{\varepsilon} + \frac{41}{128} - (\frac{313}{29})\boldsymbol{\varepsilon} + \frac{119}{44}$
4	$\mathbf{i}(-\frac{949}{4})\boldsymbol{\varepsilon} - \frac{30}{77} + \mathbf{j}(\frac{163}{43})\boldsymbol{\varepsilon} + \frac{7}{815} + \mathbf{k}(-\frac{51}{128})\boldsymbol{\varepsilon} + \frac{153}{245} - (\frac{528}{19})\boldsymbol{\varepsilon} + \frac{95}{29}$
5	$\mathbf{i}(-\frac{1696}{5})\boldsymbol{\varepsilon} - \frac{110}{201} + \mathbf{j}(\frac{247}{21})\boldsymbol{\varepsilon} + \frac{11}{456} + \mathbf{k}(-\frac{89}{24})\boldsymbol{\varepsilon} + \frac{906}{907} - (\frac{660}{13})\boldsymbol{\varepsilon} + \frac{165}{46}$
6	$\mathbf{i}(-\frac{2939}{7})\boldsymbol{\varepsilon} - \frac{79}{119} + \mathbf{j}(\frac{1279}{46})\boldsymbol{\varepsilon} + \frac{19}{358} + \mathbf{k}(-\frac{1449}{145})\boldsymbol{\varepsilon} + \frac{100}{71} - (\frac{816}{11})\boldsymbol{\varepsilon} + \frac{18}{5}$
7	$\mathbf{i}(-461\boldsymbol{\varepsilon} - \frac{487}{682}) + \mathbf{j}(\frac{575}{11})\boldsymbol{\varepsilon} + \frac{10}{101} + \mathbf{k}(-\frac{557}{32})\boldsymbol{\varepsilon} + \frac{135}{76} - (\frac{185}{2})\boldsymbol{\varepsilon} + \frac{257}{78}$
8	$\mathbf{i}(-\frac{1364}{3})\boldsymbol{\varepsilon} - \frac{115}{167} + \mathbf{j}(\frac{864}{11})\boldsymbol{\varepsilon} + \frac{31}{212} + \mathbf{k}(-\frac{347}{14})\boldsymbol{\varepsilon} + \frac{291}{145} - (\frac{705}{7})\boldsymbol{\varepsilon} + \frac{191}{70}$
9	$\mathbf{i}(-\frac{1226}{3})\boldsymbol{\varepsilon} - \frac{44}{73} + \mathbf{j}(\frac{957}{10})\boldsymbol{\varepsilon} + \frac{181}{1018} + \mathbf{k}(-\frac{817}{27})\boldsymbol{\varepsilon} + \frac{75}{37} - (\frac{694}{7})\boldsymbol{\varepsilon} + \frac{390}{191}$
10	$\mathbf{i}(-\frac{1713}{5})\boldsymbol{\varepsilon} - \frac{85}{173} + \mathbf{j}(\frac{1921}{20})\boldsymbol{\varepsilon} + \frac{20}{109} + \mathbf{k}(-\frac{713}{22})\boldsymbol{\varepsilon} + \frac{106}{57} - (\frac{715}{8})\boldsymbol{\varepsilon} + \frac{38}{27}$
11	$\mathbf{i}(-\frac{8831}{32})\boldsymbol{\varepsilon} - \frac{77}{201} + \mathbf{j}(\frac{248}{3})\boldsymbol{\varepsilon} + \frac{52}{311} + \mathbf{k}(-\frac{458}{15})\boldsymbol{\varepsilon} + \frac{73}{46} - (\frac{1476}{19})\boldsymbol{\varepsilon} + \frac{109}{119}$

A suitable initial guess for the shape parameter can be found by interpolating four poses [5] or, as we did in our example, by assigning random values to the shape parameters. Several attempts might be necessary in order to ensure good convergence. Once the evolution runs smoothly little effect on the local minimum has been observed. In the first iterations, the scaling factor λ needs to be small enough in order to compensate for large amount of changes Sp_i in the shape parameters. For the evolution to have a good flow, we found $\lambda := \max\{10\|\text{Sp}_i\|_\infty^{-1}, 1\}$ to be a good choice. With this initial setup we arrive at the computation of the foot points on $C(t)$ as described in the previous section. From relation (4) we obtain an equation of degree at most 10. Its zeros are found numerically and we can use (5) to find the parameter value of the closest point. Here, we also impose some constraints on the foot point computation in order to ensure that the poses are visited in successive order. To achieve this, after a provisional curve is evolved, two target points are chosen which are approximated best. Next interval constraints are applied to the remaining foot points such that their respective parameter values t_m are in successive order. If the computed values of foot points do not fit the constraint interval, the boundary point with the minimal distance is chosen.

The evolution rule consists of comparing coordinates of the curve velocity (3) with respect to some orthonormal basis of \mathbb{R}^{12} with the coordinates of the difference vector from foot point to target point. This produces an overconstrained system which is solved in least square sense. Hence the shape parameters Sp_{pl} approximately change with the corresponding amount and the process is repeated again starting with the foot point computation. The final results are presented in Table 2 with a three decimal digit precision. The evolution process itself is visualized in Figure 1. The target poses are labelled from 1 to 10, the angles and distances to the respective target poses are given in Table 3. It can be seen that the distances are quite

good while the orientation seems to be hard to match. The reasons for this are under investigation. We conjecture an inappropriate distribution of feature points.

Table 2 Final Shape Parameters

x_0	x_1	x_2	x_3	x_5	x_6	x_7
5.822	-0.213	0.2	-0.337	-329.055	82.644	-100.544
y_0	y_1	y_2	y_3	y_5	y_6	y_7
6.084	0.051	-0.244	0.287	-8.987	-749.392	937.288
z_0	z_1	z_2	z_3	z_5	z_6	z_7
4.926	0.061	0.181	0.384	-99.5	-423.666	34.386

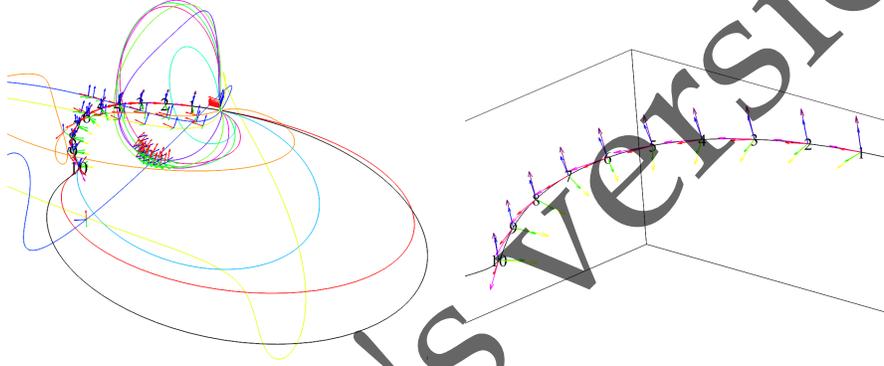


Fig. 1 TCP trajectory and orientation during the evolution process

Table 3 Variation of angle (in radians) and distance

TP_s	TP_1	TP_2	TP_3	TP_4	TP_5	TP_6	TP_7	TP_8	TP_9	TP_{10}
ϕ	0.061	0.102	0.125	0.139	0.124	0.106	0.08	0.061	0.151	0.421
t	0.922	1.488	1.38	1.182	1.522	2.689	4.914	8.26	3.736	4.336

After the motion curve $C(t) = (t - \mathbf{x})(t - \mathbf{y})(t - \mathbf{z})$ is obtained we can start the synthesis of the overconstrained 6R linkage using motion factorization [4] as explained in Section 2. First the quadratic factors M_i are computed by multiplying the curve with its quaternion conjugate:

$$C\bar{C} = (t^2 - 12.165t + 37.143)(t^2 - 11.648t + 34.116)(t^2 - 9.853t + 24.456) \quad (7)$$

By selecting the first quadratic factor from (7) polynomial division (a variant of Euclid's algorithm taking into account the non-commutativity of quaternion multi-

plication) is used to divide $C(t)$ and single out the remainder

$$(-59.057\mathbf{i}\epsilon - 0.191\mathbf{i} - 9.9\mathbf{j}\epsilon + 0.036\mathbf{j} - 13.531\mathbf{k}\epsilon + 0.134\mathbf{k} - 16.841\epsilon + 0.352)t + 347.317\mathbf{i}\epsilon + 1.143\mathbf{i} + 70.062\mathbf{j}\epsilon - 0.222\mathbf{j} + 80.891\mathbf{k}\epsilon - 0.667\mathbf{k} + 94.976\epsilon - 2.208 \quad (8)$$

The constant term h_{13} in the rightmost factor is computed as a unique root of this linear remainder polynomial:

$$h_{13} = \mathbf{i}(30.463\epsilon + 0.135) + \mathbf{j}(16.643\epsilon + 0.135) + \mathbf{k}(19.361\epsilon - 0.329) + 6.084 \quad (9)$$

After the first root is computed $C(t)$, is divided by $t - h_{13}$ and the process is iterated with the quotient and with one of the remaining quadratic factors from (7). After the second root is computed the quotient will be the last linear factor. All the possible combinations in which the quadratic factor can be chosen will produce six different open 3R kinematic chains. Suitable combinations [4] then give overconstrained 6R linkages. Four examples are depicted in Figure 2

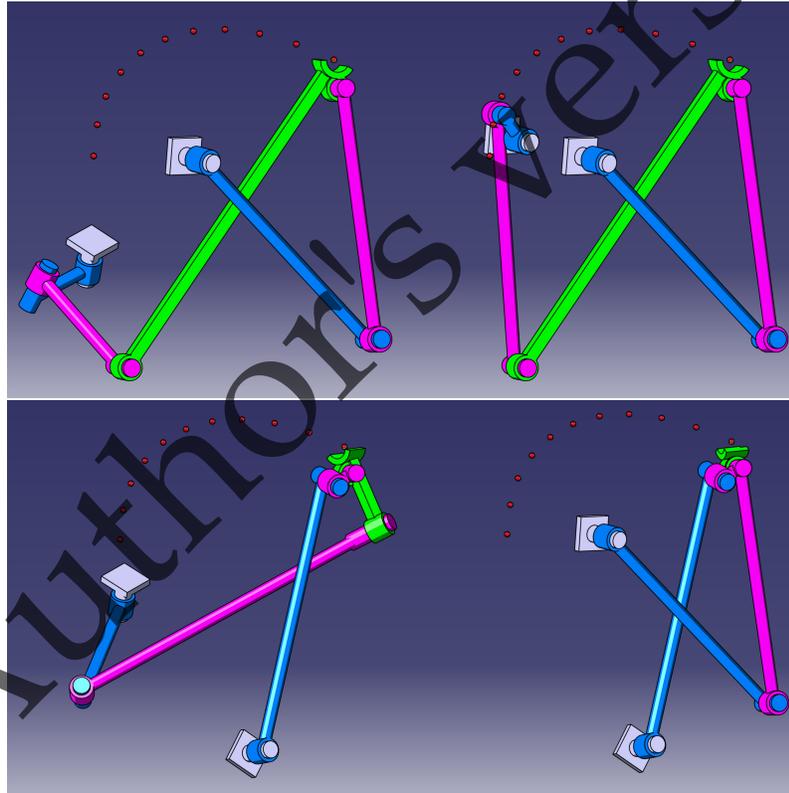


Fig. 2 Four different 6R linkages obtained

5 Conclusions

We used properties of the factorized representation of rational motions to set-up an evolution process for optimal design of corresponding linkages. The evolution gives an open kinematic chain that, if desired, can be combined with other chains obtained from different factorizations to produce overconstrained linkages. From a mechanical point of view, overconstrained linkages are robust, need minimal control elements and they are ideal for repetitive motions in an interval. In Section 4 we illustrated this process for an overconstrained 6R linkage. So far, position matching is good while matching orientations should be improved.

The construction relies on the factorized representation which helps to ensure validity of the Study condition throughout the evolution and automatically relates the rational motion to kinematic chains. Moreover, rationality allows efficient and stable computation of footpoints which is a crucial part in any evolution based mechanism synthesis.

Acknowledgements The research was supported by the Austrian Science Fund (FWF): P 26607 (Algebraic Methods in Kinematics: Motion Factorisation and Bond Theory).

References

1. Aigner M., Šír Z. and Jüttler B.: Evolution-based least-squares fitting using Pythagorean hodograph spline curves. *Computer Aided Geometric Design*, **24:6** 310–322 (2007)
2. Aigner M. and Jüttler B.: Hybrid curve fitting. *Computing* **79:2** 237–247 (2007)
3. Cabrera, J. A. Simon, A. and Prado, M.: Optimal synthesis of mechanisms with genetic algorithms. *Mechanism and Machine Theory*, **37:10** 1165–1177 (2002)
4. Hegedüs G., Schicho J., and Schröcker H.-P.: Factorization of rational curves in the Study quadric and revolute linkages. *Mechanism and Machine Theory*, **69(2)**, 142–152 (2013)
5. Hegedüs G., Schicho J., and Schröcker H.-P.: Four-Pose Synthesis of Angle-Symmetric 6R Linkages. *Journal of Mechanisms and Robotics* **7:4** 50–57 (2015)
6. Hofer M., Pottmann H.: Energy Minimizing Splines in Manifolds. *Transactions on Graphics* **23:3**, Proceedings of ACM SIGGRAPH 2004 284–293 (2004)
7. Hofer M., Pottmann H., and Ravani B.: From curve design algorithms to the design of rigid body motions. *The Visual Computer* **20:5** 279–297 (2004)
8. Hofer M., Pottmann H. and Ravani B.: Geometric design of motions constrained by a contacting surface pair. *Computer Aided Geometric Design*, **20:8** 523–547 (2003)
9. Husty M. L., Schröcker, H.-P.: Algebraic Geometry and Kinematics. In book: *Nonlinear Computational Geometry*, pp 85–107 (2009)
10. Larochelle P.M., Murray A.P. and Angeles J.: SVD and PD based projection metrics on SE(n). In: *On Advances in Robot Kinematics*, ed. by Lenarčič, J. and Galetti, C. Dordrecht, Boston and London: Kluwer Academic Publishers, pp. 13–22 (2004)
11. Paradis M. J. and Willmert K. D.: Optimal Mechanism Design Using the Gauss Constrained Method. *J. Mech., Trans., and Automation*, **105(2)**, 187–196 (1983)
12. Schröcker, H.-P. Jüttler, B. and Aigner M.: Evolving Four-Bars for Optimal Synthesis. In: *Proceedings of EUCOMES 08, The Second European Conference on Mechanism Science*, pp. 109–116 (2009)
13. Zhang, X. Zhou, J. and Ye, Y.: Optimal mechanism design using interior-point methods. *Mechanism and Machine Theory*, **31:1**, 83–98 (2000)