BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 65, No. 2, 2017
*DOI: 10.1515/bpasts-2017-0025*

# Suboptimal approximations in repeatable inverse kinematics for robot manipulators

I. DULEBA* and I. KARCZ-DULEBA

Faculty of Electronics, Wroclaw University of Science and Technology, 11/17 Janiszewski St., 50-372 Wroclaw, Poland

**Abstract.** In this paper a repeatable inverse kinematic task was solved via an approximation of a pseudo-inverse Jacobian matrix of a robot manipulator. An entry configuration to the task was optimized and a task-dependent definition of an approximation region, in a configuration space, was utilized. As a side effect, a relationship between manipulability and optimally augmented forward kinematics was established and independence of approximation task solutions on rotations in augmented components of kinematics was proved. A simulation study was performed on planar pendula manipulators. It was demonstrated that selection of an initial configuration to the repeatable inverse kinematic task heavily impacts solvability of the task and its quality. Some remarks on a formulation of the approximation task and its numerical aspects were also provided.

**Key words:** manipulator, inverse kinematics, repeatability, optimization.

## 1. Introduction

A repeatable inverse kinematic task is to plan a loop in a configuration space of a manipulator mapped, via forward kinematics, into a given loop in a taskspace. The task belongs to a broad family of robotic tasks aimed at planning and executing paths for stationary and mobile robots [7]. Repeatability offers some advantages while performing cyclic tasks (only once a collision of resulting trajectory with obstacles should be checked and gains from one-cycle trajectory optimization are multiplied). A standard technique to obtain the desirable repeatability property for redundant manipulators is to augment its kinematics [5] with extra coordinates to get nonredundant (augmented) kinematics. Because, locally and outside singular configurations, the inverse mapping of augmented kinematics is unique, a resulting trajectory forms a loop in a configuration space while tracing a loop in a task-space. Robotic researchers have tried to get even more and to couple advantages of repeatability with advantages of pseudo-inverse inverse kinematics. It is known [6] that Moore-Penrose (pseudo-) inverse, locally, minimizes velocity (displacement) in a configuration space when a velocity (displacement) in a task-space is fixed. Thus, a local motion is the most effective when planned using a pseudo-inverse of the Jacobian matrix.

This line was initialized by Roberts and Maciejewski seminal works [10]. Their approach relies on searching for functions augmenting original kinematics in quite a general form which generate a part (first columns) of the inverse of the augmented Jacobian matrix as close as possible to the pseudo-inverse matrix over a prescribed region in a configuration space. The resulting minimization task leads to equations involving partial derivatives of augmenting functions. Following this line, Tchon and coworkers [8] reformulated the repeatable kinematic task in the language of distributions and used some tools of differential geometry. Also in this case partial differential equations have to be solved. In order to avoid dealing with difficult partial differential equations and to transform the optimization task into parametric domain, augmenting kinematic functions are usually searched for as linear combinations of unknown parameters and some basis (polynomial, harmonic) functions [4].

Recently, an alternative approach to repeatable inverse kinematics was proposed which avoids a construction of augmented kinematics and designs an appropriate loop straightforward in a configuration space [1]. The approach extensively uses ideas of continuation methods [9].

In this paper we will concentrate on an optimization of the length of a loop in a configuration space for repeatable inverse kinematic task using a classical framework of augmented kinematics. The outline of the paper follows. In Section 2 a repeatable inverse kinematic task is defined and the Newton algorithm of inverse kinematics for redundant manipulators is recalled. Then, two approaches are discussed for orthonormal extensions of the Jacobian matrix at a given configuration. The first one, numerical in nature, is based on the singular value decomposition (SVD) algorithm. The second one offers analytic formulas for the extension and can be considered as an application of the Gramm-Schmidt (G-S) orthogonalization procedure applied to vectors expressed in a symbolic form. In both cases, it was observed that the extensions are not unique and all of them can be immersed into a parametric orthogonal space of appropriate size. The relationship between an augmented Jacobian matrix (resulting from the augmented kinematics) and the pseudo-inverse Jacobian matrix is recalled. In Section 2 an example is also given which illustrates why there is no ideal

*e-mail: ignacy.duleba@pwr.edu.pl

augmented kinematics that emulates the pseudo-inverse Jacobian matrix and a useful relationship is established between a determinant of the augmented Jacobian matrix and a manipulability index. In Section 3 quality functions are constructed that evaluate a distance between the pseudo-inverse Jacobian matrix and an appropriate part of the augmented Jacobian matrix at a given configuration and, then, they are extended on a given region in a configuration space. The minimization of the distance between the two Jacobian matrices defines an approximation task. The proposed quality functions differ in a complexity of optimization tasks generated and an accuracy of the approximation. In practical cases, an approximation region, in a configuration space, for repeatable inverse kinematic tasks is not known explicitly (typically assumed to be known and regular [4, 10]). In Section 4 an algorithm minimizing a trajectory length for a repeatable inverse kinematic task is provided. Simulation results of the proposed algorithm are collected in Section 5. The simulations were performed on models of pendula with the redundancy index $r$ modified in order to check various aspects of the algorithm and its parameters. Section 6 summarizes the paper.

## 2. Theory

**2.1. Repeatable inverse kinematics task.** Forward kinematics $k$ maps a configuration $q$ of a manipulator from a configuration space $Q$, into a generalized position $x$ in a task-space $X$ [11]

$$k:\ Q \ni q \to x = k(q) \in X, \tag{1}$$

where $\dim Q = n$, $\dim X = m \leq 6$. Based on kinematics (1), an $(m \times n)$ Jacobian matrix $J(q) = \partial k/\partial q$ is calculated. In the task-space a closed path (X-loop) is defined

$$\{x(s),\ s \in [0, s_{\max}],\ x(0) = x(s_{\max})\}, \tag{2}$$

where $s$ variable is usually defined as a current length of the loop starting from its initial point. A repeatable inverse kinematic task is to find a cyclic path (Q-loop) $q(\cdot)$ in the configuration space which corresponds to X-loop (2)

$$\forall_{s\in[0,s_{\max}]}\ k(q(s)) = x(s),\ \ q(0) = q(s_{\max}). \tag{3}$$

A classical repeatable inverse kinematic task assumes that an entry configuration $q_{init}$ to the loop is given $k(q_{init}) = x_0 = x(0)$ although in this paper also a free-entry configuration will be allowed. The task is trivial for non-redundant manipulators, thus only redundant manipulators will be considered with redundancy index $r$ equal to $n - m \geq 1$.

A standard way to solve the inverse kinematic task [6] is to apply the Newton algorithm

$$q_{i+1} = q_i + \xi \cdot J^{\#}(q_i)(x(s) - k(q_i)), \tag{4}$$

where $\xi$ is a small, real and positive parameter, $J^{\#} = J^T(J \cdot J^T)^{-1}$ is a pseudo-inverse of the Jacobian matrix $J$ and an initial configuration $q_0$ is fixed ($k(q_0) = x(0)$). Unfortunately, typically algorithm (4) does not generate a loop in a configuration space (3) when applied to consecutive points of X-loop (2).

In order to solve the cyclic inverse task, kinematics (1) is augmented with extra $r$ components $k_{add}(q) = (k_{m+1}(q), \ldots, k_{m+r}(q))^T$ to get non-redundant kinematics $k_{aug}(q) = (k(q)^T, k_{add}(q)^T)$ and the following version of the Newton algorithm is applied

$$q_{i+1} = q_i + \xi \cdot J^{-1}_{aug,trunc}(q_i)(x(s) - k(q_i)), \tag{5}$$

where $J^{-1}_{aug,trunc}(q_i)$ collects first $m$ columns of the inverse of the augmented Jacobian matrix $J_{aug}(q_i) = \partial k_{aug}(q)/\partial q$. The Newton algorithm (5) is run for an appropriate number of points $x(s)$ with values of $s$ increased and the initial configuration $q_0$ for the next task selected as the final configuration from the current run ($q_0 = q_{init}$ for $s = 0$). The resulting Q-loop is composed of configurations generated with the Newton algorithm (5).

**2.2. Orthonormal augmenting the Jacobian matrix.** As a pseudo-inverse of the Jacobian matrix offers desirable property of local minimization of a manipulator displacement while following X-loop, we will search for an augmentation of the matrix approximating the pseudo-inverse solution.

The first attempt will be numeric in nature. Using the SVD algorithm [3], the Jacobian matrix attains a form

$$J = U \cdot [D, 0_{m,r}] \cdot V^T, \tag{6}$$

where $0_{m,r}$ is a $(m \times r)$ block matrix composed of zeros, $U \in SO(m)$, $V \in SO(n)$ are rotational matrices in $m$ and $n$ dimensional spaces, respectively, and $D = diag(d_i)$ is a $(m \times m)$ diagonal matrix with ordered non-negative values $d_i \geq d_j$, when $i < j$. Later on we will assume that only regular configurations are considered, i.e.\ $d_m > 0$.

**Observation 1.** For a given J, SVD is not unique, as the following identity holds

$$I_{-i,m} \cdot [D, 0_{m,r}] \cdot I_{-i,n} = [D, 0_{m,r}], \tag{7}$$

where $I_{-i,k}$ denotes $k \times k$ identity matrix with the element $(i, i)$ replaced with –1. Eq. (7) when applied to Eq. (6) changes signs of all elements in the $i$-th column of matrix $U$ and the $i$-th row of matrix $V^T$. It is worth noticing that vector-rows of the matrix $D$ are perpendicular to each other.

The square root of the determinant of a manipulability matrix $M(q) = J(q) \cdot J^T(q)$ [6] is called a manipulability index

$$m(q) = \sqrt{\det(M(q))} = \prod_{i=1}^{m} d_i. \tag{8}$$

An orthonormal augmentation of the Jacobian matrix can be viewed as a purposeful extending $[D, 0_{r,m}]$ into a square matrix $(n \times n)$ while preserving the value of the manipulability index. Formally, the orthonormally augmented Jacobian $J_{aug}$ can be expressed as

$$J_{aug} = \tilde{U} \cdot \tilde{D} \cdot V^T = \begin{bmatrix} U & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} D & 0 \\ 0 & I_r \end{bmatrix} \cdot V^T = \\ = \begin{bmatrix} U & 0 \\ 0 & I_r \end{bmatrix} \begin{bmatrix} D & 0 \\ 0 & R \end{bmatrix} \cdot V^T, \tag{9}$$

where $R \in SO(r)$ is any rotational matrix, $\tilde{U} \in SO(n)$. The orthonormal augmentation means that added vectors are not only perpendicular to each other and to rows of matrix $[D, 0_{r,m}]$ but they are also a unit-length, (in the matrix $\tilde{D}$, cf. (9), matrices $I_r/R$ appear). In fact the vectors span a null space of the Jacobian matrix.

Unfortunately, SVD of the matrix $J(q)$ can be computed only numerically at a given configuration $q$. When solving a repeatable inverse kinematics, cf. Eq. (4), the SVD procedure has to be applied many times (for numerous values of $s$) as the Newton algorithm of inverse kinematics progresses. Therefore, it is desirable to have the orthonormal extension of the Jacobian matrix in a symbolic form. For this purpose, the G-S orthogonalization procedure can be modified to process vectors composed of functions rather than real-valued vectors.

**Observation 2.** Similarly to the augmented SVD, cf. (9), also the G-S orthogonalization procedure, at a given configuration $q$, does generate a unique basis of the r-dimensional ($r \geq 2$) null space of the Jacobian matrix. To generate each orthonormal basis is is enough to take any orthonormal basis and rotate it within $SO(r)$.

Note, however, that for practical robots computations using G-S are really difficult as Jacobian matrices are complicated due to many items composed of multi-term trigonometric functions (sine and cosine functions appear for rotational degrees of freedom). Moreover, in most cases transformations of coordinates $\tilde{q} = f(q)$ simplifying Jacobian matrices and, consequently, G-S procedure, are not permitted for two reasons:

1. either perpendicularity between vectors can be lost (i.e. perpendicular vectors in transformed coordinates may not display the property in the original coordinates),
2. or minimization of the energy $\dot{q}^T\dot{q}$ (a crucial assumption in deriving the pseudo-inverse Jacobian matrix) can give a different solution than minimization of $\dot{\tilde{q}}^T\dot{\tilde{q}}$.

To sum up: each of the augmenting vectors that span a basis in the null space of the Jacobian matrix should display the following properties

**Property 1:** to have a unit length,
**Property 2:** to be perpendicular to others,
**Property 3:** to be perpendicular to any row of the Jacobian matrix.

An important natural question arises: are there any functions augmenting the original kinematics with derivative w.r.t. the configuration corresponding to the added rows of the augmented Jacobian matrix and satisfying Properties 1–3?

If so, the problem of an approximation of the Jacobian matrix with appropriately defined kinematic functions would be trivial and the approximation would be exact. However, the following simple example prompts that this is not the case and ideal augmenting functions are over-constrained.

**Example 1:** for 2D planar pendulum $q = (q_1, q_2)$ with one dimensional taskspace ($k_1$ coordinate in (40)) its Jacobian matrix equals

$$J(q) = [-a_1 s_1 - a_2 s_{12}, -a_2 s_{12}] = \partial k_1 / \partial q. \tag{10}$$

The only augmenting $k_2(q)$ function should satisfy

$$\frac{\partial k_2}{\partial q_2}\Big] = [-a_2 s_{12}, a_1 s_1 + a_2 s_{12}], \quad \|\frac{\partial k_2}{\partial q}\| = 1. \tag{11}$$

From the first condition in (11)

$$k_2 = -a_2 \int s_{12} dq_1 = a_2 c_{12} + C_1(q_2) \tag{12}$$

while the second condition (11) takes the form

$$k_2 = \int (a_1 s_1 + a_2 s_{12}) dq_2 = a_1 s_1 \cdot q_2 - a_2 c_{12} + C_2(q_1), \tag{13}$$

where $C_p$ are some functions. From (12, 13) it can be deduced that there is no admissible augmenting function $k_2(q)$.

This example confirms an obvious fact that, in a typical case, more constraints imposed on a given task (loopiness in the configuration space in our case) may give worse (sometimes even no) solutions than for an un-constrained or a less constrained task.

**2.3. A relationship between an inverse of orthonormally augmented $J$ and pseudo-inverse $J^{\#}$.** The relationship will be established based on SVD. Using (6), the pseudo-inverse matrix $J^{\#}$ can be expressed as follows

$$J^{\#} = V \cdot \begin{bmatrix} D^{-1} \\ 0_{r,m} \end{bmatrix} \cdot U^T, \tag{14}$$

where $m \times m$ matrix $D^{-1} = diag(1/d_i)$. Taking inverse of $J_{aug}$, cf. Eq. (9), exploiting properties of matrices from $SO(k)$, one gets

$$J_{aug}^{-1} = V \cdot \tilde{D}^{-1} \cdot \tilde{U}^T = V \cdot \begin{bmatrix} D^{-1} & 0 \\ 0 & I_r \end{bmatrix} \cdot \begin{bmatrix} U^T & 0 \\ 0 & R^T \end{bmatrix} \\ = V \cdot \begin{bmatrix} D^{-1} & 0 \\ 0 & R^T \end{bmatrix} \cdot \begin{bmatrix} U^T & 0 \\ 0 & I_r \end{bmatrix}. \tag{15}$$

Comparing (14) with (15) one can deduce that $J^{\#}$ is formed with first $m$ columns of the inverse of the orthonormally augmented Jacobian matrix.

## 3. Optimal augmenting the Jacobian matrix

In this section an optimal approximation of the pseudo-inverse Jacobian matrix with a part of an inverse of the Jacobian matrix based on augmented kinematics will be formulated over a given region in a configuration space. Then, the task will be reformulated to simplify the optimization task and to split it into

$r$ independent, low dimensional tasks. For special augmented kinematics, described by linear or special quadratic functions, optimal solutions will be given explicitly. Some remarks will be formulated on uniqueness of the solution of the approximation task in a parametric space and on permitted transformations of coordinates.

When desirable Properties 1–3 of vectors augmenting Jacobian matrix are known, an optimization (approximation) task can be sketched: let us assume augmenting kinematic functions $k_{add}(p, q) = (k_{m+i}(p, q))$, $i = 1, \ldots, r$ in a parametric form, compute their differentials $\partial k_{m+i}(p, q)/\partial q$ and minimize

$$f(p,q) = \|J^{\#}(q) - J^{-1}_{aug,trunc}(p,q)\| \tag{16}$$

(here and later on $\|\cdot\|$ denotes the Euclidean norm) w.r.t. parameters $p$ over a given region in the configuration space $A \subset Q$. However, the minimization of (16) seems to be too computationally involved.

The other, and equivalent, approach is to formulate analytically the required Properties 1–3 to construct and to optimize a criterion function that punishes solutions violating the properties. At first, let us formally formulate required properties at one particular configuration $q$. The orthogonality within the null space of the Jacobian matrix sets the constraint

$$\left\langle \partial k_{m+i}/\partial q, \partial k_{m+j}/\partial q \right\rangle = \delta_{ij}, \;\; i,j = 1,\ldots,r, \;\; i \le j \tag{17}$$

where $\delta_{ij} = 1$ for $i = j$ and 0 otherwise. Perpendicularity to rows of the Jacobian matrix adds more constraints

$$\langle \partial k_{m+i}/\partial q, J_s \rangle = 0, \quad i = 1,\ldots,r, \quad s = 1,\ldots,m. \tag{18}$$

As optimal values of extra rows augmenting $J$ to $J_{aug}$ are known, (either from SVD or the G-S procedure) a simpler version of condition (17) is proposed

$$\left\langle \partial k_{m+i}/\partial q, J_{aug,m+j} \right\rangle = \delta_{ij}, \quad i,j = 1,\ldots,r, \quad i \le j, \tag{19}$$

to avoid using twice unknown $\partial k_{m+i}/\partial q$. In fact the condition (18) is redundant one (and can be omitted) because in an ideal case $\partial k_{m+i}/\partial q$ are the same as rows $J_{aug,m+i}$, $i = 1, \ldots, r$, thus automatically are perpendicular to rows of $J$ (because $J_{aug,m+i} \perp J$). Finally, the only active condition is given by Eq. (19) or expressed in a simpler form as

$$\|\frac{\partial k_{m+i}}{\partial q} - J_{aug,m+i}(q)\|^2 = 0, \quad i = 1,\ldots,r. \tag{20}$$

Note that by definition $\|J_{aug,m+i}(q)\| = 1$, so $\|\partial k_{m+i}/\partial q\| = 1$ to meet condition (20). As condition (20) can not be met exactly, a criterion function

$$f(p,q) = \sum_{i=1}^{r} \|\partial k_{m+i}(p,q)/\partial q - J_{aug,m+i}(q)\|^2, \tag{21}$$

is to be optimized over some region $A \subset Q$

$$F(p^\star) = \min_p \int_{A \subset Q} f(p,q)dq. \tag{22}$$

In (21) the Euclidean norm is squared to avoid the square root in the optimized function.

The minimization task (21), (22) is primarily defined over a continuous region $A \subset Q$. In a practical implementation the region $A$ can be composed of finitely many configurations in $Q$. In this case integral (22) is replaced with a sum. Moreover, because there is no need to have an augmented Jacobian matrix in a symbolic form, then the SVD procedure can be used to derive necessary rows of $J_{aug}(q)$ at each configuration $q$.

One more useful property of the optimization task (21), (22) can be noticed. In fact it can be decomposed into $r$ independent minimization tasks where $k_{m+i}(p, q)$ is searched for that minimizes the $i$-th quality function

$$f_i(p_i,q) = \|\frac{\partial k_{m+i}(p_i,q)}{\partial q} - J_{aug,m+i}(q)\|^2, \quad i = 1,\ldots,r, \tag{23}$$

over a given region $A$

$$F_i(p_i^\star) = \min_{p_i} \int_{A \subset Q} f_i(p_i,q)dq. \tag{24}$$

However, one must be aware that the simplification, due to decomposition into $r$ independent tasks, has also one disadvantage as explained in the following example.

**Example 2.** Assume that, at some configuration $q$, the optimal null space in $R^3$ is spanned by $v_1 = (1,0,0)$ and $v_2 = (0,1,0)$.

Apparently, the pair $\tilde{v}_1 = (\cos(\phi), \sin(\phi), 0)$, $\tilde{v}_2 = (-\sin(\phi), \cos(\phi), 0)$ (for small values of $\phi$) approximating $v_1$, $v_2$, seems to be better than the pair $\hat{v}_1 = \tilde{v}_1$, $\hat{v}_2 = (\sin(\phi), \cos(\phi), 0)$. Versors of the first pair are perpendicular to each other contrary to the other pair although for both pairs their pairwise distance to versors $v_1$, $v_2$ is the same.

The optimal approximation of the pseudo-inverse Jacobian will begin with linear augmenting functions (a constant term can be omitted as it is annihilated while taking derivatives)

$$k_{m+i}(p,q) = \sum_{s=1}^{n} p_{i,s} q_s \;\Rightarrow\; \frac{\partial k_{m+i}}{\partial q} = (p_{i,1},\ldots,p_{i,n}). \tag{25}$$

When an approximation region $A$ includes only one configuration $q^w$, the solution of optimization task (23), (24) is given explicitly

$$p_{i,s} = J_{aug,m+i,s}(q^w), \quad s = 1,\ldots,n. \tag{26}$$

When the region $A = \{q^1, \ldots, q^N\}$ is composed of $N$ configurations, the final solution is also given explicitly

$$p_{i,s} = \frac{1}{N} \sum_{w=1}^{N} J_{aug,m+i,s}(q^w), \quad s = 1,\ldots,n. \tag{27}$$

A slightly more complicated case appears for quadratic optimized functions, $i = 1, \ldots, r$

$$k_{m+i}(p_i, q) = \sum_{s=1}^{n} p_{i,s} q_s + \frac{1}{2} \sum_{s=1}^{n} \sum_{t=1}^{n} p_{i,st} q_s q_t, \qquad p_{i,st} = p_{i,ts} \tag{28}$$

with their derivatives

$$\frac{\partial k_{m+i}(p_i, q)}{\partial q_j} = p_{i,j} + \sum_{s=1}^{n} p_{i,sj} q_s, \quad j = 1, \ldots, n. \tag{29}$$

The vector of unknown variables $p_i$ collects variables from the linear and quadratic part $p_i = (p_{i,s}, p_{i,st})$. This time, the optimization task (23), (24) is reduced to solving a set of linear equations w.r.t. vector $p_i$ variables resulting from equilibrium conditions, cf. (23)

$$\frac{\partial f_i(p_i, q)}{\partial p_i} = 0, \quad i = 1, \ldots, r. \tag{30}$$

The set may involve many items. However, for particular functions (28) with cross variables neglected $p_{i,st} = 0$ for $s \neq t$, the solution can be obtained explicitly as computations for each functions $k_{m+i}(p, q)$, $i = 1, \ldots, r$ can be performed coordinate-wise, $j = 1, \ldots, n$ and the resulting linear equations are two dimensional only (below the optimization over the region $A$ composed of $N$ configurations $q^w$, $w = 1, \ldots, N$ was used)

$$\begin{bmatrix} N & \sum_{w=1}^{N} q_j^w \\ \sum_{w=1}^{N} q_j^w & \sum_{w=1}^{N} (q_j^w)^2 \end{bmatrix} \begin{bmatrix} p_{i,j} \\ p_{i,jj} \end{bmatrix} = \begin{bmatrix} \sum_{w=1}^{N} J_{aug,m+i,j}(q^w) \\ \sum_{w=1}^{N} J_{aug,m+i,j}(q^w) q_j^w \end{bmatrix}, \tag{31}$$

where $q_j^w$ denotes the $j$th component of vector $q^w$ and $J_{aug,m+i,j}(q^w)$ is the $j$th component of the $(m+i)$th row of the augmented Jacobian matrix evaluated at configuration $q^w$.

Now some facts concerning the optimal solution will be highlighted.

**Theorem 1.** Let us assume that an approximation task has been solved, i.e. functions $k_{add}(q) = (k_{m+i}(q))$, $i = 1, \ldots, r$ were determined. Then also functions $\tilde{k}_{add}(q) = (\tilde{k}_{m+i}(q))$, $i = 1, \ldots, r$

$$\tilde{k}_{add}(q) = R \cdot k_{add}(q) \tag{32}$$

for any constant matrix $R \in SO(r)$, are also the solutions of the approximation task.

**Proof.** A simple exercise

$$\left\| \frac{\partial k_{add}(q)}{\partial q} - J_{aug,add} \right\| = \left\| R \frac{\partial k_{add}(q)}{\partial q} - R J_{aug,add} \right\| = \left\| \frac{\tilde{k}_{add}(q)}{\partial q} - R J_{aug,add} \right\| = \left\| \frac{\tilde{k}_{add}(q)}{\partial q} - \tilde{J}_{aug,add} \right\|, \tag{33}$$

where $J_{aug,add}$ collects $r$ added rows of the Jacobian augmented matrix, and $\tilde{J}_{aug,add}$ is another orthonormal basis within the null space of $J$. □

Theorem 1 reflects the fact that infinite many orthonormal bases can be defined in, at least, a two-dimensional null space and the bases are equivalent to each other. The theorem may also prompt to search for augmenting kinematic functions invariant to rotations.

Let us also observe that a constant rotation matrix $R$ is an important assumption of Theorem 1. Otherwise, the first term in the following equation

$$\frac{\partial \tilde{k}_{add}(q)}{\partial q} = \frac{\partial R(q)}{\partial q} \cdot k_{add}(q) + R \cdot \frac{\partial k_{add}(q)}{\partial q} \tag{34}$$

will spoil the thesis of Theorem 1 as the added part of the Jacobian matrix $\partial \tilde{k}_{add}(q)/\partial q$ explicitly depends on forward kinematics of augmenting coordinates $k_{add}(q)$.

Forward kinematics is usually defined in natural coordinates tightly related with a construction of a manipulator. A redefinition of coordinates may simplify the form or complexity of the Jacobian matrix. Therefore, while searching for augmenting functions it is worth considering to redefine (transform) natural coordinates. However, this approach has its own disadvantages as illustrated with the following example.

**Example 3.** Forward kinematics of the 3D planar pendulum, cf. Fig. 1, in natural coordinates is given by (40). When consecutive coordinates $(\tilde{q}_1, \tilde{q}_2, \tilde{q}_3) = \tilde{q}$ are not defined with respect to the previous link but rather with respect to the $x$-axis of the global coordinate frame, the Jacobian matrix (for positional coordinates $(x, y)$) takes a form

$$\tilde{J} = \begin{bmatrix} -a_1 \tilde{s}_1 & -a_2 \tilde{s}_2 & -a_3 \tilde{s}_3 \\ a_1 \tilde{c}_1 & a_2 \tilde{c}_2 & a_3 \tilde{c}_3 \end{bmatrix}, \text{ where } \begin{matrix} \tilde{s}_i = \sin(\tilde{q}_i), \\ \tilde{c}_i = \cos(\tilde{q}_i). \end{matrix} \tag{35}$$

It is easy to check, via straightforward calculations, that the Jacobian matrix $\tilde{J}$ is much simpler than Jacobian $J$ of kinematics (40) with a configuration vector $\boldsymbol{q} = (q_1, q_2, q_3)$. Unfortunately,

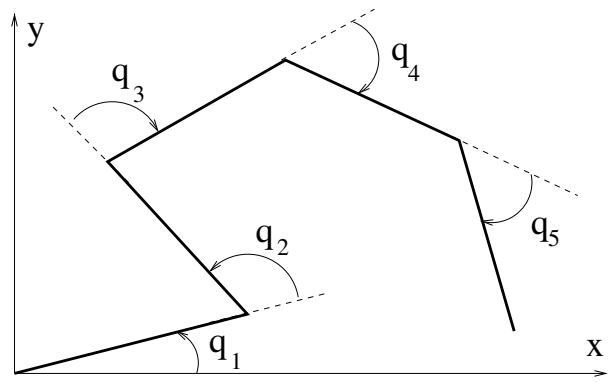


Fig. 1. Planar pendula up to 5 dofs

while designing an algorithm of the optimal approximation of pseudo-inverse $J^{\#}$ based on the Jacobian matrix $J$, cf. (4), one must be aware that the algorithm optimizes locally an energy in natural coordinates $(\sum_{i=1}^{n}\dot{q}_i^2)$ and in the considered example

$$\tilde{q} = (\tilde{q}_1, \tilde{q}_2, \tilde{q}_3) = (q_1, q_1 + q_2, q_1 + q_2 + q_3), \tag{36}$$

apparently increases the energy $\sum_{i=1}^{n}\dot{\tilde{q}}^2 > \sum_{i=1}^{n}\dot{q}_i^2$. In the new coordinates varied $q_1$ coordinate changes all components of $\tilde{q}$.

## 4. A sub-optimal solution of a repeatable inverse kinematic task

A clear disadvantage of optimization tasks presented previously is that the region $A$, over which the quality function is evaluated, is usually given explicitly (frequently as an $n$-dimensional cuboid in the configuration space [4, 10]). In practical situations, the region may not contain many (the most) of configurations generated when tracing a given loop in the task-space and it is not given explicitly but rather implicitly and ambiguously via inverse kinematics. Therefore, we propose an original algorithm to construct the region $A$ iteratively to make the optimization dedicated for a given, traced loop in the task-space.

Formally: given kinematics $k(q)$, a loop in the task-space (2) and initial configuration $q_{init}$ such that $k(q_{init}) = x(0)$, find the shortest possible loop in the configuration space $q(\cdot)$ corresponding to the loop $x(\cdot)$, cf. (2). The following algorithm solves the task iteratively using iterative approximations of the pseudo-inverse of the Jacobian matrix:

**Step 1.** Initialize the algorithm: an iteration counter $c \leftarrow 1$, configurations forming the set $A_1 = \{q_{init}\}$, and the initial loop length in the configuration space $L_0 \leftarrow \infty$.

**Step 2.** Express each augmented kinematic function $k_{m+i}(p_i, q)$, $i = 1, \ldots, r$ in a parametric form (i.e. select a class of admissible functions and the number of elements for each $k_{m+i}(p_i, q)$.

**Step 3.** Compute $J_{aug}(q)$ for configurations from the set $A_c$, cf. (6), (9).

**Step 4.** Solve $r$ optimization tasks, $i = 1, \ldots, r$ in the current iteration $c$

$$F_i^c(p_i) = \int_{A_c} f_i(p_i, q)dq, \quad F_i^c(p_i^\star) = \min_{p_i} F_i^c(p_i), \tag{37}$$

with $f_i(p_i, q)$ defined by Eq. (23). For special optimized functions derived either from Eq. (27) or Eq. (31).

**Step 5.** For the optimal $p^\star = (p_1^\star, \ldots, p_r^\star)$, which determines augmented kinematics, compute the resulting Q-loop $q(\cdot)$ (using algorithm (5)) and its length

$$L_c(p^\star) = \int_{s=0}^{s_{\max}} \sqrt{\langle \partial q(s)/\partial s, \partial q(s)/\partial s \rangle} ds. \tag{38}$$

**Step 6.** Check the stop condition: if

$$L_{c-1} - L_c \leq \delta \tag{39}$$

where $\delta$ is a given, positive-valued threshold, then stop the algorithm and output the loop $q(\cdot)$ from the previous iteration. Otherwise progress with Step 7.

**Step 7.** Discretize uniformly the loop $q(\cdot)$ with $N$ points to form the set $A_{c+1}$ for the next iteration $A_{c+1} = \{q(s_j)\}$ where $s_j = j \cdot s_{\max}/N, j = 0, \ldots, N-1$. Increase the iteration counter $c \leftarrow c + 1$ and go to Step 3.

To increase the optimization potential of the algorithm, an outer loop can be added to the algorithm aimed at varying the initial value of $q_{init}$. The selection of $q_{init}$ can be implemented with any method described in [2].

Some remarks concerning the algorithm follow:

- As the region $A_c$ is composed of finite number of points, thus integrals in (37, 38), are replaced with sums and velocity $\partial q/\partial s$ at each particular configuration is interpolated based on its neighboring configurations.
- An initial region $A_c = A_1$ was selected as a single, given, configuration $q_{init}$ because no other information about Q-loop is known when the algorithm begins. As the algorithm progresses, new approximations give rise to consecutive (known) Q-loops and discretization of the current Q-loop generated new regions $A_c$. It is also interesting to start $A_1$ with any other configuration.
- Linear augmented functions (25) are the simplest possible and very useful as a solution of the approximation task is given explicitly. However, they may not be so efficient when the region $A$ is larger. In this case, it may rarely happen that the vector of coefficients $p_i$, cf. (27), is very short to cause some numerical problems and to warn that linear augmenting functions are not rich enough. Moreover, as the quality function prefers augmented function $k_{m+i}(q)$ with $\|\partial k_{m+i}(q)/\partial q\| \simeq 1$ some other bases (like harmonic ones) can be used to search for desired augmented functions.
- In a general case, any standard optimization method can be applied to derive a vector of parameters $p_i^\star$ in Step 4. However, by selecting appropriate quality functions (linear or some special quadratic) the solution can be given explicitly to speed-up computations significantly as the optimization is performed many times.
- It is expected that only a few (two or three iterations, counted by variable $c$) are required to complete the algorithm as a variation of resulting trajectory (due to varying parameters $p$ from one iteration to another) is smaller and smaller over iterations.

## 5. Simulations

In order to test the proposed algorithm, planar pendulum manipulators, Fig. 1, up-to five degrees of freedom were selected. The manipulators are easy to visualize and a high degree of re-

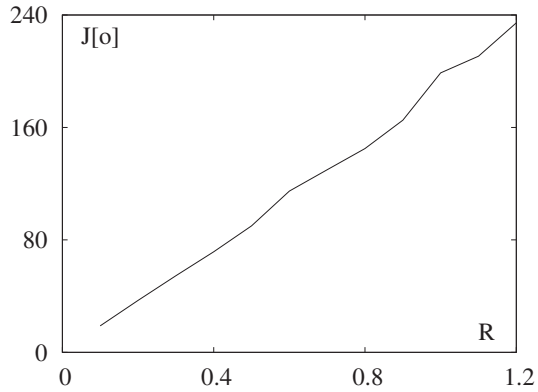


Fig. 2. Length of a Q-loop as a function of a circle radius $R$

dundancy can be obtained easily. Forward positional kinematics of the pendula are given by

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} k_1(q) \\ k_2(q) \end{bmatrix} = \\ = \begin{bmatrix} a_1 c_{1,1} + a_2 c_{1,2} + a_3 c_{1,3} + a_4 c_{1,4} + a_5 c_{1,5} \\ a_1 s_{1,1} + a_2 s_{1,2} + a_3 s_{1,3} + a_4 s_{1,4} + a_5 s_{1,5} \end{bmatrix}, \tag{40}$$

where $\boldsymbol{a} = (a_i, i = 1, \ldots, n)$ is a vector of links' lengths set to 1 in all simulations. A standard robotic convention was used to abbreviate trigonometric functions, $c_{1,i} = \cos(\sum_{j=1}^{i} q_j)$, and $s_{1,i} = \sin(\sum_{j=1}^{i} q_j)$. A circle centered at $(x_c, y_c)$ with the radius $R$ was a path to follow

$$\begin{aligned} x(s) &= x_c + R\cos(2\pi s), \\ y(s) &= y_c + R\sin(2\pi s), \quad s \in [0, 1]. \end{aligned} \tag{41}$$

If not stated otherwise, the linear approximation (25) was used.

In the first simulation, for the 4D-pendulum, fixed $(x_c, y_c) = (2.5, 0.5)$, the length of resulting loop in the configuration space was computed as a function of radius $R$. The initial configuration $q_{init}$ was also varied [2] and the plot depicted in Fig. 2 presents the shortest Q-loop for each $R$. It appears that the characteristics is almost linear. So one can expect that the length of the Q-loop scales linearly with the length of the given X-loop if only an initial configuration is free to choose.

In the second simulation (the same manipulator, and the center of the circle) and fixed radius $R = 1.25$, it was checked how the length of $Q$-loop depends on $q_{init}$. The set of possible $q_{init}$ was discretized (214 items generated). The shortest loop started at $\boldsymbol{q}_0 = (28.3°, -6.7°, -21.6°, -20°)$ (length 242.3°), while the longest one at $\boldsymbol{q}_0 = (29.7°, -49.4°, 39.7°, -20°)$ (length 411.8°). Both solutions were presented in Fig. 3. As expected, cf. Eq. (19), augmenting functions that correspond to the shortest $Q$-loop

$$\begin{aligned} k_3(q) &= 0.49q_1 - 0.76q_2 + 0.16q_3 + 0.39q_4, \\ k_4(q) &= 0.046q_1 + 0.26q_2 - 0.65q_3 + 0.71q_4, \end{aligned} \tag{42}$$

have their gradients almost unit-length $\|\partial k_2/\partial q\| = 0.995$, $\|\partial k_3/\partial q\| = 0.996$. It appears that selection of an initial entry configuration to the Q-loop is important and can significantly impact the optimal length of the loop.

In the third simulation, it was checked whether this observation is valid also for pendula with a different number of degrees of freedom. For all pendula with $n = 3, 4, 5$, $(x_c, y_c) = (0.5, 1.5)$, $R = 0.9$, while $q_{init}$ was varied. Results collected in Table 1 confirm that the observation made previously does not depend on

Table 1
The shortest and the longest $Q$-loops: varied $q_{init}$ and $n$; while $R$, $(x_c, y_c)$ fixed

| $n$ | $q_{init}$[°] | len | $q_{init}$[°] | len |
|---|---|---|---|---|
| | shortest | | longest | |
| 3 | (51, −34, −52) | 238 | (−33.8, 86.7, −37.9) | 338 |
| 4 | (72, −23, −69, −40) | 170 | (94.9, −151.8, 77, −20) | 290 |
| 5 | (−106, 92, 14, 45, 45) | 139 | (45, −112, −159, −135, 0) | 309 |

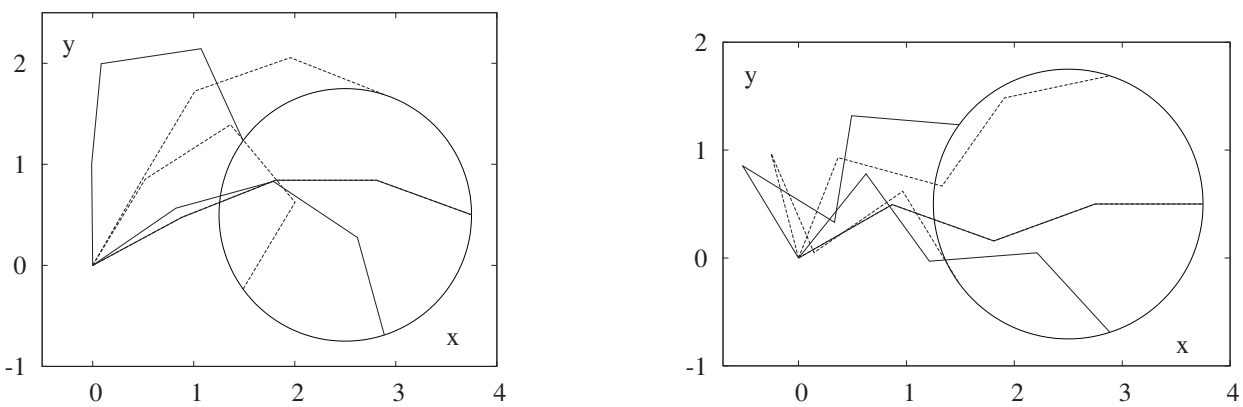


Fig. 3. Stroboscopic views of postures of 4D pendulum corresponding to the shortest and the longest path for $q_{init}$ varied, $R$ and $(x_c, y_c)$ fixed

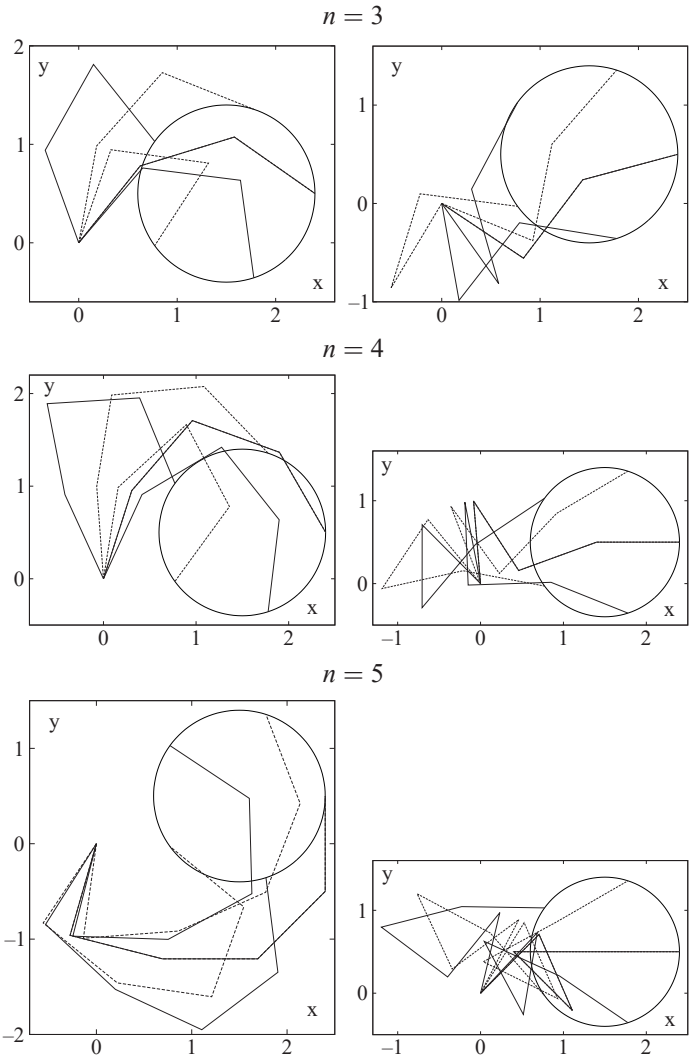


Fig. 4. Stroboscopic views of loops corresponding to the shortest (left) and the longest (right) $Q$-loops, $q_{init}$ varied

the number of degrees of freedom. The stroboscopic views of the loops corresponding to the shortest and the longest $Q$ loop are presented in Fig. 4.

In the next experiment (for the same initial data as previously) the algorithm using the pseudo-inverse Jacobian was run. Results were collected in Table 2. Obviously, for the pseudo-inverse solution the loop was broken ($q(0) \neq q(1)$). It is interesting that quality of the optimal solutions with repeatable and non-repeatable algorithm are almost the same (cf. Table 1).

Table 2
The shortest $Q$-loops obtained using pseudo-inverse Jacobian; varied $q_{init}$ and $n$; while $R$, $(x_c, y_c)$ fixed

| $n$ | $q(0)[°] \to q(1)[°]$ | len |
|---|---|---|
| 3 | $(51.3, -34.3, -52) \to (56, -45.2, -41.7)$ | 238.1 |
| 4 | $(-60.4, 45.4, 55, 40) \to (-51.4, 33.7, 53.5, 53.5)$ | 170.4 |
| 5 | $(24, -63, -51, 180, -45) \to (58, -11.7, -117, -19)$ | 130.2 |

The goal of the last simulation is to compare linear, cf. (25), with a special case of the quadratic approximation cf. (28) (with cross variables neglected). A path to follow was a circle centered at (1.5, 0.5). Simulations were carried out on pendula with $n = 3, 4, 5$ degrees of freedom and two selected radii of the prescribed path. Potential initial configurations to the Q-loop were generated. Their total number is collected in the fifth column of Table 3. Some of them were unaccept-

Table 3
A comparison of $Q$-loops obtained for the same $q_{init}$ configurations using linear and quadratic approximation functions, $n$ and $R$ varied

| $n$ | $R$ | $\Delta_{\min}[°]$ | $\Delta_{\max}[°]$ | all $q_{init}$ | $q_{init}$ accept | passed |
|---|---|---|---|---|---|---|
| 3 | 0.9 | −3.0 | 10.8 | 288 | 84 | 38 |
| 3 | 0.5 | −0.07 | 3.2 | 288 | 116 | 116 |
| 4 | 1.2 | −7.3 | 11 | 648 | 116 | 41 |
| 4 | 0.6 | −0.77 | 18.65 | 648 | 216 | 151 |
| 5 | 1.2 | −49 | 34 | 2000 | 400 | 93 |
| 5 | 0.7 | −3.4 | 27 | 2000 | 646 | 325 |

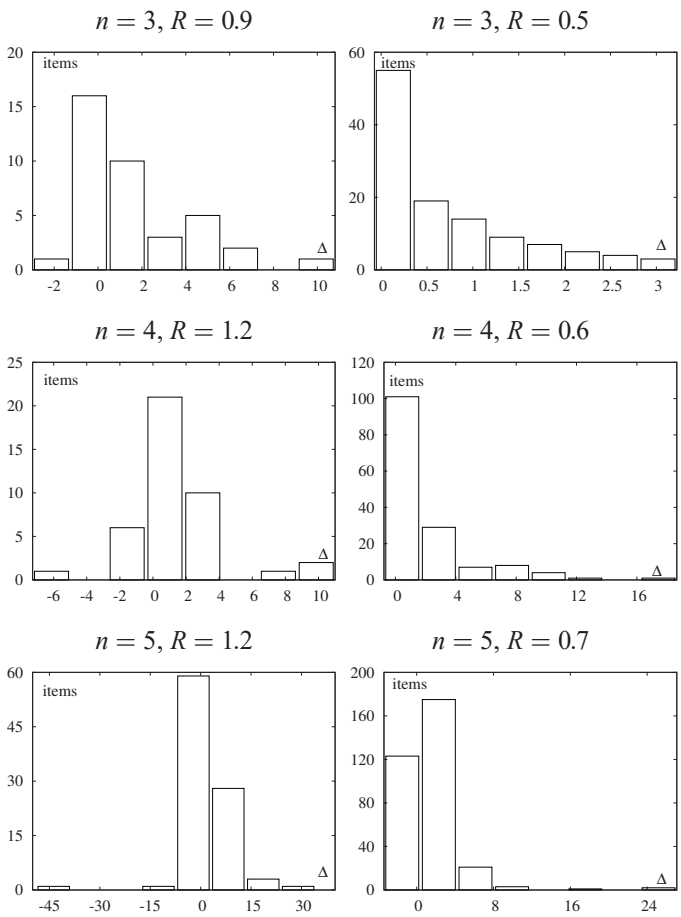


Fig. 5. Histograms of Q-loop length differences obtained from quadratic and linear approximations for the same initial configurations, $n$ and $R$ varied

able due to geometrical constraints, those acceptable were counted in the sixth column of Table 3. Then, an approximation task was run for two approximation functions (linear and quadratic one). Some solutions were unacceptable as an initial configuration acceptability does not necessary imply acceptability of the Q-loop as well (a conflict may appear somewhere along the loop, not necessarily at $s = 0$). These $q_{init}$ which successfully generated Q-loops for both functions were counted in the seventh column of Table 3. In this way with each generated $q_{init}$, two lengths of acceptable Q-loops were associated: for linear $len_1$ and quadratic $len_2$ approximation functions. Based on the values, a vector of lengths' difference was constructed $\Delta = len_2 - len_1$. The minimal and maximal component of this vector is presented in the third and the fourth column of Table 3, respectively. Histograms based on vectors $\Delta$ obtained for different values of $n$ and $R$ were presented in Fig. 3.

In most cases, it appears that the linear approximation is better than quadratic one (positive values of $\Delta$ dominate). This result seems to be unexpected but averaging over large approximation regions in a configuration space with coordinates of $q$ varied substantially may lead to inaccuracies of high degree approximations. High-degree polynomial approximating functions tend to attain large values while the optimal functions should have lengths of their gradients, cf. (17), as close to one as possible at any configuration along designed Q-loop. Exemplary optimal approximation functions for $n = 5$, $R = 1.2$ and $q_{init}$ = (133°, 172°, 9°, –36°, 0°) followed: augmented functions for the linear approximation

$$\begin{aligned} k_3(q) &= 0.74q_1 + 0.08q_2 - 0.6q_3 - 0.2q_4 - 0.19q_5, \\ k_4(q) &= -0.11q_1 + 0.43q_2 + 0.27q_3 - 0.81q_4 - 0.28q_5, \\ k_5(q) &= -0.07q_1 + 0.04q_2 + 0.08q_3 - 0.37q_4 - 0.92q_5, \end{aligned} \tag{43}$$

have lengths of their gradients very close to 1: $\|\partial k_3/\partial q\| = 0.99$, $\|\partial k_4/\partial q\| = 1$, $\|\partial k_5/\partial q\| = 1$ (the length of resulting Q-loop equals 340°). Quadratic approximating functions

$$\begin{aligned} k_3(q) &= 0.85q_1 + 0.33q_2 - 0.61q_3 - 0.36q_4 - 0.07q_5 + \\ &\quad -0.02q_1^2 - 0.13q_2^2 + 0.08q_3^2 - 0.34q_4^2 + 0.38q_5^2, \\ k_4(q) &= 0.47q_1 + 0.94q_2 + 0.54q_3 - 0.79q_4 - 0.33q_5 + \\ &\quad -0.17q_1^2 - 0.22q_2^2 - 0.14q_3^2 - 0.04q_4^2 - 0.72q_5^2, \\ k_5(q) &= 1.87q_1 + 1.1q_2 + 0.17q_3 + 0.75q_4 - 0.84q_5 + \\ &\quad -0.75q_1^2 - 0.42q_2^2 - 0.1q_3^2 + 0.49q_4^2 - 0.24q_5^2, \end{aligned} \tag{44}$$

generated Q-loop of the length 374°. Notice that in this case $\|\partial k_{m+i}/\partial q\|$, $i = 1, \ldots, r$, becomes configuration dependent and its linear part is usually longer than 1. In (44): $\|\partial k_3^{lin}/\partial q\| = 1.16$, $\|\partial k_4^{lin}/\partial q\| = 1.46$, $\|\partial k_5^{lin}/\partial q\| = 2.45$.

## 6. Conclusions

In this paper a repeatable inverse kinematic task was solved via approximating a pseudo-inverse Jacobian matrix with linear and quadratic polynomials, allowing to select freely an initial entry configuration and defining iteratively a task-dependent region in the configuration space to evaluate the quality function. Simulations confirmed a good quality of designed techniques and proved that a linear approximation function is good enough to solve the task accurately and fast. An invariance of the approximation function on constant rotations was proved and a relationship of optimal extensions of the Jacobian matrix with a manipulability index was revealed.

**Acknowledgements.** This work was supported by Wroclaw University of Science and Technology within Statutory Grant No. 0401/0209/16.

## References

[1] I. Duleba and M. Opałka, "On application of elastic band method to repeatable inverse kinematics in robot manipulators", *Journal of Automation, Mobile Robotics and Intelligent Systems*, 7(4), 5–12, (2013).

[2] I. Duleba and I. Karcz-Duleba: "A suboptimal solution of repeatable inverse kinematics in robot manipulators with a free entry configuration", *Mediterranean Conf. on Control and Automation*, Athens, 563–568, (2016).

[3] G.H. Golub and C. Reinsch, "Singular value decomposition and least squares solutions", *Numerische Mathematik* 14 (5), 403–420, (1970).

[4] J. Karpińska, "Approximation of algorithms for robot motion planning", Ph. D. dissertation, Wroclaw University of Technology, 2012, [in Polish].

[5] C.A. Klein, C. Chu-Jeng and S. Ahmed, "A new formulation of the extended Jacobian method and its use in mapping algorithmic singularities for kinematically redundant manipulators", *IEEE Trans. Robot. Autom.*, 11, 50–55, (1995).

[6] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*, Addison Wesley, New York, 1991.

[7] A. Ratajczak, "Trajectory reproduction and trajectory tracking problem for the nonholonomic systems" *Bull. Pol. Ac.: Tech.*, 64(1), 63–70, (2016).

[8] J.E. Ratajczak, K. Tchon and M. Janiak, "Approximation of Jacobian inverse kinematics algorithms: differential geometric vs. variational approach", *Journal of Intelligent and Robotic Systems*, 68 (3/4), 211–224, (2012).

[9] S. Richter and R. DeCarlo, "Continuation methods: theory and applications", *IEEE Tran. on Automatic Control*, 28(6), 660–665, (1983).

[10] R. Roberts and A.A. Maciejewski, "Repeatable generalized inverse control strategies for kinematically redundant manipulators", *IEEE Trans. on Automatic Control*, 38, 689–699, (1993).

[11] M. Spong and M. Vidyasagar, *Introduction to Robotics. Robot Dynamics and Control*, MIT Press, Cambridge, 1989.