Int. J. Appl. Math. Comput. Sci., 2000, Vol.10, No.3, 605–622

KINEMATIC APPROXIMATION OF ROBOTIC MANIPULATORS

IGNACY DULĘBA*

A kinematic approximation of the nominal, reference kinematics of a manipulator is addressed. The approximation task leads to a minimax optimization problem. Some modifications facilitating the implementation of the algorithm in search spaces of high dimensionalities are presented. A sub-gradient mesh algorithm of solving the minimax task is given in detail. The approximation of the CYBOTECH manipulator with a ULB robot is provided based on the use of the Chebyshev metric. As an alternative, some measures of proximity of manipulators are also introduced.

Keywords: manipulator, kinematics, approximation, metrics, ULB robot

1. Introduction

An engineer constructing a manipulator often faces the problem of an optimal design. To fulfil many (usually opposite) requirements is a difficult task. Practical experience and/or theoretical versatility may help to sketch an optimal construction of a manipulator. An approximation problem arises when the design of a real robot is required as close as possible to the optimal manipulator. In this paper, we concentrate on a kinematic design, when kinematic parameters of manipulators are optimized. A kinematic approximation task is to find, among a parameterized family of robots, a kinematics that is as close as possible to a given nominal one. The evaluation of a manipulator's kinematics can be based on harmonic maps (Park and Brockett, 1994), volumes of their workspace (Yang and Lee, 1984), singularity avoidance (Spong and Vidyasagar, 1989), and some other dexterity measures derived from the singular values of the Jacobian matrix of the manipulator, (Klein and Blaho, 1987).

In this paper, we evaluate kinematics according to the Chebyshev kinematic metric introduced in (Tchoń and Dulęba, 1994). This metric joins mathematical clarity with Euclidean intuitions of a distance. The metric measures the distance between a pair of robots with similar kinematic structures by the maximization of the distance between the points in their workspaces corresponding to the same configurations of robots. The resulting minimax optimization task is solved with the use of a slightly modified mesh algorithm.

* Institute of Engineering Cybernetics, Wrocław University of Technology, Janiszewskiego 11/17, 50–372 Wrocław, Poland, e-mail: iwd@ict.pwr.wroc.pl

The results of this paper are important for the kinematic design of robot manipulators. The problem can be met in CAD/CAM systems and in situations when a manipulator should be replaced by a new one. Also some useful algorithmic tools are discussed, which can be easily used in are ready to use for other applications where the minimax problem appears.

This paper is organized as follows. In Section 2, the Chebyshev kinematic approximation task is defined. In this section, necessary formulae are provided to compute the exponential coordinates. The coordinates are used in the Chebyshev metric to equip the workspace of a manipulator with a measure of proximity of points. Section 3 discusses a mesh method applied to solve the kinematic approximation task. In Section 4, some results of computer simulations are presented for the approximation of the CYBOTECH robot with a ULB one. Both the robots have six degrees of freedom to fully utilize exponential coordinates which express joint positional and rotational characteristics of the workspace. Section 5 enumerates a few disadvantages of the Chebyshev kinematic metric and introduces the Hausdorff kinematic metric and measures that can be applied in the kinematic approximation task. Section 6 concludes the paper.

2. Chebyshev Kinematic Metric and Exponential Coordinates

The kinematics of a rigid manipulator with n degrees of freedom are described by the formula, (Paul, 1981; Spong and Vidyasagar, 1989):

$$f : Q \longrightarrow X \subset \mathrm{SE}(3),$$

where Q is a joint (configuration) space, and the workspace X is a subset of the special Euclidean group SE(3) (Paul, 1981). Usually, the joint space Q is a compact subset of $\mathbb{R}^n$.

Let a pair of kinematics f_1, f_2 sharing the same joint space Q be given. For any configuration $q \in Q$ a distance between

$$f_1(q) = \begin{bmatrix} R_1(q) & T_1(q) \\ 0 & 1 \end{bmatrix}$$

and

$$f_2(q) = \begin{bmatrix} R_2(q) & T_2(q) \\ 0 & 1 \end{bmatrix}$$

is denoted by $d(f_1(q), f_2(q))$, where $R_1, R_2 \in \mathrm{SO}(3)$, $T_1, T_2 \in \mathbb{R}^3$. The Chebyshev kinematic metric maximizes the distance between the points in SE(3) corresponding to the same configurations of manipulators over their joint space:

$$\rho(f_1, f_2) = \max_{q\in Q} d\big(f_1(q), f_2(q)\big). \tag{1}$$

In order to apply (1), the distance d in SE(3) should be defined. For this purpose, we use the exponential coordinates described in (Tchoń and Dulęba, 1993; 1994).

The special Euclidean group SE(3) is both a differentiable manifold and an algebraic group with matrix multiplication, being an example of a Lie group. At the identity of the group, I_4, a vector space called the Lie algebra se(3) is well defined. The space collects the vectors tangent to the curves lying in SE(3) and passing through the identity element. On each Lie algebra the exponential mapping into the corresponding Lie group

$$\exp : \mathrm{se}(3) \to \mathrm{SE}(3) \tag{2}$$

is defined according to the formula

$$\exp(A) = I + A + A^2/2 + \cdots \tag{3}$$

Let

$$\mathrm{SE}(3) \ni s = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \quad \text{with} \quad \cos\alpha = \frac{1}{2}(\mathrm{Tr}\, R - 1),$$

where Tr is the trace operator. For $0 \le \alpha < \pi$, the map

$$\exp^{-1} :\ \mathrm{SE}(3) \longrightarrow \mathrm{se}(3)\ (\mathbb{R}^6),$$

sets in $\mathbb{R}^6$ the inverse of the exponential coordinates (r, t) according to the expression

$$\exp^{-1}\left(\begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}\right) = \begin{bmatrix} [r] & t \\ 0 & 0 \end{bmatrix}, \tag{4}$$

where

$$t = \left(I_3 - \frac{1}{2}[r] + \frac{2\sin\alpha - \alpha(1+\cos\alpha)}{2\alpha^2 \sin\alpha}[r]^2\right) T, \tag{5}$$

and

$$[r] = \frac{\alpha}{2\sin\alpha}(R - R^T) \quad \text{for} \quad 0 \le \alpha < \pi. \tag{6}$$

The rotation components of the exponential coordinates $\mathbb{R}^3 \ni r = (r_1, r_2, r_3)^T$ are uniquely derived from the skew-symmetric matrix

$$[r] = \begin{bmatrix} 0 & -r_3 & r_2 \\ r_3 & 0 & -r_1 \\ -r_2 & r_1 & 0 \end{bmatrix}.$$

The inverse of the exponential coordinates can be extended to cover also the elements of SE(3) resulting in $\alpha = \pi$ (Dulęba, 2000), i.e.

$$|r_i| = \alpha \frac{R_{ii} + 1}{2} \quad \text{for} \quad \alpha = \pi \quad \text{and} \quad i = 1, 2, 3. \tag{7}$$

Equation (7) defines only the amplitudes of r coordinates. The signs of the components are easy to determine from the entries of the matrix R lying outside the main diagonal (Dulęba, 2000).

It is worth noticing that the rotational components of the exponential coordinates are related to the well-known axis-angle representation $(\tilde{r}, \alpha)$ of the special orthogonal group SO(3) (Spong and Vidyasagar, 1989) by the formula

$$r = \alpha \tilde{r}. \quad (\|r\| = \alpha)$$

The square of the distance between the points

$$s_1 \begin{bmatrix} R_1 & T_1 \\ 0 & 1 \end{bmatrix}, \quad s_2 = \begin{bmatrix} R_2 & T_2 \\ 0 & 1 \end{bmatrix}$$

in SE(3), with the rotation and position coordinates weighted with the same coefficient, is equal to (Tchoń and Dulęba, 1994)

$$d^2(s_1, s_2) = \langle r, r\rangle + \langle t, t\rangle = \langle r, r\rangle + \langle \theta, \theta\rangle + \gamma\big(\langle r, r\rangle\langle \theta, \theta\rangle - \langle r, \theta\rangle^2\big). \tag{8}$$

To derive eqn. (8), eqns. (4)–(6) were in use. In eqn. (8), $\langle\cdot,\cdot\rangle$ denotes the inner product in $\mathbb{R}^3$ and

$$[r] = \frac{\alpha}{2\sin\alpha}\big(R_1{}^T R_2 - R_2{}^T R_1\big), \quad \theta = R_1{}^T(T_2 - T_1),$$
$$\cos\alpha = \frac{1}{2}\big(\mathrm{Tr}\,(R_1{}^T R_2) - 1\big), \quad \gamma = \alpha^{-2}\left(\left(\frac{\frac{\alpha}{2}}{\sin\frac{\alpha}{2}}\right)^2 - 1\right). \tag{9}$$

A computationally simpler form of (8) is given by the expression (Tchoń and Dulęba, 1993):

$$d^2(s_1, s_2) = \alpha^2 + \delta\|T_2 - T_1\|^2 - \frac{\delta - 1}{\|\tilde{r}\|^2}\langle\tilde{r}, \theta\rangle^2. \tag{10}$$

where

$$\delta = \left(\frac{\frac{\alpha}{2}}{\sin\frac{\alpha}{2}}\right)^2, \quad r = \frac{\alpha}{2\sin\alpha}\tilde{r}.$$

The metric based on the exponential coordinates is not a unique metric structure on SE(3). Two other well-established metrics were reported in (Park, 1995). Slightly reformulated in order to use definitions (9), they are given by the formulae

$$d^2(s_1, s_2) = c\,\alpha^2 + e\,\|T_2 - T_1\|^2 \tag{11}$$

and

$$d^2(s_1, s_2) = c\,\alpha^2 + e\,\|R_1\, R_2^T\, T_2 - T_1\|^2, \tag{12}$$

where c and e are positive constants. Expressions (11) and (12) are sums of two distances, the first one measures the proximity in the special orthogonal group SO(3), while the other is the position proximity. The algorithm of kinematic approximation developed in Section 3 also works for metrics (11) and (12).

It is well-known from the classical analysis that any non-negative function has its optima at the same points as the function squared. For computational reasons, we prefer optimization of the square of the distance to avoid computationally expensive operations, i.e.

$$\rho^2(f_1, f_2) = \max_{q\in Q} d^2\big(f_1(q), f_2(q)\big) = d^2\big(f_1(q^\star), f_2(q^\star)\big). \tag{13}$$

Equations (10) and (13) constitute the basis for the computation of the Chebyshev distance between two kinematics f_1 and f_2 such that $f_1(q) = s_1$ and $f_2(q) = s_2$. The definition (13) can be extended naturally to cover a distance between a given nominal kinematics f_{nom} and a family of kinematics $f \in \mathcal{F}$:

$$\rho^2(f_{\mathrm{nom}}, \mathcal{F}) = \min_{f\in\mathcal{F}} \rho^2(f_{\mathrm{nom}}, f).$$

Usually, the family is parameterized with kinematic (geometric) parameters like translational shifts and axis misalignment angles. For this case, the Chebyshev kinematic approximation task is defined as follows:

The Chebyshev kinematic approximation task: *For the nominal kinematics* f_{nom}*, with a joint space* Q *and a family of kinematics* $\mathcal{F} = \{f(p) \mid p \in P\}$ *(where* P *is a compact and convex set of the geometrical parameters of the family) with the same joint spaces, find a parameter* $p^\star \in P$ *such that*

$$\begin{aligned}\rho^2\big(f_{\mathrm{nom}}, f(p^\star)\big) &= \min_{p\in P} \rho^2\big(f_{\mathrm{nom}}, f(p)\big) = \min_{p\in P}\max_{q\in Q} d^2\big(f_{\mathrm{nom}}(q), f(p,q)\big) \\ &= \min_{p\in P}\max_{q\in Q} F(p,q). \end{aligned}\tag{14}$$

Equation (14) introduces a standard minimax task with the performance index $F(p,q)$ to be optimized. In Section 3, a mesh method is presented for solving the Chebyshev kinematic approximation task.

3. Mesh Method of Solving the Minimax Task

In order to solve the Chebyshev approximation task, we will apply the mesh method set forth in (Demyanov and Malozemtsev, 1972). This method belongs to the class of (sub-)gradient techniques. It relies on finding, in each iteration, a function taking a (near-)maximal value of the performance index $F(p,q)$ with respect to the configuration variables. Then the function is minimized along the direction of its anti-gradient with respect to kinematic parameters. In what follows, only the indispensable ingredients of the required mesh method will be presented.

Let $F(p,q)$ be the optimized function, $p \in P'$, $q \in Q$, and P' be an open superset of the set P. It is assumed that $F(p,q)$ is continuous on $P' \times Q$ along with its partial derivative $\partial F(p,q)/\partial p$. Our aim is to minimize the function

$$\phi(p) = \max_{q\in Q} F(p,q) \tag{15}$$

on a convex and compact set of kinematic parameters $P \subset P'$. For a compact $P \subset P'$ and for some $p_0 \in P$, the set

$$M(p_0) = \{p \in P,\ \phi(p) \le \phi(p_0)\} \tag{16}$$

is bounded and $\phi(p)$ attains its minimum value on P, i.e. there exists a $p^\star \in P$ such that $\phi(p^\star) = \min_{p\in P}\phi(p)$.

The approximation task given by eqn. (14), equivalently introduced by (15), is continuous both in the parameter and joint coordinates. The idea of the mesh method is to discretize the joint space keeping the space of parameters continuous. The continuous-discrete task is easier to solve than its continuous-continuous counterpart. Below, Theorem 1 validating the discretization of the joint space will be presented after the introduction of two convenient notations: the symbol $[i:j]$ will denote the set of integers from i to j, $\|v\| = \max_i v_i$ being the l_∞ norm.

The joint space Q is covered by an everywhere dense mesh composed of N points

$$G_N = \{q^i \in Q,\ i \in [1:N]\},$$

i.e. for any real number $\epsilon > 0$ there exists an index N_0 such that for $N > N_0$ and $\forall q \in Q$: $\min_{i\in[1:N]} \|q - q^i\| = \|q - G_N\| \le \epsilon$. Let $f^i(p) = F(p, q^i)$ $i \in [1:N]$ and

$$\phi_N(p) = \max_{i\in[1:N]} f^i(p) = \max_{q\in G_N} F(p,q). \tag{17}$$

We assume that for N large enough there exists at least one stationary point $p_N \in P$ of the function $\phi_N(p)$

$$\inf_{z\in P}\max_{i\in R(p_N)}\left\langle \frac{\partial f^i(p_N)}{\partial p}, z - p_N \right\rangle = 0,$$

where

$$R(p) = \{i,\ i \in [1:N]\ \wedge\ f^i(p) = \phi_N(p)\}. \tag{18}$$

The following result guarantees the solvability of the continuous-discrete minimax task:

Theorem 1. (Demyanov and Malozemtsev, 1972) *Any limit point of the sequence $\{p_N\}$, indexed with finer and finer meshes, is a stationary point of the function $\phi(p)$ on the set P.*

Before presenting an algorithm implementing the mesh method, we need to introduce a couple of concepts used in its description:

- the neighborhood of a point p: $P(p) = \{z \in P,\ \|z - p\| \le 1\}$,
- two sets of indices, labelling the functions which, for a current p, attain (near) maximal values of $\phi_N(p)$; the first set is introduced by (18) while the other is given by

$$R_\epsilon(p) = \{i,\ i \in [1:N]\ \wedge\ \phi_N(p) - f^i(p) < \epsilon\},$$

- auxiliary functions corresponding to these sets of indices:

$$\psi(p) = \min_{z\in P(p)}\max_{i\in R(p)}\left\langle \frac{\partial f^i(p)}{\partial p}, z - p\right\rangle,\quad \psi_\epsilon(p) = \min_{z\in P(p)}\max_{i\in R_\epsilon(p)}\left\langle \frac{\partial f^i(p)}{\partial p}, z - p\right\rangle.$$

It can be checked that $p^\star \in P$ is a stationary point of $\phi_N(p)$ on P if $\psi(p^\star) = 0$. For a fixed p, the function $\psi_\epsilon(p)$ is a non-increasing piecewise-constant function of ϵ (Demyanov and Malozemtsev, 1972). An algorithm of the mesh method, which finds a saddle point of the function $F(p,q)$, is described by the following steps:

Step 1. Set parameters $\epsilon_0 > 0$, $\rho_0 > 0$ and an initial point $p_0 \in P$.

Step 2. Let the k-th point $p_k \in M(p_0)$ (cf. (16)) be determined. If $\psi(p_k) = 0$, then p_k is a stationary point of $\phi_N(p)$ in P, so the algorithm is stopped and a local minimum is found. Additionally, when the functions $f^i(p)$ are convex, the minimum is global.

When $\psi(p_k) < 0$, a sequence of real, positive numbers $\epsilon(\nu) = \epsilon_0/2^\nu$, $\nu = 0, 1, \ldots$ is checked as long as, for the first time, the following inequality holds:

$$\psi_\epsilon(p_k) \le -\frac{\rho_0}{\epsilon_0}\epsilon(\nu). \tag{19}$$

This may happen even for $\nu = 0$. The index ν determined by eqn. (19) is denoted by ν_k. When the index is finite, the left-hand side of (19) is negative while the right-hand side tends to zero.

Step 3. Compute $z_k \in P(p_k)$ satisfying

$$\psi_{\epsilon(\nu_k)}(p_k) = \max_{i \in R_{\epsilon(\nu_k)}(p)} \left\langle \frac{\partial f^i(p_k)}{\partial p}, z_k - p_k \right\rangle.$$

Step 4. Minimize the function $\phi(p)$ along the line joining z_k and p_k,

$$p = p_k(\alpha) = p_k + \alpha(z_k - p_k), \quad 0 \le \alpha \le 1.$$

$p_k(\alpha) \subset P$ as $\alpha \in [0,1]$. We find $\alpha_k \in [0,1]$ such that

$$\phi\big(p_k(\alpha_k)\big) = \min_{\alpha \in [0,1]} \phi\big(p_k(\alpha)\big). \tag{20}$$

Then $P(p) \ni p_{k+1} = p_k(\alpha_k)$ and $\phi(p_{k+1}) < \phi(p_k)$.

Step 5. Go to Step 2.

Iterating Steps 2–5, we obtain sequences $\{p_k\}$, $\{\epsilon(\nu_k)\}$, $\{z_k\}$ and $\{\alpha_k\}$, where $p_k \in M(p_0)$, $z_k \in P(p_k)$, $k = 0, 1, \ldots,$ and

$$\phi(p_0) > \phi(p_1) > \cdots > \phi(p_k) > \cdots .$$

If the sequence $\{p_k\}$ is finite, then its last element is a stationary point of the function $\phi_N(p)$ in P. When it is infinite, it can be proved (Demyanov and Malozemtsev, 1972) that the sequence ϕ converges and $\phi(p_N) \longrightarrow \phi(p^\star)$.

In any iteration of the algorithm, $\epsilon(\nu_k)$, z_k and α_k should be found. The value of α_k is easy to obtain by the optimization of a univariate function $\Phi(\alpha) = \phi(p_k(\alpha))$ over the interval $[0,1] \ni \alpha$. A much more difficult problem is to find $\epsilon(\nu_k)$ and z_k's, as they result from solving the task

$$\max_{i\in R_\epsilon(p)} \left\langle \frac{\partial f^i(p)}{\partial p}, z-p \right\rangle \longrightarrow \min_{z\in P(p)} \tag{21}$$

for a fixed p. Let $\mathrm{Co}_\epsilon(p)$ be the convex hull spanned by vectors $\partial f^i(p)/\partial p$, $i \in R_\epsilon(p)$. Then the optimization task (21) can be rewritten as

$$\max_{v\in \mathrm{Co}_\epsilon(p)} \langle v, z-p\rangle \longrightarrow \min_{z\in P(p)} . \tag{22}$$

In the general case, the task given by (22) is far from being trivial. Fortunately, for most robotic problems the set P is a cuboid $p_i^{\min} \le p_i \le p_i^{\max}$, $i = 1,\dots,m$, where $p_i^{\min}$ and $p_i^{\max}$ are constants. For this particular case, eqn. (22) formulates a linear-programming task. To see this, we set

$$P = \left\{z \in \mathbb{R}^m,\ z_i^{\min} \le z_i \le z_i^{\max},\ i = 1,\dots,m\right\}, \tag{23}$$

and consider an extended search space composed of vectors $w = (z_1,\dots,z_m,u)^T$, where u is a free auxiliary variable. Let $w^\star = (z_1^\star,\dots,z_m^\star,u^\star)$ denote a vector minimizing the linear form

$$\mathcal{L}(w) = u$$

subject to

$$\begin{aligned} &\left\langle \frac{\partial f^i(p)}{\partial p}, z-p \right\rangle \le u, && i \in R_\epsilon(p),\\ &z_i^{\min} \le z_i \le z_i^{\max}, && i = 1,\dots,m,\\ &-1 \le z-p \le 1. \end{aligned}$$

Then $z^\star = (z_1^\star,\dots,z_m^\star)$ is a solution to the task introduced by eqn. (15). When the set of kinematic parameters P is not an m-dimensional cuboid, a more general minimax solver should be applied (Pin *et al.*, 1994).

In the next section, the mesh algorithm will be slightly modified and applied to the approximation of the CYBOTECH robot with a member of the ULB family of robots.

4. Chebyshev Approximation of the CYBOTECH Robot with a ULB Robot

4.1. Modified Mesh Method

A specific form of the set of parameters (23) encountered in robotic problems facilitates solving the approximation task. On the other hand, the task is difficult to solve in the case of multi-dimensional joint and parameter spaces. A uniform mesh covering the n-dimensional space with K trial points for each dimension consists of K^n points. For robots fully employing both the positions and orientations of their tools, the joint space should be at least six-dimensional. To make the Chebyshev approximation task computationally tractable, we should reduce the number of mesh points. Therefore, instead of considering the global mesh G_N, we form a set of local meshes $G_N^{\rm loc}$ by truncating the global mesh on a subset $S \subset Q$ of the joint space, i.e.

$$G_N^{\rm loc}(S) = G_N|_S. \tag{24}$$

For any local reduced mesh, a solution to the approximation task is to be found. Let $(p^\star, q^\star)$ be the point of minimum of the function $\phi(p)$ given by (15) and assume that $q^\star$ is not a boundary point of the current mesh $G_N^{\rm loc}(S)$. Then the solution to the Chebyshev approximation task is found. Otherwise, when $q^\star$ is located on the boundary, also a reduced mesh $G_N^{\rm loc}(S = S')$ defined by centring the mesh, for the next iteration, around $q^\star \in S' \subset Q$. The approximation problem is solved by applying a two stage iterative produce. At its first step the approximation task is solved with the current local mesh as the domain of search. Then, when an optimal point has been found, the local mesh is centered around that point and the mesh is read for the next iteration. Obviously, a stationary solution will be found as an upper-bounded and increasing sequence of current best solutions defined by the iteration process.

Besides the reduction in the number of mesh points also an altered distance between the points will be allowed to shorten the time of solving the approximation task. A mesh with a rough grid is applied first. Then, when a stationary point for this particular grid is found, the discretization is made finer and a solution is sought alter initialization at the point of the recent optimum. The process can be repeated several times. To measure a computational complexity of the Chebyshev kinematic approximation, we use the formula

$$\mathrm{cmplx} = \mathrm{iter} \times K^n, \tag{25}$$

where 'iter' is the number of evaluations of $f^i(p)$, $i \in [1:N]$, K denotes the number of points discretizing a single coordinate of the joint space (we assume that all joint coordinates are discretized with the same number of points) and n stands for the number of degrees of freedom. The measure of the computational complexity introduced by (25) is valid when the auxiliary computations necessary when computing the values of $f^i(p)$, $i \in [1:N]$ are negligible.

In order to determine α_k from (20), $f^i(p)$ should be evaluated many times (e.g. using the golden section search algorithm (Vasilev, 1988)). To reduce the computational cost of computing α_k, we propose to obtain its value based on the formula

$$\max_{i \in R_\epsilon(p_k)} f^i\big(p_k(\alpha_k)\big) = \min_{\alpha \in [0,1]} \max_{i \in R_\epsilon(p_k)} f^i\big(p_k(\alpha)\big) \tag{26}$$

instead of (20). A significant difference between (20) and (26) is that the set of functions indexed with i has the power N for (20) and $|R_\epsilon(p_k)| \ll N$ when (26) is applied. Consequently, computing α_k according to (26) is less involved. One serious drawback in using (26) instead of (20) is that the former formula may sometimes violate the condition $\phi(p_{k+1}) < \phi(p_k)$ required for convergence of the mesh algorithm. However, the advantages of the modified way of setting the value of α_k in reduction of the computational complexity are so sound that we apply (26) to compute α_k. After computing α_k, we check whether the condition $\phi(p_{k+1}) < \phi(p_k)$ is satisfied. If this is the case, the computational process progresses. Otherwise, only for one iteration, formula (26) is substituted with (20), and the condition $\phi(p_{k+1}) < \phi(p_k)$ must hold.

Our modifications of the basic mesh method to solve the minimax problem resulting from the Chebyshev kinematic approximation task are the following. One global dense mesh composed of an extremely large number of points is replaced by iteratively defined local meshes with reasonable numbers of points. The local meshes can also have varied distances between the neighbouring points so as to cover the whole search space. The computational complexity of determining the value of α_k was reduced by using formula (26) instead of (20). The last modification is introduced to extend the domain $p_k z_k$ of searching for α_k to the interval $p_k \tilde{z}_k$, where $\tilde{z}_k$ is a point at which the line initialized at p_k and passing through z_k meets the boundary of the set P.

The delineated kinematic approximation algorithm can be used not only to design a manipulator close (in the Chebyshev sense) to a nominal manipulator (to be replaced with the new design) but also to compare sub-manipulators, e.g. 3 DOF grippers to check whether they can do the same manipulations in SO(3). The square of the distance in SO(3) is given by the term $\langle r, r\rangle = \alpha^2$ in (8). The comparisons may be restricted to any compact subset of their configuration spaces.

The modified mesh method of solving the Chebyshev approximation task will be applied to approximate the CYBOTECH manipulator with a member of the ULB-family of robots.

4.2. Simulation Results

To fully utilize both the position and orientation coordinates, we chose a ULB robot with six degrees of freedom bearing the 6R kinematic structure, and a CYBOTECH manipulator with 7 degrees of freedom and 7R kinematic structure. Since the Chebyshev kinematic metric can be used only for the same kinematic structures, the last degree of freedom of the CYBOTECH robot was left immobile. The kinematics of these robots given in the standard Denavit-Hartenberg notation are provided in Table 1. The data for the ULB robot were taken from (Renders *et al.*, 1991) while for the CYBOTECH robot from (Litvin *et al.*, 1987). The robot to be approximated is the CYBOTECH with kinematics $f_1 = f_{\mathrm{nom}}$. Its approximation is sought within the family of ULB robots with kinematics $f_2(p)$. The configuration space is $\mathcal{T}^6$ and the space of parameters $(a_2, d_4, d_6)^T = (p_1, p_2, p_3)^T = p \in P \subset \mathbb{R}^3$. The mesh algorithm requires frequent evaluations of $\partial F(p,q)/\partial p$. Using (10) and (14), after simple computations we arrive at

$$\frac{\partial F(p,q)}{\partial p_i} = 2\delta \left\langle T_2 - T_1, \frac{\partial \|T_2 - T_1\|}{\partial p_i} \right\rangle - 2\frac{\delta - 1}{\|\tilde{r}\|^2}\langle \tilde{r}, \theta\rangle^2 \left\langle \tilde{r}, {R_1}^T \frac{\partial T_2}{\partial p_i} \right\rangle, \quad i = 1,2,3.$$

Table 1. Denavit-Hartenberg parameters of the CYBOTECH and ULB manipulators. The asterisk denotes the variables of motion.

i	CYBOTECH				ULB			
	a_i [cm]	d_i [cm]	$\alpha_i[°]$	θ_i	a_i [cm]	d_i [cm]	$\alpha_i[°]$	θ_i
1	0	0	90	*	0	0	−90	*
2	$a_2 = 76.2$	0	0	*	$a_2 = 35$	0	0	*
3	$a_3 = 76.2$	0	−90	*	0	0	90	*
4	0	0	−90	*	0	$d_4 = 35$	−90	*
5	0	$d_5 = 152.4$	−90	*	0	0	90	*
6	0	0	90	*	0	$d_6 = 10$	0	*
7	0	0	0	0	—	—	—	—

The steps of the mesh algorithm implemented for simulations are the following:

Step 1. Read in the initial data:

- the space of kinematic parameters $p_i^{\min}$, $p_i^{\max}$, $i = 1, 2, 3$,
- the initial point of search $p_0 \in P$,
- the number of mesh changes l,
- the initial mesh size Δq,
- the number of points discretizing each joint coordinate K, and
- the initial translation of the mesh window T.

Set $r = 0$ (the iteration counter). The initial mesh is given as

$$G_N^{\text{loc}} = \Big\{(0, q_2, q_3, q_4, q_5, q_6)^T,$$

$$q_i = T\pi/2 + j\Delta q, \quad j = 0, \ldots, K-1, \ i = 2, \ldots, 6\Big\}.$$

We set $q_1 = 0$ as both the robots have the same first rotation axis z_0 and $\rho(f', f'')$ should not depend on the choice of the first joint coordinate. For simplicity, we set the most trivial value of q_1.

Step 2. $r \longleftarrow r + 1$ (i.e. substitute $r + 1$ for r).

Step 3. For the local mesh G_N^{loc}, find a solution to the minimax problem

$$\left(p^{(r)\star}, q^{(r)\star}\right), \quad \text{i.e.} \quad F\left(p^{(r)\star}, q^{(r)\star}\right) = \min_{p\in P} \max_{q\in G_N^{\text{loc}}} d^2(p, q),$$

with the use of the mesh algorithm and modifications presented in Subsection 4.1.

Step 4. When $q^{(r)\star}$ lies on the boundary of the mesh $G_N^{\rm loc}$, centre $G_N^{\rm loc}$ around $q^{(r)\star}$, i.e.

$$G_N^{\rm loc} = \Big\{(0, q_2, q_3, q_4, q_5, q_6)^T,\ q_i = q^{(r)\star} + \big(j - \lfloor K/2 \rfloor\big)\Delta q,\quad j = 0, \dots, K-1,\ i = 2, \dots, 6\Big\}. \tag{27}$$

(here $\lfloor a \rfloor$ denotes the greatest integer which is less than or equal to a) and go to Step 2. Otherwise, continue with Step 5.

Step 5. Check the stopping condition: if the mesh changed its size l times, then stop the computations. Otherwise, make the size of the mesh twice finer $\Delta q \longleftarrow \Delta q/2$, and centre $G_N^{\rm loc}$ around $q^{(r)\star}$ in accordance with (27).

Step 6. Go to Step 2.

The above algorithm covers only a family of tasks characterized by uniform meshes (each coordinate evenly partitioned using the same number of points and the shift of the mesh window described by a constant value of T) covering sub-spaces of the joint space. We selected this family of tasks because it can be coded with only a few numbers. Computer simulations have to show the influence of some parameters of the minimax task on the resulting solution $p^\star$.

In the very first simulations, we varied the value of K, determining the number of points of the local mesh $G_N^{\rm loc}(|G_N^{\rm loc}| = K^{n-1})$, the size of the mesh Δq and the number of changes in the grid l. The remaining data were fixed as follows: the initial point $p_0 = [35, 35, 10]^T$ (nominal parameters of the ULB robot); space of parameters: $P = [32, 38] \times [32, 38] \times [7, 13]$; lengths of the mesh discretization: $\Delta q = 10°, 20°, 30°$; number of mesh changes: $l = 3 + \Delta q/10$; initial mesh window: $T = 0, 1, 2, 3$. In the simulations only formula (26) was used (with no switches to formula (20)). Consequently, cycles in computations are possible. This means that, for some k, it may happen that $\phi(p_{k+1}) \geq \phi(p_k)$ and the values of ϕ oscillate (usually between two values).

In most tables presenting the results of simulations we provide the following data: the optimal value of $\phi(p^\star)$, the optimal point in the space of kinematic parameters $p^\star$ and the configuration $q^\star$ (without coordinate $q_1 = 0$) where the optimum was located. When oscillations appear, the data of points of the circulation are shown.

At first, the influence of the number of points discretizing each single coordinate of the joint space on the quality of the solution was tested. The results are presented in Tables 2–4. The optimal point in the space of kinematic parameters does not depend on the number of points discretizing each joint coordinate (cf. Tables 2–4). The number of points in a local mesh strongly influences the complexity of the minimax task. Accordingly, it is reasonable to use local meshes with only three points for each joint coordinate. When a cycle in computations appears (cf. Table 2 for $T = 3$ and $\Delta q = 10°$), it is advised to change the initial point of search, rather than to use the computationally involved formula (20) to obtain α_k. When there are no cycles,

Table 2. Results for three points discretizing each joint coordinate.

T	$\Delta q = 10°$	$\Delta q = 20°$	$\Delta q = 30°$
0	303805 $(32, 32, 7)$ $(-75, 27, -85, 20, 25)$	302055 $(32, 32, 7)$ $(105, 35, -85, 15, 25)$	309487 $(32, 32, 7)$ $(105, 30, -90, 0, 7)$
1	104701 $(32, 32, 7)$ $(80, 145, 107, 35, 17)$	302055 $(32, 32, 7)$ $(105, 35, -85, 15, 25)$	309487 $(32, 32, 7)$ $(105, 30, -90, 0, -7)$
2	267403 $(32, 32, 13)$ $(-72, 32, -70, 145, -150)$	267293 $(32, 32, 13)$ $(105, 35, -70, 145, -150)$	276741 $(32, 32, 13)$ $(105, 30, -90, 165, -172)$
3	160100 $(38, 38, 13)$ $(-170, -20, -80, -90, -90)$ 161546 $(38, 32, 13)$ $(-150, -20, -90, -80, -100)$	161546 $(38, 32, 13)$ $(-150, -20, -90, -80, -100)$ 160100 $(38, 38, 13)$ $(-170, -20, -80, -90, -90)$	309487 $(32, 32, 7)$ $(-75, 30, -90, 0, 7)$ — — —

Table 3. Results for five points discretizing each joint coordinate.

T	$\Delta q = 10°$	$\Delta q = 20°$	$\Delta q = 30°$
0	310905 $(32, 32, 7)$ $(106, 28, -91, 1, 5)$	303020 $(32, 32, 7)$ $(107, 32, -87, 15, 25)$	309688 $(32, 32, 7)$ $(105, 30, -86, 4, -4)$
1	303020 $(32, 32, 7)$ $(107, 32, -87, 15, 25)$	309410 $(32, 32, 7)$ $(105, 30, -87, 15, 7)$	276954 $(32, 32, 13)$ $(-75, 30, -86, 165, -172)$
2	165368 $(38, 32, 13)$ $(-110, -60, -100, 130, -70)$ 162358 $(32, 38, 13)$ $(-140, -60, -100, -140, -120)$	154184 $(38, 32, 13)$ $(-100, -60, -80, 180, -60)$ 162799 $(32, 38, 7)$ $(-140, -60, -80, -120, -80)$	165587 $(32, 38, 7)$ $(180, -60, -90, -120, 180)$ 164810 $(38, 32, 13)$ $(-90, -60, -90, 180, -60)$
3	303020 $(32, 32, 7)$ $(-107, -32, -92, -15, -25)$	309410 $(32, 32, 7)$ $(-75, 30, -92, -15, -7)$	309688 $(32, 32, 7)$ $(-75, 30, -93, -4, 4)$

formulae (20) and (26) give the same optimal values. The results collected in Tables 2–4 validate the claim that some joint coordinates of the optimal configuration $q^\star$ tend to a constant value ($q_3^\star$), while the others, e.g. ($q_2^\star$), can take more than one optimal value (here $q_2^\star = 105°$ or $q_2^\star = -75°$). Since the trigonometric functions are periodic, more than just one optimal point can be expected in the joint space.

Table 4. Results for seven points discretizing each joint coordinate.

T	$\Delta q = 10^\circ$	$\Delta q = 20^\circ$	$\Delta q = 30^\circ$
0	311134 $(32, 32, 7)$ $(106, 30, -91, 2, 5)$	304010 $(32, 32, 7)$ $(107, 32, -90, 22, 25)$	310600 $(32, 32, 7)$ $(105, 30, -94, 2, 6)$
1	311139 $(32, 32, 7)$ $(106, 30, -91, -2, -4)$	276524 $(32, 32, 13)$ $(-74, 30, -92, -160, 166)$	276954 $(32, 32, 13)$ $(105, 30, -94, -165, -172)$
2	276164 $(32, 32, 13)$ $(106, 30, -93, 157, 166)$ — — —	170206 $(32, 38, 13)$ $(-140, -60, -80, -60, -60)$ 160333 $(38, 32, 7)$ $(-80, -60, -80, -60, -140)$	310600 $(32, 38, 7)$ $(-75, 30, -86, -2, -6)$ — — —
3	311134 $(32, 32, 7)$ $(-75, 40, -89, -2, -4)$	310905 $(32, 32, 7)$ $(-74, 29, -89, -1, -5)$	310600 $(32, 32, 7)$ $(-75, 30, -94, 2, 6)$

Table 5. Computational complexity of the minimax approximation task when using eqns. (20) and (26) to determine α_k.

K	$\lvert G_N^{\text{loc}} \rvert$	iter eqn. (26)	iter eqn. (20)
3	243	42	106
5	3125	32	80
7	16807	24	40

To examine the influence of the initial point p_0 on the optimal set of parameters $p^\star$, we chose the task defined by the data $\Delta q = 10^\circ$, $T = 0$, $K = 5$. For the initial points in the parameter space $(37, 37, 12)^T$, $(33, 33, 12)^T$, $(33, 33, 8)^T$, $(33, 37, 8)^T$, $(37, 33, 8)^T$, the optimal set of parameters is $p^\star = (32, 32, 7)^T$. The results do not depend on the initial point in the space of kinematic parameters because, due to the low initial resolution of the mesh, the penetration abilities of the algorithm in the joint space are high.

The aim of the next test was to evaluate the computational complexity when the parameter α optimized in each iteration of the mesh algorithm is computed according to either formula (20) or (26). As can be seen from Table 5, formula (26) speeds us computations significantly. As has been mentioned before, the results obtained with both the formulae are in most cases the same. The computational complexity is mainly influenced by the number of points discretizing each joint coordinate, K, cf. Table 5. The most computationally demanding task with $K = 7$ required a few minutes on a PC.

The next test checked the influence of varied ranges $p_i^{\min}$ and $p_i^{\max}$, $i = 1, 2, 3$ of the set of kinematic parameters on the optimal point in this set. The simulations were initialized at the point $p_0 = [32, 32, 10]^T$ with $\Delta q = 10°$ and $T = 0$. The results shown in Table 6 indicate that the optimal set of kinematic parameters is always located on the boundary of the space of kinematic parameters. This observation also confirms the results from Tables 2–4 and 6.

Table 6. Ranges of feasible kinematic parameters.

$p_1^{\min}$	32	32	25	25	32	32	32	25
$p_1^{\max}$	38	45	38	38	38	38	38	38
$p_2^{\min}$	32	32	25	32	25	32	32	32
$p_2^{\max}$	38	45	38	45	38	38	38	38
$p_3^{\min}$	7	7	5	7	7	5	7	7
$p_3^{\max}$	13	20	13	13	13	13	20	13
$p^\star$	32, 32, 7	32, 32, 7	25, 25, 5	25, 32, 7	32, 25, 7	32, 32, 5	32, 32, 7	25, 32, 7
ρ	303805	303805	281996	296552	292200	300501	303805	296552

It should be mentioned that the basic algorithm of solving the minimax task (Demyanov and Malozemtsev, 1972), cannot be applied in the kinematic approximation task due to a burst-of-computation phenomenon. The phenomenon arises when a single dense mesh in the joint space is applied to preserve the reliability of the solution. When the modifications presented in this paper are applied, locally optimal solutions are found with a reasonable amount of computations.

5. Alternative Kinematic Measures and Metrics

Besides advantages, the Chebyshev metric used in the kinematic approximation task possesses some disadvantages. It compares only the kinematics of the manipulators which belong to the same class described by three conditions: the same number of degrees of freedom, the same kinematic structure and identical joint spaces. When the first two conditions are fulfilled, the third can be satisfied by possibly shrinking the joint space of one of the manipulators. Sometimes the first two conditions are too strong to be met. For those cases, we propose a Hausdorff kinematic metric.

Let $(V \subset \mathrm{SE}(3), d)$ be a compact space equipped with the distance measure d in $\mathrm{SE}(3)$ introduced in (10). The measure of proximity of a point $x \in \mathrm{SE}(3)$ and a compact subset $A \subset \mathrm{SE}(3)$ is equal to

$$d(x, A) = \min_{y \in A} d(x, y). \tag{28}$$

The distances between two compact sets $A, B \subset \mathrm{SE}(3)$ are described by

$$d(A, B) = \max_{x \in A} d(x, B), \quad d(B, A) = \max_{x \in B} d(x, A). \tag{29}$$

The Hausdorff metric measuring the distance between sets A and B can be defined as follows:

$$h(A,B) = \max\big(d(A,B), d(B,A)\big). \tag{30}$$

As a joint space of robotic manipulators is a compact set and kinematics are continuous functions of configurations, the definition (30) can be applied to robotic manipulators by setting

$$A = f_1(Q_1), \quad B = f_2(Q_2), \tag{31}$$

Here n_1 and n_2 are degrees of freedom of the manipulators, Q_1 and Q_2 denote their joint spaces, and f_1 and f_2 describe their kinematics, respectively. The Hausdorff kinematic proximity measure of the manipulators' kinematics is introduced by

$$\rho_H(f_1, f_2) = h\big(f_1(Q_1), f_2(Q_2)\big). \tag{32}$$

Somewhat informally, we call ρ_H the Hausdorff kinematic metric, although $\rho_H(f_1, f_2)$ may vanish even when $f_1 \neq f_2$. But from the point of view of performing robotic tasks in a workspace, the two kinematics are kinematically equivalent. Equation (32) characterizes both the kinematic equations and joint spaces by measuring the distance between some points in SE(3). As opposed to the Chebyshev kinematic metric, the Hausdorff kinematic metric, can be defined for any pair of robots with no restrictions on their joint spaces.

Additionally, the following measures of the distance between two kinematics can be defined:

$$\begin{aligned} d_1(f_1, f_2) &= \int_{q_1 \in Q_1 \subset \mathbb{R}^{n_1}} \operatorname{sign}\Big(\min_{q_2 \in Q_2 \subset \mathbb{R}^{n_2}} d\big(f_1(q_1), f_2(q_2)\big)\Big)\, \mathrm{d}q_1, \\ d_2(f_2, f_1) &= \int_{q_2 \in Q_2 \subset \mathbb{R}^{n_2}} \operatorname{sign}\Big(\min_{q_1 \in Q_1 \subset \mathbb{R}^{n_1}} d\big(f_1(q_1), f_2(q_2)\big)\Big)\, \mathrm{d}q_2, \end{aligned} \tag{33}$$

and

$$m(f_1, f_2) = \max\big(d_1(f_1, f_2), d_2(f_2, f_1)\big), \tag{34}$$

where the sign function is defined classically, $d(f_1(q_1), f_2(q_2))$ being the distance between points $f_1(q_1)$ and $f_2(q_2)$ in SE(3). The measure m can be interpreted as a symmetric difference of two sets of masses $f_1(Q_1)$ and $f_2(Q_2)$, respectively. The density of a mass takes the value of zero on the set $f_1(Q_1) \cap f_2(Q_2)$ while outside this set it takes a value that depends on the number of configurations realizing a given point in SE(3).

It seems that the Hausdorff metric shadows somehow the information about the components maximized, as it evaluates only boundary points of robot workspaces. Expressions (29), (31) or (33) characterize the manipulator's kinematics better than (32). All the kinematic measures introduced in this subsection seem to be computationally more involved than the Chebyshev kinematic metric.

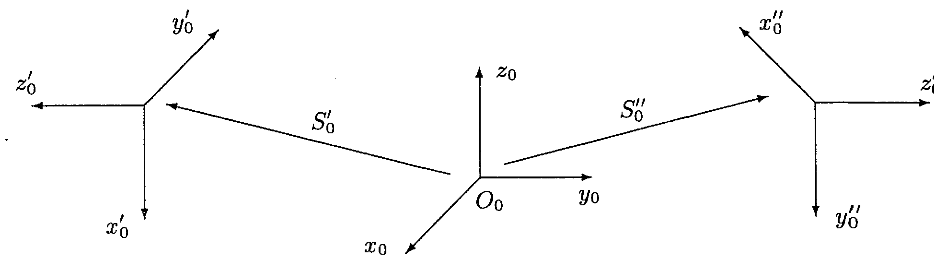

Fig. 1. The global coordinate frame and local coordinate frames of manipulators.

For the Chebyshev and Hausdorff kinematic metrics, as well as for measures (29), (31) and (33), (34), one more assumption can be avoided. It concerns the identity of the base coordinate frames $O_0x_0y_0z_0$ of the robots to be compared. To weaken this assumption, we introduce an extra zeroth degree of freedom, as depicted in Fig. 1. In this case, the approximation task requires also to determine the locations of local coordinate frames of robots $s_0^1, s_0^2 \in \mathrm{SE}(3)$ in the global coordinate frame $x_0y_0z_0$. Without loss of generality, we can set either $s_0^1 = I_4$ or $s_0^2 = I_4$, i.e. one of local coordinate frames coincides with the global coordinate frame.

6. Conclusions

In this paper, a Chebyshev kinematic approximation of robot manipulators has been proposed. The minimax problem generated by the approximation task is solved with a modified mesh algorithm. The relevant details of the algorithm together with its modifications have been given. The modifications are aimed at reducing the computational complexity of the algorithm. The approximation task has been illustrated with the approximation of the CYBOTECH robot with a member of the ULB family of robots. Those robots constitute a good choice for the approximation task as they possess at least six degrees of freedom and fully utilize motion abilities in the general workspace (a six-dimensional subset of SE(3)). The Chebyshev approximation task is computationally involved and can be solved only in an off-line mode. However, there is no need to solve this task in real time. Some conditions constituting the Chebyshev kinematic metric have been weakened and, as an alternative, the Hausdorff kinematic metric as well as a measure of the proximity of manipulators given by (33) and (34) have been introduced.

References

Demyanov W.F. and Malozemtzev W.N. (1972): *Introduction to Minmax.* — Moscow: Optim. Ops. Res. Sc. Publ. (in Russian).

Dulęba I. (2000): *Straight lines in external spaces of manipulators.* (in Polish, submitted).

Klein C. and Blaho B. (1987): *Dexterity measures for the design and control of kinematically redundant manipulators.* — Int. J. Robotics Res., Vol.6, No.2, pp.72–83.

Litvin F., Castelli V.P. and Shaheen M. (1987): *Singularities in motion and displacement functions for a 7 degree-of-freedom manipulator.* — J. Robotic Syst., Vol.4, No.3, pp.397–421.

Park F. (1995): *Distance metrics on the rigid-body motions with applications to mechanism design.* — Trans. ASME J. Mech. Design, Vol.117, No.1, pp.48–54.

Park F. and Brockett R. (1994): *Kinematic harmonic maps and optimal design of robotic mechanisms.* — Int. J. Robotics Res., Vol.13, No.1, pp.1–15.

Paul R. (1981): *Robot Manipulators: Mathematics, Programming and Control.* — Cambridge: MIT Press.

Pin F., Culioli J.-C. and Reister S. (1994): *Using minmax approaches to plan optimal task commutation configurations for combined mobile platform-manipulator systems.* — IEEE Trans. Robotics Automat., Vol.10, No.1, pp.44–53.

Renders J.M., Rossignol E., Becquet M. and Hanus R. (1991): *Kinematic calibration and geometrical parameter identification for robots.* — IEEE Trans. Robotics Automat., Vol.7, No.6, pp.721–731.

Spong M. and Vidyasagar M. (1989). *Introduction to Robotics. Robot Dynamics and Control.* — Cambridge: MIT Press.

Tchoń K. and Dulęba I. (1993): *On inverting singular kinematics and geodesic trajectory planning.* — J. Intell. Robotic Syst., Vol.8, No.3, pp.325–359.

Tchoń K. and Dulęba I. (1994): *Definition of a kinematic metric for robot manipulators.* — J. Robotic Syst., Vol.11,. No.3, pp.211–222.

Vasilev F.P. (1988): *Numerical methods of extremal problems.* — Moscow: Nauka (in Russian).

Yang D.C. and Lee T. (1984): *Heuristic combinatorial optimization in the design of manipulator workspace.* — Trans. Syst. Man Cybern., Vol.14, No.4, pp.571–580.

Received: 28 October 1999
Revised: 20 April 2000