Solutions of the inverse kinematic problem for manipulation robots based on the genetic algorithm

N O Krakhmalev, D A Korostelyov
Bryansk State Technical University, Bryansk, Russia

krakhmalevnik@yandex.ru

Abstract. Using simple examples, an unconventional method for solving the inverse kinematic problem for manipulation robots is considered. The classical nonlinear formulation of this problem is reduced to the optimization problem. To solve which it is proposed to use the stochastic method, implemented based on a genetic algorithm. As the optimization criterion (fitness function), the square of the deviation of the position of the tool center point of the manipulation robot from the given trajectory point is selected.

Various kinematic schemes of manipulation systems (flat, spatial, with kinematic redundancy) are investigated. The algorithm showed good convergence with a given accuracy, including multivariate solutions to the problem.

1. Introduction
Solution of the kinematics inverse problem consists in determining the values of the hinged (curvilinear) coordinates of the manipulation robot according to the given Cartesian (rectangular) coordinates of the characteristic point of its output link (TCP - tool center point) and is a problem on which the performance of the robot depends on a lot. The problem is such that in the mathematical description of the space surrounding the robot, including 3D models of manipulation objects (parts), various coordinate metrics are used. These metrics are chosen for reasons of convenience in describing the corresponding types of objects or spaces.

For example, to describe the surrounding space in CAD-systems use Cartesian coordinates, the same coordinates are used in the design of structures of various parts. However, the specifics of manipulative robot control is that the control models of the robots themselves use hinge coordinates. This is due to the fact that the actuators of the control system are servos, and each such drive implements the relative movement of the links in one of the hinges. For these reasons, program motion trajectories are specified in Cartesian coordinates, and for the implementation of controlled motion by the robot, these coordinates must be converted to the hinge coordinates of a particular robot.

The need to solve the inverse problem of kinematics for practical purposes arises when programming manipulation robots. In order to move the manipulated object or tool from point to point, it is necessary to form a Point-to-point (PTP) command in the robot control program. The formation of this command assumes, for example, that when programming a robot by training, first the TCP (Tool center point) of the robot is brought to the starting point, and then to the end point. In this case, the robot control system reads from the servo encoders and stores the values of the hinge coordinates for each of these points.

The trajectory of movement between given points can be constructed, for example, based on minimizing time spent by robot to move. Another thing is if between start and end points of trajectory
must have a certain, strictly defined form, for example, a straight line. Then, based on data corresponding to start and end points, an analytical or numerical (tabular) representation of line segment bounded by these points is created. Allocation on a segment of a set of points close enough to each other, and solving for each inverse kinematics, allows to generate for each point of the trajectory a vector of corresponding joint coordinates.

The traditional approach to solving the inverse kinematics problem is based on the compilation and solution of closed systems of nonlinear equations connecting curvilinear and rectangular coordinates based on the equality of the same geometric quantities expressed in different metrics. The closedness of these systems of equations implies the correspondence of the number of equations in them to the number of unknown articulated coordinates. In the case of kinematic redundancy of the manipulation system, isolation is ensured by “freezing” some articulated coordinates.

If it is not possible to express explicitly unknown hinge coordinates from the compiled system of nonlinear equations connecting these coordinates with Cartesian ones, then, in particular, numerical methods can be used to find the roots of equations. For example, the formulation of inverse kinematics problem for its numerical solution can be formulated as follows

\[
\{Y\} = \{f_i(x)\}, \quad i = (1, 2, 3). \Rightarrow \{dY\} = \sum_{j=1}^{n} \frac{\partial f_i}{\partial x_j} dx_j = [J]dx,
\]

where \(\{Y\}\) – vector (1×3) of Cartesian coordinates, \(\{x\}\) – vector (1×n) of hinge coordinates, \(f(x)\) – nonlinear functions of hinge coordinates, \([J]\) – Jacobi matrix (n×3).

Moving from the differential representation of equation (1) to the finite difference we obtain equations of form

\[
[J][\Delta x] = \{\Delta Y\},
\]

\[
\{\Delta x\} = [J]^{-1}\{\Delta Y\}.
\]

Equation (2) can be solved using numerical methods for solving systems of linear equations, for example, by the method of variable elimination (Gauss). When solving equation (3) it is necessary to use the numerical method of matrix inversion. Based on these equations an iterative algorithm for solving the inverse kinematics problem for a motion trajectory given by a set of points represented by their Cartesian coordinates can be constructed

\[
\{x^{k+1}\} = \{x^k\} + \{\Delta x^k\},
\]

where \(k=1, \ldots, N\) – number of points of trajectory partition.

The considered numerical methods have their drawbacks. Perhaps main one is the disadvantage associated with need to calculate the elements of the Jacobi matrix at each iteration step. The use of numerical methods may require a large amount of time, exceeding allowable limit for controlling robot in real time. The considered algorithm has other drawbacks. For example, the increase (accumulation) of calculation error from iteration to iteration due to expression (4).

2. Method of solution

In this article, a stochastic optimization method based on a genetic algorithm was used to solve inverse kinematics problem of manipulative robots. The genetic algorithm in its various modifications is a computational procedure based on an analogy with evolutionary processes occurring in the nature. When describing it, they use terms that have arisen in neurobiology, but they reflect the specifics of a computational algorithm. Therefore, a gene is a structure for presenting information about variables (parameters) of being researched mathematical model. The shape of gene structure depends on a type of used variables. Distinguish real, binary, symbolic and other forms of genes. A chromosome (individual) is an ordered sequence of genes, often considered as a vector, which elements are corresponding genes. A population is many individuals.
The genetic algorithm, as a rule, contains the following operators: generation of the original population, selection of parents, recombination (reproduction), mutation, selection of individuals into a new population, etc. All these operators are implemented by random functions; therefore, that is why genetic algorithms belong to stochastic methods.

Fitness function is used to assess the fitness of each individual and the population as a whole. The value of the fitness function, calculated based on the value of the genes of a particular individual, is taken into account when choosing parents and recombination. Genes are more likely to interbreed with best genes for which fitness function is better. The best value of the fitness function (Best fitness) is determined by the direction of optimization, by finding the maximum or minimum of fitness function. When analyzing the convergence process, the average value of fitness function by population (Mean fitness) is often used.

As a fitness function for solving inverse kinematics problem of manipulating robots, we will use a function of calculating the square of position deviation of the robot’s output link characteristic point from selected point on a motion path. We use the genetic algorithm to search for values of hinge coordinates minimizing fitness function using example of a flat two-link (figure 1) and three-link spatial manipulation systems (figure 2).

![Figure 1](image1.png)  ![Figure 2](image2.png)

**Figure 1.** Deviation of the position of the flat two-link manipulation system  
**Figure 2.** Deviation of the position of the spatial three-link manipulation system

For a two-link manipulation robot, the chromosome will be a vector \(X=[x_1 \; x_2]^T\) which elements are the hinged coordinates, \(x_1\) – is rotational, \(x_2\) – is translational. For these coordinates, we introduce restrictions \(0 \leq x_1 \leq 2\pi, \; 0.5 \leq x_2 \leq 1.5\). We make up a fitness function

\[
z = \left( y_1 - x_2 \cos(x_1) \right)^2 + \left( y_2 - x_2 \sin(x_1) \right)^2,
\]

where \(Y=[y_1 \; y_2]^T\) – vector of Cartesian coordinates of point selected on the trajectory.

We set coordinates of selected point \(y_1=1.0000\) m – rectangular coordinate along \(x\) axis, \(y_2=0.0000\) m – rectangular coordinate along \(y\) axis. We set the population size at 10 individuals. We run algorithm to carry out. In the eleventh generation, the calculation result was: \(z=3.9272e-09\) – value of fitness function for best individual, \(x_1=6.2830 \approx 2\pi\) rad, \(x_2=1.0000\) m (figure 3).
Figure 3. The results of calculating the fitness function

A repeated experiment produced the same result, starting algorithm for third time yielded values:

\[ z = 3.9154 \times 10^{-09}, x_1 = 0.0001 \text{ rad}, x_2 = 1.0000 \text{ m}. \]

These values of hinged coordinates also correspond to one of two possible positions of manipulation system, since in one at the same position, \( x_1 \) can take value 0 or \( 2\pi \).

After changing values of Cartesian coordinates by \( y_1 = 0.8660 \text{ m} \) and \( y_2 = 0.5000 \text{ m} \). Starting algorithm yielded values: \( z = 1.0299 \times 10^{-10}, x_1 = 0.5236 \approx \pi/6 \text{ rad}, x_2 = 1.0000 \text{ m}. \), which corresponds to solving the problem with a high degree accuracy.

We consider the work of the genetic algorithm as an example of solving inverse kinematics problem for a three-link spatial manipulation system (Fig. 3). For a three-link manipulation robot, the chromosome will be a vector \( X = [x_1, x_2, x_3]^T \) whose elements are hinged coordinates, \( x_1 \) and \( x_2 \) – rotational, and \( x_3 \) – translational. For these coordinates, we introduce restrictions \( 0 \leq x_1 \leq 2\pi, -\pi/6 \leq x_2 \leq \pi/2, 0 \leq x_3 \leq 1 \). We compose analytical expressions for determining projections of radius vector of characteristic point of last (3rd) link of manipulation system in a fixed coordinate system associated with its base.

\[
\begin{bmatrix}
x_1 \\
x_2 \\
x_3
\end{bmatrix}
= \begin{bmatrix}
-L_2 + L_3 + x_1 \sin(x_1) \sin(x_2) \\
(L_2 + L_3 + x_3) \cos(x_1) \cos(x_2)
\end{bmatrix},
\]

where \( L_i \) – lengths of links, \( i=1-3 \).

Based on expressions (6) we compose a fitness function

\[
z = (y_1 - r_x)^2 + (y_2 - r_y)^2 + (y_3 - r_z)^2,
\]

where \( Y = [y_1, y_2, y_3]^T \) – vector of the Cartesian coordinates of point selected on the trajectory.

Let us set values of link lengths \( l_1=1.0000 \text{ m}, l_2=1.0000 \text{ m}, l_3=0.0000 \text{ m} \) (the third link extends telescopically from second) and coordinates of selected point \( y_1= 1.0000 \text{ m} \) – coordinate along \( x \) axis,


\[ y_2 = 0.0000 \text{ m} \] – coordinate along \( y \) axis, \( y_3 = 0.0000 \text{ m} \) – coordinate along \( z \) axis. Set population size to 30 individuals. Run the algorithm to execute. In the twentieth generation, the calculation result was: \( z = 1.2961 \times 10^{-07} \) – value of the fitness function for best individual, \( x_1 = 0.0000 \text{ rad} \), \( x_2 = 0.0000 \text{ m} \), \( x_3 = 1.0000 \text{ m} \) (figure 4).

![Figure 4. The results of calculating the fitness function](image)

3. Conclusion
As the result of computational experiment, the obtained values of generalized, coordinates corresponding to inverse problem solution of manipulating robots’ kinematics can be found with high accuracy by minimizing proposed fitness functions based on genetic algorithm. The calculation time for each test was about a few seconds, which allows to use this method in practice when programming manipulation robots by machine learning. Similar problems were considered in [1–10].

References
[1] Chernousko F L, Bolotnik N N, Gradetsky V G Manipulation Robots. Dynamics, Control, and Optimization. CRC Press, Boca Raton (1993).
[2] Castellet A, Thomas F An Algorithm for the Solution of Inverse Kinematics Problems Based on an Interval Method / Springer. Advances in Robot Kinematics: Analysis and Control, pp 393-402 (1998).
[3] Aristidou A, Lasenby J FABRIK: A fast, iterative solver for the Inverse Kinematics problem / Elsevier Inc. Graphical Models, no. 73(5), pp. 243-260 (2011).
[4] Filipiak P, Michalak K, Lipinski P A Predictive Evolutionary Algorithm for Dynamic Constrained Inverse Kinematics Problems / International Conference on Hybrid Artificial Intelligence Systems // Springer. Lecture Notes in Computer Science book series. Vol. 7208, pp 610-621 (2012).
[5] Xiao L, Zhang Y Solving time-varying inverse kinematics problem of wheeled mobile manipulators using Zhang neural network with exponential convergence / Springer. Nonlinear Dynamics. Vol. 76, pp 1543-1559 (2014).

[6] Köker R, Çakar T, Sari Y A neural-network committee machine approach to the inverse kinematics problem solution of robotic manipulators / Springer. Engineering with Computers. Vol. 30, pp 641-649 (2014).

[7] Asadi-Eydivand M, Ebadzadeh M M, Solati-Hashjin M, Darlot C, Osman N A A Cerebellum-inspired neural network solution of the inverse kinematics problem / Springer. Biological Cybernetics, Vol. 109, pp 561-574 (2015).

[8] Ren Z, Wang Z, Sun L A hybrid biogeography-based optimization method for the inverse kinematics problem of an 8-DOF redundant humanoid manipulator / Springer. Frontiers of Information Technology & Electronic Engineering, Vol. 16, pp 607-616 (2015).

[9] Iakovlev R, Denisov A, Prakapovich R Iterative Method for Solving the Inverse Kinematics Problem of Multi-link Robotic Systems with Rotational Joints / Proceedings of 14th International Conference on Electromechanics and Robotics “Zavalishin's Readings” // Springer. Smart Innovation, Systems and Technologies book series. Vol. 154, pp 237-251 (2019).

[10] Krakhmalev O Designing Object Diagrams and the Method of Structural Mutations in Models of Robots’ Manipulation Systems/ Proceedings of 14th International Conference on Electromechanics and Robotics “Zavalishin's Readings” // Springer. Smart Innovation, Systems and Technologies book series. Vol. 154, pp 209-221 (2019).