A genetic algorithm approach to a neural-network-based inverse kinematics solution of robotic manipulators based on error minimization

Raşit Köker *

Sakarya University, Technical Education Faculty, Electronics and Computer Sciences Department, 54187 Sakarya, Turkey
International University of Sarajevo, Faculty of Engineering and Natural Sciences, Computer Sciences and Engineering Department, 71000 Sarajevo, Bosnia and Herzegovina

ARTICLE INFO

ABSTRACT

The solution of the inverse kinematics problem is fundamental in robot control. Many traditional inverse kinematics problem solutions, such as the geometric, iterative, and algebraic approaches, are inadequate for redundant robots. Recently, much attention has been focused on a neural-network-based inverse kinematics problem solution in robotics. However, the precision of the result obtained from a neural network requires improvement for certain sensitive tasks. In this paper, neural network and genetic algorithms are used together to solve the inverse kinematics problem of a six-joint Stanford robotic manipulator to minimize the error at the end effector. The proposed hybrid approach combines the characteristics of neural networks and evolutionary techniques to obtain more precise solutions. Three Elman neural networks were trained using separate training sets because one of the sets yields better results than the other two. The floating-point portions of each network were placed in the initial population of the genetic algorithm with the floating-point portions from randomly generated solutions. The end-effector position error was defined as the fitness function, and the genetic algorithm was implemented. Using this approach, the floating-point portion of the neural-network result was improved by up to ten significant digits using a genetic algorithm, and the error was reduced to micrometer levels. These results were compared with those from studies in the literature and found to be significantly better.

© 2012 Elsevier Inc. All rights reserved.

1. Introduction

Robots are currently used in industrial and medical applications where high accuracy, repeatability, and stability of the operations are required [60]. As a result of developments in modern manufacturing technologies, micro-assembly is also becoming more important for certain sensitive tasks such as semiconductor processing and assembly, small parts assembly, and precision material processing. Micro-assembly involves the joining of parts that have at least one dimension that is less than one millimeter and that must be placed with micrometer accuracy. Although many industrial robots operate with high precision, research is ongoing in the development of robotic manipulators to further increase the precision, specifically in areas such as the design of grippers, actuator control systems, and vision systems for more precise robotic applications in micro-assembly [3,5,15,25,56]. For example, to accomplish micro-assembly applications, the hybrid parallel robot Micabo has been designed and constructed at the Institute of Machine Tools and Production Technology at TU-Braunschweg [15,56].

---

* Address: Sakarya University, Technical Education Faculty, Electronics and Computer Sciences Department, 54187 Sakarya, Turkey. Tel.: +90 532 408 18 72; fax: +90 264 295 64 24.

E-mail address: rkoker@sakarya.edu.tr

In parallel with these developments, the solution of the inverse kinematics problem, which is one of the most critical problems in robotics, must also be obtained with high precision. Robot kinematics handles the mapping between joint space (θ) and Cartesian space (x, y, z), where θ represents the positions of the joints of a robotic manipulator and (x, y, z) represent the position of the end effector of the manipulator. Furthermore, in many applications, the orientation also must be mapped with the (x, y, z) coordinates. The kinematics analysis is performed in two ways in robotics, with direct kinematics and inverse kinematics. Direct kinematics can be defined as the mapping from joint space to Cartesian space, whereas inverse kinematics is defined as the mapping from Cartesian space to joint space [52,63].

The inverse kinematics problem is usually more complex for redundant robots. Traditionally, three models are used to solve the inverse kinematics problem: the geometric [17,37] algebraic [12,20,41,50], and iterative [32] models. Each model contains certain disadvantages for solving the inverse kinematics problem. For instance, closed-form solutions are not guaranteed for the algebraic methods, and closed-form solutions for the first three joints of the robot must exist geometrically when the geometric method is used. Similarly, the iterative inverse kinematics solution method converges to only one solution that depends on the starting point. Additionally, these traditional solution methods may have a prohibitive computational cost because of the high complexity of the geometric structure of the robotic manipulators [31,35]. For these reasons, researchers have focused on solving the inverse kinematics problem using artificial neural networks.

Many papers have been published related to the neural-network-based inverse kinematics solution for robotic manipulators [4,10,24,26,27,31,33,35,42–44,48,58]. Tejomurtula and Kak [58] proposed a solution for the inverse kinematics problem for a three-joint robotic manipulator based on structured neural networks that can be trained quickly to overcome the disadvantages of the backpropagation algorithms, such as training time and accuracy. Karlık and Aydın presented a study of the inverse kinematics solution for a six-joint robot based on identifying the best neural-network configuration. This group found that designing six separate neural networks with two hidden layers for each output yields a better solution than designing a single neural network with six outputs [31]. Oyama and colleagues [48] presented modular neural network architectures for learning in the inverse kinematics model. Their method is based on DeMers' method, which involves a number of experts, an expert selector, an expert generator, and a feedback controller that can accommodate the nonlinearities in the kinematic system. Their method contains certain limitations; for example, the inverse kinematics computation procedure is highly complex, and the learning speed is low. However, root mean square (RMS) hand position errors of less than 10 mm were achieved. Mayorga et al. [44] presented a novel neural-network approach for fast inverse kinematics calculations and efficient singularity avoidance or prevention for redundant robots based on a set of bounded geometrical concepts used to determine the characteristic matrices. Their algorithm enables the implementation of fast and robust real-time algorithms for safe robot path generation. A neural-network approach using the backpropagation algorithm was presented by Bingul and Ertunc [4] for the inverse kinematics solution of an industrial robotic manipulator without an analytical inverse kinematics solution. The disadvantages of their approach are the large errors in the joint angles and the inability of this approach to provide multiple solutions to the inverse kinematics problem. Köker proposed the use of parallel neural networks to compute the inverse kinematics of the robots. The main aim in this proposed method was to select the best output of the networks in the parallel block to obtain the most reliable solution for a 6-degree-of-freedom (dof) robot [35]. Hasan et al. presented an inverse kinematics solution for a 6-dof robotic manipulator [26,27]. In their first study [26], an adaptive learning strategy using an artificial neural network was proposed to control the motion of a 6-dof robotic manipulator and to overcome difficulties in solving the inverse kinematics problem such as singularities and uncertainties. In their approach, a network was trained to learn a desired set of joint angle positions from a set of given end-effector positions, and the experimental results showed good mapping over the working area of the robot. After 8×10^6 iterations, the absolute error percentages for joints 1–6 were 3.635%, 3.66%, 5.31%, 1.73%, 3.435%, and 6.1%, respectively. The researchers also provided a graphical presentation of these errors by iteration number. They published a second paper [27] based on using artificial neural networks to learn robot system characteristics rather than specifying an explicit robot system model to overcome singularities and uncertainties; this method was implemented for another type of 6-dof robotic manipulator model. After 3×10^5 iterations, the total error percentages for the test dataset for joints 1–6 were 0.915%, 0.135%, 0.57%, 4.79%, 4.81%, and 1.11%, respectively.

Researchers have also focused on using genetic algorithms to obtain inverse kinematics solutions for robots [11,30,46,47,57]. Kalra and colleagues [30] used an evolutionary approach based on a single-level real coded genetic algorithm to solve the multimodal inverse kinematics problem for industrial robots. In their method, the fitness function is defined in a manner that requires separate evaluation of the positional error of the robot and the total joint displacement. They obtained an acceptable distribution of population members around the multiple inverse kinematics solutions using a niche strategy for the binary tournament selection operator and mating restriction. The simulation experiments were performed on a SCARA robot and a PUMA robot using the niche strategies proposed by Oei et al. [47] and Deb and Agrawal [11]. This group also presented minimum positioning errors of the wrist for the obtained values of the joint variables by using these two niche strategies to construct a table. The position errors obtained from the niche-strategy simulations described by Oei et al. [47] and Deb and Agrawal [11] for the SCARA robot were in the ranges of 0.40–2.19 mm and 0.14–0.69 mm, respectively. The minimum positioning errors obtained using the niche strategy proposed by Deb and Agrawal [11] were less than those obtained using the niche strategy of Oei et al. [47]. Furthermore, in Kalra and colleagues' paper [30], the positioning errors for the PUMA robot using the niche strategy of Deb and Agrawal [11] were also within the range of 0.36–1.75 mm. During the simulation studies, the optimum solutions for the SCARA robot were obtained after 100 generations, and the optimum solutions for the PUMA robot were obtained after only 300 generations. Tabandeh et al. [57] presented a genetic algo-