Solving the forward kinematics problem in parallel manipulators using an iterative artificial neural network strategy

Pratik J. Parikh · Sarah S. Lam

Abstract In this paper we address the forward kinematics problem of a parallel manipulator and propose an iterative neural network strategy for its real-time solution to a desired level of accuracy. Parallel manipulators are closed kinematic structures that possess requisite rigidity to yield a high payload to self-weight ratio. Because of this unique feature, they have been employed in manufacturing, flight simulation systems, and medical robotics. However, it is this closed kinematic structure that has led to difficulty in their kinematic control, especially the forward kinematics control. The iterative neural network strategy we propose employs a trained neural network and an error compensation algorithm in the feedback loop. The proposed strategy was tested with data from a real-world flight simulation system. Results show that solutions with a maximum error in the position and orientation parameters of 0.25 mm and 0.01°, respectively, can be achieved in less than five iterations (or about 1 second). Because of the nature of this strategy, it is possible to implement it in a hardware form, which can result in a multi-fold reduction in the solution time for the same accuracy level.

Keywords Kinematics · Neural networks · Parallel manipulators · Gough–Stewart platform

1 Introduction

Manipulators have been employed for operations that include picking and placing objects, imitating a human movement, and operation in hazardous regions. Typically, a (robotic) manipulator is a mechanism consisting of kinematic links or chains. This assemblage of links could exist in two basic designs: open loop and closed loop mechanisms. Open loop mechanisms (structures) are formed through a series of kinematic links connected end-to-end (e.g., serial manipulators) and represent most of today’s robotic manipulators with structures similar to human arm, as shown in Fig. 1(b). Closed loop mechanism consists of a mobile platform with an end-effector that is connected to the base through three or more articulated links as shown in Fig. 1(a) [1]. A few synonyms of parallel manipulators are parallel robots, parallel kinematic machines or hexapods, which all are closed-loop kinematic structures. A linear actuator within each link enables link length changes. The link lengths are changed to control the final position and orientation of the mobile end-effector.

Applications that require high rigidity, precision in final positioning, fast dynamic response, and large load to weight ratios are best suited for parallel manipulators. Specific application domains include motion base in flight simulation systems and a variety of medical and manufacturing applications.

One of the key research issues in this domain is solving the kinematic problems. These problems deal with establishing the relationship between the mechanism’s position, velocity, and acceleration and the mechanism parameters like link lengths and angles. There are two types of kinematic problems that exist for any serial or parallel manipulator: inverse kinematics problem (IKP) and forward kinematics problem (FKP). The IKP refers to mapping the task space to
the joint space; i.e., finding the joint angles or link lengths given the final position and orientation of the manipulator. Mathematically, the mapping function \( f \) for IKP is represented function as \( f : \{ \text{task space} \} \rightarrow \{ \text{joint space} \} \). The FKP describes the final position and orientation based on a given set of joint angles or lengths, thus, mapping the joint space to the task space. Mathematically, the mapping function for FKP is represented as \( f : \{ \text{joint space} \} \rightarrow \{ \text{task space} \} \). Though the IKP for a parallel manipulator has a closed-form mathematical solution, the FKP lacks one. This is because the FKP represents, in general, an under-defined problem, where the number of equations is less than the number of unknowns. Moreover, the high degree of non-linearity in the kinematic formulations adds to the complexity. In this direction, several researchers have focused on developing strategies to solve the FKP. Our focus in this paper is to solve the FKP for parallel manipulators using a neural network-based iterative strategy.

The rest of the paper is organized as follows. In Sect. 2 we summarize various approaches that have been proposed to-date for solving the FKP. In Sect. 3 we provide a brief overview of neural networks, which lays the basis of Sect. 4 in which we introduce the iterative neural network strategy. In Sect. 5 we briefly summarize the system we considered to test our proposed strategy. In Sect. 6 we demonstrate the applicability of our strategy in effectively and efficiently solving the FKP. Finally, in Sect. 7 we present our conclusions.

2 Literature review

The FKP in parallel manipulators has been approached by many researchers in the past. However, due to the lack of an effective solution strategy, efforts in this direction are still continuing. Four classes of methodologies have been employed: analytical approaches, use of additional sensors or transducers, numerical methods, and neural network-based approaches.

The analytical approaches to solving the FKP are as follows: (a) a system of nonlinear equations can be formulated and converted to a system of kinematic equations to a higher degree polynomial with one unknown, and then solved using a numerical technique like Newton–Raphson method, etc. [2, 3, 4], (b) the position and orientation parameters can be decoupled, thereby, reducing the complexity and speeding up the solution process [5], or (c) a quaternion approach can be employed to obtain a closed-form solution for certain class of parallel manipulators [6]. Other analytical approaches that have been suggested include continuation method [7], elimination method [8, 9], and interval analysis [10]. Zhao et al. [11] propose a forward kinematics model with natural coordinates for the Gough–Stewart manipulator and other spatial parallel mechanisms. The main advantage of this method is that the derivative matrix of the constraint equations only consists of linear or constant elements, which are relatively easy to solve. The authors indicated that their approach can achieve real-time solutions, but they did not report the computation time. It is worth noting that most of these analytical approaches were devised for special configurations of parallel manipulators, and that it was not clearly shown how to generalize these approaches to other configurations of parallel manipulators. Moreover, for some of these approaches the computation times are not suitable for real-time operation. For example, Merlet’s [10] interval analysis approach requires at least 3–4 seconds on a 16 cluster 2 GHz computer system, while on a single computer it is above 100 seconds.

Installation of additional sensors or transducers on the manipulator itself was proposed to obtain more information about the system (manipulator) state [12]. The additional cost of the sensors has made this the least desirable approach to solving the FKP. To address this issue of generalization, numerical approaches have been proposed, whereby the kinematics problem is formulated such that it could be solved using any of the available numerical techniques like the Newton–Raphson method [13, 14]. However, a lack of