



# Forward kinematics analysis of a novel 3-DOF parallel manipulator

X. Wu<sup>a,b,\*</sup> and Z. Xie<sup>a</sup>

a. *State Key Laboratory of Mechanical Transmission, Chongqing University, Chongqing 400044, China.*

b. *Department of Mechanical and Manufacturing Engineering, Aalborg University, Aalborg 9200, Denmark.*

Received 14 April 2017; received in revised form 24 July 2017; accepted 6 January 2018

## KEYWORDS

Parallel manipulator;  
 Kinematics analysis;  
 Neural network;  
 Ant colony  
 optimization;  
 Newton iterative  
 method.

**Abstract.** A novel spatial parallel manipulator designed to assemble diagnostic instruments in SG-III is introduced in this paper. Firstly, resorting to screw theory, mobility analysis is presented for this manipulator. Then, the inverse kinematics problem is determined by the method of RPY transformation with the singularity analyzed. As a key issue in parallel manipulators, it is more difficult to solve the forward kinematics problem, since it is highly nonlinear and coupled. In this work, three different approaches are presented to deal with this issue, namely, the back propagation neural network, the simplified ant colony optimization, and the proposed improved Newton iterative method. Simulation of each approach is conducted, and their merits and demerits are compared in detail. It is concluded that the improved Newton iterative method, which can provide good initial iteration values, shows the best performance in estimation of the nonlinear forward kinematic mapping of the considered parallel manipulator.

© 2019 Sharif University of Technology. All rights reserved.

## 1. Introduction

Parallel Manipulators (PMs) are closed-loop mechanical structures composed of a Moving Platform (MP) coupled to a fixed base by serial limbs. Compared with serial manipulators, PMs own the advantages of higher mechanical rigidity, lighter weight, larger load-weight ratio, better orientation precision, suitable positional actuators arrangement, and stable capacity [1-3] and they have drawn the attention of many researchers and industries during the past decades.

As well known, PMs with 6-DOF have many advantages; but, 6-DOF is not always needed in many industrial applications and PMs with less DOF will be suitable instead. Many PMs with less 3-DOF have been presented in recent years, such as Delta

and Tsai mechanisms with three pure translational DOFs [3,4], T2R1-type (two translations and one rotation) PMs providing planar motion [2,5], T1R2-type spatial PMs [6-8], spherical PMs with three pure rotational DOFs [9,10], etc.

Sufficient efforts have been put to kinematics analysis, as a key issue in PMs, which includes inverse and forward kinematics. Inverse Kinematics (IK) analysis of PMs involves mapping a known pose of the end effector on the input of each actuator, which is usually straightforward and simple, while the Forward Kinematics (FK) problem is highly nonlinear and coupled [11,12]. Generally, traditional methods for solving FK problem can be divided into two major categories, i.e., the closed-form solutions [6,13-17] and the numerical approaches [7,18-24]. The mainly used closed-form solutions include analytical method [6], Gröbner basis method [13], algebraic elimination method [14], dual quaternions method [15], Geometric approach [16], vector approach [17], etc. These methods need to build a complicated mathematical model, while the

\*. *Corresponding author.*

*E-mail address: wuxycqu@163.com (X. Wu)*

computational process is time-consuming. Moreover, their generalization is not very good due to the configuration difference of PMs, which means they could not be applied to all PMs. To address these issues, numerical approaches are proposed, among which the Newton iterative method [7,21] is the most commonly employed one. With the development of intelligence algorithms, such as artificial neural network [12,18–20], genetic algorithm [22], particle swarm optimization [23], support vector regression [24], simulated annealing algorithm [5], etc., they have also been taken into account for FK analysis.

In order to find a method with good performance to solve the FK problem of the proposed novel PM, several approaches are taken into consideration, which include the Back Propagation Neural Network (BPNN), the simplified Ant Colony Optimization (ACO), and the improved Newton iterative method. Comparisons among these approaches are analyzed in detail.

The rest of this paper is organized as follows. Section 2 contains a brief description of the novel 3-DOF PM. The kinematics analyses, including mobility, inverse kinematics, forward kinematics, and singularity, are presented in Section 3. Three approaches, namely, the BPNN, the simplified ACO, and the improved Newton iterative method, are taken into consideration for FK analysis in Sections 4 to 6, respectively. Conclusions are provided in the final section.

## 2. Manipulator description

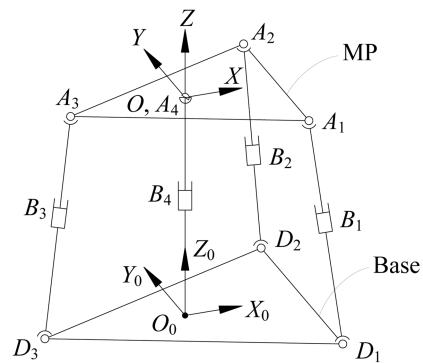
SG-III is one of the largest Inertial Confinement Fusion (ICF) facilities in the world, which is a football-stadium-size 48-beam laser constructed to create fusion conditions with controllable laboratory conditions [25]. As shown in Figures 1 and 2, a precision assembly platform is designed to install, adjust, and uninstall a series of diagnostic instruments in the ICF facility, and this assembly platform is mainly made up of three components, namely, the special fixture,



**Figure 1.** The assembly operation of diagnostic instruments.



**Figure 2.** The platform for the assembly of diagnostic instruments.



**Figure 3.** Schematic diagram of the 3-SPS/PU PM.

the 3-SPS/PU PM, and the omnidirectional vehicle.

Having a closed-loop structure, as depicted in Figure 3, the considered 3-SPS/PU manipulator consists of three identical SPS kinematic chains and one PU kinematic chain, all connecting the MP to the base. Here, S, P, and U represent the spherical, prismatic, and universal joints, respectively, and the three prismatic joints in the SPS kinematic chains are chosen as the active joints. For the purpose of kinematic analysis, the fixed frame  $\{O_0\}$  is attached to the isosceles triangle base with its origin located at the midpoint of the vertical line of  $D_1D_2$ ,  $X_0$ -axis is pointed along vector  $\overrightarrow{D_3O_0}$  and  $Y_0$ -axis is pointed in the direction of vector  $\overrightarrow{D_1D_2}$ , and the local frame  $\{O\}$  is attached to the MP and defined in a similar way.

## 3. Kinematics analyses

In the kinematics analyses, mobility, inverse kinematics, forward kinematics, and singularity are considered.

### 3.1. Mobility analysis

As well known, it is convenient to calculate the number of DOFs of common spatial mechanisms by traditional methods, but they cannot indicate the properties of the DOFs, i.e., whether they are translational or rotational DOFs [7]. Therefore, screw theory [6] is applied to analyze the mobility of the proposed 3-SPS/PU PM.

Since the three SPS kinematic chains have identical topologies, we only take the first SPS kinematic

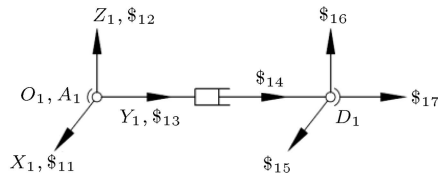


Figure 4. Kinematic chain of SPS limb.

chain for an example. Local frame  $\{O_1\}$  is attached to spherical joint  $A_1$ , as described in Figure 4. Here, complex joints can be replaced with some basic joints with single DOF, such as  $S \equiv \overline{RRR}$ ,  $U \equiv \overline{RR}$ , where  $R$  is a revolute joint. Then, all the kinematic pairs are represented with screws, and the kinematic screw system of the first SPS limb under the local frame can be expressed as follows:

$$\begin{cases} \$11 = (1, 0, 0; 0, 0, 0) \\ \$12 = (0, 0, 1; 0, 0, 0) \\ \$13 = (0, 1, 0; 0, 0, 0) \\ \$14 = (0, 0, 0; 0, 1, 0) \\ \$15 = (1, 0, 0; 0, 0, c) \\ \$16 = (0, 0, 1; a, 0, 0) \\ \$17 = (0, 1, 0; 0, 0, 0) \end{cases} \quad (1)$$

where  $a$  and  $c$  are constants, which can be neglected.

It is easy to observe that reciprocal screws of the SPS limb do not exist, which means the limb does not exert any constraint on the MP. Employing the same method for the PU limb, it can be found that there exist two constraint forces passing through the universal joint center parallel to axes  $X_0$  and  $Y_0$ . There also exists a constraint couple parallel to axis  $Z_0$ . In other words, the rotation motion about axis  $Z_0$  and translation motion along axes  $X_0$  and  $Y_0$  of the MP are constrained.

To sum up, there does not exist any common constraint between the PU limb and the SPS limbs. Thus, common constraint  $\lambda$  of the considered PM is zero, i.e., this mechanism is still a six-order screw system. The rotation of the SPS limb about its own axis does not affect the movement of the MP, which means it will produce a passive DOF. Usually, an SPS limb could be replaced with an UPS non-redundant limb. However, note that all the six spherical joints in this work are the same, which makes little batch manufacturing possible; moreover, interchangeability of the system will also increase. Thus, SPS limbs are suitable for the proposed PM.

The number of DOFs of this PM can be derived according to the modified G-K criterion.

$$M = d(n - g - 1) + \sum_{i=1}^g f_i + v - \xi. \quad (2)$$

Here,  $M$  is the number of DOFs of the mechanism,  $d$

stands for the order of the mechanism,  $n$  is the total number of components,  $g$  represents the number of joints,  $f_i$  is the DOF of the  $i$ th joint,  $v$  denotes the number of redundant constraints, and  $\xi$  represents the passive DOF. In the proposed PM,  $d = 6$ ,  $n = 9$ ,  $g = 11$ ,  $v = 0$ , and  $\xi = 3$ . Accordingly, it is calculated that  $M = 3$ , i.e., this PM only possesses three DOFs on account of the constraint of the PU limb; these DOFs include one translation along  $Z_0$ -axis and two rotations about  $X_0$ -axis and  $Y_0$ -axis.

### 3.2. Inverse kinematics

As with the proposed mechanism, the IK problem involves mapping a known pose (position and orientation) of the MP on the actuated input of each SPS limb. It can be efficiently determined by the method of RPY (roll, pitch, and yaw) transformation, which is usually applied to describe the IK problem of PMs. Here, the given pose  $(\alpha, \beta, z)$  represents the pose of the MP with respect to the fixed frame. Accordingly, the position vector of point  $A_i$  in the fixed frame is:

$$\mathbf{a}_i = \mathbf{R}\mathbf{a}'_i + \mathbf{p}, \quad i = 1, 2, 3, \quad (3)$$

where  $\mathbf{a}'_i$  is the position vector of point  $A_i$  in the local frame,  $\mathbf{R}$  is the rotation matrix, and  $\mathbf{p}$  is position vector of the operation point  $A_4$ .

$$\mathbf{a}'_1 = [d_1, -d_2, 0]^T,$$

$$\mathbf{a}'_2 = [d_1, d_2, 0]^T,$$

$$\mathbf{a}'_3 = [-d_1, 0, 0]^T,$$

$$\mathbf{p} = [0, 0, z]^T,$$

$$\mathbf{R} = \mathbf{R}(y, \beta) \mathbf{R}(x, \alpha)$$

$$= \begin{bmatrix} \cos \beta & \sin \beta \sin \alpha & \sin \beta \cos \alpha \\ 0 & \cos \alpha & -\sin \alpha \\ -\sin \beta & \cos \beta \sin \alpha & \cos \beta \cos \alpha \end{bmatrix}.$$

Next, the IK problem of the considered mechanism can be expressed as follows:

$$l_i = \|\mathbf{a}_i - \mathbf{d}_i\| = f_i(\alpha, \beta, z), \quad i = 1, 2, 3, \quad (4)$$

where  $\mathbf{d}_i$  is the position vector of point  $D_i$  in the fixed frame.

$$\mathbf{d}_1 = [d_3, -d_4, 0]^T,$$

$$\mathbf{d}_2 = [d_3, d_4, 0]^T,$$

$$\mathbf{d}_3 = [-d_3, 0, 0]^T.$$

By expanding Eq. (4), the IK solutions own a more detailed description as in Eqs. (5) shown in Box I.

In this case,  $d_i$  is determined by geometrical constraints and design dimensions, where  $d_1 = 750$  mm,  $d_2 = 215$  mm,  $d_3 = 780$  mm, and  $d_4 = 260$  mm.

$$l_1 = \sqrt{(d_1 \cos \beta - d_2 \sin \beta \sin \alpha - d_3)^2 + (d_4 - d_2 \cos \alpha)^2 + (z - d_1 \sin \beta - d_2 \cos \beta \sin \alpha)^2}, \quad (5a)$$

$$l_2 = \sqrt{(d_1 \cos \beta + d_2 \sin \beta \sin \alpha - d_3)^2 + (d_2 \cos \alpha - d_4)^2 + (z - d_1 \sin \beta + d_2 \cos \beta \sin \alpha)^2}, \quad (5b)$$

$$l_3 = \sqrt{(d_3 - d_1 \cos \beta)^2 + (z + d_1 \sin \beta)^2}. \quad (5c)$$

## Box I

**3.3. Forward kinematics**

Contrary to IK problem, the FK problem needs to determine the pose of the MP with given actuated inputs. IK analysis is the foundation for further study of FK problem. In order to get the closed-form solution to FK, algebraic elimination method is considered.

By the pose adjusting requirement, the motion ranges of the MP are indicated as Eq. (6); the motion ranges of  $X_0$  and  $Y_0$ -rotations are about  $\pm 5^\circ$ , while  $Z_0$ -translation varies from 365 mm to 485 mm.

$$\begin{cases} -5^\circ < \alpha < 5^\circ \\ -5^\circ < \beta < 5^\circ \\ 365 \text{ mm} < z < 485 \text{ mm} \end{cases} \quad (6)$$

It can be obtained from the above that  $z + d_1 \sin \beta > 0$ , and in view of Eq. (5c), we have:

$$z = \sqrt{l_3^2 - (d_3 - d_1 \cos \beta)^2} - d_1 \sin \beta. \quad (7)$$

Referring to Eqs. (5a) and (5b), the following equation can be obtained

$$l_1^2 - l_2^2 = 4d_2 \sin \alpha (d_3 \sin \beta - z \cos \beta). \quad (8)$$

Then, substituting Eq. (7) into Eq. (8) gives Eq. (9) as shown in Box II.

Since  $-5^\circ < \alpha < 5^\circ$ , we achieve:

$$\cos \alpha = \sqrt{1 - \sin^2 \alpha}. \quad (10)$$

Now, both  $\alpha$  and  $z$  can be eliminated and analytically expressed by  $\beta$ . In the following, by substituting Eqs. (7), (9), and (10) into Eq. (5a), we can obtain a very huge and complex transcendental equation for  $\beta$  and  $l_i$ . Obviously, it is difficult to solve  $\beta$  from this transcendental equation, which means it is impractical to get a closed-form solution to the FK problem of the considered PM.

**3.4. Singularity analysis**

For general PMs, singularities are configurations where the MP gains or loses DOF [26]. In order to determine these configurations, the relationship between the velocity of the MP and the velocities of the actuators should be identified. As shown in Figure 3, the position vector of the operation point  $A_4$  can be expressed as:

$$\mathbf{p} = \mathbf{d}_i + l_i \mathbf{n}_i - \mathbf{R} \mathbf{a}'_i, \quad i = 1, 2, 3, \quad (11)$$

where  $\mathbf{n}_i$  is the unit vector parallel to the translation direction of corresponding prismatic joint. Differentiating both sides of Eq. (11) leads to:

$$\dot{\mathbf{p}} = \boldsymbol{\omega}_i \times l_i \mathbf{n}_i + \dot{l}_i \mathbf{n}_i - \boldsymbol{\omega}_p \times (\mathbf{R} \mathbf{a}'_i), \quad i = 1, 2, 3, \quad (12)$$

where  $\dot{\mathbf{p}} = [0, 0, \dot{z}]^T$  is the translational velocity of the operating point on the MP,  $\boldsymbol{\omega}_i$  is the angular velocity of limb  $i$ , and  $\boldsymbol{\omega}_p = [\dot{\alpha}, \dot{\beta}, 0]^T$  is the angular velocity of the MP. Dot multiplication by  $\mathbf{n}_i$  yields:

$$\dot{l}_i = \mathbf{n}_i \cdot \dot{\mathbf{p}} + \mathbf{n}_i \cdot (\boldsymbol{\omega}_p \times (\mathbf{R} \mathbf{a}'_i)), \quad i = 1, 2, 3. \quad (13)$$

Let  $\dot{\mathbf{x}} = [\dot{z}, \dot{\alpha}, \dot{\beta}]^T$  and  $\dot{\mathbf{q}} = [\dot{l}_1, \dot{l}_2, \dot{l}_3]^T$  be the velocity vectors of the MP and the actuated joints, respectively. Accordingly, Eq. (13) can be rewritten in a matrix form as:

$$\mathbf{J}_q \dot{\mathbf{q}} = \mathbf{J}_x \dot{\mathbf{x}}, \quad (14)$$

where  $\mathbf{J}_x$  and  $\mathbf{J}_q$  are the direct and inverse Jacobian matrices, respectively:

$$\mathbf{J}_q = \mathbf{E}_{4 \times 4}, \quad (15)$$

$$\mathbf{J}_x = \begin{bmatrix} \mathbf{n}_1 \mathbf{k} & (\mathbf{R} \mathbf{a}'_1 \times \mathbf{n}_1) \cdot \mathbf{i} & (\mathbf{R} \mathbf{a}'_1 \times \mathbf{n}_1) \cdot \mathbf{j} \\ \mathbf{n}_2 \mathbf{k} & (\mathbf{R} \mathbf{a}'_2 \times \mathbf{n}_2) \cdot \mathbf{i} & (\mathbf{R} \mathbf{a}'_2 \times \mathbf{n}_2) \cdot \mathbf{j} \\ \mathbf{n}_3 \mathbf{k} & (\mathbf{R} \mathbf{a}'_3 \times \mathbf{n}_3) \cdot \mathbf{i} & (\mathbf{R} \mathbf{a}'_3 \times \mathbf{n}_3) \cdot \mathbf{j} \end{bmatrix} \quad (16)$$

$$\sin \alpha = \frac{l_1^2 - l_2^2}{4d_2(d_3 \sin \beta - (\sqrt{l_3^2 - (d_3 - d_1 \cos \beta)^2} - d_1 \sin \beta) \cos \beta)}. \quad (9)$$

## Box II

As reported in [27], three different types of singularities are identified. The first type of singularities, termed the inverse kinematic singularities, occurs when  $\mathbf{J}_q$  is not invertible. The second type, called direct kinematic singularities, arises when  $\mathbf{J}_x$  is not invertible. The third type of singularities, termed the combined singularities, occurs when both  $\mathbf{J}_x$  and  $\mathbf{J}_q$  are singular. Obviously, there only exist direct kinematic singularities since  $\det(\mathbf{J}_q) = 1$ . The undesired singularities can occur when  $\mathbf{J}_x$  is not a full-rank matrix, and some singular configurations are derived in the following cases:

1. When the MP falls on top of the base, i.e.,  $\mathbf{n}_i \mathbf{k} = 0$  ( $i = 1, 2, 3$ ), the first column of the matrix  $\mathbf{J}_x$  will vanish, while the MP gains one DOF. The MP can still move along the  $Z_0$ -axis even when all the actuators are locked. This singular configuration is shown in Figure 5(a);
2. When  $D_2$  coincides with  $D_1$  and  $\alpha = \pm 90^\circ$ , as shown in Figure 5(b), the MP can still move along the  $Y_0$ -axis even without actuating the actuators;
3. When one of the vectors  $\mathbf{a}_i$  ( $i = 1, 2, 3$ ) is located in the base plane, as shown in Figure 5(c), one row of the matrix  $\mathbf{J}_x$  will vanish, which yields  $\det(\mathbf{J}_x) = 0$ ;
4. When two of the vectors  $\mathbf{n}_i$  ( $i = 1, 2, 3$ ) concur with the common point  $A_4$ , as shown in Figure 5(d), we have  $\mathbf{R}\mathbf{a}'_i \times \mathbf{n}_i = 0$ . Accordingly, two rows of the last two columns of the matrix  $\mathbf{J}_x$  will vanish; the rank of the matrix  $\mathbf{J}_x$  will be smaller than three;
5. When  $d_1/d_3 = d_2/d_4$ , while the MP and the base are parallel, the MP can still perform a small rotation about the instantaneous point  $G_2$ , even when all the actuators are locked, as depicted in Figure 5(e);

6. When the MP and the base are congruent and parallel to each other, as shown in Figure 5(f), the MP can still move along the direction norm to  $Z_0$ -axis even with all actuators locked.

Therefore, in the above six cases, singularities of the considered 3-SPS/PU PM occur. Moreover, the workspace expressed by Eq. (6) is free from singularity.

#### 4. BPNN for forward kinematics

Neural Network (NN) is an adaptive and intelligent algorithm inspired by human brain, which can approximate an unknown system within numerous interconnected neurons. It possesses many useful capabilities and properties: input-output mapping, nonlinearity, adaptability, fault tolerance, evidential response, and uniformity of analysis and design [18]. The well-known NN is BPNN, which is trained with supervision, and gradient-descent technique is employed to minimize the error. BPNN involves two passes through the network; the network's output activities are generated by the forward pass and the backward pass propagates the error initially found in the output nodes back through the network to assign error to each node that contributed to the initial error [19]. BPNN also owns strong nonlinearity mapping ability, great self-studying and self-adjusting capability, and excellent generalization performance. It is considered as a universal approximator, essential for function-approximation problems.

Usually, learning samples should be prepared before determining the structure of BPNN. In this work, learning samples are comprised of abundant input-output sets; the inputs  $\mathbf{q} = [l_1, l_2, l_3]^T$  are the actuated inputs to the SPS limbs and the outputs

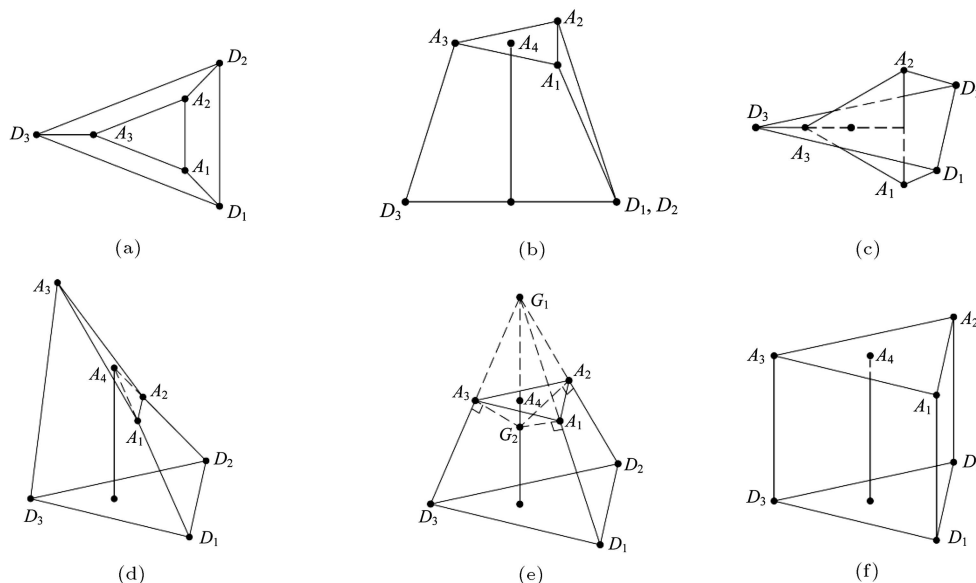
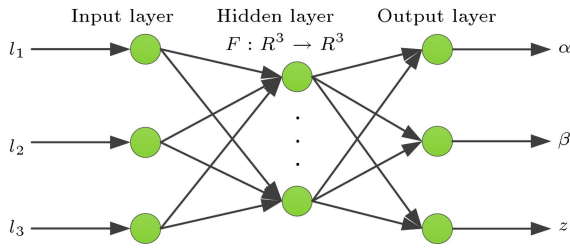


Figure 5. Singular configurations of the 3-SPS/PU PM.



**Figure 6.** Structure of BPNN with one hidden layer.

$\mathbf{x} = [z, \alpha, \beta]^T$  are the pose of the MP. All the samples can be derived directly from the IK solutions.

It can be learnt from Eq. (6) that the orders of magnitude of different data are not the same, which may cause great errors in the predictions; therefore, min-max normalization processing is carried out as

$$x'_r = \frac{x_r - x_{\min}}{x_{\max} - x_{\min}}, \quad (17)$$

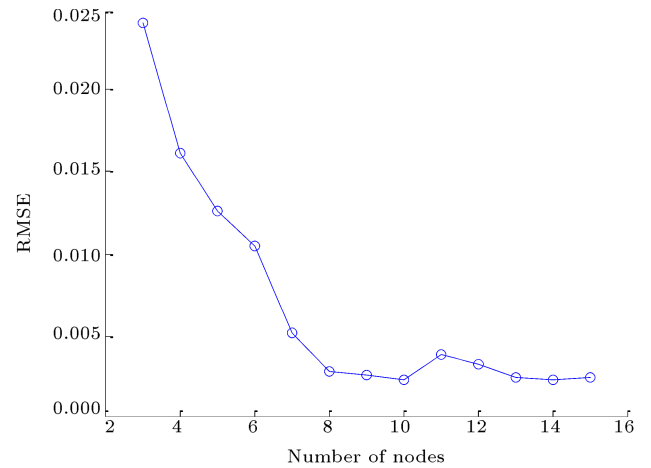
where  $x_r$  is the original value,  $x_{\min}$  and  $x_{\max}$  are the minimum and maximum values of the data, and  $x'_r$  is the value after normalization processing, which belongs to the interval  $[0, 1]$ .

As soon as learning samples are ready, we proceed to choose the structure of BPNN. Generally, BPNN will perform better with more hidden layers, but it is inevitably time-consuming; moreover, it is difficult to determine the detailed structure, so we just take BPNN with one hidden layer into account, as depicted in Figure 6. Since there is no theoretical method to determine the number of nodes in the hidden layer, empirical formula and trial and error method [20] will be employed to deal with this issue, and following rules are taken into consideration.

$$\begin{cases} H = O + 0.75I \\ H < 2I \end{cases} \quad (18)$$

where  $H$ ,  $I$ , and  $O$  are the numbers of nodes in the hidden layer, the input layer, and the output layer, respectively.

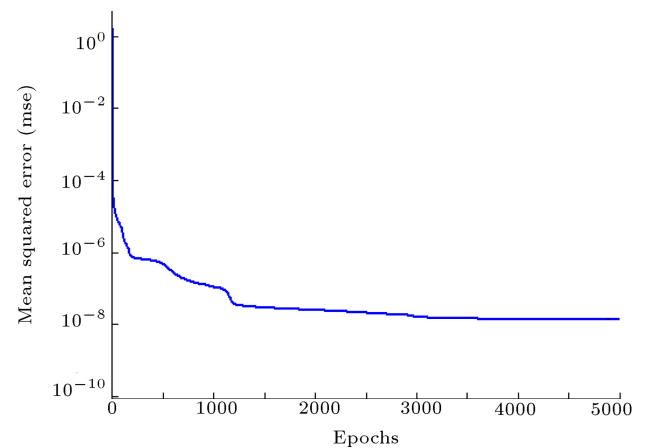
It is noticeable that the result calculated from Eq. (18) is used as a reference, so BPNNs with different nodes in the hidden layer should also be tested; their performance is represented by root mean square error (RMSE) as depicted in Figure 7. The BPNN with the best performance, as indicated, would be selected, in which the network with fewer hidden layer nodes will be the best choice, since the number of weights as well as the training time of the network will increase with more nodes in the hidden layer [12]. In this case, we choose 1500 samples for network training and 100 samples for validating, and the RMSE performance is obtained after 4000 training cycles. Moreover, tangent sigmoid function is used as activation function of the hidden layer and linear function for the output layer,



**Figure 7.** RMSE under different nodes in the hidden layer.

and Levenberg-Marquardt algorithm is employed to adjust the weights of the network. It can be learnt from Figure 7 that the network with 10 nodes in the hidden layer is the best choice.

Since the structure is determined, the prediction accuracy of BPNN mainly relies on the size of the learning samples. In this case, the tested input sample  $\mathbf{q}_0 = [374.338875 \text{ mm}, 389.064158 \text{ mm}, 483.366307 \text{ mm}]^T$  is derived directly from IK solutions corresponding to the given pose  $\mathbf{x}_0 = [2^\circ, 4^\circ, 430 \text{ mm}]^T$  of the MP. As shown in Table 1, aiming at the tested sample  $\mathbf{q}_0$ , the accuracy of the outputs solved by different sample sizes is compared. It is found that prediction error decreases with increment of the sample number, and it tends to be relatively steady when the sample number reaches about 1500. Since the accuracy will not increase significantly and the computations are inevitably time-consuming as sample number increases, 1500 learning samples are enough for network training. The prediction accuracy appears to be relatively stable after about 4000 learning iterations under this situation, as depicted in Figure 8. In this case, running the



**Figure 8.** Training performance of BPNN.

**Table 1.** Results of BPNN with different sample sizes.

Sample number	Calculated results			Absolute error		
	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)
300	2.006529	4.002701	430.12277	$6.5 \times 10^{-3}$	$2.7 \times 10^{-3}$	$1.2 \times 10^{-1}$
600	1.999497	4.002852	430.024402	$5.0 \times 10^{-4}$	$2.9 \times 10^{-3}$	$2.4 \times 10^{-2}$
1000	1.999932	3.999797	430.005106	$6.8 \times 10^{-5}$	$2.0 \times 10^{-4}$	$5.1 \times 10^{-3}$
1500	2.000063	3.999890	430.003115	$6.3 \times 10^{-5}$	$1.1 \times 10^{-4}$	$3.1 \times 10^{-3}$
2000	1.999660	3.999552	430.004254	$3.4 \times 10^{-4}$	$4.5 \times 10^{-4}$	$4.3 \times 10^{-3}$

**Table 2.** Results of BPNN for FK.

No.	Target values			Calculated results			Absolute error		
	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)
1	0	2	400	0.000444	1.999810	400.006470	$4.4 \times 10^{-4}$	$1.9 \times 10^{-4}$	$6.5 \times 10^{-3}$
2	1	3	415	1.000500	2.999625	415.005649	$5.0 \times 10^{-4}$	$3.8 \times 10^{-4}$	$5.6 \times 10^{-3}$
3	2	3	435	2.000254	2.999745	434.998385	$2.5 \times 10^{-4}$	$2.6 \times 10^{-4}$	$1.6 \times 10^{-3}$
4	3	5	445	3.002727	4.999393	444.998175	$2.7 \times 10^{-3}$	$6.1 \times 10^{-4}$	$1.8 \times 10^{-3}$
5	5	5	460	4.999837	4.999585	459.996612	$1.6 \times 10^{-4}$	$4.2 \times 10^{-4}$	$3.4 \times 10^{-3}$

simulation on a PC with a 2.6 GHz processor and 8 GB of RAM, the time required for network training is 80.1941 s, while the time for calculating the FK problem of the aiming sample  $\mathbf{q}_0$  is 0.6192 s. Solutions for the other five chosen samples are shown in Table 2.

## 5. Simplified ACO for forward kinematics

In this section, FK analysis based on ACO is adopted. General definition of ACO is first introduced, from which the simplified ACO for FK analysis is derived.

### 5.1. Basic ACO

Swarm intelligence algorithms are relatively new approaches for problem solving that are inspired by the group behaviors of insects or other animals. By observing the foraging behavior of ant colony, Dorigo and Blum [28] found that ant colony could always find the shortest route between the nest and the food source, which was mainly based on their indirect communication by depositing pheromone on the trails, and ACO was proposed by this enlightenment. ACO is proved to be a useful approach to solve complicated combinatorial optimization problems; its first application was to find the shortest route to link a number of cities, which was also called the traveling salesman problem.

Similar to other swarm intelligence optimization algorithms, heuristic search method is employed by ACO, where positive feedback and distributed cooperation mechanism are utilized to find the optimal route. Moreover, action choice and pheromone update rule are the two key rules. As defined by Eq. (19), action choice rule indicates the selection probability of the  $k$ th ant

located at node  $i$  choosing to move to the next node  $j$  at time  $t$ :

$$P_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)}{\sum_{s \in \text{allowed}_k} \tau_{is}^\alpha(t) \eta_{is}^\beta(t)}, & j \in \text{allowed}_k \\ 0 & \text{others} \end{cases} \quad (19)$$

where parameters  $\alpha$  and  $\beta$  are positive constants used to amplify the influence of pheromone concentrations,  $\text{allowed}_k$  denotes the neighborhood of node  $i$  with respect to ant  $k$ ,  $\tau_{ij}(t)$ , and  $\eta_{ij}(t)$  represent the pheromone and heuristic information. The pheromone update rule is expressed as follows:

$$\tau_{ij}(t+1) = (1 - \rho) \tau_{ij}(t) + \Delta \tau_{ij}(t, t+1), \quad (20)$$

$$\Delta \tau_{ij}(t, t+1) = \sum_{k=1}^m \Delta \tau_{ij}^k(t, t+1), \quad (21)$$

where  $\Delta \tau_{ij}^k(t, t+1)$  is the amount of pheromone deposited by the  $k$ th ant on path  $(i, j)$  at time interval  $(t, t+1)$ , which depends on the ants' performance,  $m$  denotes the number of ants,  $\Delta \tau_{ij}(t, t+1)$  represents the increment of pheromone, and  $(1 - \rho)$  denotes the evaporation rate as  $\rho$  is the pheromone decay parameter.

### 5.2. Application to forward kinematics

It is well known that basic ACO can be successfully used to tackle discrete problems, but it is not suitable for continuous optimization problems. However, there are classes of problems that require choosing values for continuous domains, such as the FK problems of PMs; thus, basic ACO should be extended to continuous

domains. Generally, extending basic ACO to solve continuous optimization problems can be attained by two major methods; the first one divides the continuous domain into several regions, while the second one uses a continuous probability distribution instead of a discrete one [29]. In addition, Xie et al. [8] proposed a simplified ACO for the FK analysis of a 3-RPS PM, which was inspired by the ACO, but did not follow it exactly. Especially for transition probability, both pheromone density and its update rule are redefined, and the simplified ACO will be employed to solve the FK problem of the considered PM.

Like the basic ACO, the simplified one is an effective method for function optimization problems, where the optimization objective should be defined firstly. The essence of the FK problem is to solve a set of nonlinear equations, which can be obtained from Eqs. (5a) to (5c) and expressed as follows:

$$l_i - f_i(\alpha, \beta, z) = 0, \quad (i = 1, 2, 3). \quad (22)$$

Accordingly, the nonlinear equations can be transformed into objective optimization problem as in Eq. (23), i.e., minimum value of objective function  $f(\alpha, \beta, z)$  is taken as the aim of the optimization process:

$$\min f(\alpha, \beta, z) = \sum_{i=1}^3 (l_i - f_i(\alpha, \beta, z))^2. \quad (23)$$

Since the objective function is determined, the constraint conditions should also be defined, which can be derived from the required workspace of the considered PM as expressed in Eq. (6).

The detailed procedure of simplified ACO employed for FK analysis is taken as follows:

**Step 1:** Initialize the parameters. Distribute the ant colony randomly in the workspace to obtain its initial position.

**Step 2:** Determine the density of pheromone with respect to the  $i$ th ant after moving  $k$  times, as expressed by Eq. (24), and record the optimum value  $\tau_{opt}$  (the maximum value of  $\tau_{i,k}$ ). Unlike in the basic ACO, density of pheromone in this method denotes the pheromone that the ant deposits on its position.

$$\tau_{i,k} = -f(\alpha_i, \beta_i, z_i). \quad (24)$$

**Step 3:** Calculate the transition probability  $P_{i,k}$  of ant  $i$  after moving  $k$  times; the transition probability in the simplified ACO is defined as:

$$P_{i,k} = \frac{\tau_{opt} - \tau_{i,k}}{\tau_{opt}}. \quad (25)$$

Local searching should be executed if  $P_{i,k} < P_c$ ; otherwise, global searching is taken, where  $P_c$  is a constant defined beforehand.

**Step 4:** Recalculate the density of pheromone  $\tau_{i,k}^{new}$  in the new position after local searching or global searching. The ant will move to the new position if  $\tau_{i,k}^{new} > \tau_{i,k}$ , otherwise it will remain unchanged.

**Step 5:** Once iteration is completed, update the density of pheromone of all the ants according to Eq. (26), where pheromone decay parameter  $\rho$  is a predefined constant.

$$\tau_{i,k+1} = (1-\rho)\tau_{i,k} + \tau_{i,k}^{new}. \quad (26)$$

**Step 6:** Iterate Step 3 to Step 5 until iteration times,  $n_c$ , is greater than prescriptive maximum iteration times  $n_{max}$ .

Here, initial test input sample,  $\mathbf{q}_0$ , is taken as an example to illustrate the calculation procedure. To begin with, the ant colony is distributed randomly in the workspace, as shown in Figure 9(a), where “\*” denotes the position of the ant, the pseudo-colored planes represent the lower limit of workspace of corresponding DOF, and the pseudo-color indicates the variation of the objective function across the workspace. The ant colony will move over and over guided by the pheromone to search the best solutions, and the ants start to gather together after 100 iterations as depicted in Figure 9(b); the ant colony finds the best solutions  $[2.000707^\circ, 3.999660^\circ, 430.000804 \text{ mm}]^T$  after 250 iterations as the ants concentrate on the same position, as shown in Figure 9(c). Convergence rate of this simplified ACO can be indicated by the variation of optimum value of pheromone as depicted in Figure 9(d); it tends to be stable after about 250 iterations and the time required for the simulation is 2.6524 s.

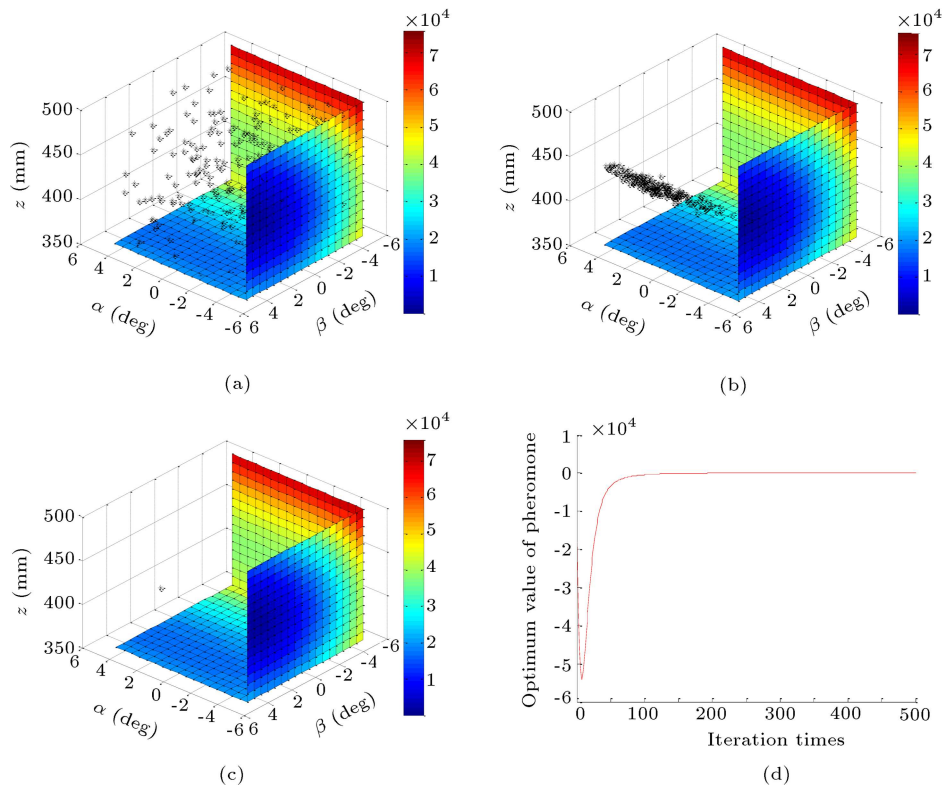
In order to test and verify the effectiveness of the simplified ACO, the same five chosen samples as discussed in Section 4 are taken into consideration and the results are shown in Table 3.

## 6. Improved Newton iterative method for forward kinematics

Newton iterative method is a relatively conventional and mature method for solving nonlinear problems, and it is employed by many researchers to solve the FK problem of common PMs. Unlike the two approaches discussed above, initial iteration values should be chosen before data processing. Here, the calculated results within Newton iterative method aiming at the tested input sample,  $\mathbf{q}_0$ , corresponding to different initial iteration values are shown in Table 4.

It can be learnt from the fifth set of data in Table 4 that initial iteration values will affect the calculated results, and they may lead to local optimal values; thus, it is very important to obtain good initial iteration values.





**Figure 9.** Simplified ACO for the FK problem: (a) Initial position distribution of the ant colony, (b) position distribution of the ant colony after 100 iterations, (c) position distribution of the ant colony after 250 iterations, and (d) optimum value of pheromone.

**Table 3.** Results of the simplified ACO for FK.

No.	Target values			Calculated results			Absolute error		
	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)
1	0	2	400	-0.000894	1.999991	400.000569	$8.9 \times 10^{-4}$	$9.0 \times 10^{-6}$	$5.7 \times 10^{-4}$
2	1	3	415	1.000058	2.999680	415.000142	$5.8 \times 10^{-5}$	$3.2 \times 10^{-4}$	$1.4 \times 10^{-4}$
3	2	3	435	2.000159	2.999536	435.000848	$1.6 \times 10^{-4}$	$4.6 \times 10^{-4}$	$8.5 \times 10^{-4}$
4	3	5	445	2.999281	5.000331	445.000245	$7.2 \times 10^{-4}$	$3.3 \times 10^{-4}$	$2.5 \times 10^{-4}$
5	5	5	460	4.999436	5.000051	459.999375	$5.6 \times 10^{-4}$	$5.1 \times 10^{-5}$	$6.3 \times 10^{-4}$

**Table 4.** Results of Newton iterative method for FK.

No.	Initial iteration values			Calculated results			Absolute error		
	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)
1	-5	0	370	1.999998	3.999999	430.000005	$2.0 \times 10^{-6}$	$1.0 \times 10^{-6}$	$5.0 \times 10^{-6}$
2	-1	1	390	1.999999	3.999998	430.000002	$1.0 \times 10^{-6}$	$2.0 \times 10^{-6}$	$2.0 \times 10^{-6}$
3	0	2	420	1.999999	3.999999	430.000001	$1.0 \times 10^{-6}$	$1.0 \times 10^{-6}$	$1.0 \times 10^{-6}$
4	2	3	450	1.999998	4.000000	430.000001	$2.0 \times 10^{-6}$	0	$1.0 \times 10^{-6}$
5	3	5	480	3.000000	5.000000	480.000000	1.0	1.0	$5.0 \times 10^1$

In order to overcome this drawback, we propose the improved Newton iterative method, by which good initial iteration values can be obtained. Since  $\alpha$  and  $\beta$  are small, we can perform the following first-order small-angle approximations:  $\sin \alpha \approx \alpha$ ,  $\cos \alpha \approx 1$ ,

$\sin \beta \approx \beta$ ,  $\cos \beta \approx 1$ ; note that angles  $\alpha$  and  $\beta$  are in radians in this section. Thus, Eqs. (5a), (7), and (9) can be rewritten as:

$$z = \sqrt{l_3^2 - (d_3 - d_1)^2} - d_1 \beta, \quad (27)$$

$$\alpha = \frac{l_1^2 - l_2^2}{4d_2(d_3\beta - \sqrt{l_3^2 - (d_3 - d_1)^2} + d_1\beta)}. \quad (28)$$

Referring to Eqs. (5a) and (5b), we can also obtain the following equation:

$$\begin{aligned} \frac{l_1^2 + l_2^2}{2} = & (d_1 - d_3)^2 + (d_4 - d_2)^2 + (z - d_1\beta)^2 \\ & + d_2^3\alpha^2(\beta^2 + 1). \end{aligned} \quad (29)$$

Substituting Eqs. (27) and (28) into Eq. (29) yields:

$$\begin{aligned} 4d_1E^2\beta^4 - 4d_1CE(2d_1 + E)\beta^3 + (AE^2 + D \\ + 4d_1^2C^2 + 8d_1EC^2 + E^2C^2)\beta^2 \\ - 2C(AE + 2d_1C^2 + EC^2)\beta + AC^2 \\ + D + C^4 = 0, \end{aligned} \quad (30)$$

where:

$$A = (d_1 - d_3)^2 + (d_4 - d_2)^2 - (l_1^2 + l_2^2)/2,$$

$$C = \sqrt{l_3^2 - (d_1 - d_3)^2},$$

$$D = (l_1^2 - l_2^2)^2/16,$$

$$E = d_1 + d_3.$$

Eq. (30) is a polynomial of degree 4 in  $\beta$ , which can be solved by Ferrari's Method or by resorting to the built-in function Roots in Matlab. Accordingly, substituting

$\beta$  into Eqs. (28) and (29), we can get the solutions to  $\alpha$  and  $z$ , where the solutions within the workspace are the initial iteration values needed. Aiming at the same tested sample,  $\mathbf{q}_0$ , good initial iteration values  $\mathbf{x}_0^c = [1.9931^\circ, 3.9949^\circ, 430.1412 \text{ mm}]^T$  can be obtained. Then, putting  $\mathbf{x}_0^c$  into the original Newton iterative method, we can get the FK solutions as  $[1.9999996^\circ, 3.9999999^\circ, 430.0000011 \text{ mm}]^T$ . It should be noted that the following equation is chosen as the convergence criterion:

$$\begin{cases} |z_{n+1} - z_n| < eps \\ |\alpha_{n+1} - \alpha_n| < eps \\ |\beta_{n+1} - \beta_n| < eps \end{cases} \quad (31)$$

where  $z_n$ ,  $\alpha_n$ , and  $\beta_n$  are the solutions after iterations for  $n$  times;  $z_{n+1}$ ,  $\alpha_{n+1}$ , and  $\beta_{n+1}$  are the solutions after iterations for  $n + 1$  times; and  $eps = 1 \times 10^{-5}$  is chosen as the stopping criterion. By this method, the final FK solutions are obtained after iterations for 3 times, and the computational time required is 1.1635 s when running the simulation on the same PC.

In order to validate the effectiveness of this method, the solutions to the other six chosen samples are shown in Table 5. The previous five samples in the table are within the workspace, while the sixth sample is beyond the workspace. It can be learnt that the samples within the workspace can obtain good initial iteration values, and the accuracy of the calculated results is high enough. Meanwhile, the sample beyond the workspace cannot get good initial iteration values, leading to a poor computational accuracy, which means

**Table 5.** Results of the improved Newton iterative method with good initial iteration values.

No.	Target values (Initial iteration values)			Calculated results			Absolute error		
	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)	$\alpha$ (deg)	$\beta$ (deg)	$z$ (mm)
1	0 ( 0	2 1.9994	400 400.0346 )	0	1.9999998	400.0000027	0	$2.0 \times 10^{-7}$	$2.70 \times 10^{-6}$
2	1 (0.9982	3 2.9979	415 415.0782)	0.9999999	3.0000000	414.9999998	$1.0 \times 10^{-7}$	0	$2.0 \times 10^{-7}$
3	2 (1.9961	3 2.9976	435 435.0802)	2.0000000	2.9999999	435.0000003	0	$1.0 \times 10^{-7}$	$3.0 \times 10^{-7}$
4	3 (2.9830	5 4.9800	445 445.2237)	2.9999975	4.9999992	445.0000098	$2.5 \times 10^{-6}$	$8.0 \times 10^{-7}$	$9.8 \times 10^{-6}$
5	5 (4.9672	5 4.9878	460 460.2467)	4.9999961	4.9999994	460.0000082	$3.9 \times 10^{-6}$	$6.0 \times 10^{-7}$	$8.2 \times 10^{-6}$
6	45 (61.17889	45 43.6536	450 439.7541)	45.130702	44.977326	450.263333	$1.3 \times 10^{-1}$	$2.3 \times 10^{-2}$	$2.6 \times 10^{-1}$

the improved Newton iterative method cannot be applied to PMs with motions of great rotation angles.

## 7. Conclusions

In this work, a novel PM with 3-SPS/PU structure designed to assemble diagnostic instruments in SG-III was proposed. On the basis of screw theory, mobility analysis was presented. Then, the IK analysis was derived in closed-form by the method of RPY transformation. By the singularity analysis, some singular configurations were revealed, which should be avoided in design.

In order to find a method with good performance to solve the FK problem of the proposed novel PM, three different approaches, namely, the BPNN, the simplified ACO, and the improved Newton iterative method, were taken into consideration. By comparison, it was found that each of these three approaches had advantages and drawbacks. The prediction accuracy of BPNN was the lowest among the three methods, which only reached the level of  $10^{-3}$ . Moreover, sufficient learning samples were required to reach the expected precision, and it needed a long time for network training. However, it possessed the highest computational speed after the network training was finished, which took only 0.6192 s. Likewise, in the simplified ACO, both computational accuracy and computational time were at the mid-level, but it was more convenient to be applied to the common FK problem, since it did not need to choose initial iteration values or train the learning samples. Finally, the improved Newton iterative method, which could provide good initial values, had the highest computational accuracy, which reached the level of  $10^{-6}$ , and its computational speed was high, which required 1.1635 s. To sum up, the improved Newton iterative method would be the best choice to solve the FK problem of the proposed mechanism. It can also be applied to solve the FK problem of similar PMs with motions of small rotation angles.

## Acknowledgment

This research was funded by Chongqing Research Program of Basic Research and Frontier Technology (No. cstc2016jcyjA0453), the National Natural Science Foundation of China (No. 51605059 & U1530138), and the National Key Research and Development Program (No.2016YFE0113600). The first author acknowledges the scholarship from Chinese Scholarship Council to support his study at Aalborg University, Denmark.

## Nomenclature

$l_i$  Actuated input of the  $i$ th SPS limb

$d_1$	Displacement between points $A_3$ and $A_4$
$d_2$	Half of the displacement between points $A_1$ and $A_2$
$d_3$	Displacement between points $D_3$ and $O_0$
$d_4$	Half of the displacement between points $D_1$ and $D_2$
$\mathbf{p}$	Position vector of point $O$ in the fixed frame $\{O_0\}$
$\mathbf{R}$	Rotation matrix from the local frame $\{O\}$ to the fixed frame $\{O_0\}$ ,
$\mathbf{a}_i, \mathbf{a}'_i$	Position vectors of $A_i$ in the local frame $\{O\}$ and the fixed frame $\{O_0\}$
$z$	Coordinate of MP with respect to the fixed frame $\{O_0\}$
$\alpha, \beta$	Orientation angles of MP with respect to the fixed frame $\{O_0\}$

## References

- Gallardoalvarado, J., Aguilarnajera, C.R., Casiquerosas, L., Ricomartinez, J.M., and Islam, M.N. "Kinematics and dynamics of 2(3-RPS) manipulators by means of screw theory and the principle of virtual work", *Mechanism and Machine Theory*, **43**(10), pp. 1281-1294 (2008).
- Staicu, S. "Inverse dynamics of the 3-PRR planar parallel robot", *Robotics and Autonomous Systems*, **57**(5), pp. 556-563 (2009).
- Varedi, S.M., Daniali, H.M., and Ganji, D.D. "Kinematics of an offset 3-UPU translational parallel manipulator by the homotopy continuation method", *Nonlinear Analysis: Real World Applications*, **10**(3), pp. 1767-1774 (2009).
- Pierrot, F., Reynaud, C., and Fournier, A. "DELTA: a simple and efficient parallel robot", *Robotica*, **8**(2), pp. 105-109 (1990).
- Chandra, R. and Rolland, L. "On solving the forward kinematics of 3RPR planar parallel manipulator using hybrid metaheuristics", *Applied Mathematics and Computation*, **217**(22), pp. 8997-9008 (2011).
- Li, B., Li, Y., and Zhao, X. "Kinematics analysis of a novel over-constrained three degree-of-freedom spatial parallel manipulator", *Mechanism and Machine Theory*, **104**, pp. 222-233 (2016).
- Liping, W., Huayang, X., Liwen, G., and Yu, Z. "A novel 3-PUU parallel mechanism and its kinematic issues", *Robotics and Computer-Integrated Manufacturing*, **42**, pp. 86-102 (2016).
- Xie, Z., Liang, H., and Song, D. "Forward kinematics of 3-RPS parallel mechanism based on a continuous ant colony algorithm", *China Mechanical Engineering*, **26**(6), pp. 799-803 (2015).

9. Gosselin, C.M. “On the kinematic design of spherical 3-DOF of parallel manipulators”, *International Journal of Robotics Research*, **12**(4), pp. 394-402 (1993).
10. Wu, G. and Zou, P. “Comparison of 3-DOF asymmetrical spherical parallel manipulators with respect to motion/force transmission and stiffness”, *Mechanism and Machine Theory*, **105**, pp. 369-387 (2016).
11. Sadjadian, H. and Taghirad, H.D. “Comparison of different methods for computing the forward kinematics of a redundant parallel manipulator”, *Journal of Intelligent & Robotic Systems*, **44**(3), pp. 225-246 (2005).
12. Zhang, D. and Lei, J. “Kinematic analysis of a novel 3-DOF actuation redundant parallel manipulator using artificial intelligence approach”, *Robotics and Computer-Integrated Manufacturing*, **27**(1), pp. 157-163 (2011).
13. Gan, D., Liao, Q., Dai, J.S., Wei, S., and Seneviratne, L.D. “Forward displacement analysis of the general 6-6 Stewart mechanism using Gröbner bases”, *Mechanism and Machine Theory*, **44**(9), pp. 1640-1647 (2009).
14. Lee, T.Y. and Shim, J.K. “Forward kinematics of the general 6-6 Stewart platform using algebraic elimination”, *Mechanism and Machine Theory*, **36**(9), pp. 1073-1085 (2001).
15. Zhou, W., Chen, W., Liu, H., and Li, X. “A new forward kinematic algorithm for a general Stewart platform”, *Mechanism and Machine Theory*, **87**, pp. 177-190 (2015).
16. Dong, X., Yu, J., Chen, B., and Zong, G. “Geometric approach for kinematic analysis of a class of 2-DOF rotational parallel manipulators”, *Chinese Journal of Mechanical Engineering*, **25**(2), pp. 241-247 (2012).
17. Ruggiu, M. and Kong, X. “Mobility and kinematic analysis of a parallel mechanism with both PPR and planar operation modes”, *Mechanism and Machine Theory*, **55**, pp. 77-90 (2012).
18. Parikh, P.J. and Lam, S.S. “Solving the forward kinematics problem in parallel manipulators using an iterative artificial neural network strategy”, *International Journal of Advanced Manufacturing Technology*, **40**(5-6), pp. 595-606 (2009).
19. Rahmani, A. and Ghanbari, A. “Application of neural network training in forward kinematics simulation for a novel modular hybrid manipulator with experimental validation”, *Intelligent Service Robotics*, **9**(1), pp. 79-91 (2016).
20. Shahamiri, S.R. and Salim, S. “Real-time frequency-based noise-robust automatic speech recognition using multi-nets artificial neural networks: A multi-views multi-learners approach”, *Neurocomputing*, **129**, pp. 199-207 (2014).
21. Yang, C., Zheng, S., Jin, J., Zhu, S., and Han, J. “Forward kinematics analysis of parallel manipulator using modified global Newton-Raphson method”, *Journal of Central South University of Technology*, **17**(6), pp. 1264-1270 (2010).
22. Ferrari, D. and Giberti, H. “A genetic algorithm approach to the kinematic synthesis of a 6-DOF parallel manipulator”, *IEEE Conference on Control Applications*, France, Antibes, pp. 222-227 (2014).
23. Rokbani, N. and Alimi, A.M. “Inverse kinematics using particle swarm optimization, a statistical analysis”, *Procedia Engineering*, **64**, pp. 1602-1611 (2013).
24. Morell, A., Tarokh, M., and Acosta, L. “Solving the forward kinematics problem in parallel robots using support vector regression”, *Engineering Applications of Artificial Intelligence*, **26**(7), pp. 1698-1706 (2013).
25. Xiong, Z., Wang, H., Cao, T., Yuan, X., Yao, C., Zhang, Z., Zhou, M., and Ma, G. “Error analysis on assembly and alignment of laser optical unit”, *Advances in Mechanical Engineering*, **7**(7), pp. 1-10 (2015).
26. Sanchezalonso, R.E., Gonzalezbarbosa, J., Castillo-castaneda, E., and Gallardoalvarado, J. “Kinematic analysis of a novel 2 (3-RUS) parallel manipulator”, *Robotica*, **34**(10), pp. 2241-2256 (2016).
27. Gosselin, C. and Angeles, J. “Singularity analysis of closed-loop kinematic chains”, *IEEE Transactions on Robotics and Automation*, **6**(3), pp. 281-290 (1990).
28. Dorigo, M. and Blum, C. “Ant colony optimization theory: a survey”, *Theoretical Computer Science*, **344**(2-3), pp. 243-278 (2005).
29. Xiao, J. and Li, L. “A hybrid ant colony optimization for continuous domains”, *Expert Systems with Applications*, **38**(9), pp. 11072-11077 (2011).

## Biographies

**Xiaoyong Wu** is a lecturer in Chongqing University of Technology, China. He received the BSc degree from Wuhan Polytechnic University in 2013, and the PhD degree in Mechanical Engineering from Chongqing University in 2018. His main research interests are theory of mechanism and parallel manipulators.

**Zhijiang Xie** is currently a Professor in the State Key Laboratory of Mechanical Transmission, Chongqing University, China. He received the BSc degree from Henan Polytechnic University, and the MSc and PhD degrees from Chongqing University. His main research interests include designing mechanical systems, robotics, mechatronics, and fault diagnosis.