Robotica (2009) volume 27, pp. 67–78. © 2008 Cambridge University Press
doi:10.1017/S0263574708004530 Printed in the United Kingdom

Error analysis and optimal design of a class of translational parallel kinematic machine using particle swarm optimization

Qingsong Xu and Yangmin Li*

Department of Electromechanical Engineering, Faculty of Science and Technology, University of Macau, Av. Padre Tomás Pereira, Taipa, Macao SAR, P.R. China

(Received in Final Form: February 23, 2008. First published online: April 15, 2008)

SUMMARY

In this paper, the optimization of architectural parameters for a class of translational parallel kinematic machine (PKM) is performed with the particle swarm optimization (PSO) to achieve the best accuracy characteristics. The conventional error transformation matrix (ETM) is derived based on the differentiation of kinematic equations, and a new error amplification index (EAI) over a usable workspace is proposed as an error performance index for the optimization. To validate the efficiency of the PSO method, both the traditional direct search method and the genetic algorithm (GA) are implemented as well. The simulation results not only show the advantages of PSO method for the architectural optimization, but also reveal the necessity to introduce the EAI for the optimal design. And the results are valuable for architectural design of the PKM for machine tool applications.

KEYWORDS: Parallel manipulators; Error model; Accuracy; Workspace; Optimal design.

1. Introduction

Parallel manipulators possess outstanding merits over the conventional serial ones in terms of accuracy, velocity, stiffness, and payload capacity, therefore they are widely used in industries.[1,2] In recent years, less-DOF (degree-of-freedom) parallel manipulators with less than six-DOF have been paid great attention since they have such extra advantages as the simplicity in structure and cost reduction in manufacture.[3] It is deserved to mention that translational parallel manipulators (TPMs) have specific applications that need pure translational motions in case of a motion simulator and a positioning tool on an assembly line. Nowadays various TPMs have been designed and proposed in the literature, and the type syntheses of the TPM with three translational motions have been conducted by some insightful investigations.[4–6]

In designing and controlling a parallel manipulator, the accuracy performance is one of the most important factors to be taken into account, particularly for those which are used as automatic machine tools. Therefore, it is necessary to investigate the error issues of parallel kinematic machines (PKMs) thoroughly. The accuracy of a PKM relies on many factors in terms of actuator control errors, installation errors, manufacturing tolerances and clearances, and architecture design, etc. In addition, the accuracy of a PKM changes with the configuration of the manipulator, which makes the design of a PKM complicated when the accuracy property is considered.

With respect to the approaches for the error modeling of a parallel manipulator, the error model can be derived through the D–H method,[7] virtual leg concept,[8] and kinematic equations differentiation,[9] etc., where the last method is more intuitive and is adopted to obtain the error transformation matrices (ETMs) of several types of PKMs.[10]

As far as the works related to the error evaluation, some performance indices have been proposed in the literature. Most assessments are based upon the singular value decomposition of the ETM. For instance, similar to the manipulability ellipsoid, the error ellipsoid method[9] was presented to find the magnitude of the errors in different directions, and the error ellipsoid volume[11] was adopted to predict the PKM errors. Moreover, just like the condition number of the Jacobian matrix, the condition number of the ETM has been utilized as a measure of the relative error amplification, which also characterizes the isotropy of the error ellipsoid. In addition, the maximum singular value of the ETM was used as the maximum value of error amplification factors (EAFs) to estimate the errors.[11] However, any single one of these proposed performance indices cannot fully characterize the error performance, and it has been shown that different indices lead to totally different architectures of a parallel manipulator after an optimal design.[11] In order to ensure the accuracy of the manipulation of the PKM anywhere in the workspace, a mixed error index is proposed in this paper to make a compromise between the maximum singular value and condition number of the ETM. And the feasibility of the introduced index is verified by numerical simulation studies.

Concerning the architecture optimization of PKMs, the best performances in terms of dexterity, workspace, and stiffness were considered to achieve the optimal kinematic parameters[12–15] However, there were relatively limited investigations on the optimal design of a PKM for the best accuracy property[11] in spite of accuracy being also an important performance for PKMs. This paper is mainly concentrated on the error analysis and architecture optimization of a class of translational parallel kinematic machine with 3-cylindrical-revolute-cylindrical (CRC) limbs

* Corresponding author. E-mail: ymli@umac.mo

for the best accuracy characteristics. The 3-CRC PKM is a modified version of the 3-PRC (3-prismatic-revolute-cylindrical) TPM formerly investigated by the authors.[16,17] And a 3-CRC PM is formed by adding one revolute (R) joint to each kinematic linkage of the original 3-PRC manipulator so as to eliminate the overconstraints, because the 3-PRC mechanism may not move or the manipulator cannot work properly if certain kinematic errors exist. In this paper, the accuracy of the 3-CRC PKM is analyzed since no works have been made on this topic yet. Particularly, the particle swarm optimization (PSO) method is introduced for the optimal design of the PKM dimensions. Although PSO owns certain advantages in terms of easy implementation and efficient computation, the applications of PSO on the problems of manipulator optimal design are still limited.[18] To the knowledge of the authors, there are no other researchers addressing this area so far.

The remainder of this paper is organized in the following way. A brief survey on optimization methods is outlined in Section 2, and the architecture and kinematic description of the PKM are presented in Section 3. Section 4 is focused on the error modeling and error evaluation of the 3-CRC PKM. Then, the PKM architectural parameters are optimized in Section 5 via three different approaches with respect to various objectives, where the discussions on simulation results are provided and some guidelines for the design of the PKM are also given. Finally, the concluding remarks are summarized in Section 6.

2. Survey on Optimization Methods

In view of the present optimization problem, it is extremely difficult to express the error performance index into an analytical form, i.e., there is no analytical expression for the objective function. Thus, it can be regarded as a nonlinear continuous optimization problem.

The traditional optimization routine utilizes a local search procedure based on the gradient of the objective function to search for the optimum. Regarding the objective function without an explicit expression of the gradient, it can be calculated by a direct search method such as the well-known Nelder–Mead simplex (NMS) algorithm. However, this technique heavily depends on good starting points, and may fall into local optima.

On the other hand, as a global method for solving both constrained and unconstrained optimization problems based on natural evolution, the genetic algorithm (GA) can be applied to solve a variety of optimization problems that are not well suited for standard optimization algorithms. These include the problems where the objective function has discontinuous, non-differentiable, stochastic, or highly nonlinear features.[19]

Moreover, the PSO can also be employed to solve a variety of optimization problems. As a form of swarm intelligence, PSO is a relatively new algorithm proposed by Kennedy and Eberhart.[20] It is a population-based stochastic optimization technique inspired by the social behavior of bird flocking or fish schooling. Since PSO is originally introduced for the optimization of continuous nonlinear functions, it has been successfully applied to many other problems such as discrete optimization, artificial neural network training, fuzzy system control, and mobile robot navigation.

A PSO system is initialized with a population of random solutions and searches for optima by updating generations, it is similar to evolutionary computation techniques such as GA. However, compared to the GA, PSO has no evolutionary operators in terms of crossover and mutation. Hence, from the viewpoint of programming, the advantages of PSO are easy implementation and fewer adjustable parameters.[21]

In the PSO, the population is called a swarm and the individuals (i.e. the search points) are called particles. Each particle moves with an adaptable velocity within the search space, and retains a memory of the best position it has ever found. Finally, it knows where the best solution encountered by any other particles in the search space, and will then modify its direction toward its own best position and the global best position, which will provide some forms of convergence in searching. Regarding a D-dimensional search space and a swarm consisting of N particles, the ith particle is represented by a D-dimensional vector $X_i = (x_{i1}, x_{i2}, \cdots, x_{iD})$, the velocity of this particle is a D-dimensional vector $V_i = (v_{i1}, v_{i2}, \cdots, v_{iD})$, and the best previous position encountered by this particle is described by $P_i = (p_{i1}, p_{i2}, \cdots, p_{iD})$. Let g represents the index of the particle which attains the best previous position among all the particles in the swarm, and k denotes the iteration counter. Then, the swarm is manipulated in accordance with the following equations:[22]

$$V_i(k+1) = wV_i(k) + c_1 r_1[P_i(k) - X_i(k)] + c_2 r_2[P_g(k) - X_i(k)], \tag{1}$$

$$X_i(k+1) = X_i(k) + V_i(k+1), \tag{2}$$

where w is the inertial weight, c_1 and c_2 are the acceleration constants called cognitive and social parameters, respectively, r_1 and r_2 are random numbers uniformly distributed between 0 and 1, and the particle index $i = 1, 2, \cdots, N$. The selection of the above parameters has been widely studied in the relevant literature,[22] and the pseudocode for a PSO algorithm is elaborated in Fig. 1.

It is noticeable that some other approaches such as the interval analysis means[14] can also be applied for the architecture optimization of a PKM. Here, for the main reason of verifying the efficiency of the PSO approach, only the three methods mentioned above are employed and

```
Initialize the Population Do{
    For $i = 1$ to Population Size{
        Calculate fitness value
        If the fitness value is better than the best one (pBest) in history
        then set current value as the new pBest
    }
    Choose the particle with the best fitness value of all the particles as gBest
    For $i = 1$ to Population Size{
        Calculate new velocity in accordance with Eq. (1)
        Update particle position in accordance with Eq. (2)
    }
} Until termination criterion is met
```

Fig. 1. The pseudocode of a PSO procedure.

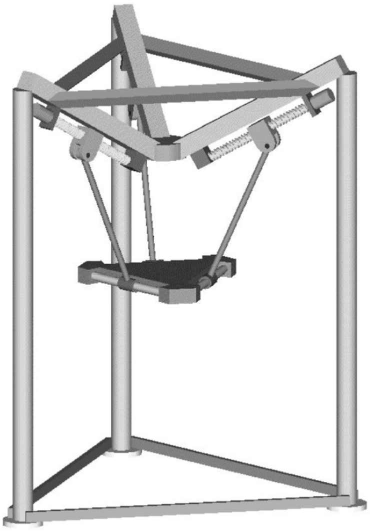


Fig. 2. CAD model of a 3-CRC PKM.

compared for the optimization of a class of 3-CRC PKM in the following discussions.

3. Architecture and Kinematics Description of a 3-CRC PKM

The CAD model of a 3-CRC PKM is shown in Fig. 2 and the schematic diagram is described in Fig. 3. The PKM consists of a mobile platform, a fixed base, and three limbs with identical kinematic structure. Each limb connects the mobile platform to the fixed base by a C (cylindrical) joint, a R (revolute) joint, and another C joint in sequence, where each C joint can be considered as the combination of a P (prismatic) joint with a coaxial R joint. In addition, the P joint on the fixed base is driven by a lead screw linear actuator.

In order to prevent the mobile platform from changing its orientation, it is necessary for the joint axes within the same limb to satisfy certain geometric conditions: Briefly, the axial directions of the passive R and C joints within each limb are parallel to each other. As far as the mobility analysis of a 3-CRC PKM is concerned, the DOF number of the manipulator can be determined by the general Grübler–Kutzbach criterion as:

$$F = \lambda(n - j - 1) + \sum_{i=1}^{j} f_i = 6 \times (8 - 9 - 1) + 15 = 3, \tag{3}$$

where λ represents the dimension of task space, n is the number of links, j is the number of joints, and f_i denotes the DOF of joint i.

Equation (3) only indicates that a 3-CRC PKM possesses 3-DOF. By resorting to the screw theory, we can generate the DOF properties of the PKM which represents three translations in space. Actually, the 3-CRC PKM is constructed by adding a R joint to each limb of a 3-PRC PKM so as to eliminate the overconstraints of the original 3-PRC parallel manipulator.[16] And it has been demonstrated in the previous works of the authors[23] that the added R joints bring no effect on either mobility or kinematics of the original 3-PRC manipulator since the added joints are idle actually.

For the sake of analysis, as shown in Figs. 3 and 4, we assign a fixed Cartesian frame $O\{x, y, z\}$ at the centered point O of the fixed base, and a moving frame $P\{u, v, w\}$ on the triangle mobile platform at the centered point P, where the z- and w-axes are perpendicular to the platform, and the x- and y-axes are parallel to the u- and v-axes, respectively.

In addition, the ith limb D_iB_i $(i = 1, 2, 3)$ with the length of l $(l_1 = l_2 = l_3 = l)$ is connected to the mobile platform at B_i which is a point on the axis of the ith C joint. B_i' denotes the point on the mobile platform that is coincident with the initial position of B_i, and the three points B_i', for $i = 1, 2$, and 3, lie on a circle of radius b. The three rails M_iN_i intersect one another at point E and intersect the x–y plane at points A_1, A_2, and A_3 locating on a circle of radius a. The sliders of P joints D_i are restricted to move along the rails between M_i and N_i. Moreover, the axis of P joint is perpendicular to the axes of R and C joints within the ith limb, and the angle α is measured from the fixed base to rails M_iN_i, which is defined as the layout angle of actuators. In order to obtain a compact architecture, the value of α is designed within the range of [0°, 90°]. Besides, angle φ_i is defined from the x-axis to $\overrightarrow{OA_i}$ in the fixed frame, and also from the u-axis to $\overrightarrow{PB_i'}$ in the moving frame. Without loss of generality, let the x-axis

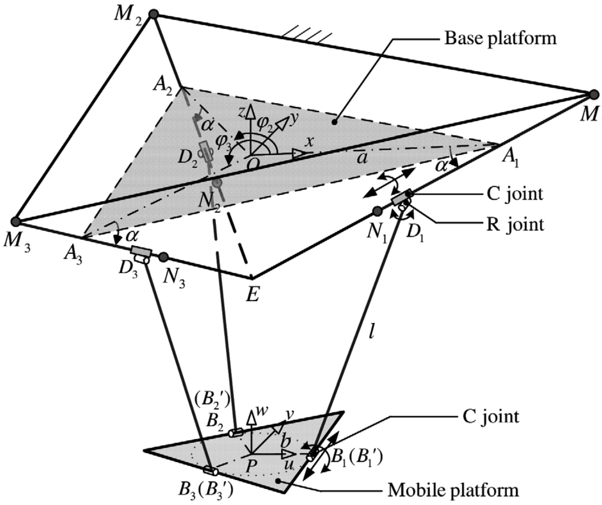


Fig. 3. Schematic representation of a 3-CRC PKM.

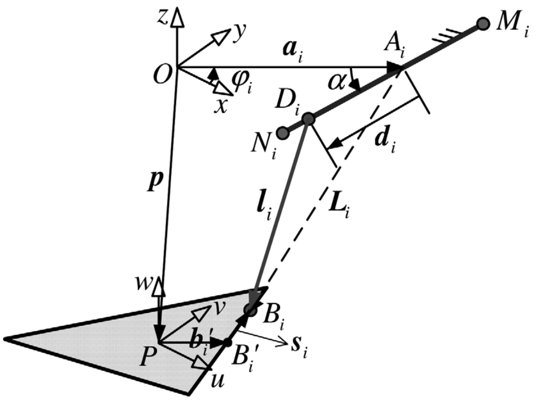


Fig. 4. Vector representation of one kinematic chain.

point along $\overrightarrow{OA_1}$, and the u-axis along $\overrightarrow{PB'_1}$. Then, we have $\varphi_1 = 0°$. Additionally, let A_i locate the midpoint of M_iN_i and let $d_{\max}$ and $s_{\max}$ denote the stroke limits for the linear actuators and passive C joints, respectively, then we have

$$-d_{\max} \leq d_i \leq d_{\max}, \tag{4}$$

$$-s_{\max} \leq s_i \leq s_{\max}, \tag{5}$$

where d_i represents a linear displacement of the ith linear actuator and s_i is the stroke of the ith C joint, for $i = 1$, 2, and 3. In order to achieve a symmetric workspace of the manipulator, both $\Delta A_1A_2A_3$ and $\Delta B_1B_2B_3$ are assigned to be equilateral triangles, then $\varphi_2 = 120°$ and $\varphi_3 = 240°$. Additionally, the passive R and C joint axis directions are designed to be perpendicular to the corresponding leg direction within the same limb for the convenience of assembly.

Since the added R joints bring no impact on the kinematics of the original 3-PRC PKM, the kinematic solutions and workspace of a 3-CRC PKM are the same to those of a 3-PRC PKM. The workspace can be obtained by taking into account the inverse kinematics solutions along with the consideration of physical constraints,[16] and it has been shown that the workspace does not contain any void. In addition, the PKM can be designed with singularities free by the mechanism design rules proposed in our previous works.[16] Then the error analysis for a 3-CRC PKM is carried out in the following discussions.

4. Error Analysis

In this section, the error model for a translational 3-CRC PKM is derived with the generation of the total ETM which transforms all error sources to the pose error of the mobile platform, and an accuracy performance index is introduced for the optimal design.

4.1. Error modeling

The ith kinematic chain of a 3-CRC PKM is shown in Fig. 4, referring to which, the position vectors of points A_i and B'_i with respect to frames O and P, respectively can be written as ${}^O\boldsymbol{a}_i$ and ${}^P\boldsymbol{b}'_i$, where a leading superscript indicates the coordinate frame with respect to which a vector is expressed. For brevity, the leading superscript will be omitted whenever the coordinate frame is the fixed frame, e.g., $\boldsymbol{a}_i = {}^O\boldsymbol{a}_i$.

Generally, the position and orientation of the mobile platform with respect to the fixed frame can be described by a position vector $\boldsymbol{p} = [p_x \quad p_y \quad p_z]^T = \overrightarrow{OP}$, and a 3×3 rotation matrix $\boldsymbol{R}$. Although $\boldsymbol{R}$ becomes an identity matrix, i.e., $\boldsymbol{R} = \boldsymbol{I}$ due to the mobile platform possesses only a translational motion, the rotation matrix is still explicitly written for the sake of error analysis. Then, the position vector $\boldsymbol{b}'_i = \boldsymbol{R}\,{}^P\boldsymbol{b}'_i$.

Let $\boldsymbol{k}_i$ be a unit vector along the leg direction $\overrightarrow{D_iB_i}$, $\boldsymbol{t}_i$ denotes the unit vector pointing along the rail M_iN_i, and $\boldsymbol{c}_i$ represents a unit vector parallel to the axes of the passive R and C joints in ith limb, respectively. With reference to Fig. 4, a vector-loop equation can be written for the ith limb as:

$$\boldsymbol{a}_i + d_i\boldsymbol{t}_i + l_i\boldsymbol{k}_i = \boldsymbol{p} + \boldsymbol{R}\,{}^P\boldsymbol{b}_i, \tag{6}$$

where ${}^P\boldsymbol{b}_i = {}^P\boldsymbol{b}'_i + s_i\,{}^P\boldsymbol{c}_i$.

A differential error model can be obtained by differentiating Eq. (6) as follows:

$$\delta\boldsymbol{a}_i + \delta d_i\boldsymbol{t}_i + d_i\delta\boldsymbol{t}_i + \delta l_i\boldsymbol{k}_i + l_i\delta\boldsymbol{k}_i = \delta\boldsymbol{p} + \delta\boldsymbol{R}\,{}^P\boldsymbol{b}_i + \boldsymbol{R}\delta{}^P\boldsymbol{b}_i, \tag{7}$$

where

$$\delta{}^P\boldsymbol{b}_i = \delta{}^P\boldsymbol{b}'_i + \delta s_i\,{}^P\boldsymbol{c}_i + s_i\delta{}^P\boldsymbol{c}_i, \tag{8}$$

and the derivative of the rotation matrix can be expressed by

$$\delta\boldsymbol{R} = \delta\boldsymbol{\theta} \times \boldsymbol{R} = \begin{bmatrix} 0 & -\delta\theta_z & \delta\theta_y \\ \delta\theta_z & 0 & -\delta\theta_x \\ -\delta\theta_y & \delta\theta_x & 0 \end{bmatrix}\boldsymbol{R}, \tag{9}$$

with $\delta\boldsymbol{\theta} = [\delta\theta_x \ \delta\theta_y \ \delta\theta_z]^T$ denoting a vector of small rotations around the fixed reference frame.

Substituting Eq. (9) into Eq. (7) and in view of $\boldsymbol{b}_i = \boldsymbol{R}\,{}^P\boldsymbol{b}_i$, one can generate

$$\delta\boldsymbol{a}_i + \delta d_i\boldsymbol{t}_i + d_i\delta\boldsymbol{t}_i + \delta l_i\boldsymbol{k}_i + l_i\delta\boldsymbol{k}_i = \delta\boldsymbol{p} + \delta\boldsymbol{\theta} \times \boldsymbol{b}_i + \boldsymbol{R}\delta{}^P\boldsymbol{b}_i. \tag{10}$$

Dot-multiplying both sides of Eq. (10) by $\boldsymbol{k}_i$, leads to

$$\begin{aligned}\boldsymbol{k}_i^T\delta\boldsymbol{a}_i + \boldsymbol{k}_i^T\boldsymbol{t}_i\delta d_i + \boldsymbol{k}_i^T(d_i\delta\boldsymbol{t}_i) + \delta l_i \\ = \boldsymbol{k}_i^T\delta\boldsymbol{p} + (\boldsymbol{b}_i \times \boldsymbol{k}_i)^T\delta\boldsymbol{\theta} + \boldsymbol{k}_i^T\boldsymbol{R}\delta{}^P\boldsymbol{b}'_i + \boldsymbol{k}_i^T\boldsymbol{R}(s_i\delta{}^P\boldsymbol{c}_i),\end{aligned} \tag{11}$$

where the properties of $\boldsymbol{k}_i^T\boldsymbol{k}_i = 1$ and $\boldsymbol{k}_i^T\delta\boldsymbol{k}_i = 0$ are considered.

Rearranging Eq. (11) allows the derivation of the expression in a matrix form:

$$\boldsymbol{J}\,\delta\boldsymbol{X} = \delta\boldsymbol{q} + \boldsymbol{J}_1\delta\boldsymbol{l} + \boldsymbol{J}_2\delta\boldsymbol{a} + \boldsymbol{J}_2\delta\boldsymbol{d} + \boldsymbol{J}_3\delta\boldsymbol{b}' + \boldsymbol{J}_3\delta\boldsymbol{s}, \tag{12}$$

where

$$\delta\boldsymbol{X} = [\delta\boldsymbol{p}^T \quad \delta\boldsymbol{\theta}^T]^T \in \Re^{6\times1}, \tag{13}$$

$$\delta\boldsymbol{q} = [\delta d_1 \quad \delta d_2 \quad \delta d_3]^T \in \Re^{3\times1}, \tag{14}$$

$$\delta\boldsymbol{l} = [\delta l_1 \quad \delta l_2 \quad \delta l_3]^T \in \Re^{3\times1}, \tag{15}$$

$$\delta\boldsymbol{a} = [\delta\boldsymbol{a}_1^T \quad \delta\boldsymbol{a}_2^T \quad \delta\boldsymbol{a}_3^T]^T \in \Re^{9\times1}, \tag{16}$$

$$\delta\boldsymbol{d} = [d_1\delta\boldsymbol{t}_1^T \quad d_2\delta\boldsymbol{t}_2^T \quad d_3\delta\boldsymbol{t}_3^T]^T \in \Re^{9\times1}, \tag{17}$$

$$\delta\boldsymbol{b}' = [\delta{}^P\boldsymbol{b}_1'^T \quad \delta{}^P\boldsymbol{b}_2'^T \quad \delta{}^P\boldsymbol{b}_3'^T]^T \in \Re^{9\times1}, \tag{18}$$

$$\delta\boldsymbol{s} = [s_1\delta\boldsymbol{c}_1^T \quad s_2\delta\boldsymbol{c}_2^T \quad s_3\delta\boldsymbol{c}_3^T]^T \in \Re^{9\times1}, \tag{19}$$

and

$$J = \begin{bmatrix} \dfrac{\boldsymbol{k}_1^T}{\boldsymbol{k}_1^T\boldsymbol{t}_1} & \dfrac{(\boldsymbol{b}_1 \times \boldsymbol{k}_1)^T}{\boldsymbol{k}_1^T\boldsymbol{t}_1} \\ \dfrac{\boldsymbol{k}_2^T}{\boldsymbol{k}_2^T\boldsymbol{t}_2} & \dfrac{(\boldsymbol{b}_2 \times \boldsymbol{k}_2)^T}{\boldsymbol{k}_2^T\boldsymbol{t}_2} \\ \dfrac{\boldsymbol{k}_3^T}{\boldsymbol{k}_3^T\boldsymbol{t}_3} & \dfrac{(\boldsymbol{b}_3 \times \boldsymbol{k}_3)^T}{\boldsymbol{k}_3^T\boldsymbol{t}_3} \end{bmatrix} \in \Re^{3\times 6}, \quad (20)$$

$$\boldsymbol{J}_1 = \begin{bmatrix} \dfrac{1}{\boldsymbol{k}_1^T\boldsymbol{t}_1} & 0 & 0 \\ 0 & \dfrac{1}{\boldsymbol{k}_2^T\boldsymbol{t}_2} & 0 \\ 0 & 0 & \dfrac{1}{\boldsymbol{k}_3^T\boldsymbol{t}_3} \end{bmatrix} \in \Re^{3\times 3}, \quad (21)$$

$$\boldsymbol{J}_2 = \begin{bmatrix} \dfrac{\boldsymbol{k}_1^T}{\boldsymbol{k}_1^T\boldsymbol{t}_1} & 0 & 0 \\ 0 & \dfrac{\boldsymbol{k}_2^T}{\boldsymbol{k}_2^T\boldsymbol{t}_2} & 0 \\ 0 & 0 & \dfrac{\boldsymbol{k}_3^T}{\boldsymbol{k}_3^T\boldsymbol{t}_3} \end{bmatrix} \in \Re^{3\times 9}, \quad (22)$$

$$\boldsymbol{J}_3 = -\begin{bmatrix} \dfrac{\boldsymbol{k}_1^T\boldsymbol{R}}{\boldsymbol{k}_1^T\boldsymbol{t}_1} & 0 & 0 \\ 0 & \dfrac{\boldsymbol{k}_2^T\boldsymbol{R}}{\boldsymbol{k}_2^T\boldsymbol{t}_2} & 0 \\ 0 & 0 & \dfrac{\boldsymbol{k}_3^T\boldsymbol{R}}{\boldsymbol{k}_3^T\boldsymbol{t}_3} \end{bmatrix} \in \Re^{3\times 9}. \quad (23)$$

It should be noticed that in Eq. (12), $\delta\boldsymbol{X}$ is a vector of the platform pose errors, $\delta\boldsymbol{q}$ represents a vector of the actuator errors including the control error and backlash of lead screws, $\delta\boldsymbol{l}$ denotes a length error vector of the legs subject to manufacturing tolerance and imperfect installation with the passive R and C joints, $\delta\boldsymbol{d}$ and $\delta\boldsymbol{s}$ describe vectors of straightness errors for the actuator axis and the passive C joint axis, $\delta\boldsymbol{a}$ and $\delta\boldsymbol{b}'$ are vectors of joint installation errors of points A_i and B_i, respectively. The errors $\delta\boldsymbol{l}$, $\delta\boldsymbol{d}$, $\delta\boldsymbol{s}$, $\delta\boldsymbol{a}$, and $\delta\boldsymbol{b}'$ are called nonactuation errors which can be partly compensated in a controller model by some calibration methods.[11]

When the manipulator is away from singularities, Eq. (12) can be expressed as

$$\delta\boldsymbol{X} = \boldsymbol{E}\,\delta\boldsymbol{e}, \quad (24)$$

where

$$\boldsymbol{E} = \boldsymbol{J}^+[\boldsymbol{I} \vdots \boldsymbol{J}_1 \vdots \boldsymbol{J}_2 \vdots \boldsymbol{J}_2 \vdots \boldsymbol{J}_3 \vdots \boldsymbol{J}_3] \in \Re^{6\times 42} \quad (25)$$

is defined as the total ETM with $\boldsymbol{J}^+$ denoting the pseudo inverse of matrix $\boldsymbol{J}$, and

$$\delta\boldsymbol{e} = [\delta\boldsymbol{q} \vdots \delta\boldsymbol{l} \vdots \delta\boldsymbol{a} \vdots \delta\boldsymbol{d} \vdots \delta\boldsymbol{b}' \vdots \delta\boldsymbol{s}] \in \Re^{42\times 1} \quad (26)$$

is defined as the error source vector.

4.2. *Error evaluation*

In order to use the ETM for the characterization of the error transformation, the total ETM should be normalized.[11] Firstly, the individual error sources can be normalized using the corresponding maximum error values as:

$$\delta\boldsymbol{q} = ||\delta\boldsymbol{q}||_{\max}\delta\bar{\boldsymbol{q}}, \quad (27)$$

$$\delta\boldsymbol{l} = ||\delta\boldsymbol{l}||_{\max}\delta\bar{\boldsymbol{l}}, \quad (28)$$

$$\delta\boldsymbol{a} = ||\delta\boldsymbol{a}||_{\max}\delta\bar{\boldsymbol{a}}, \quad (29)$$

$$\delta\boldsymbol{d} = ||\delta\boldsymbol{d}||_{\max}\delta\bar{\boldsymbol{d}}, \quad (30)$$

$$\delta\boldsymbol{b}' = ||\delta\boldsymbol{b}'||_{\max}\delta\bar{\boldsymbol{b}'}, \quad (31)$$

$$\delta\boldsymbol{s} = ||\delta\boldsymbol{s}||_{\max}\delta\bar{\boldsymbol{s}}, \quad (32)$$

where "$||\cdot||_{\max}$" represents the maximum value of each error source and "$\bar{\cdot}$" denotes a normalized vector. Besides, the maximum errors for the linear actuators and leg lengths can be simplified as $||\delta\boldsymbol{q}||_{\max} = |\delta q|_{\max}$ and $||\delta\boldsymbol{l}||_{\max} = |\delta l|_{\max}$, respectively.

Additionally, an observation of the units of matrix $\boldsymbol{J}$ in Eq. (20) reveals that the first three columns are dimensionless while the last three are related to the length units which are introduced by the position vector $\boldsymbol{b}_i$. It is necessary to homogenize the units of the Jacobian matrices so as to generate a total error transformation matrix invariant of the length units adopted. Here, the mobile platform radius b is chosen as the characteristic length[12] to obtain the dimensionally homogeneous Jacobian matrix $\boldsymbol{J}_h$, i.e.,

$$\boldsymbol{J}_h = \boldsymbol{J}\,\boldsymbol{H}, \quad (33)$$

where the 6×6 diagonal matrix

$$\boldsymbol{H} = \mathrm{diag}\left\{1,\ 1,\ 1,\ \frac{1}{b},\ \frac{1}{b},\ \frac{1}{b}\right\}. \quad (34)$$

Hence Eq. (24) becomes

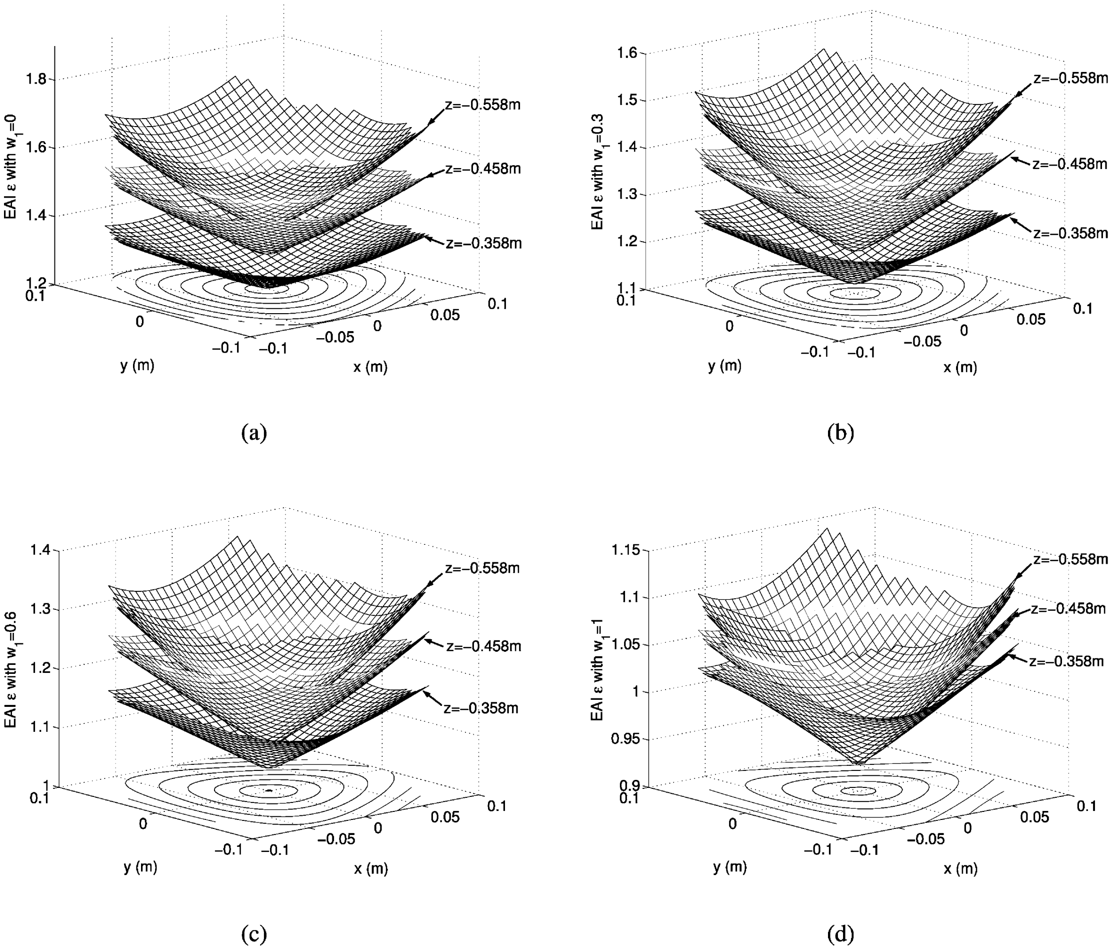
$$\delta\boldsymbol{X} = \bar{\boldsymbol{E}}\,\delta\bar{\boldsymbol{e}}, \quad (35)$$

where

$$\bar{\boldsymbol{E}} = \boldsymbol{J}_h^+[|\delta q|_{\max}\boldsymbol{I} \vdots |\delta l|_{\max}\boldsymbol{J}_1 \vdots ||\delta\boldsymbol{a}||_{\max}\boldsymbol{J}_2 \vdots ||\delta\boldsymbol{d}||_{\max}\boldsymbol{J}_2 \vdots ||\delta\boldsymbol{b}'||_{\max}\boldsymbol{J}_3 \vdots ||\delta\boldsymbol{s}||_{\max}\boldsymbol{J}_3], \quad (36)$$

$$\delta\bar{\boldsymbol{e}} = [\delta\bar{\boldsymbol{q}} \vdots \delta\bar{\boldsymbol{l}} \vdots \delta\bar{\boldsymbol{a}} \vdots \delta\bar{\boldsymbol{d}} \vdots \delta\bar{\boldsymbol{b}'} \vdots \delta\bar{\boldsymbol{s}}], \quad (37)$$

with $\boldsymbol{J}_h^+$ denoting the pseudo inverse of the matrix $\boldsymbol{J}_h$.



(a)

(b)

(c)

(d)

Fig. 5. The distributions of EAI with (a) $w_1 = 0$, (b) $w_1 = 0.3$, (c) $w_1 = 0.6$, and (d) $w_1 = 1.0$ in three planes at different heights.

The error transformation can be characterized mainly by the singular values of $\overline{\boldsymbol{E}}$, i.e., $\sigma_{\overline{\boldsymbol{E}}}$, which can be regarded as the error amplification factor (EAF).[11] As far as the singular values $(\sigma_{\overline{\boldsymbol{E}}})_i$ for the 6×42 nonsymmetric matrix $\overline{\boldsymbol{E}}$ is concerned, it can be calculated by $(\sigma_{\overline{\boldsymbol{E}}})_i = \sqrt{(\lambda_{\overline{\boldsymbol{E}}})_i}$, where $(\lambda_{\overline{\boldsymbol{E}}})_i$ denotes the ith nonzero eigenvalue for the matrix $\overline{\boldsymbol{E}}^T\overline{\boldsymbol{E}}$. However, due to the computational errors introduced by the numerical methods, quite a number of singular values are very small and close to zero. In the following discussions, only the singular values that are larger than 1.0E−6 are taken into consideration.

According to various tasks to be performed, different EAFs can be adopted for the evaluation of error properties. In order to ensure the accuracy of the manipulation of the PKM anywhere in the workspace, the maximum value of $\sigma_{\overline{\boldsymbol{E}}}$ over the workspace, i.e.,

$$\eta_1 = \max\{\sigma_{\overline{\boldsymbol{E}}}\} \tag{38}$$

should be smaller than a specified value. Thus, it seems that the η_1 is the most important index for the PKM. Additionally, in order to ensure the uniformity of error amplification of a 3-CRC PKM all over the workspace, the condition number of ETM,

$$\eta_2 = \frac{\max\{\sigma_{\overline{\boldsymbol{E}}}\}}{\min\{\sigma_{\overline{\boldsymbol{E}}}\}}, \tag{39}$$

i.e., the ratio of the maximum to minimum values of $\sigma_{\overline{\boldsymbol{E}}}$ should be as small as possible.

Furthermore, for the sake of evaluating the error characteristics of a 3-CRC PKM in a global sense, a mixed error amplification index (EAI) is proposed as follows:

$$\varepsilon = w_1\eta_1 + (1 - w_1)\eta_2, \tag{40}$$

where the weight parameter w_1 ($w_1 \in [0\ \ 1]$) describes the proportion of η_1 in the mixed index. Since the EAI is the compromise of the two EAFs η_1 and η_2, when $w_1 = 0$, EAI $= \eta_2$, and in case of $w_1 = 1$, EAI$= \eta_1$. The smaller the EAI is, the lower pose error becomes. Therefore, EAI can be used as an accuracy criteria for the architecture optimization of the 3-CRC PKM.

4.3. A case study

As an example, the architectural parameters of a 3-CRC PKM are designed as $\alpha = 45°$, $a = 0.5$ m, $b = 0.3$ m, l =0.5 m, $s_{\max} = 0.1$ m, and $d_{\max} = 0.2$ m, the distributions of the EAI for $w_1 = 0$, 0.3, 0.6, and 1.0 in three planes at heights of $z = -0.358$ m, -0.458 m (home position), and -0.558 m are illustrated in Fig. 5(a)–(d), respectively.

It is observed that the distributions of EAI with different values of the weight number in each x–y plane are all 120° symmetrical about the axial directions of three P joints. In addition, the worst EAI occurs around the boundary of

Table I. Maximum error values.

$\lvert\delta q\rvert_{\max}$	$\lvert\delta l\rvert_{\max}$	$\lvert\lvert\delta \boldsymbol{a}\rvert\rvert_{\max}$	$\lvert\lvert\delta \boldsymbol{d}\rvert\rvert_{\max}$	$\lvert\lvert\delta \boldsymbol{b}'\rvert\rvert_{\max}$	$\lvert\lvert\delta \boldsymbol{s}\rvert\rvert_{\max}$
0.2 mm	0.5 mm	0.5 mm	0.5 mm	0.5 mm	0.5 mm

the workspace since the manipulator approaches singularity when it comes near the workspace boundary. Moreover, the largest value of EAI for any values of w_1 occurs on the lowest height plane in the workspace. Along with the increasing of the weight number w_1 from 0 to 1, the EAI diminishes monotonously as shown in Fig. 5.

5. Optimization of a 3-CRC PKM via Different Approaches

The three approaches in terms of NMS method, GA, and PSO have been implemented for the architectural optimization of a class of 3-CRC PKM in order to generate the lowest value of the maximum EAI within the workspace.

5.1. Design variables and objective function

The architectural parameters of a 3-CRC PKM involve the layout angle of actuators (α), sizes of the fixed base platform (a) and the mobile platform (b), and the length of legs (l). In order to perform the architecture optimization independent of the dimension of each design candidate, the last three parameters are scaled by the stroke limit of the linear actuators $d_{\max} = S$. Thus, the design variables become α, $\frac{a}{S}$, $\frac{b}{S}$, and $\frac{l}{S}$. To ensure a real geometry and compact structure of the manipulator, the design variables are limited within the parameter spaces of $0^\circ \le \alpha \le 90^\circ$, $1.5 \le \frac{a}{S} \le 5.0$, $1.2 \le \frac{b}{S} \le 5.0$, and $1.0 \le \frac{l}{S} \le 5.0$, respectively. Additionally, for a 3-CRC PKM, let the home position of the mobile platform be in the case of mid stroke of linear actuators, i.e, $d_i = 0$ ($i = 1, 2$, and 3). The predicted maximum error source values subject to the control and calibration accuracy are elaborated in Table I.

The objective function for minimization is chosen based on the introduced EAI, i.e.,

$$f(n) = \max_{i \in U}\{\varepsilon_i\}, \qquad (41)$$

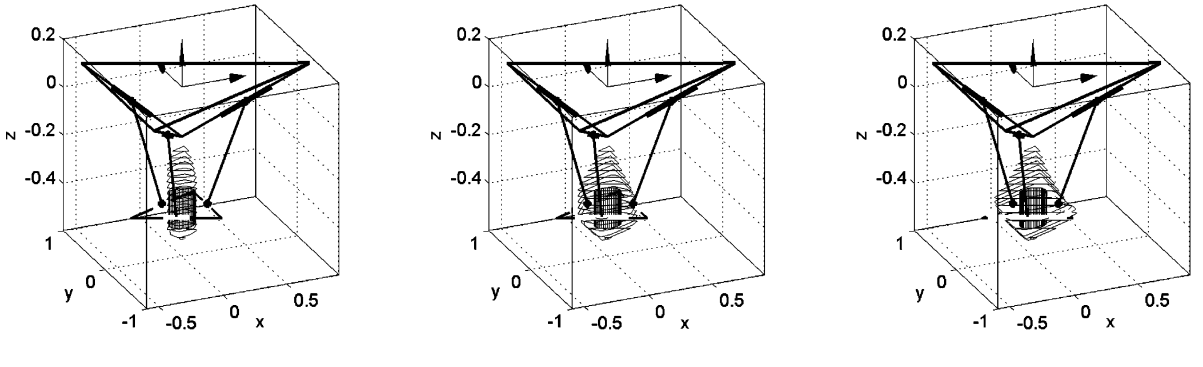
where $n = [\alpha \;\; \frac{a}{S} \;\; \frac{b}{S} \;\; \frac{l}{S}]^T$ denotes the design variables, and ε_i is the EAI at the ith position within the usable workspace (U) of a 3-CRC PKM with a particular set of parameters (n). Then, the optimization problem can be summarized as follows:

- Minimize: $f(n)$
- Variables to be optimized: $n = [\alpha \;\; \frac{a}{S} \;\; \frac{b}{S} \;\; \frac{l}{S}]^T$
- Subject to: $\alpha \in [0^\circ \;\; 90^\circ]$, $\frac{a}{S} \in [1.5 \;\; 5.0]$, $\frac{b}{S} \in [1.2 \;\; 5.0]$, and $\frac{l}{S} \in [1.0 \;\; 5.0]$

Once the stroke limit $d_{\max}$ of the linear actuator is given as S, the reachable workspace of the PKM with different values of the passive C joint stroke limit $s_{\max}$ can be illustrated in Fig. 6. It can be observed that once the actuator's stroke is determined, the workspace volume shrinks as $s_{\max}$ descends. On the other hand, the workspace grows up as the stroke value $s_{\max}$ increases. However, the workspace cannot increase infinitely since the stroke $s_{\max}$ is restricted by the size of the mobile platform. In order to install the passive C joint on the equilateral triangle shape mobile platform, the stroke is limited by $s_{\max} \le b \tan(\pi/3)$. Here, the extreme value $s_{\max} = \sqrt{3}\, b$ is selected to obtain a larger reachable workspace.

From Fig. 5, one can observe that the PKM has a poor accuracy property around the workspace boundary. Hence, it is reasonable to restrict the PKM to perform tasks in a usable workspace, i.e., a central subworkspace inside the reachable workspace, which is far away from singularities. According to the tasks to be carried out by the PKM, there may be several ways to define the shape of the subworkspace. Here, the usable workspace is designed to be a cylinder since the cross section of the central part of the reachable workspace takes on a hexagon shape. Besides, for the simplicity of design and analysis, the size of the cylinder shape usable workspace is assigned with the radius of $S/2$ and height of $2S/3$, whose geometry center (with the height of z_0) locates at the home position point of the mobile platform. Thus, the usable workspace lies in almost the center of the reachable workspace, and these two types of workspace are shown in Fig. 6 as well.

Again, from the distributions of EAI over the reachable workspace with different values of the weight number in Fig. 5, it can be deduced that, for a particular geometry of the robot, the largest value of EAI, i.e., $f(n)$, arises



(a) $s_{\max} = \frac{1}{2}d_{\max}$. (b) $s_{\max} = d_{\max}$. (c) $s_{\max} = \frac{3}{2}d_{\max}$.

Fig. 6. Illustration of the workspace for a 3-CRC PKM.

on the boundary of the bottom circular planes of cylinder usable workspace, i.e., a circle of radius $S/2$ at the height of $z_0 - S/3$. Instead of the time-consuming searching over the entire usable workspace, we can solely calculate the EAI on the edge of the circular bottom plane in order to evaluate $f(n)$. Moreover, only one third of a circle, i.e., a 120° arc can be considered since the distributions of EAI within the workspace have a symmetry at 120° when looked down, just similar to the shape of reachable workspace. These deductions will greatly simplify the procedure of accuracy evaluation in terms of computational time. Thus, in the numerical analysis, the performance of one PKM is evaluated on one third of a circle by a discretization method where the 120° arc is sampled by an interval of 10°.

5.2. Optimization settings

Concerning the NMS method, the starting guesses vector $n_0 = [\alpha_0 \ \frac{a_0}{S} \ \frac{b_0}{S} \ \frac{l_0}{S}]^T$ for the four variables is selected at random from the search space. The termination tolerances for n and $f(n)$ are assigned to be 1.0E−6.

Regarding the GA approach, the normalized geometric selection is adopted, and genetic operators are chosen to be non-uniform mutation with the ratio of 0.08 and arithmetic crossover with the ratio of 0.8, respectively. In addition, the population size is assigned as 30 and the maximum generation number is set to 2000, respectively.

In order to apply PSO for the PKM architectural optimization, several fundamental issues are required to be determined at first. In the current four-dimensional optimization problem, a particle can be described by $X_i = (x_{i1}, x_{i2}, x_{i3}, x_{i4})$ with the particle velocity $V_i = (v_{i1}, v_{i2}, v_{i3}, v_{i4})$, which corresponds to a set of the PKM design variables $(\alpha, \frac{a}{S}, \frac{b}{S}, \frac{l}{S})$. In the nth generation, there are N particles $pop^n = (X_1, X_2, \cdots, X_N)$, where the population size N is also set to be 30 for the current problem. The inertia weight w determines the impact of previous velocities on the current velocity, and the initial and final values are selected as 0.9 and 0.4, respectively, where 500 epoches are allowed to take from the initial value to the final one linearly. In addition, the local and global acceleration constants are assigned as $c_1 = 2.0$ and $c_2 = 2.0$, respectively. As far as the termination criterion is concerned, three items are set up. One criterion is the maximum number of iterations (2000) for the optimization procedure, another one is the minimum global error gradient (1.0E−6) which is the error between two neighboring $gBest$, and the third one is the maximum number of iterations without error change which is chosen as 300.

Additionally, both the GA and PSO are initialized with random start values within the search space. The three optimization programs are developed in MATLAB environment, and a total of 10 independent runs are carried out on a personal computer (Intel Pentium-4, 3.0 GHz CPU, 512 MB RAM) for each approach.

5.3. Discussions on optimization results

Generally, due to the stochastic properties of the NMS, GA, and PSO approaches, i.e., they make random choices, the results obtained from each run are not the same exactly, as shown in Tables II, III, and IV, respectively, where the optimizations are performed with respect to the weight number $w_1 = 0.6$. In addition, for the reason of comparison, the total calculation time, the best, worst, mean, median, and standard deviation of the fitness values for the 10 optimization results generated by each method are elaborated in Table V.

Table II. Optimization results of 10 independent runs from NMS optimization.

Result set	α (rad)	a/S	b/S	l/S	$f(n)$
1	0.8793	4.1110	2.2997	2.5673	1.0534
2	0.1097	5.0000	2.1707	3.8248	1.0625
3	0.8656	4.9087	2.8685	2.7942	1.0300
4	0.8612	5.0000	2.8535	2.8990	**1.0283**
5	1.5708	4.4400	2.2690	3.3463	1.0527
6	0.8293	4.9981	2.6183	3.1430	1.0298
7	1.3479	5.0000	2.8008	3.2319	1.0344
8	0.7017	4.9999	3.3218	2.5258	1.0685
9	0.7120	5.0000	2.8312	2.9979	1.0307
10	1.4139	5.0000	2.7146	3.4079	1.0360

Table III. Optimization results of 10 independent runs from GA optimization.

Result set	α (rad)	a/S	b/S	l/S	$f(n)$
1	0.8605	4.9113	2.7002	2.9618	1.0301
2	0.8347	5.0000	2.7062	3.0565	1.0286
3	0.8037	5.0000	2.4915	3.2824	1.0305
4	0.8216	5.0000	2.6178	3.1495	1.0293
5	0.8030	5.0000	2.5262	3.2490	1.0302
6	0.8686	4.9724	2.8436	2.8777	**1.0284**
7	0.8187	5.0000	2.6105	3.1582	1.0294
8	0.7938	4.9995	2.3634	3.4125	1.0320
9	0.8384	4.9310	2.5713	3.1183	1.0307
10	0.8747	4.9118	2.8858	2.7754	1.0294

Table IV. Optimization results of 10 independent runs from PSO optimization.

Result set	α (rad)	a/S	b/S	l/S	$f(n)$
1	0.8784	5.0000	2.9687	2.7784	1.0277
2	0.8753	5.0000	2.9586	2.7899	1.0277
3	0.8782	5.0000	2.9644	2.7827	1.0277
4	0.8795	5.0000	2.9697	2.7768	1.0277
5	0.8770	5.0000	2.9674	2.7805	1.0277
6	0.8762	5.0000	2.9622	2.7859	1.0277
7	0.8782	5.0000	2.9636	2.7835	1.0277
8	0.8790	5.0000	2.9705	2.7764	1.0277
9	0.8789	5.0000	2.9687	2.7781	1.0277
10	0.8779	5.0000	2.9688	2.7787	1.0277

It is observed from Table II that the 10 sets of optimization results obtained by the NMS approach are very different, which mainly come from the reason that the NMS algorithm is very sensitive to the initial guesses. With respect to the GA outputs as shown in Table III, we can see the optimization results are more acceptable compared to the NMS method, since the GA results agree more with one another. On the contrary, regarding the PSO approach, since it is nonsensitive to the initial starts and is efficient in searching,

Table V. Comparison of optimization results from the three different approaches.

	NMS	GA	PSO
Total time (min)	4.8	20.6	53.2
Best fitness	1.0283	1.0284	1.0277
Worst fitness	1.0685	1.0320	1.0277
Mean fitness	1.0426	1.0299	1.0277
Standard deviation	0.0151	0.0011	0.0000

all of the 10 independent optimizations produce the same results approximately as shown in Table IV. And the standard deviation of the fitness values is much less than 0.5E−5 and very small compared with other two approaches as indicated by Table V. A comparison of the best fitness value obtained by the three approaches reveals that the PSO method gives better fitness value (1.0277) than the other two approaches (1.0283 and 1.0284), which results in a manipulator whose parameters may be described by any set of the 10 results in Table IV.

Furthermore, from the GA results in Table III, we can see that some results with closed fitness values correspond to very different architecture parameters (e.g., the first and fifth set of results). One possible reason is that the combination of some architecture parameters leads to almost the same objective function values. Another possible explanation is that the fitness value is not clear enough, and the objective function should be amplified (e.g., multiplied by a factor of 1.0E+6), for the sake of exposing the small value (behind the radix point) of the original objective function clearly. Fortunately, in view of the PSO results in Table IV, we can observe that although the same fitness value corresponds to different architecture parameters, the differences between the parameters are very slight and can be neglected in some situations. For instance, if one number after the radix point is required, all optimized architecture parameters become the same set, i.e., $\alpha = 0.9$, $a/S = 5.0$, $b/S = 3.0$, and $l/S = 2.8$. In consequence, since the relatively reliable results based on the current objective function can be created by the PSO method, the efforts to modify the objective function in Eq. (41) may be saved in a sense.

Moreover, the convergent processes of the GA and PSO are illustrated in Fig. 7. It is noticeable that the negative sign ahead of the fitness values is used to convert a minimization problem into a maximization one for the convenience of computation. It can be observed that convergence rate of the PSO method is sharper and quicker than that of the GA method. In view of the computational time as shown in Table V, we can see that the PSO requires the longest time to output the 10 sets of results, which is almost 2.6 times longer than the GA approach and 10 times slower than the NMS algorithm.

An overview of the simulation results demonstrates that the PSO method has no sensitivity to the initial conditions like the NMS means and has a better convergence rate than a GA procedure, which is at the expense of a longer calculation time. Considering that it can give the results with the best fitness value, PSO is the best among the three methods for the architectural optimization of a 3-CRC PKM provided that a longer computational time is allowed. For this reason, the PSO is employed in the following studies.

5.4. *Comparison studies*

To illustrate the validity of the proposed EAI, the 3-CRC PKM is optimized with respect to different values of the weight number w_1. The optimization results for 11 values of w_1 are elaborated in Table VI, which shows that the values of $f(n)$ decrease monotonously along with the increasing of the weight number, as previously predicted by Fig. 5. It should be noted that $\max\{\eta_1\}$ and $\max\{\eta_2\}$ represent the maximum value of EAFs η_1 and η_2 over the entire usable workspace, respectively, which are calculated and selected from the boundary circle on the bottom plane of the cylinder usable workspace.

Additionally, it is observed that the optimization with $w_1 = 0$, i.e., to solely minimize $\max\{\eta_2\}$, results in a PKM with $\max\{\eta_1\} = 0.9735$, which is larger than the value of 0.9728 generated from the optimization of minimizing $\max\{\eta_1\}$. On the other hand, the optimization with $w_1 = 1$, i.e., to minimize $\max\{\eta_1\}$ only, leads to a PKM with $\max\{\eta_2\} =$

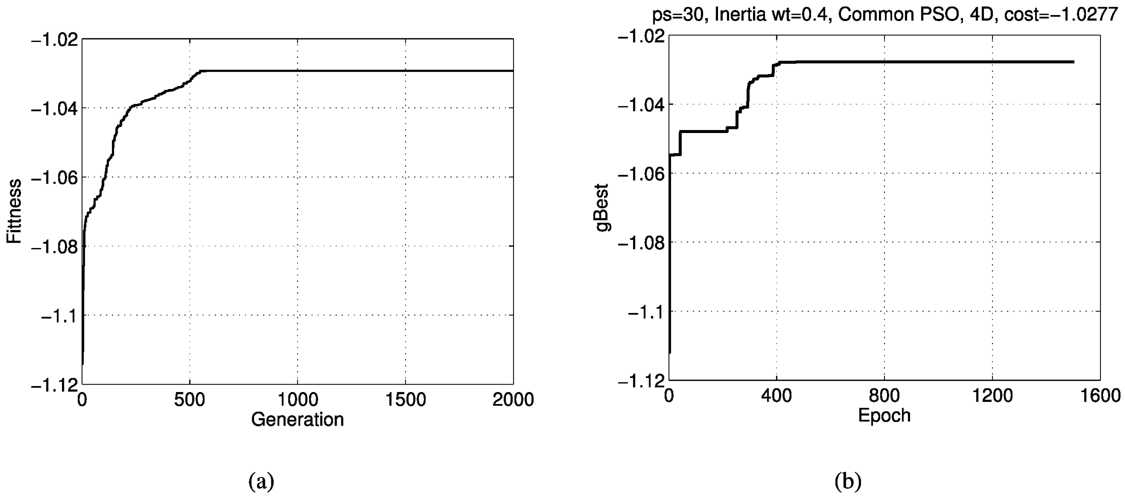


Fig. 7. The convergence processes of (a) GA and (b) PSO optimization.

1.1118, which is greater than the value of 1.1093 generated from the optimization of minimizing $\max\{\eta_2\}$. Consequently, the optimization performed for the best EAF η_1 results in a PKM with worse EAF η_2, and vice versa.

This comparison study reveals in part that it is necessary to introduce a mixed performance index (EAI) for the architectural optimization of a 3-CRC PKM by making a better compromise between the two different EAFs. And the emphasis of the compromise heavily depends on specific objectives as elaborated by the following discussions.

In order to show the effectiveness of the proposed optimization method, the accuracy of the 3-CRC PKM is assessed by the statistical simulation way, which is similar to the approach adopted by other researches.[11] During the simulation, a maximum steady state control error is assigned for $\delta\boldsymbol{q}$ and an allowable manufacturing tolerance is assumed for $\delta\boldsymbol{l}$. The random vectors of $\delta\boldsymbol{a}$, $\delta\boldsymbol{d}$, $\delta\boldsymbol{b}'$, and $\delta\boldsymbol{s}$ are uniformly distributed within an error sphere with the center located at the nominal site of a point and the radius being the maximum allowable installation error. All of the length errors are supposed to follow uniform distribution with the limits specified by the maximum error described in Table I. In addition, the corresponding pose errors of the PKM are calculated by the error model in Eq. (24). The maximum error over the usable workspace can be obtained by the same procedure employed to evaluate the objective function. For a particular PKM, a total of 10,000 sets of random values for $\delta\boldsymbol{q}$, $\delta\boldsymbol{l}$, $\delta\boldsymbol{a}$, $\delta\boldsymbol{d}$, $\delta\boldsymbol{b}'$, and $\delta\boldsymbol{s}$ are selected for the simulation study. And the average value of the 10,000 maximum pose errors is calculated as the maximum error over the usable workspace.

With the variation of the weight value w_1, the architectural parameters of the optimized 3-CRC PKMs are shown in Table VI. In addition, we can define the position and orientation errors of the PKM as:

$$\delta p = \sqrt{(\delta p_x)^2 + (\delta p_y)^2 + (\delta p_z)^2}\,, \tag{42}$$

$$\delta \theta = \sqrt{(\delta \theta_x)^2 + (\delta \theta_y)^2 + (\delta \theta_z)^2}\,. \tag{43}$$

Then, the pose errors of the PKM versus the weight number w_1 are depicted in Fig. 8 along with the assignment of $S = 100\,\text{mm}$.

Table VI. Comparison of optimization results with respect to different objective functions.

w_1	α (rad)	a/S	b/S	l/S	$f(n)$	$\max\{\eta_1\}$	$\max\{\eta_2\}$
0	0.9137	5.0000	3.0206	2.7085	1.1093	**0.9735**	1.1093
0.1	0.9062	5.0000	3.0068	2.7261	1.0958	0.9734	1.1093
0.2	0.8987	5.0000	3.0021	2.7350	1.0822	0.9733	1.1094
0.3	0.8957	5.0000	2.9920	2.7463	1.0686	0.9732	1.1094
0.4	0.8913	5.0000	2.9876	2.7530	1.0550	0.9732	1.1095
0.5	0.8839	5.0000	2.9760	2.7683	1.0413	0.9731	1.1095
0.6	0.8770	5.0000	2.9674	2.7805	1.0277	0.9730	1.1096
0.7	0.8679	5.0000	2.9438	2.8082	1.0140	0.9730	1.1098
0.8	0.8595	5.0000	2.9098	2.8456	1.0003	0.9729	1.1100
0.9	0.8482	5.0000	2.8557	2.9041	0.9866	0.9729	1.1102
1.0	0.8360	5.0000	2.7524	3.0120	0.9728	0.9728	**1.1118**

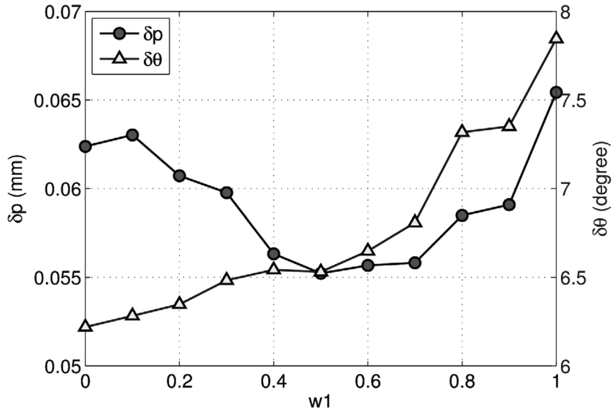


Fig. 8. Position and orientation errors vs. the weight number.

The first observation of Fig. 8 is that the orientation error increases as the weight number rises. Whereas the position error does not have a simple variation tendency. A detailed comparison of pose errors of PKM optimized for different objectives, i.e., different w_1 values, allows the generation of the following deductions:

(a) The optimization with respect to the maximum singular value (η_1) of the EAM only, i.e., $w_1 = 1$, results in a PKM possessing both the largest position error and largest orientation error.
(b) The position error has relatively small value within the range of [0.4, 0.7], and reaches to the smallest value around the weight value of $w_1 = 0.5$.
(c) The orientation error decreases monotonously as the reduction of the weight value and attains the lowest value in case of $w_1 = 0$.
(d) The volumetric errors have more influences on the orientation error than those on the position error.

Besides, the numerical simulation exhibits that the volumetric errors are reflected in the pose errors of the PKM, which shows intuitively the effect of the ETM as represented in Eq. (25). And the errors influence the position displacements and cause rotation displacements of a 3-CRC PKM as well.

In addition to that, the simulation results show that the optimization with emphasis only on the maximum singular value (η_1) or the condition number (η_2) leads to a PKM with relatively large pose errors, which also demonstrates in part the necessity to introduce the new EAI for the PKM optimization. Moreover, the simulation results give some useful guidelines for the optimal design of a 3-CRC PKM. That is, in order to obtain a PKM with both lower position errors and smaller orientation errors, the weight number (w_1) should be chosen around the range between 0.4 and 0.6, i.e., with almost equal emphases on the error amplification factors of η_1 and η_2.

The schematic diagram of a PKM optimized with respect to the weight value $w_1 = 0.5$ is illustrated in Fig. 9. It can be recognized that the accuracy property may not be the sole design criterion for a general PKM design optimization problem. For a practical application, the 3-CRC PKM is preferred to be designed along with other performances in terms of stiffness, workspace, and dexterity taken into

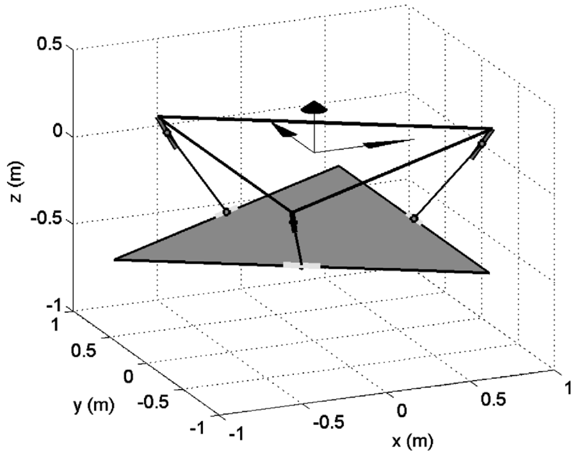


Fig. 9. Schematic diagram of an optimized 3-CRC PKM with $w_1 = 0.5$.

consideration simultaneously. And such a multi-objective optimization for a 3-CRC PKM is planned in the next step of our research.

6. Conclusions

The optimal design of a 3-CRC PKM is carried out for the best accuracy performance in this paper. The dimensionally homogeneous total ETM relating each error source to the manipulator pose errors is established in detail. Through a survey of the existing error evaluation approaches, a new mixed EAI making a compromise between the maximum singular value and condition number of the ETM over a usable workspace is introduced, which can be adopted as the performance index of the PKM for machine tool applications. In order to obtain an optimal accuracy property over the usable workspace, which is defined as a cylinder inscribed at the center of the reachable workspace, the architectural parameters of the PKM have been optimized by employing three different approaches of NMS, GA, and PSO, respectively. The simulation results show that the PSO approach can generate the best PKM parameters with a longer calculation time, and it is the best method for the optimization since it has no sensitivity to the initial conditions like the NMS method, and possesses a better convergence rate than a GA procedure provided that a longer computational time is available. Furthermore, the PSO is adopted for the optimization of the PKM with respect to different objectives, and the optimization results reveal that the optimization for any one of the individual objectives results in a PKM with relatively large pose errors, while the optimization based on the proposed mixed EAI may lead to a PKM possessing a comparatively high accuracy over the usable workspace. For the sake of obtaining a PKM with the smallest pose errors, the weight number in the EAI should be chosen around 0.4–0.6.

The main contribution of the paper includes the error modeling of a 3-CRC PKM and the architecture optimization with respect to a proposed mixed error index via the efficient PSO approach. The results presented in this paper provide a sound base for the design of a 3-CRC PKM prototype with accuracy properties taken into account. Moreover, the methodology proposed here can be easily extended to the optimum design of a class of PKMs as well. Our future works include the optimal design of a 3-CRC PKM with the consideration of other required performances, and the comparison of the PSO method with other promising approaches for the optimization problem.

Acknowledgments

The authors appreciate the fund support from the research committee of University of Macau under grant no.: RG065/06-07S/08T/LYM/FST and Macao Science and Technology Development Fund under grant no.: 069/2005/A.

References

1. G. Yang, I.-M. Chen, W. Lin and J. Angeles, “Singularity analysis of three-legged parallel robots based on passive-joint velocities,” *IEEE Trans. Robot. Autom.* **17**(4), 413–422 (2001).
2. Y. Wang, “An Incremental Method for Forward Kinematics of Parallel Manipulators,” *Proceedings of IEEE International Conference on Robotics, Automation and Mechatronics* (2006), Bangkok, Thailand, pp. 243–247.
3. J. S. Dai, Z. Huang and H. Lipkin, “Mobility of overconstrained parallel mechanisms,” *ASME J. Mech. Des.* **128**(1), 220–229 (2006).
4. J. M. Hervé and F. Sparacino, “Structural Synthesis of Parallel Robots Generating Spatial Translation,” *Proceedings of 5th International Conference on Advanced Robotics* (1991), Pisa, Italy, pp. 808–813.
5. X. Kong and C. M. Gosselin, “Type synthesis of 3-DOF translational parallel manipulators based on screw theory,” *ASME J. Mech. Des.* **126**(1), 83–92 (2004).
6. C. C. Lee and J. M. Hervé, “Translational parallel manipulators with doubly planar limbs,” *Mech. Mach. Theory* **41**(4), 433–455 (2006).
7. J. Wang and O. Masory, “On the Accuracy of a Stewart Platform—Part I: The Effect of Manufacturing Tolerances,” *Proceedings of IEEE International Conference on Robotics and Automation* (1993), Alanta, Georgia, USA, pp. 114–120.
8. H. S. Kim and Y. J. Choi, “The kinematic error bound analysis of the Stewart platform,” *J. Robot. Syst.* **17**(1), 63–73 (2000).
9. T. Ropponen and T. Arai, “Accuracy Analysis of a Modified Stewart Platform Manipulator,” *Proceedings of IEEE International Conference on Robotics and Automation* (1995), Nagoya, Japan, pp. 521–525.
10. T. Li and P. Ye, “The Measurement of Kinematic Accuracy for Various Configurations of Parallel Manipulators,” *Proceedings of IEEE International Conference on Systems, Man and Cybernetics* (2003), Washington, DC, USA, pp. 1122–1129.
11. J. Ryu and J. Cha, “Volumetric error analysis and architecture optimization for accuracy of HexaSlide type parallel manipulators,” *Mech. Mach. Theory* **38**(3), 227–240 (2003).
12. O. Ma and J. Angeles, “Optimum Architecture Design of Platform Manipulators,” *Proceedings of 5th International Conference on Advanced Robotics* (1991), Pisa, Italy, pp. 1130–1135.
13. X.-J. Liu, J. Wang, K.-K. Oh and J. Kim, “A new approach to the design of a DELTA robot with a desired workspace,” *J. Intell. Robot. Syst.* **39**(2), 209–225 (2004).
14. F. Hao and J.-P. Merlet, “Multi-criteria optimal design of parallel manipulators based on interval analysis,” *Mech. Mach. Theory* **40**(2), 157–171 (2005).
15. Y. Li and Q. Xu, “A new approach to the architecture optimization of a general 3-PUU translational parallel manipulator,” *J. Intell. Robot. Syst.* **46**(1), 59–72 (2006).
16. Y. Li and Q. Xu, “Kinematic analysis and design of a new 3-DOF translational parallel manipulator,” *ASME J. Mech. Des.* **128**(4), 729–737 (2006).

17. Q. Xu and Y. Li, “Design and analysis of a new singularity-free three-prismatic-revolute-cylindrical translational parallel manipulator,” *Proc. Inst. Mech. Eng. Part C—J. Mech. Eng. Sci.* **221**(5), 565–577 (2007).
18. Q. Xu and Y. Li, “Stiffness Optimization of a 3-DOF Parallel Kinematic Machine Using Particle Swarm Optimization,” *Proceedings of IEEE International Conference on Robotics and Biomimetics* (2006), Kunming, China, pp. 1169–1174.
19. D. Zhang, Z. Xu, C. M. Mechefske and F. Xi, “Optimum design of parallel kinematic toolheads with genetic algorithms,” *Robotica* **22**(1), 77–84 (2004).
20. J. Kennedy and R. C. Eberhart, “Particle Swarm Optimization,” *Proceedings of International Conference on Neural Networks* (1995), Perth, Australia, pp. 1942–1948.
21. B. Birge, “PSOt—A Particle Swarm Optimization Toolbox for Use with Matlab,” *Proceedings of IEEE Swarm Intelligence Symposium* (2003), Indianapolis, Indiana, USA, pp. 182–186.
22. M. Clerc and J. Kennedy, “The particle swarm-explosion, stability, and convergence in a multidimensional complex space,” *IEEE Trans. Evol. Comput.* **6**(1), 58–73 (2002).
23. Q. Xu and Y. Li, “An investigation on mobility and stiffness of a 3-DOF translational parallel manipulator via screw theory,” *Robot. Comput.-Integrated Manufact.*, **24**(3), 402–414 (2008).