Robotica (2010) volume 28, pp. 349–357. © Cambridge University Press 2009
doi:10.1017/S0263574709005657

Design, analysis, and stiffness optimization of a three degree of freedom parallel manipulator

Zhen Gao†,‡, Dan Zhang§*, Xiaolin Hu§ and Yunjian Ge†

†*Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei, Anhui Province 230031, P. R. China*
‡*Department of Automation, University of Science and Technology of China, Hefei, Anhui Province 230027, P. R. China*
§*Faculty of Engineering and Applied Science, University of Ontario Institute of Technology, Oshawa, Ontario, Canada L1H 7K4*

(Received in Final Form: April 6, 2009. First published online: May 5, 2009)

SUMMARY
This paper proposed a novel three degree of freedom (DOF) parallel manipulator—two translations and one rotation. The mobility study and inverse kinematic analysis are conducted, and a CAD model is presented showing the design features. The optimization techniques based on artificial intelligence approaches are investigated to improve the system stiffness of the proposed 3-DOF parallel manipulator. Genetic algorithms and artificial neural networks are implemented as the intelligent optimization methods for the stiffness synthesis. The mean value and the standard deviation of the global stiffness distribution are proposed as the design indices. Both the single objective and multi-objective optimization issues are addressed. The effectiveness of this methodology is validated with Matlab.

KEYWORDS: Parallel manipulator; Stiffness optimization; Artificial intelligence approaches; degree of freedom; Parallelogram structure.

1. Introduction

Over the past decades, parallel manipulators have received more and more attention from researchers and manufacturers, as parallel manipulators provide the advantages of high rigidity, low inertia, easy for force feedback control, and they have a wide range of applications in the fields of assembly, manufacturing, underground projects, space technologies, and biology projects.[1–6] Examples of this type of robot can be found in the motion platform for pilot training simulators and the positioning device for high-precision surgical tools. Recently, researchers have been trying to utilize these advantages to develop parallel robot-based multi-axis machining tools and precision assembly tools.

Since most machining operations only require a maximum of five axes, new configurations with less than six axes (degree of freedom) would be more appropriate.[7,8] Tripod is one type of parallel kinematic machine (PKM) with three degree of freedom (DOF), and it can implement the basic three axes machining,[9] by combining with the XY gantry, it can conduct 4-DOF or 5-DOF machining. Therefore, tripod mechanism is targeted as the main configuration for the development of parallel manipulators. There are few manipulators with 3-DOF that can combine the spatial rotational and translational motion.[10] A tripod parallel kinematics structure, of which the end-effecter (the moving platform) has three independent motions—two pure translations and one pure rotation—will be proposed and studied in this paper.

Past research and development efforts with parallel manipulators have shown the ever-increasing demand on the robot's rigidity, which is directly related to the system stiffness. In order to increase the production, a parallel manipulator which is capable of high-speed operations with optimal rigidity is necessary. In this paper, the mean value and the standard deviation of the stiffness distribution of the manipulator are proposed as the design indices.

Many researchers have studied on the issue of optimal design of robot manipulators.[11–14] Zhao *et al.*[15] exploited the least number method of variables to optimize the leg length of a spatial parallel manipulator for the purpose of obtaining a desired dexterous workspace. Stock and Miller[16] presented a method for multidimensional kinematic optimization of the linear Delta robot architecture's geometry. A utility objective function was formulated incorporating two performance indices, including manipulability and space utilization. Rout and Mittal[17] proposed the experimental approach for the optimization of the dimension of 2-DOF R-R planar manipulator. Ceccarelli and Lanni[18] investigated the multi-objective optimization problem of a general 3R manipulator for prescribed workspace limits and numerically using an algebraic formulation. As the primary components of artificial intelligence approaches, genetic algorithms and artificial neural networks play the important roles in various fields of science and technology. In this paper, the two methods are applied as the optimization criteria for the stiffness synthesis.

Since the traditional optimization methods can only handle a few geometric variables due to the lack of convergence of the optimization algorithm. However, Genetic algorithms have applied the powerful and broadly applicable stochastic search methods and optimization techniques, and they can escape from local optima.[19] Therefore, genetic algorithms have been selected as the best candidate to address the convergence issue and are suitable for stiffness optimization of the proposed 3-DOF parallel manipulator. Neural networks possess the capability of complex function approximation and generalization by simulating the basic

*Corresponding author. E-mail: dan.zhang@uoit.ca

https://doi.org/10.1017/S0263574709005657 Published online by Cambridge University Press

functionality of the human nervous system in an attempt to partially capture some of its computational strengths. Since the solution of objective function must be solved before using genetic algorithms, neural networks will be conducted to represent the objective functions of the two performance indices for a 3-DOF parallel manipulator. On the investigation of multi-objective optimization problem in design process, since it is impossible to maximize or minimize all objective function values if they are conflicting with each other, the trade-off process should be executed. This methodology will pave the way for providing not only the effective guidance, but also a new approach of dimensional synthesis for the optimal design of general parallel mechanisms.

The remainder of the paper is organized as follows. In Section 2, the structure of the tripod parallel manipulator and its parametric description is introduced. In Section 3, the inverse kinematics and velocity equation are derived based on the geometrical structure. Design process of CAD model including leg structure and platform structure is described in Section 4. In Section 5, the optimization criteria of system stiffness are developed. In Section 6, the process for the applications of the genetic algorithms and neural networks is conducted in order to optimize the performance indices of the proposed parallel manipulator. Both the single objective optimization and the multi-objective optimization are implemented and compared. Finally, the conclusions are given in Section 7.

2. Structure of the Parallel Manipulator

The new 3-DOF parallel manipulator is composed of a base structure, a moving platform and three legs connecting the base and platform. Among those three legs, two of them are in same plane and consist of identical planar four bar parallelograms as chains connected to the moving platform by revolute joints, while the third leg is one rectangular bar connected to the moving platform by a spatial joint. There is one revolute joint on the top end of each leg, and the revolute joint is linked to the base by an active prismatic joint.

The CAD model of the 3-DOF parallel mechanism is shown in Fig. 1.

The DOF for a closed-loop kinematic chain can be determined using the Chebychev–Grübler–Kutzbach formula:[20]

$$M = d(n - g - 1) + \sum_{i-1}^{g} f_i, \tag{1}$$

where M is the system DOF of the assembly or mechanism, d is the order of the system ($d = 3$ for planar motion and $d = 6$ for spatial motion), n is the number of links including the frames, g is the number of joints, and f_i is the number of DOFs for the ith joint, With Eq. (1), the DOF of the proposed parallel manipulator is

$$M = 6(8 - 9 - 1) + (5 + 5 + 5) = 3. \tag{2}$$

In Fig. 1, the parallelograms play the role of improving the kinematics performance and the leg stiffness can be increased largely.[7] In regard to the types of actuated joints, they can

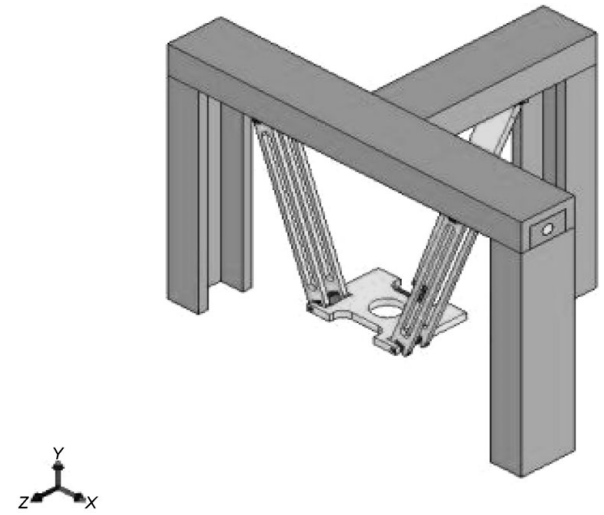


Fig. 1. CAD modeling of 3-DOF parallel manipulator.

be either revolute or prismatic. Since the prismatic joints can easily achieve high accuracy and heavy loads, the majority of the 3-DOF parallel mechanism in reality use actuated prismatic joints.

The output link of a planar parallelogram mechanism will remain in a fixed orientation with respect to its input link, and the parallelogram can ensure the desired output, in terms of translation and rotation. The advantages of the proposed parallel manipulator are:

(1) The use of the parallelogram structure can greatly increase the stiffness of the legs;
(2) Two identical chains offer good symmetry;
(3) The joint which connects the third leg and the moving platform gives the rotation about y-axis with respect to reference frame attached to the end-effecter.

3. Kinematics Analysis and Modeling

3.1. Inverse kinematics

A kinematics model of the manipulator is shown in Fig. 2. The vertices of the moving platform are $p_i (i = 1, 2, 3)$, and the vertices of the base are $b_i (i = 1, 2, 3)$. A global reference system $O : O - xyz$ is located at the point of intersection b_1b_2 and Ob_3. Another reference system, called the moving frame $O' : O' - x'y'z'$, is located at the center of p_1p_2 on the moving platform.

The given position and orientation of the end-effecter (the moving platform) is specified by its three independent motions: y, z pure translations, and ϕ pure rotation about y-axis. The position is given by the position vectors $(O')_O$, and the orientation is given by rotation matrix Q as follows:

$$(O')_O = (x \quad y \quad z)^T, \tag{3}$$

where $x = 0$, and

$$Q = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix}, \tag{4}$$

https://doi.org/10.1017/S0263574709005657 Published online by Cambridge University Press

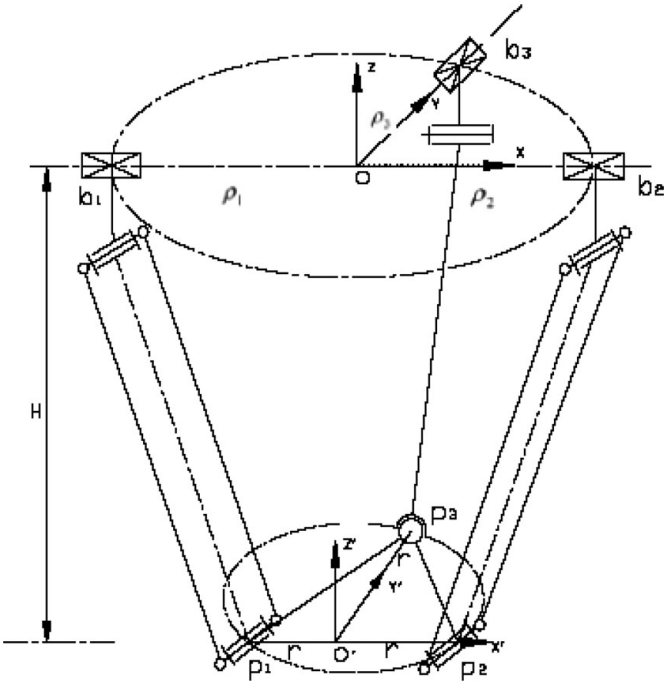


Fig. 2. Schematic representation.

where the angle ϕ is the rotation about y-axis. The coordinates of the point p_i in reference system O' can be described by the vector $(p_i)_{O'}(i = 1, 2, 3)$:

$$(p_1)_{O'} = \begin{pmatrix} -r \\ 0 \\ 0 \end{pmatrix} (p_2)_{O'} = \begin{pmatrix} r \\ 0 \\ 0 \end{pmatrix} (p_3)_{O'} = \begin{pmatrix} 0 \\ r \\ 0 \end{pmatrix}. \quad (5)$$

The vectors $(b_i)_O(i = 1, 2, 3)$ in frame $O : O - xyz$ will be defined as position vectors of joints:

$$(b_1) = \begin{pmatrix} -\rho_1 \\ 0 \\ 0 \end{pmatrix} (b_2) = \begin{pmatrix} \rho_2 \\ 0 \\ 0 \end{pmatrix} (b_3) = \begin{pmatrix} 0 \\ \rho_3 \\ 0 \end{pmatrix}. \quad (6)$$

The vector $(p_i)_O(i = 1, 2, 3)$ in frame $O : O - xyz$ can be written as

$$(p_i)_O = Q(p_i)_{O'} + (O')_O. \quad (7)$$

That is

$$(p_1)_O = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix} \begin{pmatrix} -r \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ y \\ z \end{pmatrix} = \begin{pmatrix} -r\cos\phi \\ y \\ r\sin\phi + z \end{pmatrix}, \quad (8)$$

$$(p_2)_O = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix} \begin{pmatrix} r \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ y \\ z \end{pmatrix} = \begin{pmatrix} r\cos\phi \\ y \\ -r\sin\phi + z \end{pmatrix}, \quad (9)$$

$$(p_3)_O = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix} \begin{pmatrix} 0 \\ r \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ y \\ z \end{pmatrix} = \begin{pmatrix} 0 \\ r + y \\ z \end{pmatrix}. \quad (10)$$

The inverse kinematics of the manipulator can be solved by applying the following constraint equation:

$$\|p_i - b_i\| = L. \quad (11)$$

Hence, one can obtain the required actuator inputs:

$$\rho_1 = \sqrt{L^2 - y^2 - (z + r\sin\phi)^2} + r\cos\phi, \quad (12)$$

$$\rho_2 = \sqrt{L^2 - y^2 - (z - r\sin\phi)^2} + r\cos\phi, \quad (13)$$

$$\rho_3 = \sqrt{L^2 - z^2} + y + r. \quad (14)$$

3.2. Velocity equation

Equations (12)–(14) can be differentiated with respect to time to obtain the velocity equations:

$$(\rho_1 - r\cos\phi)\dot{\rho}_1 + y\dot{y} + (z + r\sin\phi)\dot{z} + r(\rho_1\sin\phi + z\cos\phi)\dot{\phi} = 0, \quad (15)$$

$$(\rho_2 - r\cos\phi)\dot{\rho}_2 + y\dot{y} + (z - r\sin\phi)\dot{z} + r(\rho_2\sin\phi - z\cos\phi)\dot{\phi} = 0, \quad (16)$$

$$[\rho_3 - (y + r)]\dot{\rho} + [\rho_3 - (y + r)]\dot{y} + z\dot{z} = 0. \quad (17)$$

Rearranging Eqs. (15)–(17) we have

$$A\dot{\rho} = B\dot{p}, \quad (18)$$

where $\dot{\rho}$ is the vector of input velocities defined as

$$\dot{\rho} = (\dot{\rho}_1, \dot{\rho}_2, \dot{\rho}_3)^T \quad (19)$$

and $\dot{p}$ is the vector of output velocities defined as

$$\dot{p} = (\dot{y}, \dot{z}, \dot{\phi})^T. \quad (20)$$

Matrices A and B can be expressed as

$$A = \begin{bmatrix} r\cos\phi - \rho_1 & 0 & 0 \\ 0 & r\cos\phi - \rho_2 & 0 \\ 0 & 0 & y + r - \rho_3 \end{bmatrix}, \quad (21)$$

https://doi.org/10.1017/S0263574709005657 Published online by Cambridge University Press

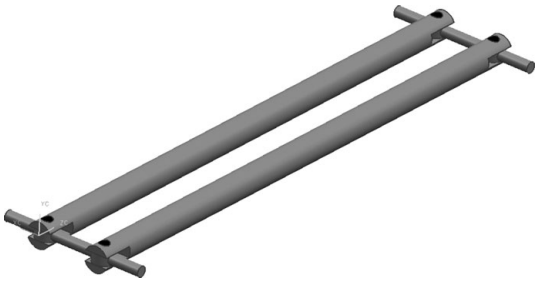


Fig. 3. The CAD model of the parallelogram legs.

$$B = \begin{bmatrix} y & z + r\sin\phi & r(\rho_1 \sin\phi + z\cos\phi) \\ y & z - r\sin\phi & r(\rho_2 \sin\phi - z\cos\phi) \\ y + r - \rho_3 & z & 0 \end{bmatrix}. \tag{22}$$

The Jacobian matrix of the manipulator can be written as

$$J = A^{-1}B \quad \text{or} \quad K = J^{-1} = B^{-1}A. \tag{23}$$

4. Design of the Manipulator

The legs of parallel manipulator are directly related to motion, acceleration, singularities, and other performance characteristics. Two identical legs utilized in the design use planar parallelograms as shown in Fig. 3. The parallelogram leg is connected to a linear motion (prismatic) actuator on one end, and connected to the platform on the other end. The complete joint structure of the parallelogram leg is PR(Pa)R.

The third leg is attached to the moving platform via a spherical as shown in Fig. 4, the joint arrangement of the third leg is PRS.

In the current design, the base is simply a T-shape platform (Fig. 5). This allows the two legs with parallelogram to act as a coupled pair and provide rigid support to the moving platform.

5. Design Optimization

5.1. Optimization principles

The purpose of optimization design is to enhance the performance indices by adjusting the structure parameters. We propose the mean value and the standard deviation of the global stiffness as the design indices in this paper. It is noted that the trace of the matrix is an invariant of the matrix, so the distribution of the system stiffness (matrix) is the distribution of the trace. The mean value represents the average stiffness of the parallel robot manipulator over the workspace, while the standard deviation indicates the stiffness variation relative to the mean value. In general, the higher the mean value the less the deformation, and the lower the standard deviation the more uniform the stiffness distribution over the workspace. In this paper, the suitability of these design indices for the system stiffness will be examined by developing their relationship with the stiffness of links and joints. We will further study a design optimization based on the stiffness indices. A multi-objective optimization problem will be defined. Ideally, it may require that the mean value should be a maximum, but the standard deviation is a minimum. However, these two goals could be in conflict, so a trade-off process (e.g., Pareto set theory) will be considered.

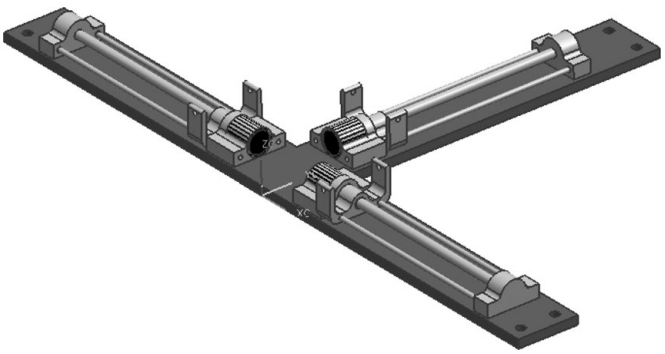


Fig. 5. The base platform with tracks mounted.

The goal of structure parameters design, which is also called dimensional synthesis, is to confirm the best geometric configuration according to objective function and geometric restriction.

Since only a few geometric parameters can be handled due to the lack of convergence, this arises from the fact that traditional optimization methods use a local search by a convergent stepwise procedure, e.g., gradient, Hessians, linearity, and continuity, which compares the values of the next points and moves to the relative optimal points.[21] Global optima can be found only if the problem possesses certain convexity properties which essentially guarantee that any local optima are a global optimum. In other words, conventional methods are based on a point-to-point rule; it has the danger of falling in local optima. The genetic algorithms are based on the population-to-population rule; it can escape from local optima.

For the implementation of genetic algorithms, one problem is how to model the objective function. It is very difficult and time-consuming exercise especially when the parameters are multifarious and the objective functions are too complex that genetic algorithm cannot work well based on the analytical expression of the performance indices. In artificial neural networks implementation, knowledge is represented as numeric weights, which are used to gather the relationships between data that are difficult to realize analytically. The network parameters can be iteratively adjusted to



Fig. 4. The CAD model of the third leg.

https://doi.org/10.1017/S0263574709005657 Published online by Cambridge University Press

minimize the sum of the squared approximation errors using a gradient descent method, thereby being utilized to represent the system stiffness for the 3-DOF parallel manipulator.

5.2. Solution of system stiffness

Stiffness is a very important factor in many applications including machine tool design, as it affects the precision of machining. Induced vibration is explicitly linked to machine tool stiffness. For a metal cutting machine tool, high stiffness allows higher machining speeds and feeds while providing the desired precision, thus reduce vibration (such as chatter). Therefore, to build and study a general stiffness model of parallel mechanisms is very important for machine tool design.

From the viewpoint of mechanics, the stiffness is the measurement of the ability of a body or structure to resist deformation due to the action of external forces. The stiffness of a parallel mechanism at a given point of its workspace can be characterized by its stiffness matrix. This matrix relates the forces and torques applied at the gripper link in Cartesian space to the corresponding linear and angular Cartesian displacements.

Two main methods have been used to establish mechanism stiffness models. The first one is called matrix structural analysis, which models structures as a combination of elements and nodes. The second method relies on the calculation of the parallel mechanism's Jacobian matrix, which is adopted in this paper.

The velocity relationship of parallel mechanisms can be written as

$$\dot{\theta} = J\dot{x}, \tag{24}$$

where $\dot{\theta}$ is the vector of joint rates, and $\dot{x}$ is the vector of Cartesian rates, a six-dimensional twist vector containing the velocity of a point on the platform and its angular velocity. Matrix J is the Jacobian matrix in Eq. (23).

From Eq. (24), one can conclude that

$$\delta\theta = J\delta x, \tag{25}$$

where $\delta\theta$ and δx represent joint and Cartesian infinitesimal displacements, respectively. Then, one can get the stiffness of this mechanism using the principle of kinematic/static duality. The forces and moments applied at the gripper under static conditions are related to the forces or moments required at the actuators to maintain the equilibrium by the transpose of the Jacobian matrix J. one can then write

$$F = J^T f, \tag{26}$$

where f is the vector of actuator forces or torques, and F is the generalized vector of Cartesian forces and torques at the gripper link. The actuator forces and displacements can be related by Hooke's law, one has

$$f = K_J \delta\theta. \tag{27}$$

Here K_J is the joint stiffness matrix of the parallel mechanism, with $K_J = \mathrm{diag}[k_1, \ldots, k_n]$, where each of the actuators in the parallel mechanism is modeled as an elastic component. k_i is a scalar representing the joint stiffness of each actuator, which is modeled as linear spring. Substituting Eq. (25) into Eq. (27), one obtains

$$f = K_J J\delta x. \tag{28}$$

Then, substituting Eq. (28) into Eq. (26) yields

$$F = J^T K_J J\delta x. \tag{29}$$

Hence, K_C, the stiffness matrix of the mechanism in the Cartesian space is then given by the following expression:

$$K_C = J^T K_J J. \tag{30}$$

Particularly, in the case for which all the actuators have the same stiffness, i.e., $k = k_1 = k_2 = \cdots = k_n$, Eq. (30) will be reduced to

$$K_C = kJ^T J. \tag{31}$$

Furthermore, the diagonal elements of the stiffness matrix are used as the system stiffness value. These elements represent the pure stiffness in each direction, and they reflect the rigidity of machine tools more clearly and directly. The objective function for mean value and standard deviation of system stiffness can be written as:

$$\mu\text{-}stiffness = E(tr(K_C)), \tag{32}$$

$$\sigma\text{-}stiffness = D(tr(K_C)), \tag{33}$$

where $E(\cdot)$ and $D(\cdot)$ represent the mean value and the standard deviation respectively, and tr is the trace of the stiffness matrix K_C.

6. Simulation

6.1. Optimization of mean value of system stiffness

In order to obtain the optimal system stiffness of the 3-DOF parallel manipulator, three geometrical parameters are selected as optimization parameters. The vector of optimization variables are

$$s = \{L, h, r\}, \tag{34}$$

where L is link length, h is the height of the moving platform, and r is radii of fixed platform, and their bounds are

$$L \in [1, 2]\,m,\ h \in [0.5, 0.8]\,m,\ r \in [0.1, 0.3]\,m.$$

The standard back propagation learning algorithm, as the most popular training method for feedforward neural network, is based on the principle of steepest descent gradient approach to minimize a criterion function representing the instantaneous error between the actual outputs and the predicted outputs.

https://doi.org/10.1017/S0263574709005657 Published online by Cambridge University Press

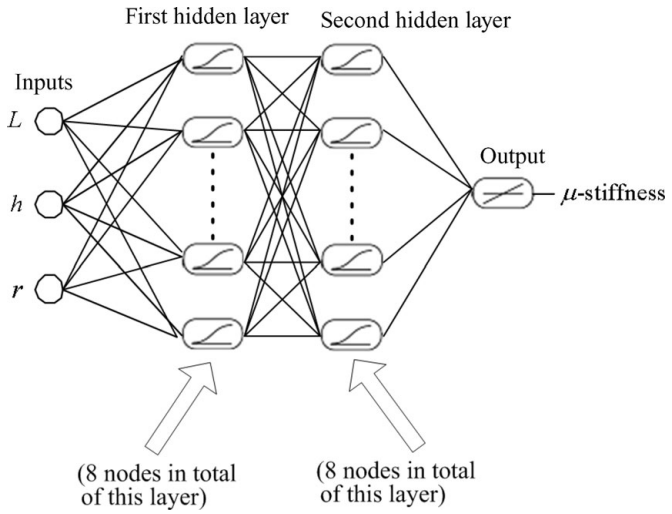


Fig. 6. The topology of feed forward neural network for the solution of μ-stiffness.

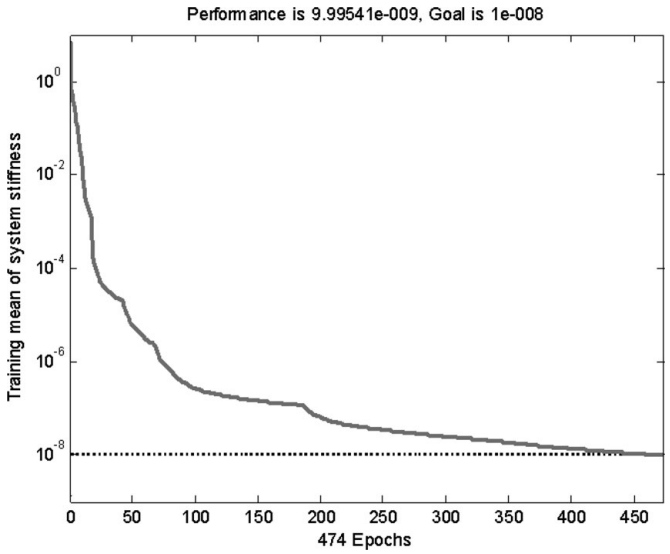


Fig. 7. Network training of the objective function about μ-stiffness.

The criterion function can be expressed as follows:

$$E = \sum_{k=1}^{K} E_k^2 = \frac{1}{2NK} \sum_{k=1}^{K} \left[\sum_{i=1}^{N} (y_{ik} - t_{ik})^2 \right], \qquad (35)$$

where K is the number of output neurons, N is the vector dimension, and y_{ik}, t_{ik} are the predicted outputs and actual outputs of the kth output neuron of the ith input dimension, respectively.

The basic training steps of back propagation neural network are included as follows:

(1) Initialize the weights and bias in each layer with small random values to make sure that the weighted inputs of network wouldn't be saturated.
(2) Confirm the set of input/output pairs and the network structure. Set some related parameters, i.e., the desired minimal E, the maximal iterative times and the learning speed.
(3) Compare the actual output with desired network response and calculate the deviation.
(4) Train the updated weights based on criterion function in each epoch.
(5) Continue the above two steps until the network satisfy the training requirement.

Figure 6 shows the topology of network developed as the objective function to model the analytical solution of mean value of the system stiffness (μ-stiffness). In this case, two hidden layers with sigmoid transfer function are established in which eight neurons exist respectively. The input vectors are the random arrangement of discretization values from the three structure variables.

Figure 7 illustrates the training result using standard back propagation learning algorithm, where the solid curve denotes the quadratic sum of output errors with respect to ideal output values. After training for 474 times, target goal error is arrived.

The genetic algorithms can be implemented to search for the best solutions after the trained neural network is ready

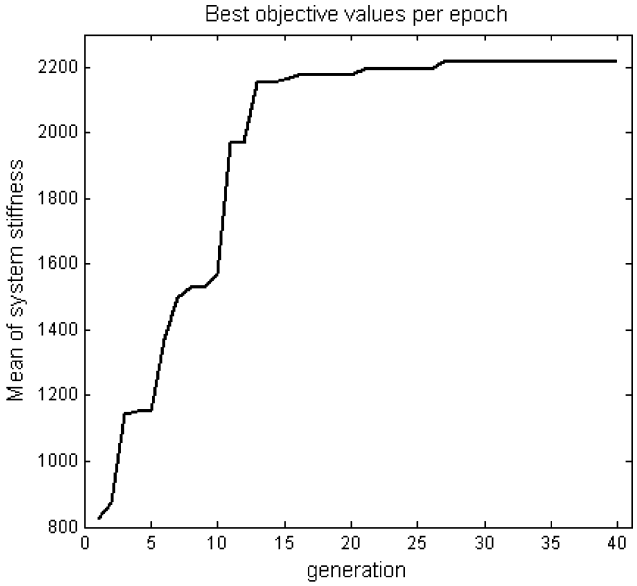


Fig. 8. The evolution of μ-stiffness.

for the objective function. To avoid the time-consuming iterative operation using traditional technologies, the issue of stiffness optimization can be converted into network optimization. Figure 8 shows the evolution process of the best individual (network) based on genetic algorithms. The optimal μ-stiffness value is 2218.

The evolution of μ-stiffness arises from the optimization of architecture and behavior variables in the implementation process of genetic algorithm as shown in Fig. 9. By adjusting the three parameters simultaneously with genetic operators such as selection, crossover and mutation, the optimal objective is obtained. The final values of three parameters searched by genetic algorithm are

$$s = \{L, h, r\} = \{1m,\ 0.8m,\ 0.3m\}.$$

6.2. *Optimization of variance of system stiffness*

The topology of neural networks is similar with the above instance for the optimization of variance of system stiffness

https://doi.org/10.1017/S0263574709005657 Published online by Cambridge University Press

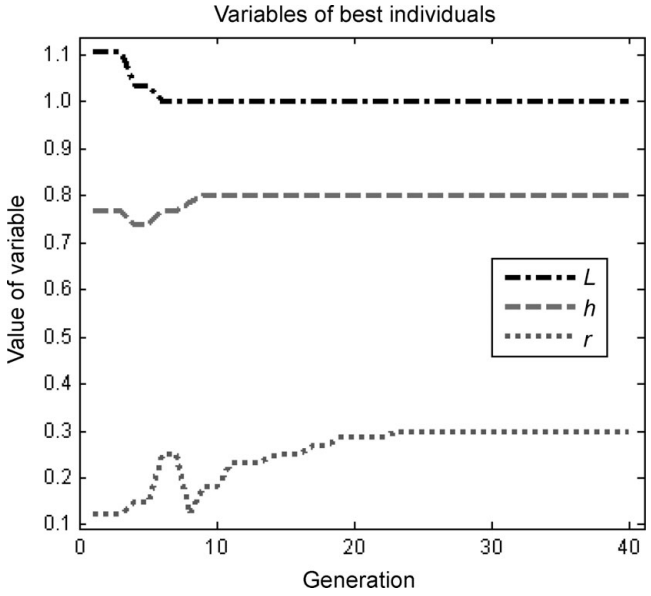


Fig. 9. The evolution of geometrical parameters for μ-stiffness optimization.

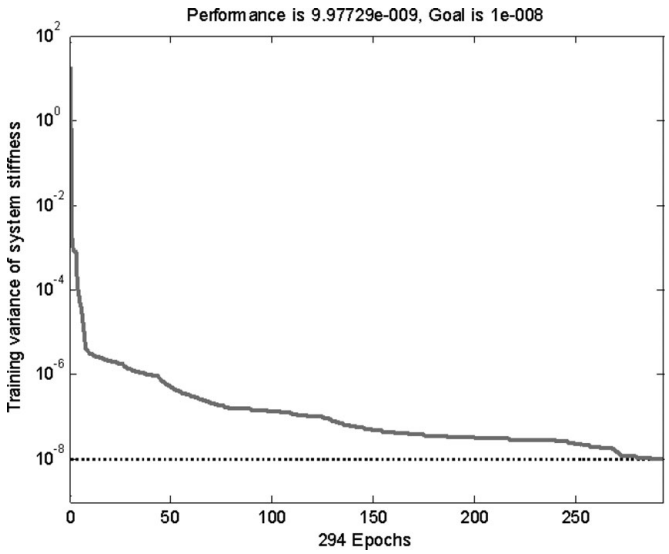


Fig. 10. Network training of the objective function about σ-stiffness.

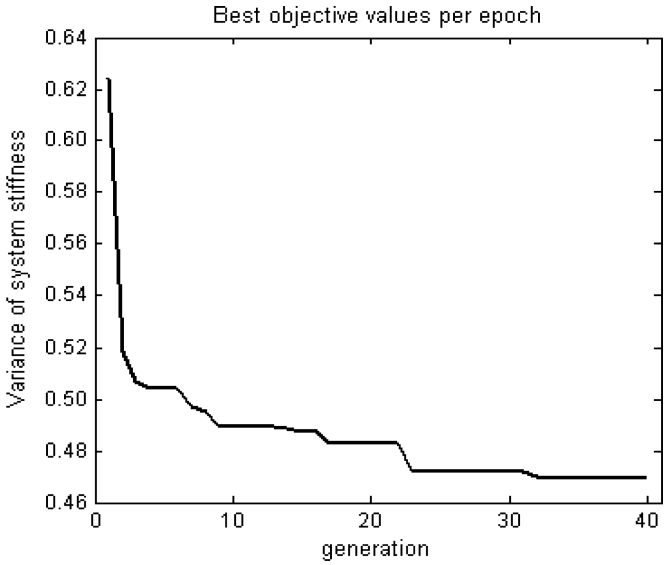


Fig. 11. The evolution of σ-stiffness.

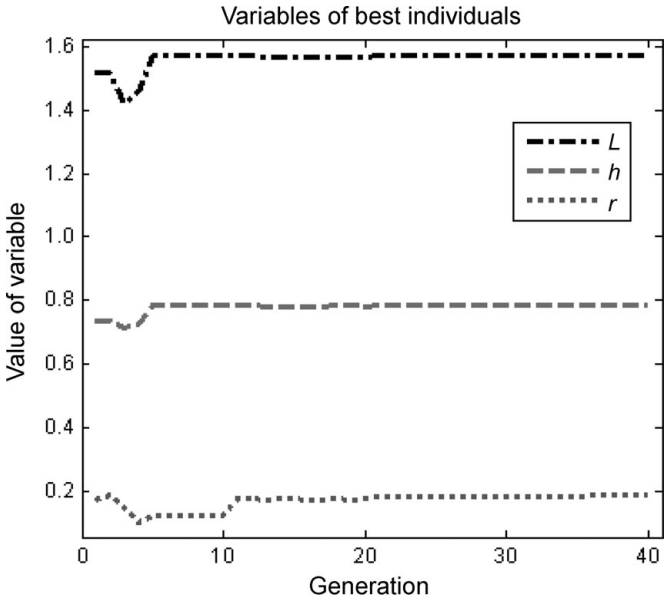


Fig. 12. The evolution of geometrical parameters for σ-stiffness optimization.

(σ-stiffness), in which the only difference is that there are 12 neurons exist in each hidden layer. Figure 10 illustrates the training result of the objective function about σ-stiffness with the back propagation algorithm, where the solid curve denotes the quadratic sum of output errors with respect to ideal output values. After training for 294 times, target goal error is arrived.

The optimization process of σ-stiffness with genetic algorithm is illustrated in Fig. 11. After global stochastic search for 40 generations, optimal σ-stiffness value is convergent at 0.47.

The evolution of geometrical parameters for σ-stiffness optimization is described in Fig. 12. Compared with Fig. 9, it can be found that the corresponding convergent points of the three parameters in these two figures are not the same, i.e.,

$$s = \{L, h, r\} = \{1.5718m, 0.78579m, 0.18845m\}$$

In other words, the two objective functions are conflicted with each other. This issue will be addressed in the following section where multi-objective optimization based on Pareto-optimal solution is conducted.

6.3. Multi-objective optimization

Multi-objective optimization problems consist of simultaneously optimizing several objective functions that are quite different from those of single objective optimization. One single global optimal search is enough for single-objective optimization task. However, in a multi-objective optimization

https://doi.org/10.1017/S0263574709005657 Published online by Cambridge University Press

problem, it requires to find all possible tradeoffs among multiple objective functions that usually conflict with each other. The set of Pareto optimal solutions is generally used for decision maker.

The basic concept of multi-objective optimization is the concept of domination.[22] In the issue of maximizing the k objective functions, decision vector (sets of variable) $x^* \in F$ is the Pareto-optimal solution if no other decision vectors satisfy both the following conditions:

$$f_i(x) \geq f_i(x^*), \forall i \in \{1, 2, \ldots, k\}, \quad (36)$$

$$f_j(x) > f_j(x^*), \exists j \in \{1, 2, \ldots, k\}. \quad (37)$$

With the same method, if both of the following conditions are true, decision vector x dominates y in the maximization issue, noted by $x > y$. That is:

$$f_i(x) \geq f_i(y), \forall i \in \{1, 2, \ldots, k\}, \quad (38)$$

$$f_j(x) > f_j(x^*), \exists j \in \{1, 2, \ldots, k\}. \quad (39)$$

According to the Eqs. (36)–(39), Pareto-optimal set can be defined as: if there is no solution in the search space which dominates any member in the set P, then the solutions belonging to the set P constitute a global Pareto-optimal set. The Pareto optimal set yields an infinite set of solutions, from which the desired solution can be chosen. In most cases, the Pareto optimal set is on the boundary of the feasible region. Typical application of Pareto based approach can be found in ref. [23]. Since the implementation of genetic algorithms—including selection, crossover, and mutation operation—focuses on the whole colony which is consisted by all individuals, and generally, Pareto-optimal solutions for multi-objective optimization issues are a multidimensional set. Therefore, genetic algorithms are the effective method to address the Pareto-optimal solutions of multi-objective optimization issues.

Following initial parameters of Pareto based on genetic algorithms are set before implementation:

- Number of sub-population = 3
- Number of individuals in each sub-population = {50, 30, 40}
- Mutation range = 0.01
- Mutation precision = 24
- Max generations for algorithm termination = 200.

After optimization, the possible optimal solutions in the whole solution space are obtained without combining all the objective functions into a single one by weighting factors. Figure 13 shows the Pareto optimal frontier sets in which the designers can intuitively determine the final solutions depending on their preferences. Hence, the analysis process and cycle time is reduced. From this picture, the trade-off between the objectives of μ-stiffness and σ-stiffness is demonstrated in the distributing trend of these Pareto points for compromisingly selecting. If any other pair of design variables is chosen from the lower/right area of Fig. 13, its corresponding values will locate an inferior point with respect to the Pareto frontier. Besides, the upper/left side is the inaccessible area of all the possible solution pairs. That

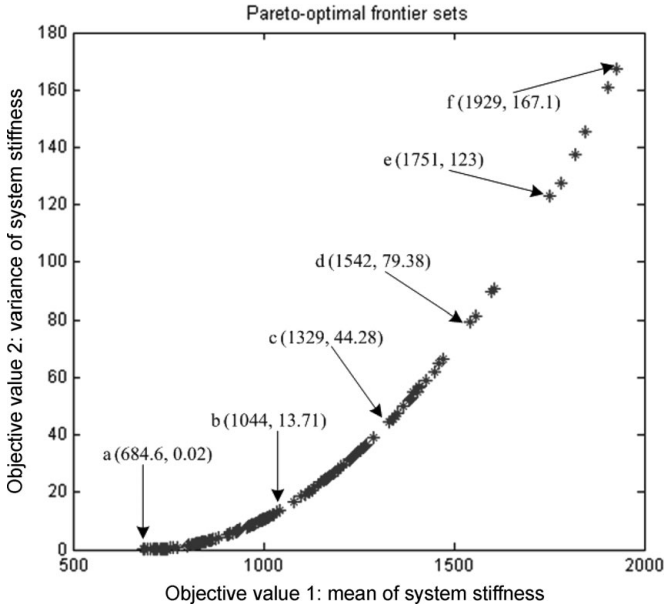


Fig. 13. Pareto-optimal frontier sets.

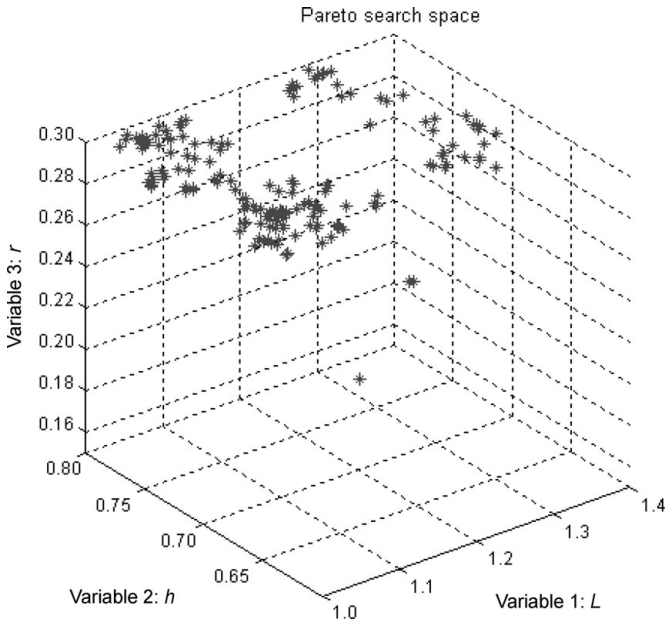


Fig. 14. Pareto search space: best individuals at end of optimization.

is why Pareto solutions are called Pareto-optimal frontier sets. Furthermore, there are six classic optimized design points named from "a" to "f" whose corresponding objective values and design parameters are shown in Table I. Figure 14 illustrates the solution distribution in the 3-dimensional Pareto search space.

It shows that a set of satisfied optimal solutions which provide enough information about alternative solutions for the decision maker with great diversity can be obtained with Pareto based genetic algorithms. Therefore, the simulation shows the efficiency of the proposed single objective and multi-objective optimization design methodology of the 3-DOF parallel manipulator.

https://doi.org/10.1017/S0263574709005657 Published online by Cambridge University Press

Table I. Objective functions values from Pareto sets and corresponding design variables.

	μ-stiffness/σ-stiffness	$L(m)$	$h(m)$	$r(m)$
a	684.6/0.02	1	0.5	0.28
b	1044/13.71	1.2369	0.78385	0.29945
c	1329/44.28	1.1236	0.78216	0.29778
d	1542/79.38	1.0289	0.75123	0.2995
e	1751/123	1.0361	0.78402	0.29789
f	1929/167.1	1.0219	0.79323	0.29191

7. Conclusions

This paper focused on the design, analysis, and optimization of a spatial 3-DOF parallel manipulator. The mechanism has a tripod structure using parallelogram mechanisms for desired motion. The proposed manipulator can be utilized in many industrial applications including high-speed surface finishing operations. The kinematic analysis of this mechanism has been conducted. Artificial neural networks and genetic algorithms as the optimization approaches are implemented simultaneously for the system stiffness synthesis. Both single objective optimization and multi-objective optimization are demonstrated to validate the results of the artificial intelligence approaches.

Acknowledgments

The authors would like to gratefully acknowledge the financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC) through Discovery Grant. The authors are also thankful to Mr. Jason Parise and Mr. Qi Shi for the CAD models.

References

1. Dan Zhang and Clément M. Gosselin, "Parallel kinematic machine design with kinetostatic model," *Robotica* **4**(7), 429–438 (2002).
2. J. A. Carretero, R. P. Podhorodeski, M. N. Nahon and C. M. Gosselin, "Kinematic analysis and optimization of a new three degree-of-freedom spatial parallel manipulator," *ASME J. Mech. Des.* **122**, 17–24 (2000).
3. G. R. Dunlop and T. P Jones, "Position analysis of a two DOF parallel mechanism-Canterbury tracker," *Mech. Mach. Theory* **34**, 599–614 (1999).
4. Dan Zhang and L. Wang, "PKM capabilities and applications exploration in a collaborative virtual environment," *Robot. Comput.-Integr. Manuf.* **22**, 384–395 (2006).
5. J. A. Soons, "On the Geometric and Thermal Errors of a Hexapod Machine Tools, Parallel Kinematic Machines: Theoretical Aspects and Industrial Requirements," **In**: Advanced Manufacturing Series (C. R. Boer, L. Molinari-Tosatti and K. S. Smith, eds.), (Springer-Verlag, 1999) pp. 151–170.
6. Dan Zhang and Clément M. Gosselin, "Kinetostatic modeling of parallel mechanisms with a passive constraining leg and revolute actuators," *Mech. Mach. Theory* **37**, 599–617 (2002).
7. Z. M. Bi, S. Y. T. Lang, Dan Zhang, P. E. Orban and M. Verner, "An integrated design toolbox for tripod-based parallel kinematic machines," *J. Mech. Des.*: *Trans. ASME*, **129**, 799–807 (2007).
8. Dan Zhang and L. Wang, "Conceptual development of an enhanced tripod mechanism for machine tool," *Robot. Comput.-Integr. Manuf.* **21**, 318–327 (2002).
9. Stefan Staicu, Dan Zhang and Radu Dan Rugescu, "Dynamic modelling of a 3-DOF parallel manipulator using iterative matrix relations," *Robotica* **24**(01), 125–130 (2006).
10. Xin-Jun Liu, Jinsong Wang and Gunter Pritschow, "A new family of spatial 3-DoF fully-parallel manipulators with high rotational capability," *Mech. Mach. Theory* **40**, 475–494 (2005).
11. Dan Zhang, L. Wang and E. Esmailzadeh, "PKM capabilities and applications exploration in a collaborative virtual environment", *Robot. Comput.-Integr. Manuf.* **22**, 384–395 (2006).
12. Paulo Roberto Bergamaschi, Antonio Carlos Nogueira and Sezimaria de Fatima Pereira Saramago, "Design and optimization of 3R manipulators using the workspace features," *Appl. Math. Comput.*, **172**, 439–463 (2006).
13. S. Kucuk and Z. Bingul, "Robot Workspace Optimization Based on a Novel Local and Global Performance Indices," *IEEE ISIE 2005*, Dubrovnik, Croatia (June 20–23, 2005) pp. 1593–1598.
14. Alexander Yu, Ilian A. Bonev and Paul Zsombor-Murray, "Geometric approach to the accuracy analysis of a class of 3-DOF planar parallel robots," *Mech. Mach. Theory* (2007) Available online.
15. Jing-Shan Zhao, Sheng-Lan Zhang, Jing-Xin Dong, Zhi-Jing Feng and Kai Zhou, "Optimizing the kinematic chains for a spatial parallel manipulator via searching the desired dexterous workspace," *Robot. Comput.-Integr. Manuf.* **23**, 38–46 (2007).
16. Michael Stock and Karol Miller, "Optimal kinematic design of spatial parallel manipulators: Application to linear delta robot," *J. Mech. Des.*: *Trans. ASME.* **125**, 292–301 (June 2003).
17. B. K. Rout and R. K. Mittal, "Parametric design optimization of 2-DOF R–R planar manipulator: A design of experiment approach," *Robot. Comput.-Integr. Manuf.* **24**, 239–248 (2008).
18. Marco Ceccarelli and Chiara Lanni, "A multi-objective optimum design of general 3R manipulators for prescribed workspace limits," *Mech. Mach. Theory* **39**, 119–132 (2004).
19. J. H. Holland, *Adaptation in Natural and Artificial Systems* (The University of Michigan Press, Ann Arbor, MI, 1975).
20. Dan Zhang, Kinetostatic Analysis and Optimization of Parallel and Hybrid Architectures for Machine Tools *Ph.D. Thesis* (Quebec, QC, Canada: Laval University, 2000).
21. C. M. Gosselin and M. Guillot, "The synthesis of manipulators with prescribed workspace," *J. Mech. Des.*: *Trans. ASME* **113**, 451–455 (1991).
22. N. Garcıa-Pedrajas, C. Hervas-Martınez and J. Munoz-Perez, "Multi-objective cooperative coevolution of artificial neural networks," *Neural Netw.* **15**, 1259–1278 (2002).
23. Shantanu Gupta, Rajiv Tiwari and Shivashankar B. Nair, "Multi-objective design optimization of rolling bearings using genetic algorithms,". *Mech. Mach. Theory* **42**, 1418–1443 (2007).

https://doi.org/10.1017/S0263574709005657 Published online by Cambridge University Press