

---

---

# Novel Manipulability for Parallel Mechanisms with Prismatic-Revolute Actuators, GA-DP Trajectory Planning Application

M. Badrikouhi<sup>a,\*</sup> and M. Bamdad<sup>a,\*\*</sup>

<sup>a</sup>*Shahrood University of Technology, Shahrood, 3619995161 Iran*

<sup>\*</sup>*e-mail: miladbadykoochi@shahroodut.ac.ir*

<sup>\*\*</sup>*e-mail: bamdad@shahroodut.ac.ir*

Received April 22, 2020; revised February 6, 2021; accepted February 7, 2021

**Abstract**—Performance measures have been utilized for numerous applications in robotics research. In this paper, the manipulability and dexterity indices have been focused. These indices have been used successfully when parallel robots have either completely revolute or prismatic joints. But, in a manipulator with both revolute and prismatic joints, these may not be used because of the dimensional inconsistency of the components in the Jacobian matrix. Moreover, in the kinematic or dynamic case, for manipulability and dexterity analysis, the Euclidean norm for the velocity inputs is considered while the velocity inputs for active joints are generally independent; therefore, the manipulability and dexterity indices are explained based on the infinity-norm. Thus, the final solution is represented by the new concept of mixed manipulability polytope. Next, an example of the 3-RPC parallel robot is analyzed based on the notion of this index. As an application, the index is applied for trajectory planning in multi-objective optimal control. For this, one of the direct methods of optimal control, dynamic programming (DP), is selected. But for the DP, the genetic algorithm (GA) is used for searching the optimum path. This analysis is applied for a novel PU-3RPR redundant parallel manipulator. First, the parallel mechanism is introduced, and the kinematics and Jacobian analysis is discussed. Then the objective function that minimizes the pseudo kinetic energy and maximizes the index, is adopted for an optimization problem. Finally, the trajectory of the PU-3RPR redundant parallel mechanism is planned by the proposed GA-DP method.

**Keywords:** parallel mechanism, manipulability, trajectory planning, dynamic programming, optimal control, genetic algorithm

**DOI:** 10.3103/S0025654421020023

## 1. INTRODUCTION

Performance indices for measuring robot manipulator's design performance have been commonly studied in [1, 2]. Indeed, the most general and useful of these indices widely used, are the inverse condition number, dexterity, and manipulability. These indices are frequently used since they are easy to calculate and prepare knowledgeable vision into various designs. As mentioned before, for the parallel manipulators, the application of these indices are increasing recently. For example, Kucuk and Bingul surveyed these indices to optimize the workspace of parallel mechanisms. It is pointed out that the manipulability index cannot be an appropriate index for ill-conditioning Jacobian matrices [3]. The dexterity of seven 3-DOF planar parallel mechanisms with two kinematic chains is evaluated by Kucuk [4] with two aims: decreasing the possibility of interference and growing the robot's workspace. Zhang investigates the dexterity performance index of a PS-RRS-2RUS parallel structure. Finally, the mobility analysis and the workspace of the mechanism with fixed orientation are computed, and yet, its dexterity is estimated [5]. Though, in some instances, these performance measures cannot take the whole complication of a manipulator's kinematic and dynamic characteristics. Among, most remarkable example are the robots which have mixed rotational and translational DOF. In these cases, the Jacobian matrix has inhomogeneous elements [6, 7]. In the case of existing prismatic and revolute joints, Jacobian meets the problem of inhomogeneity. Gosselin resolves the inhomogeneity problem of Jacobian by another formula of the end-effector velocity. This method was developed to evaluate the force transmission capability of a parallel kinematic manipulator [8].

Additionally, other research methods suggested separating the translational and the rotational element of the Jacobian to define isolated performance measures [9]. Also, Mirshekari et al. [2] have resolved the problem of inhomogeneity by defining the characteristic length, but this idea lacks a direct geometric interpretation. Compare to the methods mentioned above, Mansouri resolved the puzzle of inhomogeneity by the power manipulability idea [10]. Hence, the apparent power manipulability was introduced to produce a manipulability index, which is equivalent to translation and rotation. Bowling and Khatib suggested dynamic capability equations to model the relationship between actuator torque capacities and the acceleration and force capabilities, because they treat linear and angular quantities in a consistent and physically meaningful way [7, 11]. Also, Kim and Ryu [12] present a new formulation of a dimensionally homogeneous Jacobian matrix for parallel manipulators with a planar mobile platform by using three end-effector points that are coplanar with the mobile platform joints. Altuzarra et al. [13] (numerically) and Pond et al. [14] (algebraically), on the other hand, obtained a square Jacobian with homogeneous units. They used a velocity equation that relates all kinematic variables of the mechanism, including active and passive joint velocities.

Except for power manipulability, the work by Kim and Ryu, and the method by Altuzarra and Pond, all of the solutions cannot solve the problem of inhomogeneity in the case of existing both types of actuators. Despite all the solutions, the researchers discussed the effectiveness and accuracy of conventional manipulability and dexterity when they have applied these indices to practical design problems. A comprehensive overview is represented in [15]. This research has presented that even when all the actuators are the same, the inverse condition number and manipulability cannot comprehensively and correctly figure the main aspects of manipulator performance. Standard indices are inadequate for specifying general design characteristics based on worst-case performance. In general, in the mentioned researches, for investigating the manipulability and condition number, the 2-norm is used while the infinity-norm addresses the limitations of the 2-norm [16]. Other kinds of performance indices have been established recently and pointed out in reference [17]. Screw theory is applied in this research to generate kinematics and force indices.

This paper aims to determine some restrictions on the traditional indices and suggest the indices that eliminate these restrictions. The authors of this paper provide a method for evaluating indices when translational and rotational actuators exist in a manipulator. Also, when translational task space and rotational task space exist, this method can be practiced by defining angular and axial manipulability. Moreover, utilizing infinity-norm, which is inspired by olds [16], this technique considered the whole robot joint space.

After describing the new index, this can be applied for problems such as structural design, robot analysis, and trajectory planning. One of the strategies in path planning is improving the available path based on the potential function. Using redundancy, the path is bettered according to the designer's aim [18, 19]. This function is the new manipulability index in this work. The trajectory planning problem can be stated as an OCP, which is commonly solved by direct methods. The direct method transforms the trajectory planning problem into a parameter optimization problem accomplished by discretizing dynamic variables containing states and controls [20]. The dynamic programming approach can be used for trajectory planning based on the minimum energy objective function. Solutions include iterative dynamic programming and energy optimization of trajectories for high-level scheduling in Refs. [21, 22] for four six-link manipulators. The energy functions in [21] are based on a similar time scaling of the manipulator trajectory. In [22], it is presented a dynamic time scale to provide an optimal energy trajectory for the operations in a manufacturing sequence. DP is a time-consuming algorithm. As a solution to the DP problem, instead of searching throughout the whole area, the influential GA can be applied. In this paper, the parameter optimization problem which is resulted from DP is solved by GA.

Therefore the article is structured as follows: the manipulability is concisely explained in Section 2, which will start generating the idea of the new manipulability and dexterity definition. In part 3, the proposed plan is discussed. Then the reformulation based on infinity-norm and final solution based on mixed manipulability polytope are represented. Also, the idea is extended to manipulators with rotational and translational DOF. In Section 4, a 3-PRC parallel manipulator ((R) revolute, (P) prismatic, and (C) cylindrical) is introduced then studied as a case study. In part 5, the validity of the new index is examined. In Section 6, trajectory planning for the redundant robot is discussed. First, improving the trajectory in a multi-priority task control by redundancy is explained. Then the dynamic programming method is wholly explored. In part 7, the GA-DP technique is studied. In part 8, the new PU-3RPR redundant parallel platform is introduced, and the forward kinematic and Jacobian of the platform are surveyed. At last, the results for trajectory planning for this redundant robot by the GA-DP method are expressed. Finally, the paper's conclusion and outlook are explained.

## 2. MANIPULABILITY

The manipulability has been presented to investigate the kinematic transmissibility of a manipulator. For manipulability calculation, it is supposed 2-norm for the joint space velocity; therefore, the joint velocities are dependent on each other. Manipulability results from the ellipsoid definition, as follows:

$$m = \sqrt{\det \mathbf{J}\mathbf{J}^T} = \prod_{i=1}^n \sigma_i, \quad (2.1)$$

where  $\sigma_i$  is the  $i$ th singular value of the Jacobian matrix. The robot can take a considerable manipulability in a specified configuration. Nevertheless, it may not generate the same velocity in all directions. Thus the kinematic isotropy or the kinetostatic conditioning index (KCI) is defined while  $\kappa^{-1}$  is computed as the inverse of the Jacobian condition number, named dexterity.

$$\kappa = \|\mathbf{J}\|_2 \|\mathbf{J}^{-1}\|_2, \quad (2.2)$$

$$\kappa^{-1} = \frac{\sigma_{\min}}{\sigma_{\max}}. \quad (2.3)$$

## 3. THE PROPOSED IDEA

### 3.1. The Problem

In the manipulators with both rotational and prismatic joint actuation, the conventional manipulability or dexterity indices, due to the dimensional inconsistency of the Jacobian matrix, may not be used [6, 23]. Moreover, for the indices' analysis, the 2-norm for the velocity inputs are considered while the velocity inputs for active joints are generally independent. Some possibilities are not covered in the analysis [15, 16]; this is explained in the next part. Thus, the appropriate norm for the vector space of velocity inputs for the active joints is the infinity-norm. Also, for the indices developed in screw theory, although this approach has been used in recent years, it is more complicated. It has a specific application, and the natural simplicity of the indices is missed [16]. Consequently, the new index as one solution for these problems is developed further.

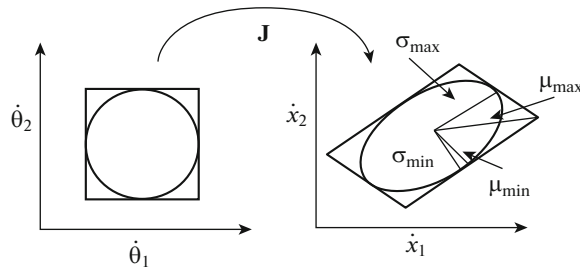
### 3.2. The Solution

As mentioned before,  $\kappa^{-1}$  has been used to identify the KCI or dexterity of the manipulator in a particular configuration. However, according to [15], in the condition number analysis, it is used the 2-norm to quantify the magnitude of the end-effector velocity and the joint velocities. But the bounds of the active joint velocity are independent of each other; consequently, the infinity-norm is selected for the vector of active joint velocities. When measuring the KCI of an orthogonal Cartesian 2-DOF robot, this issue is manifested. As presented in [15], Jacobian matrix is the identity matrix; and this manipulator is fully isotropic. But, if the infinity-norm is employed when selecting a unit velocity for the joints, the minimum possible end-effector velocity is 1 (one of the joints has unit velocity, and the other has zero velocity), though the maximum feasible end-effector velocity is  $\sqrt{2}$  (both joints have unit speed and the end-effector moving at a  $45^\circ$  angle relative to the joints). In this evident and insignificant instance,  $\kappa^{-1}$  does not agree with any of the preceding major physical design specifications; therefore, this index has a deficiency in practical specification-based design [16]. In general, the KCI,  $\kappa^{-1}$ , and manipulability have failed for designing manipulators based on worst-case conditions. It can be resolved to use the appropriate norm [15]. Hence, through the infinity-norm for active joint velocities, the ellipsoid converts to polytope (see Fig. 1) and a new index  $\mu$  for dexterity can be extracted. First,  $\mu_{\min}$  and  $\mu_{\max}$  are defined as follows:

$$\mu_{\min} = \min_{\|\dot{\theta}\|_{\infty}=1} \|\dot{x}\|_2 = \min_{\|\dot{\theta}\|_{\infty}=1} \|\mathbf{J}\dot{\theta}\|_2, \quad (3.1)$$

$$\mu_{\max} = \max_{\|\dot{\theta}\|_{\infty}=1} \|\dot{x}\|_2 = \max_{\|\dot{\theta}\|_{\infty}=1} \|\mathbf{J}\dot{\theta}\|_2, \quad (3.2)$$

$\dot{\theta}$  is the velocity vector of the joint space,  $\mu_{\max}$  and  $\mu_{\min}$  are the maximum and minimum distance from the origin in the end-effector velocity space (see Fig. 1). They are the maximum and minimum capability of



**Fig. 1.** 2-norm limits (Left) and infinity-norm limits (Right) on the joint velocity space. Resulted manipulability ellipsoid and polytope are presented for the Jacobian [16].

the manipulator for moving in an arbitrary direction. The distances depend on the Jacobian. The definition of a new dexterity is as follows:

$$\iota = \frac{\mu_{\min}}{\mu_{\max}} = \frac{\min_{\|\dot{\theta}\|_{\infty}=1} \|\mathbf{J}\dot{\theta}\|_2}{\max_{\|\dot{\theta}\|_{\infty}=1} \|\mathbf{J}\dot{\theta}\|_2}, \quad (3.3)$$

where  $\iota$  is the new dexterity index, therefore, when a manipulator is in the singular configuration,  $\iota$  becomes zero. For a completely isotropic manipulator  $\mu_{\min} = \mu_{\max}$  makes  $\iota = 1$  (see Fig. 1).

Also, the Jacobian matrix's inhomogeneity appears in coexisting rotational and prismatic joint actuation. To tackle this problem, the authors offer a new approach. If the manipulator has two linear DOF ( $\dot{x}_1, \dot{x}_2$ ) and four actuators, two rotational [ $\dot{\theta}_1, \dot{\theta}_2$ ] (rad/s) and two linear [ $\dot{q}_1, \dot{q}_2$ ] (m/s), the relation between joint velocities and end-effector velocity is:

$$\dot{\mathbf{x}} = [\mathbf{J}_q \mathbf{J}_\theta] [\dot{\mathbf{q}} \dot{\boldsymbol{\theta}}]^T, \quad (3.4)$$

$$\dot{\mathbf{q}}_{\min} \leq \dot{\mathbf{q}} \leq \dot{\mathbf{q}}_{\max}, \quad \dot{\boldsymbol{\theta}}_{\min} \leq \dot{\boldsymbol{\theta}} \leq \dot{\boldsymbol{\theta}}_{\max}.$$

A solution that is represented here is evaluating  $\mathbf{J}_q$  and  $\mathbf{J}_\theta$  separately, then analyzing them altogether. Thus in this state, there is no inhomogeneity in each of the two Jacobian matrices. For the system describing in Eq. (3.4), there are two spaces. The rotational and linear velocity joint space is transmitted to the task space by  $\mathbf{J}_\theta$  and  $\mathbf{J}_q$ , respectively. Similar to the analysis of conventional manipulability, rotational and linear joint velocities,  $\dot{\boldsymbol{\theta}}$  (rad/s) and  $\dot{\mathbf{q}}$  (m/s), are bounded in  $[-1,1]$ . Therefore, the joint space is two squares that are converted into two polytopes by  $\mathbf{J}_\theta$  and  $\mathbf{J}_q$  [16]. Each polytope represents the admissible velocity with the same unit in task space. The new idea suggests that manipulability is composed of two polytopes. Depends on rotational or linear speed, one polytope can move on the perimeter of another polytope; Thus, the admissible distribution of the robot's speed is obtained in Fig. 2. Consequently, the area of mixed ellipsoids directly relates to the robot manipulability, such as conventional manipulability index analysis. Also, dexterity is the maximum distance from the perimeter of the mixed figure of ellipsoids to the minimum one (see Fig. 2). By this definition, the problem of inhomogeneity is merely solvable

$$m = \text{Area of the mixed manipulability polytope}$$

$$\zeta = \frac{|\dot{\mathbf{r}}|_{\min}}{|\dot{\mathbf{r}}|_{\max}} \quad (3.5)$$

in which  $m$  is manipulability and  $\zeta$  is dexterity.

### 3.3. Special Case, Rotational And Linear DOF Problem

When the robot has mixed axial and angular DOF, then the Jacobian is as follows:

$$\begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{vq} & \mathbf{J}_{v\theta} \\ \mathbf{J}_{\omega q} & \mathbf{J}_{\omega\theta} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\boldsymbol{\theta}} \end{bmatrix}, \quad (3.6)$$

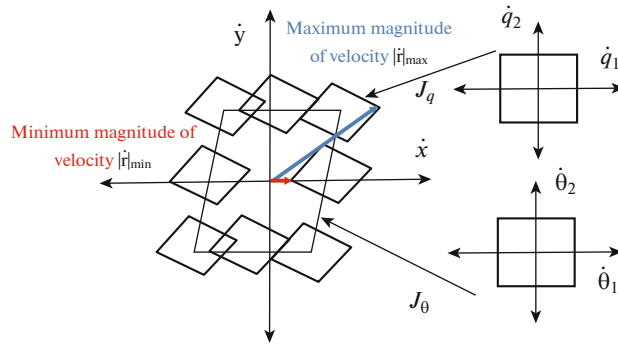


Fig. 2. Dexterity and manipulability. The manipulability is the whole area of the resulted figure, and the dexterity is the minimum magnitude of the velocity to the maximum one.

where  $\omega$  and  $v$  are the rotational and linear velocity. Also,  $\dot{\theta}$  (rad/s) and  $\dot{q}$  (m/s) are the rotational and linear DOF. The Jacobian  $J$  is inhomogeneous. The idea is to separate angular and axial terms and the new definition of angular and axial manipulability. Accordingly, the first column space of Eq. (3.6) presents the axial manipulability  $v = [J_{vq} J_{v\theta}][\dot{q} \dot{\theta}]^T$  and the second one shows angular manipulability  $\omega = [J_{\omega q} J_{\omega \theta}][\dot{q} \dot{\theta}]^T$ . Thus, in this case, the analysis is the same as subsection 3.2 for manipulability. According to manipulator design, the robot parameters can be set to gain maximum angular or axial manipulability. In section 5, the authors will apply this new method for the parallel 3-RPC-manipulator with three similar linear DOF and both prismatic and revolute actuators.

#### 4. PARALLEL 3-RPC-MANIPULATOR, A CASE FOR MANIPULABILITY ANALYSIS

For surveying the method which is introduced in subsection 3.3, a 3-RPC parallel mechanism is introduced. This robot includes an end-effector that is coupled to a base platform by three similar kinematic chains. Each chain consists of a revolute (R), a prismatic (P), and a cylindrical (C) joint providing a translational movement. This parallel kinematic manipulator (PKM) provides mixed translational and/or rotational actuation. This robot has three translational DOF. To certify a completely translational movement, the joint alignment must be selected regularly. Kong and Gosselin [24] represent the rules for the RPC-robot. In each limb, the axes of the revolute joints must be parallel. But, for the structure of the PKM, which has three similar chains, revolute joint axes should not be parallel. Moreover, the prismatic joints' orientation and the revolute joints axes should not be perpendicular [25].

##### 4.1. Forward Kinematics

According to the manipulator kinematics, the closed-loop relation can be written as follows:

$$p = a_i + d_i - {}^P b_i - s_i = a_i + d_i d_{i0} - {}^P b_i - s_i s_{i0}, \tag{4.1}$$

where  $a_i$  is the vector pointing from  $R_0$  to the first revolute joint of limb  $i$ , and  ${}^P b_i$  is the vector from point  $P$  to the origin of the cylindrical joint (see Fig. 3). Both vectors are constant throughout the robot movement where  $d_i$  is the vector from the revolute joint of limb  $i$  to the present position of the cylindrical joint.  $d_{i0}$  is the unit vector of limb  $i$ , where  $d_i$  is referred to its length. The stroke length of the cylindrical joint of limb  $i$  is given by the vector  $s_i$ . The vectors  $a_i$  and  ${}^P b_i$  are known due to geometrical consideration. Also, the unit vector of a cylindrical joint axis and the displacement of the cylindrical joint of limb  $i$  are specified by  $s_{i0}$  and  $s_i$  [26].

With this assumption, we have  $\dot{p} = J\dot{q}$  which  $J$  is the Jacobian matrix. In this section, three manipulators are examined. For prismatic actuators, through the velocity loop closure, we have:

$$\dot{p} = (d_{10}^T \ d_{20}^T \ d_{30}^T)^{-T} (\dot{d}_1 \ \dot{d}_2 \ \dot{d}_3)^T \tag{4.2}$$

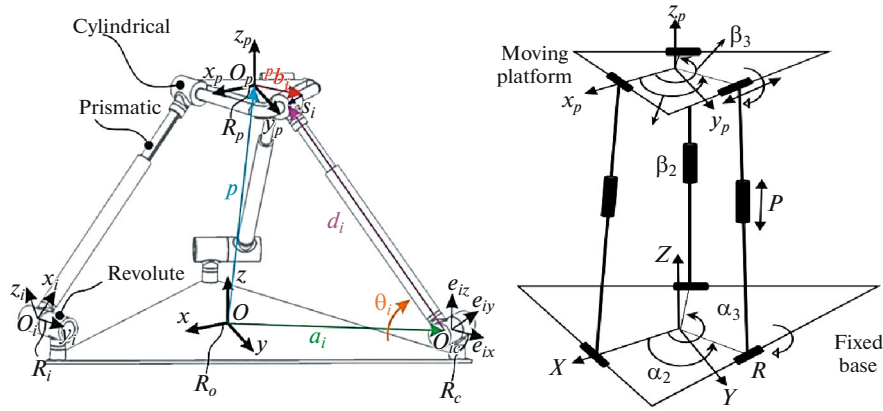


Fig. 3. 3-RPC manipulator [27].

and for rotary actuators:

$$\dot{\mathbf{p}} = \begin{pmatrix} (\mathbf{e}_{1y} \times \mathbf{d}_{10})^T & (\mathbf{e}_{2y} \times \mathbf{d}_{20})^T & (\mathbf{e}_{3y} \times \mathbf{d}_{30})^T \\ d_1 & d_2 & d_3 \end{pmatrix}^{-T} (\dot{\theta}_1 \ \dot{\theta}_2 \ \dot{\theta}_3)^T. \quad (4.3)$$

To find the Jacobian  $\mathbf{J}$  for the 3-RPC with both translational and rotary actuators, it is necessary to both above matrices are combined:

$$\dot{\mathbf{p}} = \begin{pmatrix} \mathbf{d}_{10}^T & \mathbf{d}_{20}^T & \mathbf{d}_{30}^T & \frac{(\mathbf{e}_{1y} \times \mathbf{d}_{10})^T}{d_1} & \frac{(\mathbf{e}_{2y} \times \mathbf{d}_{20})^T}{d_2} & \frac{(\mathbf{e}_{3y} \times \mathbf{d}_{30})^T}{d_3} \end{pmatrix}^{-T} (\dot{d}_1 \ \dot{d}_2 \ \dot{d}_3 \ \dot{\theta}_1 \ \dot{\theta}_2 \ \dot{\theta}_3)^T. \quad (4.4)$$

The maximal stroke length is 0.4 m. Therefore, the overall length of the prismatic joint changes between 0.4 m and 0.8 m. The base platform's radius set to  $r_a = 0.5$  m, and the radius of the end-effector is  $r_b = 0.2$  m [26]. Table 1 shows the geometric and kinematic parameters. The cubic form for the workspace is selected. The length and width of this cube are set to 0.15 m. Also, this cube's height is 0.1 m from 0.4 to 0.6m in the  $z$ -direction of the reference coordinate. The maximum velocity of the prismatic and revolute joint is set to 0.25 m/s and 0.5 rad/s, respectively.

#### 4.2. Manipulability Analysis

Considering the joint velocity limit, it can be compared to the manipulability of PKMs with a different configuration of actuators. The manipulability, in this case, is the volume of a resulted cube. For this mechanism by separating the rotational and translational part, it results:

$$\dot{\mathbf{p}} = \begin{pmatrix} \mathbf{d}_{10}^T & \mathbf{d}_{20}^T & \mathbf{d}_{30}^T \end{pmatrix}^{-T} \begin{pmatrix} \dot{d}_1 \\ \dot{d}_2 \\ \dot{d}_3 \end{pmatrix} + \begin{pmatrix} (\mathbf{e}_{1y} \times \mathbf{d}_{10})^T & (\mathbf{e}_{2y} \times \mathbf{d}_{20})^T & (\mathbf{e}_{3y} \times \mathbf{d}_{30})^T \\ d_1 & d_2 & d_3 \end{pmatrix}^{-T} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{pmatrix}. \quad (4.5)$$

$$\dot{\mathbf{p}} = \mathbf{J}_d \dot{\mathbf{d}} + \mathbf{J}_\theta \dot{\boldsymbol{\theta}}$$

Like Section 3.2, Jacobian is separated into two parts,  $\mathbf{J}_d$  and  $\mathbf{J}_\theta$ , representing translational and angular Jacobian. Due to the existence of three DOF in rotation and translation, each joint velocity space is a

Table 1. Configuration and parameters of triangular PKM.

Frame parameters	Platform parameters
$\mathbf{a}_i = a_i(\cos \alpha_i \ \sin \alpha_i \ 0)^T$	$\mathbf{b}_i = b_i(\cos \beta_i \ \sin \beta_i \ 0)^T$
$\alpha = \begin{bmatrix} 0 & \frac{2\pi}{3} & \frac{4\pi}{3} \end{bmatrix}$	$\beta = \begin{bmatrix} 0 & \frac{2\pi}{3} & \frac{4\pi}{3} \end{bmatrix}$

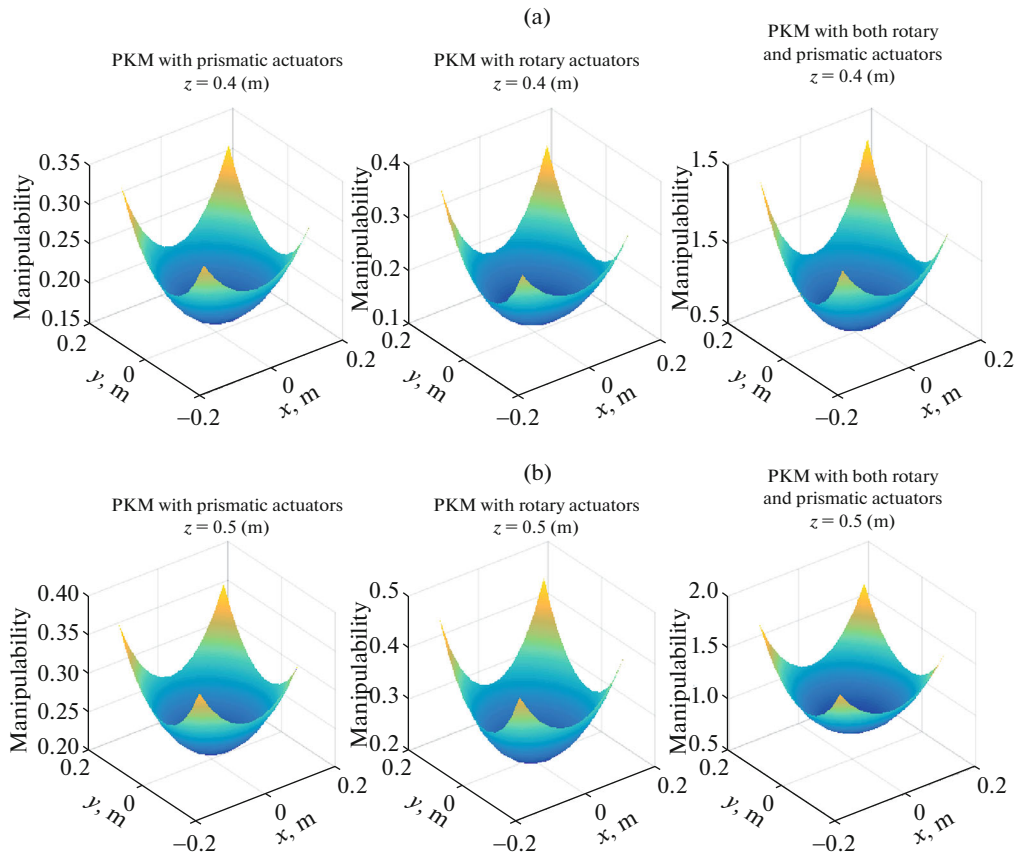


Fig. 4. The manipulability of three types of PKMs. (a)  $z = 0.4m$ ; (b)  $z = 0.5m$ .

cube. Like Fig. 2, these two cubes convert to another two cubes by  $\mathbf{J}_d$  and  $\mathbf{J}_\theta$  in the PKM velocity space, and the final manipulability is the volume of mixed two cubes. One cube moves around the outer surface of another cube. Therefore the manipulability with both types of actuators is calculated and can be compared with two other PKMs in different configurations. As an example, Figs. 4a and 4b show the results for the manipulability of three types of PKMs in  $z = 0.4 m$  and  $z = 0.5 m$ .

For the global index, the new index should be integrated into the workspace and then its average can be calculated.

$$GMI = \frac{\int_W m dW}{\int_W dW} \tag{4.6}$$

The global manipulability index of three PKMs is shown in Table 2. In this case, it can be seen the manipulability is improved significantly close to four times for a redundant PKM, which has three rotational and three translational actuators relative to other PKMs.

Table 2. The manipulability of PKMs based on the new index.

PKMs	Global manipulability index
PKM with prismatic actuators	0.2412
PKM with rotary actuators	0.2375
PKM with both rotary and prismatic actuators	0.9536

**Table 3.** The validity of the new index compared with the conventional index

Z- direction	$z = 0.4$ m	$z = 0.44$ m	$z = 0.48$ m	$z = 0.52$ m	$z = 0.56$ m	$z = 0.6$ m
Conventional manipulability index	0.334	0.367	0.404	0.444	0.489	0.537
New manipulability index	8.018	8.807	9.695	10.676	11.748	12.909
Conventional dexterity index	0.032	0.028	0.025	0.022	0.019	0.017
New dexterity index	0.527	0.506	0.489	0.474	0.462	0.452

## 5. VERIFICATION OF THE NEW INDICES

For the validity of the new index, 3-RPC PKM with translational actuators is considered. This PKM has three translational actuators, and it can be analyzed with conventional indices, but for validity, it is analyzed with the new index, too. In this case, the indices' value is different, but the indices' trend must be close to each other. It is supposed that joint three is separated from two other joints. As mentioned before, the Jacobian is presented in Eq. (5.1).

$$\mathbf{J} = (\mathbf{d}_{10}^T \mathbf{d}_{20}^T \mathbf{d}_{30}^T)^{-T}. \quad (5.1)$$

For examination of the new index,  $d_3$  is separated from two other angles. The result is:

$$\dot{\mathbf{p}} = (\mathbf{d}_{10}^T \mathbf{d}_{20}^T)^{-T} [\dot{d}_1 \ \dot{d}_2]^T + \mathbf{d}_{30}^{-1} \dot{d}_3. \quad (5.2)$$

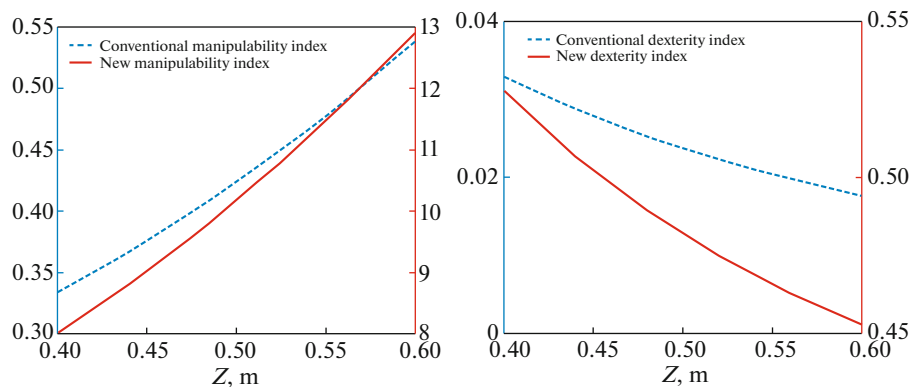
As an example, for the new index, it is considered that  $\dot{d}_1$  and  $\dot{d}_2$  is between  $-1$  to  $1$  m/s and  $\dot{d}_3$  is between  $-0.25$  to  $0.25$  m/s. The new index and the conventional index are surveyed for 3-RPC robot moving from  $z = 0.4$  m to  $z = 0.6$  m. The resulted manipulability and dexterity measures for the two indices are shown in Table 3. When the height is increased from  $z = 0.4$  to  $z = 0.6$ , the analytical outcomes from the conventional index indicate that the manipulability of the manipulator is improved while the dexterity is decreased. This result can be seen for the new manipulability and dexterity index in Fig. 5. Therefore we can find the best configuration of the robot concerning both approaches. Thus, the new indices are reliable for parallel manipulators with mixed prismatic and revolute joints.

## 6. APPLICATION FOR TRAJECTORY PLANNING

### 6.1. Improving The Trajectory In A Multi-Priority Task Control

Multi-priority task control is a well-known frame to manipulate the tasks in kinematically redundant robots. Suppose that there are two tasks. The main task is indicated by the first order  $y_1 \in R^{m_1}$ , and the sub-task is specified by the second manipulation order  $y_2 \in R^{m_2}$ . The basic formulation between the joint velocity vector  $\dot{\mathbf{l}} \in R^n$  and the first task  $\mathbf{y}_1$ , and the second subtask  $\mathbf{y}_2$  is:

$$\mathbf{y}_1 = f(\mathbf{l}), \quad \mathbf{y}_2 = g(\mathbf{l}) \quad (6.1)$$

**Fig. 5.** Trend comparison for the manipulability and dexterity.



in which  $\mathbf{l} = [l_1 \ l_2 \ l_3 \ d]^T$  is the joint variables. If we suppose  $\dot{\mathbf{y}}_d$ , the desired path as the first task, we have  $\dot{\mathbf{y}}_d = \mathbf{J}\dot{\mathbf{l}}$ . In this case, the general solution for  $\dot{\mathbf{l}}$  is obtained from the following equation:

$$\dot{\mathbf{l}} = \mathbf{J}^+ \dot{\mathbf{y}}_d + (\mathbf{I} - \mathbf{J}^+ \mathbf{J})k_d, \quad (6.2)$$

where  $k_d$  is a positive constant and  $\mathbf{J}^+$  is the pseudo-inverse of Jacobian. Consequently, we can define two tasks. The first term on the right-hand side of Eq. (6.2) is the joint velocity to reach the requested trajectory,  $y_d(t)$ . While there are various solutions for  $\dot{\mathbf{l}}$  satisfying Eq. (6.2), the second term gives a solution to make the second task  $g(\mathbf{l})$ , as high as possible. This is possible by setting the parameter  $k_d$  by a designer. Therefore, an application for a new manipulability index, Eq. (3.5), is designing a specific path for maximizing the manipulability.

### 6.2. Dynamic Programming

There are two techniques for solving the optimal control problem (OCP), one is Pontryagin's Minimum Principle (PMP), and the other is Bellman's dynamic programming. The main concept of DP is a discrete, multistage optimization problem. In each finite time intervals, a decision of the design variables is selected based on the cost function from the sum of possible decisions. The chief issue of DP is based on the principle of optimality [28].

### 6.3. Optimal Control Solution Using Dynamic Programming

Let us evaluate the OCP of a discrete-time dynamic. Let the system be defined as  $\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k), k)$ . The cost function is defined as:

$$J_i(\mathbf{x}(k_i)) = J = S(\mathbf{x}(k_f), k_f) + \sum_{k=i}^{k_f-1} V(\mathbf{x}(k), \mathbf{u}(k)), \quad (6.3)$$

where  $\mathbf{x}(k)$ ,  $\mathbf{u}(k)$  are the  $n$  and  $r$  state and control vectors, respectively. We displayed the dependency of  $J$  to the start time ( $k$ ) and state vector ( $\mathbf{x}(k)$ ). We should apply the principle of optimality to obtain the optimal law,  $\mathbf{u}^*(k)$ ,  $k = 1, \dots, k_f - 1$ , that provides the optimal trajectory  $\mathbf{x}^*(k)$ ,  $k = 1, \dots, k_f$ , for the system  $\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k))$ . Let us suppose that we estimated the optimal control for all states, times, and costs from  $k+1$  to  $k_f$ . Then, at each step  $k$ , we utilize the principle of optimality to express as:

$$J_k^*(\mathbf{x}(k)) = \min_{\mathbf{u}(k)} [V[\mathbf{x}(k), \mathbf{u}(k)] + J_{k+1}^*(\mathbf{x}(k+1))]. \quad (6.4)$$

Equation (6.4) is the analytical form of the principle of optimality as exerted to the plant for searching the optimal control law. Also, it is named the functional equation of dynamic programming [28]. Thus we designate the DP method for the continuous-time plant. For a continuous-time system, the procedure is to discretize the system and apply DP to it. For discretization, we can utilize the famous Euler approach [29].

## 7. GA-DP METHOD

We remark that the DP procedure is a computationally intensive technique, particularly with the growth in the system's order and the number of time steps. Therefore Bellman named this increased computational burden as a "curse of dimensionality" [28]. However, with the remarkable improvements in high-speed computational tools, the DP method's application is increased. As a solution, instead of searching throughout the whole search space in every state and time steps, the powerful GA search method can be used. GA method is used in many types of research as a method of solving similar OCP in trajectory planning [30]. The details of the GA method are pointed out in Ref. [30]. In this paper, the GA and DP method is combined for solving the OCP.

As can be seen, if we have one variable  $q$  ( $q \in 0, 1, 2, 3, 4, 5$ ) with time  $t$  ( $t \in 0, 1, 2, 3, 4, 5$ ), the search zone can be divided into many areas (see Fig. 6a). In this meshgrid, as an example,  $q_{12}$  means the value 1 of variable  $q$  in discrete-time 2. If there is a continuous-time system, it can be discretized to get the discrete-time system. There are many trajectories between point 1 ( $q_{00}$ ), and point 2, ( $q_{55}$ ). For example, the green path can be considered as ( $q_{00}, q_{12}, q_{23}, q_{34}, q_{45}, q_{55}$ ). The DP searches all of the possibilities, which are  $6^8$  possible trajectories, based on the OCP's objective function. But in GA-DP, it is considered the primary population of trajectories. By mutation, crossover, and natural selection on these primary populations,

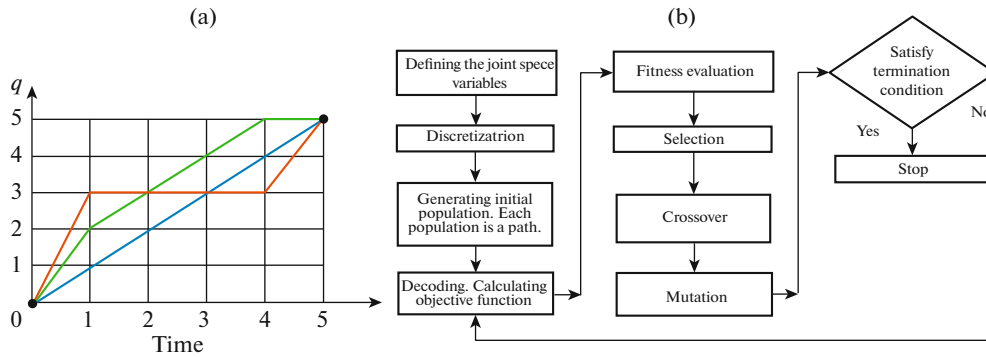


Fig. 6. (a) Meshgrid of the search space; (b) flowchart of the GA-DP for trajectory planning

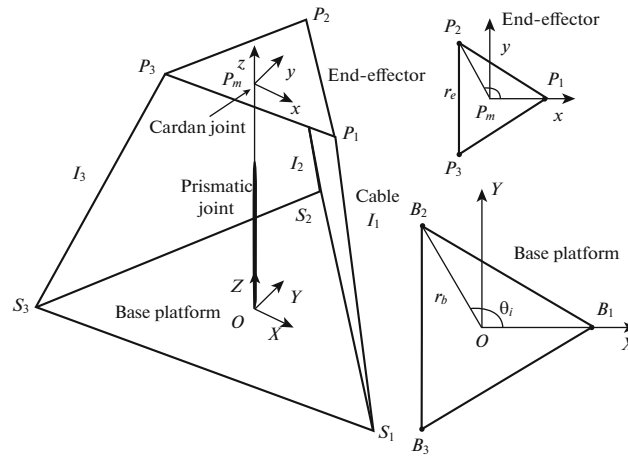


Fig. 7. The structure of the parallel PU-3RPR robot

the next trajectories are built, and the GA is going on until finding the optimum solution (see Fig. 6b). By this algorithm, GA finds the optimum solution at a lower time than conventional search. This method is applicable when the variables and time steps are very much, so the curse of dimensionality is solved.

### 8. PU-3RPR PARALLEL PLATFORM

To implement the strategies, a parallel PU-3RPR cable platform is selected. The PU-3RPR is shown in Fig. 7. This platform has three-degree-of-freedom with one degree of redundancy. It is inspired by the robot of Khakpour et al. [31]. But due to the use of modifications made to it, it is considered a new platform. In this structure, the prismatic actuator is fixed to the base platform and attached to the moving platform using a Cardan joint. It moves the end-effector up and down, and three cables are designed to guide the moving platform. The pneumatic cylinder, by handling the weight of the end-effector, improves the robot’s dynamic agility while no need to use high tensioners for guiding the end-effector through the cables. As mentioned, the cylinder is connected to the mobile platform through a universal connection. Thus it does not have a roll movement about the Z direction. The choice of three cables instead of two cables is also due to redundancy in design.

#### 8.1. Kinematic

For analysis, the XYZ reference frame is connected to the base at point O, and the xyz is connected to the Pm, the center of mass of the end effector. The coordinates of the end effector are expressed in [z, α, β]<sup>T</sup>. z denotes the height of the center of mass of the end effector, α denotes the pitch angle of the end effector around the x-axis, and β denotes the yaw angle of the end effector around the y-axis. In reverse kinematic,

these coordinates are considered. The joint coordinate is  $[l_1 \ l_2 \ l_3 \ d]^T$ .  $l_i$  is the length of the cables, and  $d$  is the position of the prismatic joint,  $P_m$ . Since the jack is fixed to the base,  $z = d$ . To obtain the length of the cables, we form the following equations:

$$\mathbf{OS}_i + \mathbf{S}_i \mathbf{P}_i = \mathbf{OP}_i \rightarrow \mathbf{l}_i = \|\mathbf{S}_i \mathbf{P}_i\| = \|\mathbf{OP}_i - \mathbf{OS}_i\|, \quad (8.1)$$

$\mathbf{OS}_i$  is a constant vector. Also,  $\mathbf{OP}_i = \mathbf{z} + \mathbf{P}_m \mathbf{P}_i$ , in which  $\mathbf{z}$  is known. For  $\mathbf{P}_m \mathbf{P}_i$ , we have  $\mathbf{P}_m \mathbf{P}_i = \mathbf{R}_m^0 \mathbf{P}_i$ .  $\mathbf{R}_m^0 = \mathbf{R}_{PR} = \mathbf{R}_y(\beta) \mathbf{R}_x(\alpha)$  is the pitch-roll rotation matrix of the moving platform relative to the base. Also,  $\mathbf{P}_i$  is the primary position of the vector  $\mathbf{P}_m \mathbf{P}_i$ . Thus the inverse kinematic equations are:

$$d = z, \quad \mathbf{l}_i = \|\mathbf{S}_i \mathbf{P}_i\| = \|\mathbf{z} + \mathbf{R}_y(\beta) \mathbf{R}_x(\alpha) \mathbf{P}_i - \mathbf{OS}_i\|. \quad (8.2)$$

### 8.2. Jacobian

By velocity loop closure, the Jacobian is calculated.  $\mathbf{I}, \mathbf{J}, \mathbf{K}$  are the unit vectors of the fixed frame, and  $\mathbf{i}, \mathbf{j}, \mathbf{k}$  are the unit vectors of the moving frame, which is attached to the moving platform. The joint velocities and the end effector velocity is explained in the following equation:

$$\dot{\mathbf{i}} = [\dot{l}_1 \ \dot{l}_2 \ \dot{l}_3 \ \dot{d}]^T, \quad \dot{\mathbf{x}} = [\dot{z} \ \dot{\alpha} \ \dot{\beta}]^T, \quad \dot{\mathbf{i}} = \mathbf{J} \dot{\mathbf{x}}. \quad (8.3)$$

First, we notice the next equation established in every closure loop of the parallel platform.

$$\mathbf{OP}_m + \mathbf{P}_m \mathbf{P}_i = \mathbf{OS}_i + \mathbf{S}_i \mathbf{P}_i. \quad (8.4)$$

In the next step, we differentiate from each term of Eq. (8.4). First, by considering prismatic joint,  $\left(\frac{d\mathbf{OP}_m}{dt}\right)_{\text{fix}} = \dot{z} \mathbf{K}$ . With pitch – yaw rotation, we have:

$$\left(\frac{d\mathbf{P}_m \mathbf{P}_i}{dt}\right)_{\text{fix}} = \boldsymbol{\Omega} \times \mathbf{P}_m \mathbf{P}_i, \quad \boldsymbol{\Omega} = \dot{\alpha} \mathbf{I} + \dot{\beta} \mathbf{J}. \quad (8.5)$$

Also,  $\mathbf{OS}_i$  is a constant vector. As a result  $\left(\frac{d\mathbf{OS}_i}{dt}\right)_{\text{fix}} = \dot{z} \mathbf{K}$ . At last:

$$\left(\frac{d\mathbf{S}_i \mathbf{P}_i}{dt}\right)_{\text{fix}} = \left(\frac{\partial \mathbf{S}_i \mathbf{P}_i}{\partial t}\right)_{\text{mov}} + \boldsymbol{\Omega} \times \mathbf{S}_i \mathbf{P}_i. \quad (8.6)$$

The unit vector in the direction of the cables is  $\mathbf{s}_i$ . The angular velocity vector of the cables  $\boldsymbol{\Omega}_i$  is perpendicular to  $\mathbf{s}_i$ , and cables do not rotate around themselves. As a result:

$$\frac{d\mathbf{l}_i \mathbf{s}_i}{dt} = \dot{l}_i \mathbf{s}_i + \boldsymbol{\Omega}_i \times \mathbf{l}_i \mathbf{s}_i. \quad (8.7)$$

Therefore the final equation for velocity loop closure is rewritten as:

$$\dot{z} \mathbf{K} + (\dot{\alpha} \mathbf{I} + \dot{\beta} \mathbf{J}) \times \mathbf{P}_m \mathbf{P}_i = \dot{l}_i \mathbf{s}_i + \boldsymbol{\Omega}_i \times \mathbf{l}_i \mathbf{s}_i. \quad (8.8)$$

For calculating  $\mathbf{J}$ , the cross product of the vector  $\mathbf{s}_i$  satisfies the following equation.

$$\dot{z} \mathbf{K} \cdot \mathbf{s}_i + ((\dot{\alpha} \mathbf{I} + \dot{\beta} \mathbf{J}) \times \mathbf{P}_m \mathbf{P}_i) \cdot \mathbf{s}_i = \dot{l}_i \mathbf{s}_i \cdot \mathbf{s}_i + (\boldsymbol{\Omega}_i \times \mathbf{l}_i \mathbf{s}_i) \cdot \mathbf{s}_i. \quad (8.9)$$

It is considered  $\mathbf{P}_m \mathbf{P}_i = \mathbf{E}_i$ . And simplifies to:

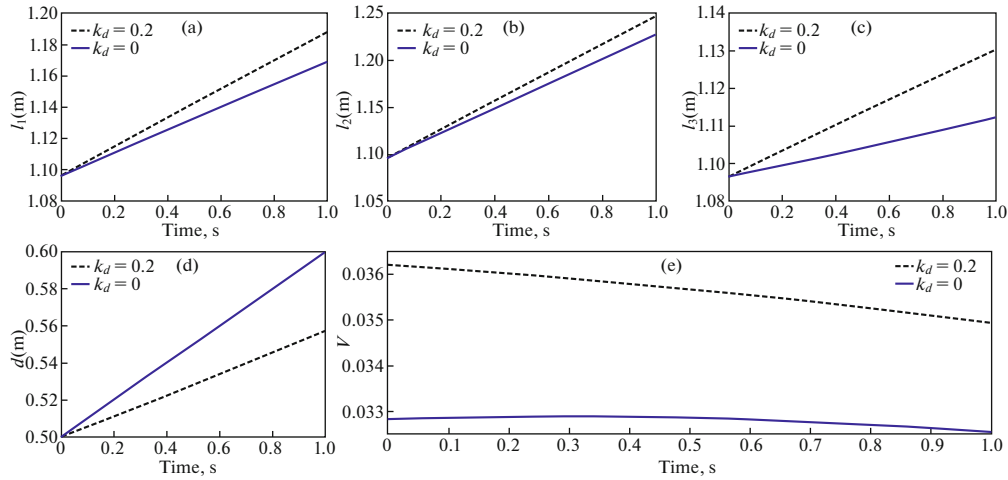
$$\dot{z} \mathbf{K} \cdot \mathbf{s}_i + \dot{\alpha} (\mathbf{I} \times \mathbf{E}_i) \cdot \mathbf{s}_i + \dot{\beta} (\mathbf{J} \times \mathbf{E}_i) \cdot \mathbf{s}_i = \dot{l}_i. \quad (8.10)$$

Rewriting in a matrix form results in:

$$[\mathbf{K} \cdot \mathbf{s}_i | (\mathbf{I} \times \mathbf{E}_i) \cdot \mathbf{s}_i | (\mathbf{J} \times \mathbf{E}_i) \cdot \mathbf{s}_i] \cdot [\dot{z} \ \dot{\alpha} \ \dot{\beta}]^T = \dot{l}_i \quad (8.11)$$

and finally for  $i = 1, 2, 3$ , the manipulator Jacobian matrix can be derived as:

$$\mathbf{J} = \begin{bmatrix} \mathbf{K} \cdot \mathbf{s}_1 & (\mathbf{I} \times \mathbf{E}_1) \cdot \mathbf{s}_1 & (\mathbf{J} \times \mathbf{E}_1) \cdot \mathbf{s}_1 \\ \mathbf{K} \cdot \mathbf{s}_2 & (\mathbf{I} \times \mathbf{E}_2) \cdot \mathbf{s}_2 & (\mathbf{J} \times \mathbf{E}_2) \cdot \mathbf{s}_2 \\ \mathbf{K} \cdot \mathbf{s}_3 & (\mathbf{I} \times \mathbf{E}_3) \cdot \mathbf{s}_3 & (\mathbf{J} \times \mathbf{E}_3) \cdot \mathbf{s}_3 \\ 1 & 0 & 0 \end{bmatrix}. \quad (8.12)$$



**Fig. 8.** (a) Length of the cable 1; (b) length of the cable 2; (c) length of the cable 3; (d) cylinder's position; (e) objective function  $V$ .

### 8.3. Optimal Trajectory by Solving the Multi-Objective OCP with GA-DP Method

The optimization problem considered in this section can be written as:

$$\text{Minimize } V = \int_0^1 \left( \frac{1}{2} (\dot{l}_1^2 + \dot{l}_2^2 + \dot{l}_3^2 + \dot{d}^2) + \frac{n}{m} \right) dt \text{ Subject to:}$$

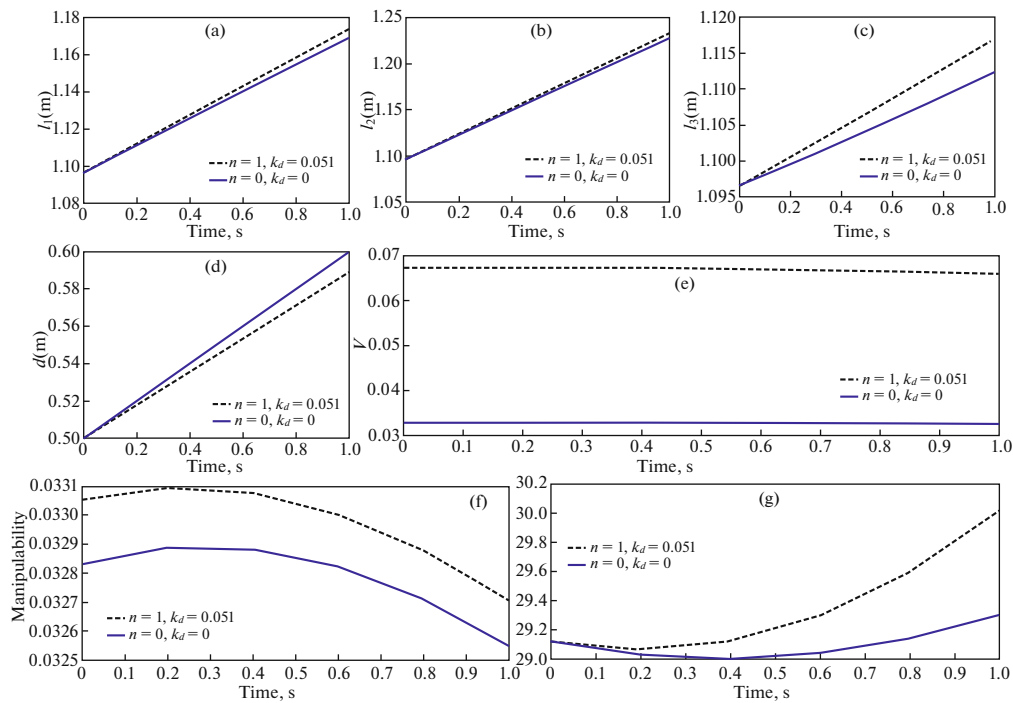
$$\mathbf{y}_d = \begin{cases} z = 0.1t + 0.5 \\ \alpha = \frac{\pi}{18}t \\ \beta = 0 \end{cases} \quad 0 \leq t \leq 1, \quad (8.13)$$

$$\dot{\mathbf{l}} = \mathbf{J}^+ \dot{\mathbf{y}}_d + (\mathbf{I} - \mathbf{J}^+ \mathbf{J}) \mathbf{k}_d \quad 0 \leq k_d \leq 1,$$

which  $\dot{\mathbf{l}}$  is the joint velocity,  $[\dot{l}_1, \dot{l}_2, \dot{l}_3, \dot{d}]^T$  and  $m$ , is the new manipulability index from Eq. (3.5). Also, the positive constant  $n$  is a design coefficient. The manipulator passes  $\mathbf{y}_d$ , but its configuration changes to improve the second subtask,  $V$ . The optimization process which is the path planning by GA-DP finds the parameter  $k_d$  for optimizing the subtask  $V$ . Minimizing the integral of pseudo kinetic energy with maximizing the manipulability is considered as the multi-objective function for the second subtask. The pseudo kinetic energy criterion was suggested by Whitney and applied in the book of Nakamura [32]. It is defined as the quadratic form of the joint velocities of the manipulator. Also, the inverse of manipulability is considered as a second term in the multi-objective function. In this simulation, one goal is decreasing the pseudo kinetic energy, and the second is increasing manipulability; therefore, normalization is needed for unifying the effect of the two goals. The size of  $n$  is arranged for considering the importance of the manipulability relative to pseudo kinetic energy.

Two scenarios for  $n = 0, 1$  are simulated. Minimum pseudo kinetic energy is considered in the first simulation and the second one, solve the minimum pseudo kinetic energy and maximum manipulability with the factor  $n = 1$ . By discretization,  $k_d$  as a design parameter and  $t$ , the DP method can solve this optimization problem. But instead of searching every state and time, GA-DP searches the best redundancy factor,  $k_d$ , to minimize Eq. (8.13). The lower and upper bound of  $k_d$  is 0 and 1. The details of the method are discussed in Section 7. Due to verify solution, the result for  $k_d = 0.2$  is plotted in Fig. 8. It is shown that the objective function  $V$  in Eq. (8.13) is increased toward optimum value  $k_d = 0$ .

In the second simulation, minimum control effort, and maximum manipulability with the factor  $n = 1$  leads to an optimized solution  $k_d = 0.051$ . It is compared with  $k_d = 0$  for  $n = 0$ . As can be seen in Fig. 9, the joint position is changed; the cost function  $V$  and pseudo kinetic energy are increased. Maximizing manipulability in equation (8-13), leads to increase in pseudo kinetic energy.



**Fig. 9.** (a) Length of the cable 1; (b) length of the cable 2; (c) length of the cable 3; (d) cylinder's position; (e) objective function  $v$ ; (f) pseudo kinetic energy; (g) Manipulability

## 9. CONCLUSIONS

This article illustrates the performance index for the parallel actuated manipulators regarding coexisting translational and rotational actuators. As a result of inhomogeneity, general Jacobian based performance indices are not appropriate for these manipulators. Consequently, the new manipulability and dexterity concept is applied to estimate the performance of these PKMs. This approach uses infinity-norm instead of 2-norm; therefore, the old manipulability and dexterity index is upgraded. This concept is used very simply, based on standard analysis, and this is no need for defining new analysis. As a case study, the manipulability of three types of 3-RPC parallel mechanisms with different actuator configurations is analyzed and compared. Also, this index is validated successfully in Fig. 5; in the next section, the application of this new index in the trajectory planning by a GA-DP method for a parallel redundant platform is discussed. First, the role of redundancy in improving robot performance is explained, then the DP method for solving OCP is expressed. The DP has the problem of the curse of dimensionality; thus, for improving the DP method, the GA algorithm is combined with the DP. Then the new PU-3RPR platform is entirely introduced as a case for surveying this method. Finally, optimal trajectory planning by solving the multi-objective OCP with the GA-DP method for a parallel redundant PU-3RPR platform is expressed in Section 8.3. For the objective function of the OCP, the new manipulability is used. The results are shown in Fig. 8 and Fig. 9. By the GA-DP method, it is possible to survey various objective functions, including the new manipulability in the path planning problem.

## REFERENCES

1. M. Bamdad and M. M. Bahri, "Kinematics and manipulability analysis of a highly articulated soft robotic manipulator," *Robotica* **37**, 868–882 (2019).
2. E. Mirshekari, A. Ghanbarzadeh, and K. H. Shirazia, "Structure comparison and optimal design of 6-RUS parallel manipulator based on kinematic and dynamic performances," *Lat. Am. J. Solids Struct.* **13**, 2414–2438 (2016).
3. S. Kucuk and Z. Bingul, "Comparative study of performance indices for fundamental robot manipulators," *Rob. Auton. Syst.* **54**, 567–573 (2006).
4. S. Kucuk, "A dexterity comparison for 3-DOF planar parallel manipulators with two kinematic chains using genetic algorithms," *Mechatronics* **19**, 868–877 (2009).

5. C. Zhang, Y. Chen, H. Xu, et. al., “Kinematic modeling and dexterity evaluation of a PS-RRS-2RUS parallel manipulator used for controllable pitch propeller,” *Mech. Based Des. Struct. Mach.* **46**, 533–551 (2018).
6. S. Shayya, S. Krut, O. Company, et al. “On the performance evaluation and analysis of general robots with mixed dofs,” in *Intelligent Robots and Systems (IROS 2014), Chicago, IL, 2014*, pp. 490–497.
7. A. Bowling and O. Khatib, “Dynamic loading criteria in actuator selection for desired dynamic performance,” *Adv. Rob.* **17**, 641–656 (2003).
8. S.-G. Kim and J. Ryu, “Force transmission analyses with dimensionally homogeneous jacobian matrices for parallel manipulators,” *J. Mech. Sci. Technol.* **18**, 780–788 (2004).
9. G. Boschetti, R. Rosa, and A. Trevisani, “Parallel robot translational performance evaluation through direction-selective index (DSI),” *J. Rob.* **2011** (2011).
10. I. Mansouri and M. Ouali, “The power manipulability—A new homogeneous performance index of robot manipulators,” *Rob. Comput. Integr. Manuf.* **27**, 434–449 (2011).
11. A. P. Bowling, “Dynamic performance, mobility, and agility of multilegged robots,” *J. Dyn. Syst., Meas., Control.* **128**, 765–777 (2006).
12. S. G. Kim and J. Ryu, “New dimensionally homogeneous Jacobian matrix formulation by three end-effector points for optimal design of parallel manipulators,” *IEEE Trans. Rob. Autom.* **19**, 731–736 (2003).
13. O. Altuzarra, O. Salgado, V. Petuya, and A. Hernández, “Point-based Jacobian formulation for computational kinematics of manipulators,” *Mech. Mach. Theory.* **41**, 1407–1423 (2006).
14. G. Pond and J. A. Carretero, “Dexterity measures and their use in quantitative dexterity comparisons,” *Meccanica* **46**, 51–64 (2011).
15. J. P. Merlet, “Jacobian, manipulability, condition number, and accuracy of parallel robots,” *Journal of Mechanical Design.* **128**, 199–206 (2006).
16. K. C. Olds, “Global Indices for kinematic and force transmission performance in parallel robots,” *IEEE Transactions on Robotics.* **31**, 494–500 (2015).
17. Q. Meng, F. Xie, X.-J. Liu, and Y. Takeda, “Screw Theory-Based Motion/Force Transmissibility Analysis of High-Speed Parallel Robots With Articulated Platforms,” *Journal of Mechanisms and Robotics.* **12** (2020).
18. T. Yoshikawa, *Foundations of Robotics: Analysis and Control* (MIT press, 1990).
19. Y. Nakamura and H. Hanafusa, “Optimal redundancy control of robot manipulators,” *Int. J. Rob. Res.* **6**, 32–42 (1987).
20. T. Chettibi, H. Lehtihet, M. Haddad, and S. Hanchi, “Minimum cost trajectory planning for industrial robots,” *Eur. J. Mech. A/Solids.* **23**, 703–715 (2004).
21. O. Wigström and B. Lennartson, “Energy optimization of trajectories for high level scheduling,” in *2011 IEEE International Conference on Automation Science and Engineering*, Trieste, 2011, pp. 654–659.
22. O. Wigstrom, B. Lennartson, A. Vergnano, and C. Breitholtz, “High-level scheduling of energy optimal trajectories,” *IEEE Trans. Autom. Sci. Eng.* **10**, 57–64 (2013).
23. C. M. Gosselin, “The optimum design of robotic manipulators using dexterity indices,” *Rob. Auton. Syst.* **9**, 213–226 (1992).
24. X. Kong and C. M. Gosselin, “Type synthesis of 3-DOF translational parallel manipulators based on screw theory,” *J. Mech. Des.* **126**, 83–92 (2004).
25. X. Kong and C. M. Gosselin, *Type Synthesis of Parallel Mechanisms* Vol. 33 (Springer, 2007).
26. I. Prause, M. Lorenz, and B. Corves, “Kinetostatic analysis of the translational RPC-manipulator with different actuator and frame configurations,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)* (Seattle, 2015), pp. 1605–1612.
27. I. Prause and B. Corves, “Dynamic modeling of the RPC-manipulator with prismatic or revolute joint actuation for different frame configurations,” in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Hamburg, 2015), pp. 4105–4112.
28. D. S. Naidu, *Optimal Control Systems* (CRC Press, 2002).
29. W. Alt, U. Felgenhauer, and M. Seydenschwanz, “Euler discretization for a class of nonlinear optimal control problems with control appearing linearly,” *Comput. Optim. Appl.* **69**, 825–856 (2018).
30. V. Azimirad and H. Shorakaei, “Dual hierarchical genetic-optimal control: A new global optimal path planning method for robots,” *J. Manuf. Syst.* **33**, 139–148 (2014).
31. H. Khakpour, L. Birglen, and S.-A. Tahan, “Analysis and optimization of a new differentially driven cable parallel robot,” *J. Mech. Rob.* **7**, 034503 (2015).
32. Y. Nakamura, *Advanced Robotics: Redundancy and Optimization* (Addison-Wesley, Inc., 1990).