Scientia Iranica B (2013) 20(6), 1742–1752

SCIENTIA IRANICA

Sharif University of Technology

Scientia Iranica

Transactions B: Mechanical Engineering

www.scientiairanica.com

Kinematic control of a new hyper-redundant manipulator with lockable joints

A. Taherifar, H. Salarieh* and A. Alasty

Center of Excellence in Design, Robotics and Automation (CEDRA), Department of Mechanical Engineering, Sharif University of Technology, Tehran, P.O. Box 11155-9567, Iran.

Received 8 June 2012; received in revised form 20 April 2013; accepted 25 June 2013

KEYWORDS

Redundant manipulator; Kinematic control; Lockable joints; Tendon-actuated manipulator.

Abstract. Kinematic control of a special hyper-redundant manipulator with lockable joints is studied. In this manipulator, the extra cables are replaced by a locking system to reduce the weight of the structure and the number of actuators. This manipulator has discrete and continuous variables due to its locking system. Therefore, a hybrid approach has been adopted in control. At first the forward kinematics and velocity kinematics of this manipulator are derived, and then a novel closed-loop control algorithm is presented. This algorithm consists of decision making, an inner loop controller, and kinematic calculation blocks. The decision making block is the logical part of the control scheme in which suitable switches will be chosen. The control block uses the end-effector position feedback to generate appropriate commands. The performance of the proposed hybrid control scheme in position tracking is assessed for several trajectories.

1. Introduction

Kinematically redundant manipulators have more Degrees Of Freedom (DOF) than required for determining the position and orientation of the end-effector [1]. Due to their high flexibility, they have a great potential to work in fully constrained, complex and hazardous environments, such as nuclear reactors and space stations [2]. They can also be used for surgery, exploration, and rescue.

In the past two decades, many redundant manipulators with various structures and different types of actuation have been proposed. Actuation with cables is one common type of manipulator operation, which is called tendon-actuation [3]. Some other manipulators [4] use pneumatic actuators. Li et al. [5] designed a novel hyper redundant manipulator, named RT1 in which all of the DOFs are actuated by only one motor via especially designed hinge bar universal joints; as a result, the manipulator weight is greatly reduced. Ananiev et al. [6] also designed a single-motor driven construction of a hyper-redundant robot using the same method used in [5], but, instead of hinge bar universal joints, they used flexible shafts. Todd and Drozda [7,8] developed a tendon-actuated robot, and put all of the actuators out of the manipulator in order to decrease the weight of the robot.

Hirose and Ma [9] implemented a coupled tendon driven manipulator with a specific tendon traction force mechanism in which a pair of tendons is pulled by base actuators via pulleys mounted on the base-side joint. Kimura and [10] designed a 15-DOF manipulator, which has 7 joint modules and one prismatic joint on the bottom; each joint module has 2 actuators. Chirikjian and Burdick [11] developed a 30-DOF robot consisting of 10 identical 3-DOF truss modules. Shammas et al. [12] designed a new 3-DOF joint for snake-like robots using an angular bevel gear train. Jones et al. [13] designed a novel continuum manipulator.

*. *Corresponding author. Tel.: +98 21 66165538*
E-mail addresses: taherifar@mech.sharif.ir (A. Taherifar); salarieh@sharif.edu (H. Salarieh); aalasti@sharif.edu (A. Alasty)

The design features two concentric flexible cylinders, with a pneumatically actuated inner tube and tendons fixed to an outer cylinder. Ning et al. [14-16] built a new 3-D modular hyper-redundant manipulator. All the joints of this manipulator are passive and state controllable and share a common input introduced by wire-driven control. The trajectory planning of redundant manipulators is also an important area of research. Marcos et al. [17] studied the trajectory planning of such manipulators using the genetic algorithm.

A novel, tendon-actuated, hyper redundant manipulator was introduced by Honorary [18] in 2009. The novelty of this tendon-actuated manipulator is due to the use of a locking mechanism in the joints, which makes it controllable with only three cables. Therefore, by releasing and locking the joints in an arbitrary sequence and by stretching the cables, the configuration of the manipulator changes. This mechanism of actuation has discrete and continuous variables, which makes the control problem sophisticated. In this paper, the modeling and kinematic control of this manipulator are studied.

In order to have a deep understanding of the control scheme presented in the following sections, it is essential to give a brief description of the manipulator. Each module of the manipulator (links and joints) consists of three hydraulic jacks, which are arranged in a parallel configuration (3RPS (3-Revolute-Prismatic-Spherical)). The lengths of these jacks are either fixed (locked) or varying (unlocked) due to the solenoid state. The solenoid valves are the main components of the locking mechanism which are put in the hydraulic circuit, as shown in Figure 1. Once they are chosen to be unlocked, the length of the jacks can be controlled by the wires.

Actuators are three cables which pass through joints all along the arm, as shown in Figure 2. When a link is locked, the position and orientation of that link remains unchanged. At any time, all links of the manipulator are locked except one. Whenever the cables are moved, the manipulator will tilt to its new position while all other locked links move as a rigid body. Lockable joints decrease the number of actuators

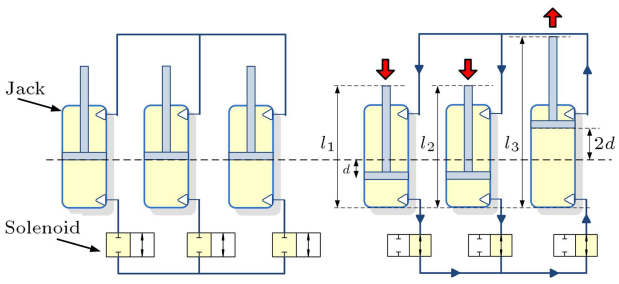

Figure 1. The hydraulic circuit of locking system in each link.

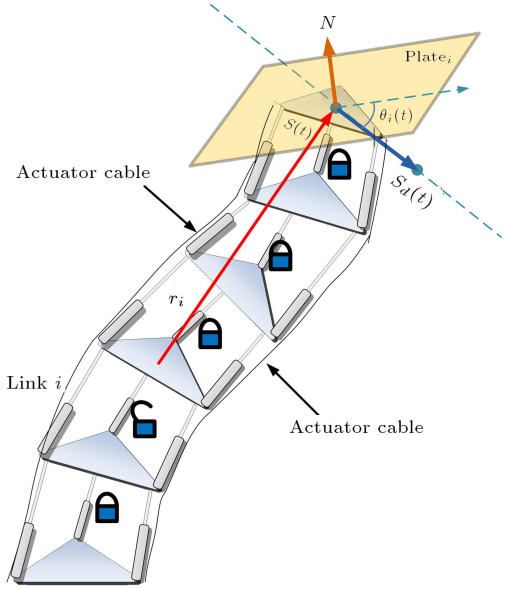

Figure 2. The schematic of the cable-actuated hyper redundant manipulator. The ith link is unlocked and the others are locked.

and reduce the total weight and cost. An eight-DOF prototype has been designed and made at Sharif University for practical experiments. Shafaei [19] optimized and utilized the manipulator in 2011. Although the structure of the manipulator is simplified by this novel idea, the path planning and control problem are enormously complicated. Taherifar et al. [20] has studied the path planning of the planar manipulator using the Particle Swarm Optimization (PSO) technique. In [20], inverse kinematics is solved with continuous PSO and the sequence of switching is optimized with the discrete PSO algorithm. The minimum time and minimum switch path planning of this manipulator are also studied in [21]. The minimum switch path will decrease vibration and energy consumption.

In previous designs of this category, one set of cables was used to control the orientation of each joint. For example, to control a joint with two degrees of freedom, at least 3 cables were needed. So, if the manipulator has 4 joints, there should be at least 12 cables to control the arm. In this design, a lock mechanism in each joint is used to lock all degrees of freedom except 2 degrees, which can be controlled by 3 cables extended along the whole arm. Pulling the cables will create torque along the arm. By choosing which lock to be opened, the torque created by the cables will change the orientation of that joint. In this paper, the modeling and control of the robot have been investigated.

The 3RPS mechanism has 3 degrees of freedom (Figure 3). In addition, the hydraulic circuit shown in this figure adds a constraint to the mechanism that reduces one degree of freedom when all solenoid valves

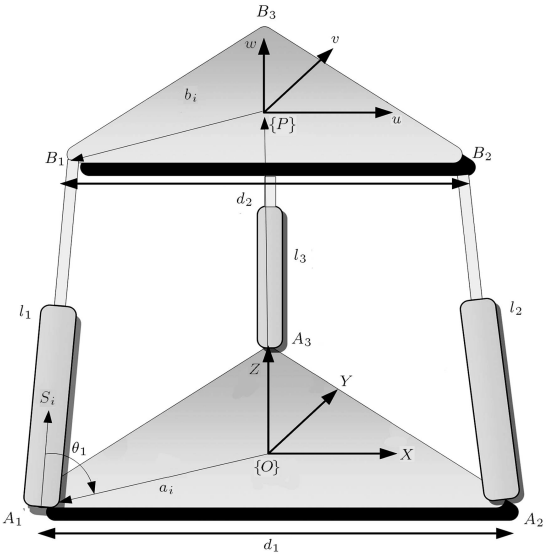

Figure 3. Schematic of the link mechanism for the redundant manipulator with RPS mechanism.

are open, that is:

$$l_1 + l_2 + l_3 = \text{const}, \tag{1}$$

where l_1, l_2 and l_3 are the lengths of the jacks. When solenoid valves are closed, the mechanism is completely locked.

In the next section, the forward kinematics and velocity kinematics of this manipulator are derived. A closed loop control scheme is proposed for the hyper-redundant manipulator with lockable joints in Section 3. The control scheme presented in this paper is based on kinematic equations. In Section 4, the results of the simulation for several trajectories are presented. Finally, the conclusion is given in Section 5.

2. Kinematics

2.1. Forward kinematics

The forward kinematics and Jacobian analysis of a manipulator are necessary for kinematic control of the manipulator. It has also an important role to play in switch decision making. The transformation matrix, ${}^{O}_{P}\mathbf{T}$, relates the coordinate system, $\{O\}$, on the lower platform to the coordinate system, $\{P\}$, on the upper platform of a link, as shown in Figure 3.

In the case where n modules are connected to each other, as shown in Figure 2, one can determine the position of the end-effector by multiplying the transformation matrices according to the following equation:

$${}^{0}\mathbf{P} = {}^{0}_{1}\mathbf{T}\ {}^{1}_{2}\mathbf{T}\cdots{}^{n-1}_{n}\mathbf{T}\ {}^{n}\mathbf{P}, \tag{2}$$

where ${}^{0}\mathbf{P}$ and ${}^{n}\mathbf{P}$ are the position of the end effector in the base frame and the nth link frame, respectively.

According to Figure 3, the position vector of the upper end of the limbs, B_1, B_2 and B_3 can be written as:

$$B_1 = \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix}, \qquad B_2 = \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix}, \qquad B_3 = \begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix}. \tag{3}$$

The distance between the upper ends of the adjacent limbs is constant in any configuration, that is:

$$\begin{cases} (x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2 = d_2^2 \\ (x_1 - x_3)^2 + (y_1 - y_3)^2 + (z_1 - z_3)^2 = d_2^2 \\ (x_2 - x_3)^2 + (y_2 - y_3)^2 + (z_2 - z_3)^2 = d_2^2 \end{cases}, \tag{4}$$

where d_2 is the upper platform length. Now, x_i and y_i in Eq. (4) are substituted in terms of l_1, l_2, l_3 and θ_1, θ_2, θ_3. Finally, the kinematic equations of a link are deriven:

$$\begin{cases} d_1^2 - \sqrt{3}d_1 l_1 \cos(\theta_1) - \sqrt{3}d_1 l_2 \cos(\theta_2) \\ \quad + l_1 l_2 \cos(\theta_1)\cos(\theta_2) + l_1^2 \\ \quad - 2l_1 l_2 \sin(\theta_1)\sin(\theta_2) + l_2^2 = d_2^2 \\ \\ d_1^2 - \sqrt{3}d_1 l_1 \cos(\theta_1) - \sqrt{3}d_1 l_3 \cos(\theta_3) \\ \quad + l_1 l_3 \cos(\theta_1)\cos(\theta_3) + l_1^2 \\ \quad - 2l_1 l_3 \sin(\theta_1)\sin(\theta_3) + l_3^2 = d_2^2 \\ \\ d_1^2 - \sqrt{3}d_1 l_2 \cos(\theta_2) - \sqrt{3}d_1 l_3 \cos(\theta_3) \\ \quad + l_2 l_3 \cos(\theta_2)\cos(\theta_3) + l_2^2 \\ \quad - 2l_2 l_3 \sin(\theta_2)\sin(\theta_3) + l_3^2 = d_2^2 \end{cases} \tag{5}$$

Assuming the jack lengths, l_1, l_2 and l_3 , are given, one can solve Eq. (5) for θ_1, θ_2 and θ_3. Then, the transformation matrix, ${}^{P}_{O}\mathbf{T}$, can simply be deriven as:

$${}^{P}_{O}\mathbf{T} = \begin{bmatrix} {}^{P}_{O}\mathbf{R}_{(3\times3)} & {}^{O}\mathbf{P}_{\text{Borg}(3\times1)} \\ [0]_{(1\times3)} & 1 \end{bmatrix}, \tag{6}$$

$${}^{P}_{O}\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 & \mathbf{R}_2 & \mathbf{R}_3 \end{bmatrix},$$

$${}^{O}\mathbf{P}_{\text{Borg}} = \begin{bmatrix} \dfrac{\mathbf{P}_1 + \mathbf{P}_2 + \mathbf{P}_3}{3} \end{bmatrix}, \tag{7}$$

$$\mathbf{R}_1 = \frac{(B_2 - B_1)}{|B_2 - B_1|}, \qquad \mathbf{R}_2 = \frac{(B_3 - \frac{B_2 + B_1}{2})}{\left|B_3 - \frac{B_2 + B_1}{2}\right|},$$

$$\mathbf{R}_3 = \mathbf{R}_1 \times \mathbf{R}_2. \tag{8}$$

So, according to the above equations, the position and the orientation of the end effector could be calculated in terms of jack lengths. The forward kinematic

equations in this section will be used to solve the velocity kinematics of the arm in the next section.

2.2. Jacobian analysis of manipulator

2.2.1. Link Jacobian

In this section, the velocity kinematics of the manipulator is investigated and the Jacobian matrix of a single link is derived. To develop the kinematic relation of the parallel manipulators, the screw theory is usually a convenient method. In contrast, the Denavit-Hartenberg coordinates are mostly used for serial manipulators. As explained before, the manipulator considered in this paper is a serial-parallel manipulator, in which the parallel mechanisms are connected to each other serially. This kind of manipulator has higher stiffness and accuracy compared with serial manipulators, and a larger workspace compared with the parallel manipulator [22]. Gallardoet et al. derived the velocity and acceleration kinematic relations for a 2(3RPS) mechanism by means of the screw theory and the principle of virtual work [23]. In this paper, an N(3RPS) mechanism is considered, but just one link is unlocked at each moment. Therefore, the kinematic relation might be developed in another way for convenience. According to Figure 3, the OABP kinematic loop relation is written as:

$$\overline{OA_i} + \overline{A_iB_i} = \overline{OP} + \overline{PB_i}, \tag{9}$$

where all of the vectors are presented in XYZ coordinates. The derivative of the above kinematic loop is:

$$v_p + \omega_B \times b_i = l_i\omega_i \times s_i + \dot{l}_i s_i, \tag{10}$$

where s_i is the unit vector along the ith leg of the mechanism. The rotational velocity of the ith leg is denoted by ω_i. To eliminate ω_i from Eq. (10), s_i is multiplied to both sides of it:

$$s_i.v_p + (b_i \times s_i).\omega_B = \dot{l}_i. \tag{11}$$

If Eq. (11) is written for three legs and arranged in a matrix;

$$J_{\text{link}}\dot{x}_m = \dot{l}, \tag{12}$$

where J_{link} is the link Jacobian:

$$J_{\text{link}} = \begin{bmatrix} s_1^T & (b_1 \times s_1)^T \\ s_2^T & (b_2 \times s_2)^T \\ s_3^T & (b_3 \times s_3)^T \end{bmatrix}_{3\times 6},$$

$$\dot{x}_m = \begin{bmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{z}_m \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \qquad \dot{l} = \begin{bmatrix} \dot{l}_1 \\ \dot{l}_2 \\ \dot{l}_3 \end{bmatrix}.$$

The link Jacobian matrix maps the upper platform velocity ($\dot{x}_m$) to the legs velocity ($\dot{l}$).

2.2.2. Manipulator Jacobian

The kinematic control of the manipulator requires the transformation between the velocity of legs and velocity of the end-effector. Therefore, the Jacobian matrix of the whole manipulator is studied in this section.

During the movement of the ith link, the upper and lower links are locked and act as a rigid body. The velocity of the end-effector is a combination of linear velocity, v_i, and rotational velocity, ω_i, of the upper platform of the ith link. One can combine these velocities according to the simple relation; $V = v_i + \omega_i \times r_i$, where r_i is the position vector of the end-effector in the ith link coordinating system. This equation is written in matrix form as:

$$J_i\dot{x}_i = \dot{S} \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & r_{iz} & -r_{iy} \\ 0 & 1 & 0 & -r_{iz} & 0 & r_{ix} \\ 0 & 0 & 1 & r_{iy} & -r_{ix} & 0 \end{bmatrix}}_{J_i} \begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{z}_i \\ \omega_{ix} \\ \omega_{iy} \\ \omega_{iz} \end{bmatrix}$$

$$= \underbrace{\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix}}_{\dot{S}}, \tag{13}$$

where J_i maps the velocity of the upper platform of the ith link to the velocity of the end-effector. The end-effector velocity is denoted by $\dot{S}$. The desired map between the velocity of the legs and end effector velocity is obtained as:

$$^iJ_{pi} = J_iJ^+_{\text{link}}, \qquad ^0J_{pi} = {}^0_iR\,{}^iJ_{pi},$$

$$\dot{S}_{3\times 1} = \underbrace{{}^0J_{pi}}_{3\times 3}\dot{l}_{3\times 1}, \tag{14}$$

where J^+_{link} is the pseudo-inverse of J_{link}. By multiplying the Jacobian $^iJ_{pi}$ to the rotation matrix, ${}^0_iR$, this relationship is expressed in the base coordinate system, where ${}^0_iR$ is the rotation matrix between the ith link and the base coordinate system, and $^0J_{pi}$ is the Jacobian matrix presented in the base coordinate system. It is worth mentioning that the ith coordinate system is not moving, so, the derivative of the rotation matrix will not enter the formulation.

As described in the introduction, the locking system in each link creates a relation between the lengths of legs. The constraint equation is expressed as $l_1 + l_2 + l_3 = \text{Const}$, and the derivative of it yields:

$$\dot{l}_3 = -\dot{l}_1 - \dot{l}_2. \tag{15}$$

This equation expresses that the velocity of the third leg is dependent on the other two legs velocity. According to Eq. (15), the dimension of the Jacobian must be changed as follows:

$$^0J_{pi} = \begin{bmatrix} ^iJ_1 & ^iJ_2 & ^iJ_3 \end{bmatrix}_{3\times 3},$$

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} ^iJ_1 & ^iJ_2 & ^iJ_3 \end{bmatrix} \begin{bmatrix} \dot{l}_1 \\ \dot{l}_2 \\ -\dot{l}_1 - \dot{l}_2 \end{bmatrix}, \tag{16}$$

Eq. (16) can be rewritten as:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} ^iJ_1 - ^iJ_3 & ^iJ_2 - ^iJ_3 \end{bmatrix} \begin{bmatrix} \dot{l}_1 \\ \dot{l}_2 \end{bmatrix},$$

$$^0J_{pi}^{\text{new}} = \begin{bmatrix} ^iJ_1 - ^iJ_3 & ^iJ_2 - ^iJ_3 \end{bmatrix}, \tag{17}$$

where $^0J_{pi}^{\text{new}}$ is the Jacobian of the manipulator. The manipulator Jacobian matrix is 3×2 dimensional and transforms the velocity of the ith link leg to the velocity of the end-effector. In order to derive the Jacobian, it is necessary to know exactly the current configuration of the manipulator. The current configuration is calculated by forward kinematic equations. Actually, the lengths of the cables are measured by encoders, and Eqs. (5) and (2) are used, respectively, to compute the elements of the Jacobian matrix.

In the following section, the kinematic formulation is used for kinematic control of the robot.

3. Control scheme

In this section, the kinematic control of the manipulator is investigated. The kinematic control is similar to the path planning algorithm that generates the desired length of wires. The difference is that the path planning is an offline algorithm, but the control scheme presented in this part is an online algorithm. The most challenging part of the online control in this special manipulator is the selection of an optimum switch, which will be described later. In order to implement this strategy, an internal control is required to assure that the wires are in tension. The internal control prevents the wires from looseness. In other words, two wires are length controlled and the third one is force controlled. The wires tensions are measured via single axis load cells. A PD controller changes the length of the third wire to keep the average force of the wires in a positive value. This positive value can be $(F_{\max} + F_{\min})/2$ to increase the resultant torque capacity of the wires. $F_{\min}$ and $F_{\max}$ are the minimum and maximum allowable force of wires. This controller acts in parallel with the control scheme presented in this section and there will be no interference. A detailed description is given in [18].

In the control of the manipulators, the kinetics of the robot should be considered and the required torque for actuators should be checked, generally. In this paper, kinematic control is considered. In fact, the kinetic analysis of this manipulator is complicated, and this assumption, i.e. kinematic control without considering kinetics, is undertaken to simplify the problem. It is assumed that the actuators are powerful enough to generate the required torque and they can follow the desired velocity with desired acceleration. It is also assumed that wires have not any longitudinal flexibility. If these two assumptions hold, the problem can be reduced to kinematics.

There are three main parts in the control block diagram of the manipulator. In the first block, named the decision making block, the proper switch will be chosen. Indeed, this block selects an appropriate link to be unlocked. The second block is a Cartesian controller which produces a proper velocity vector for the end-effector ($\dot{S}$) to reduce the end-effector position error with maximum possible velocity. The input of this block is the desired trajectory (S_d) and the position of the end-effector (S). The third block, inverse kinematics, is a mapping between the legs and the end-effector velocity, and was explained in the previous section. The block diagram of this control scheme is shown in Figure 4.

In the following parts, each block is described in detail.

3.1. Decision making

This section presents the logical part of the whole algorithm. The logical part is one of the main parts in hybrid control algorithms and, generally, no systematic design method exists for it. In this part, an innovative algorithm for switch selection is presented. At each instant of time, a link must be selected and unlocked for the movement. The criterion of selection is to obtain proper tracking and smoothness in motion.

Consider a predetermined trajectory, S_d and $\dot{S}_d$, for the end-effector. If the ith link of the manipulator moves and the other links stay unchanged, the direction of the end-effector movement is a function of $\dot{l}_1$ and $\dot{l}_2$ of that link, and it keeps moving in a plane at each moment. The normal vector of this plane might be determined by cross-multiplying the direction of two arbitrary vectors on the plane. By substituting $\dot{l}_2 = 1$, $\dot{l}_1 = 0$ and $\dot{l}_2 = 0$, $\dot{l}_1 = 1$ in Eq. (17), these vectors are obtained as:

$$N_{\text{plane } i} = N_1 \times N_2, \qquad N_1 = {}^iJ_1 - {}^iJ_3,$$

$$N_2 = {}^iJ_2 - {}^iJ_3. \tag{18}$$

This plane is unique at this moment for each link. According to Figure 2, the desired vector, $d(t)$, is

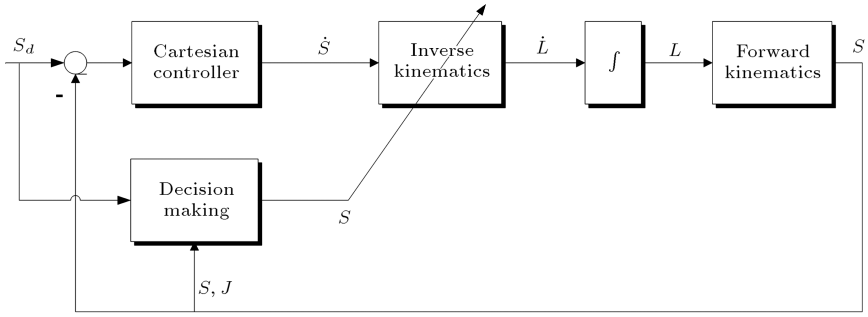

Figure 4. Block diagram of kinematic control of redundant manipulator with lockable joints.

defined by:

$$d(t) = S_d(t) - S(t), \tag{19}$$

where $S(t)$ is the current position of the end-effector, which can be determined, in practice, for example, by image processing. The angle between $d(t)$ and the normal vector of the plane for each link ($N_{\text{plane } i}$) is the criterion of link selection:

$$\theta_i(t) = 90 - \cos^{-1}\left(\frac{N_{\text{plane } i} \times d(t)}{\|N_{\text{plane } i}\| \, \|d(t)\|}\right) (\text{deg}). \tag{20}$$

Now, the link which has the minimum angle, $\theta_i(t)$, is chosen as the best link. At each instant of time, if the mentioned special link is unlocked, the direction of movement will have the most proximity to the desired direction, therefore, that special link is a suitable choice.

This algorithm considers the angle between the possible directions of motion and the desired direction and chooses a link which provides the minimum angle. In other words, the algorithm uses the desired point of the next step and calculates the angle between the desired direction of the end-effector and the direction of the end effector when a certain link is unlocked and chooses the minimum angle. Using the algorithm presented in this section, online trajectory tracking can be achieved.

By this method of selection, a problem may occur. The lengths of jacks may reach their extremes. In other words, a link becomes saturated, while the decision making algorithm keeps selecting that saturated link as the best choice. To avoid this problem, the selection of that link should be forbidden for several iterations. By this method, other links are selected instead of that saturated link, and after some iteration, the situation of the manipulator is changed in such a way that the saturated link escapes from its extreme location.

3.2. Controller

The controller is defined in Cartesian space and generates an appropriate input for the inverse kinematic equations. The controller uses a combination of the error and its integration. Consider the error equation as:

$$\dot{e} + k_p e + k_I \int_0^t e dt = 0, \tag{21}$$

where $e = S_d - S$, and k_p and k_l are proportional and integral gains, respectively. They should be adjusted by trial and error to get the best convergence speed and avoid chattering, i.e. the high frequency vibrations in the output. The error, e, is a 3×1 vector, and each controller gain is a 3×3 matrix. Eq. (21) can be rewritten in the following form:

$$\dot{S} = \dot{S}_d + k_p(S_d - S) + k_I \int_0^t (S_d - S) dt. \tag{22}$$

In this equation, if $S_d = S$, the magnitude of $\dot{S}$ will approach $\dot{S}_d$.

3.3. Inverse kinematic

When a suitable link is chosen by the decision making block and the Jacobian matrix is calculated and a proper velocity input is determined by the controller, everything is ready to find the legs velocities. Rewriting the inverse form of Eq. (17), the velocity of legs can be determined easily by:

$$\begin{bmatrix} \dot{l}_1 \\ \dot{l}_2 \end{bmatrix} = \left[{}^0 J_{pi}^{\text{new}}\right]^+ \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix}, \tag{23}$$

where:

$${}^0 J_{pi}^{\text{new}} = \begin{bmatrix} {}^i J_1 - {}^i J_3 & {}^i J_2 - {}^i J_3 \end{bmatrix},$$

denotes the final Jacobian matrix. In this equation, the desired velocities in x, y and z directions are given, and the velocities of two legs must be determined. So, the Jacobian is not square and pseudo-inverse techniques should be used for inverse calculation. Here, the

minimum damping square method is used to prevent singularity and increase the velocity norm; other techniques can be found in [24,25]:

$$\dot{l}^{(\lambda)} = \left([^0J_{pi}^{\text{new}}]^T[^0J_{pi}^{\text{new}}] + \lambda^2 I\right)^{-1}[^0J_{pi}^{\text{new}}]^T\dot{S}, \qquad (24)$$

where λ is damping ratio. By substituting $\dot{S}$ from Eq. (22), the following equation is obtained:

$$\dot{l}^{(\lambda)} = \left([^0J_{pi}^{\text{new}}]^T[^0J_{pi}^{\text{new}}] + \lambda^2 I\right)^{-1}[^0J_{pi}^{\text{new}}]^T$$
$$(\dot{S}_d + k_p(S_d - S) + k_I\int_0^t (S_d - S)dt). \qquad (25)$$

This equation is used to determine the legs velocities, and then the lengths of legs are determined by integration. After that, the next iteration starts from the forward kinematic calculation. It is worth mentioning that the leg length and the leg velocity are equivalent to the cable length and the cable velocity, respectively.

4. Results

The results are presented for an 8-DOF manipulator (4 links). The specification and parameters of the manipulator are shown in Table 1.

In the first part, a circular trajectory is considered in the x-y plane. The trajectory specifications and algorithm parameters are shown in Table 2.

The controller gains are adjusted by trial and error in which the integration gain is set to zero at first and a proper proportional gain is obtained. Then, the steady state error has been decreased by increasing the integration gain. Finally, the following gains are used:

$$k_p = \text{diag}[100, 100, 100]\ \text{sec}^{-1},$$
$$k_I = \text{diag}[30, 30, 40]\ \text{sec}^{-2}. \qquad (26)$$

The simulation is performed; Figure 5 shows the final configuration of the manipulator. The desired and actual paths of the end-effector motion are also shown in Figure 5. The starting point of motion is set out of the desired path and the manipulator approaches it gently. It should be noted that the step size of the solver will change the quality of solution greatly. As the step size decreases, the smoothness of the trajectory and the number of switches will grow.

Figure 6 shows the desired and actual positions of the end-effector in x, y and z directions, and also the position error trajectories. The average errors in these directions are: -1.96×10^{-4}, -3.87×10^{-4} and 0.0052 meters, respectively. It can be observed that the manipulator motion has some chattering in 0 to 5 seconds and the number of switches has increased. The reason is that the manipulator could not follow the desired trajectory by unlocking a certain link. On the other hand, the angles between $d(t)$ and $N_{\text{plane } i}$ for all links are too small. As a result, the manipulator has to switch quickly between different links to keep the error nearby zero. The average velocity, errors in x, y and z directions are -0.0013, -0.0016 m, -1.46×10^{-5} meters per second and the relative error is 2%. The actual trajectory consists of 215 switches, the area between the desired and actual trajectory is 0.1269 square meters and the maximum cable speed is 0.0277 meters per second. Figure 7 shows the end-effector motion path clearly. If the step size increases,

Table 1. The manipulator specification and parameters.

Manipulator parameters	(mm)
The length of fixed platform d_1	100
The length of moving platform d_2	70
Minimum leg length $l_{\min}$	100
Maximum leg length $l_{\max}$	150
The summation of leg length	375

Table 2. Circular trajectory specification and algorithm parameters.

Trajectory specification and algorithm parameter	Value
Step size (s)	0.008
Damping ratio-λ	1
Circular trajectory radius (m)	0.15
Circular trajectory height (m)	0.43
Simulation time (s)	20

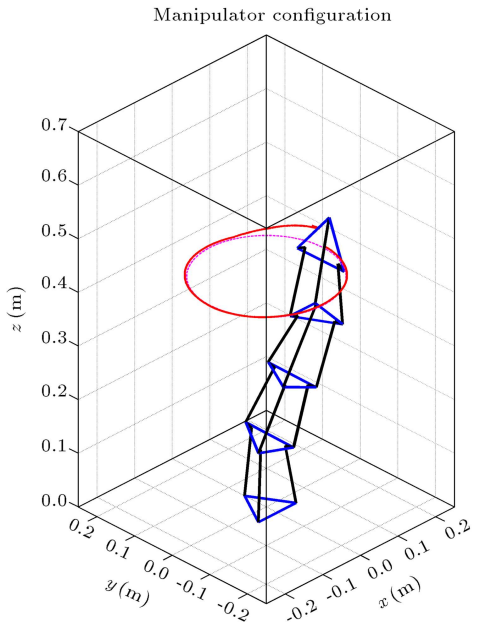

Figure 5. Manipulator configuration at starting point.

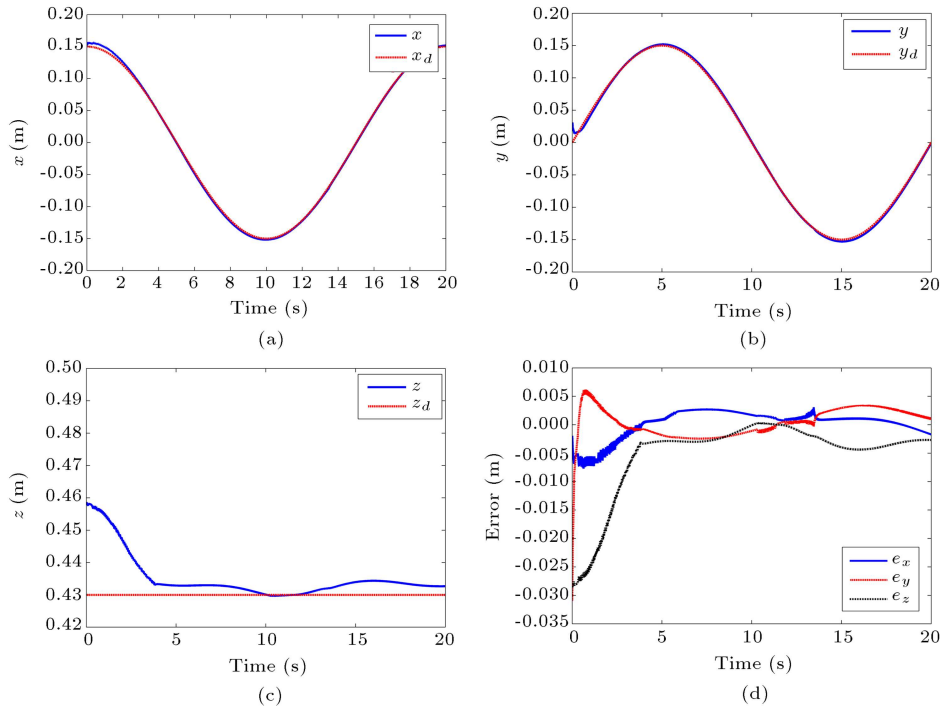

Figure 6. The desired and actual position of the end-effector in (a) x direction, (b) y direction and (c) z direction. (d) The position error trajectories.

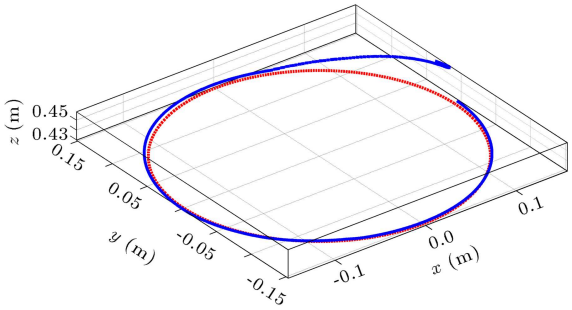

Figure 7. The desired and actual path of the end-effector.

the number of switches and the quality of motion will decrease. For example, if the step size changes to 0.05 (s), the number of switches will become 81.

In the following part, the rose of three pedals is considered as a reference trajectory. The polar and Cartesian formulations of this trajectory are given in Eq. (27):

$$r(t) = A\cos(3\theta(t)), \qquad \theta(t) = \frac{2\pi t}{T},$$

$$x(t) = r(t)\cos(\theta(t)), \qquad y(t) = r(t)\sin(\theta(t)),$$

$$\dot{x}(t) = \dot{r}(t)\cos(\theta(t)) - r\dot{\theta}(t)\sin(\theta(t)),$$

$$\dot{y}(t) = \dot{r}(t)\sin(\theta(t)) + r\dot{\theta}(t)\cos(\theta(t)). \qquad (27)$$

By derivation of x and y, with respect to time, the end-effector desired velocities in x and y directions are determined. This trajectory is located in a plane parallel to the x-y plane and at 0.45 m height. A and T in Eq. (27) are set to 0.15 meters and 50 seconds, respectively. The controller gains are the same as the previous part, except that k_p is doubled.

Figure 8 shows the end-effector actual and desired paths and the configuration of the manipulator at the starting point. The manipulator has switched 309

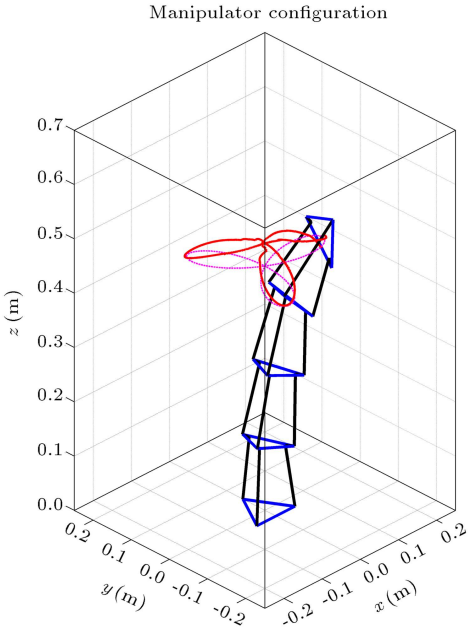

Figure 8. Manipulator configuration at starting point.

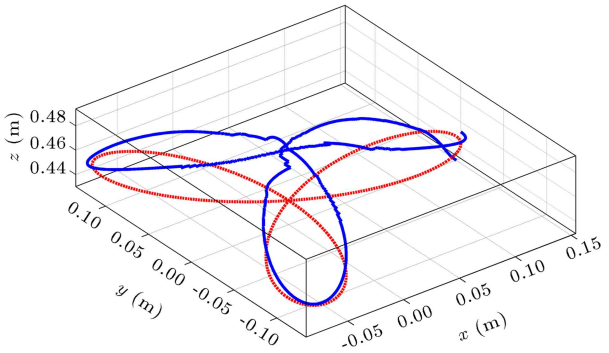

Figure 9. The desired and actual path of the end-effector.

times between different links. Increasing the proportional gain of the controller will increase the switching numbers directly. The maximum cable velocity is 0.1466 meters per second, which reduces by decreasing the time step. Figure 9 shows the actual and desired trajectories of the end-effector in detail. It can be observed that the actual trajectory is not coincident with the desired trajectory at $x = y = 0$. The reason is that the manipulator has to unlock the lower links and move in the opposite direction to eliminate the error caused by previous movements. On the other hand, the manipulator is forced to follow a predefined velocity (Eq. (27)). Consequently, the position error in the z direction could not be eliminated in this trajectory. This problem may be solved by introducing a more complicated decision-making algorithm. The switch selection algorithm presented in this paper just uses the next desired point. If the algorithm were able to consider the whole desired trajectories, the z direction error might be reduced, which is investigated in future work.

Figure 10 shows the desired and actual positions of the end-effector in x, y and z directions, and the position error trajectories. It can be observed that the tracking performance is excellent in the x and y directions and acceptable in the z direction. The average error in the z direction is 0.0191 meters, which is equivalent to 4% of desired magnitude. Table 3 shows the average position and velocity errors in this trajectory for two different time steps. According to the results presented in this table, reducing the time step will increase the number of switches and reduce the average errors in x and y directions. When the time step decreases, the manipulator is allowed to change its moving link in one second (more than before), so the accuracy of motion and the number of switches will increase.

5. Conclusions

This paper presents a novel lockable redundant mechanism produced by making a chain of elementary blocks constituted of 3RPS parallel structures. The legs of the 3RPS mechanism are hydraulic jacks whose outputs are interconnected. This way, it is possible to achieve a coupling between the 3 translations of jacks, which results in reducing the number of degrees of freedom of the platform to two. Also, by using 3 solenoid valves, it is possible to lock the elementary 3RPS

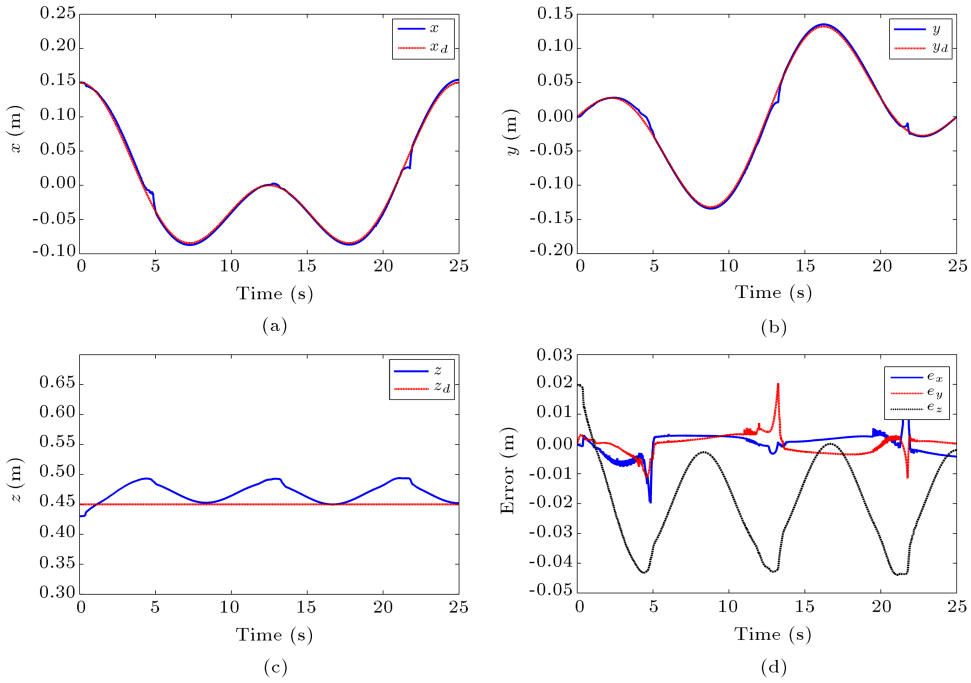

Figure 10. The desired and actual position of the end-effector in (a) x direction, (b) y direction and (c) z direction. (d) The position error trajectories.

Table 3. Results of simulation in the rose of three pedals trajectory in two different step sizes.

Error evaluation criterion	Step size=0.008 s	Step size=0.05 s
Average position error in x direction (m)	-0.00008	-0.00083
Average position error in y direction (m)	-0.00003	-0.00052
Average position error in z direction (m)	0.0191	0.0108
Average velocity error in x direction (m/s)	0.0001	0.0001
Average velocity error in y direction (m/s)	0.00002	-0.0012
Average velocity error in z direction (m/s)	-0.00087	-0.0036
Switches number	309	131

block in its current configuration. The actuation of the robot is performed using 3 cables passing through the robot. The locking systems avoid multiplying the number of cable triplets by the number of elementary blocks. One of the achievements of this study is in proposing a control strategy for selecting properly the best link to be unlocked for controlling the situation of the manipulator. It is chosen according to a maximum manipulability criterion with respect to the desired trajectory. Simulation results illustrate the efficiency of the control strategy. The control scheme is also used for tracking a circular trajectory and the rose of three pedals trajectory. The results are discussed for different time steps. Dynamic modeling should be done in future work, and the required torques of the actuators should also be computed to assure that the manipulator can follow the desired path with this strategy.

References

1. Zhang, Y., Wang, J. and Xia, Y. "A dual neural network for redundancy resolution of kinematically redundant manipulators subject to joint limits and joint velocity limits", *IEEE Transactions on Neural Networks*, **14**(3), pp. 658-667 (2003).
2. Choset, H. and Henning, W. "A follow the leader approach to serpentine robot motion planning", *ASCE Journal of Aerospace Engineering*, **12**(2), pp. 65-73 (1999).
3. Hayashi, A., Park, J. and Kuipers, B.J. "Toward planning and control of highly redundant manipulators", *5th IEEE International Symposium on Intelligent Control*, pp. 683-688 (1990).
4. Wilson, J. and Mahajan, U. "The mechanics and positioning of highly flexible manipulator limbs", *Journal of Mechanisms, Transmissions and Automation in Design*, **111**(2), pp. 232-237 (1989).
5. Li, Y., Ma, P., Qin, C., Gao, X., Wang, J. and Zhu, H. "Design and study of a novel hyper-redundant manipulator", *Robotica*, **21**, pp. 505-509 (2003).
6. Ananiev, A., Kalaykov, I., Petrov, E. and Hadjiysk, B. "Single-motor driven construction of hyper-redundant robot", *International Conference on Mechatronics and Robotics*, Aachen, Germany (2004).
7. Todd, D., *Fundamentals of Robot Technology: An Introduction to Industrial Robots, Teleoperators, and Robot Vehicles*, Halsted Press New York, NY, USA (1986).
8. Drozda, T. "The spine robot... the verdict's yet to come", *Manufacturing Engineering*, **93**(3), pp. 110-112 (1984).
9. Hirose, S. and Ma, S. "Coupled tendon-driven multi-joint manipulator", *IEEE International Conference on Robotics and Automation* (1993).
10. Kimura, S., Tsuchiya, S., Takegai, T. and Iizuka, N. "Hyper redundant modular manipulator arm", *IECON 2000, 26th Annual Conference of the IEEE Industrial Electronics Society*, pp. 2467-2472 (2000).
11. Chirikjian, G. and Burdick, J.W. "Design and experiments with a 30 DOf robot", *IEEE International Conference on Robotics and Automation*, Atlanta, GA (1993).
12. Shammas, E., Wolf, A., Brown, H.B. and Choset, J.H. "New joint design for three-dimensional hyper redundant robots", *IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, Las Vegas, Nevada (2003).
13. Jones, B.A., McMahan, W. and Walker, I. "Design and analysis of a novel pneumatic manipulator", *Proceedings of 3rd IFAC*, Australia (2004).
14. Ning, K.J. and Worgotter, F. "A novel concept for building a hyper-redundant chain robot", *IEEE Transactions on Robotics*, **25**(6), pp. 1237-1248 (2009).
15. Ning, K.J. and Wörgötter, F. "To paint what is seen: A system implementation of a novel conceptual hyper-redundant chain robot with monocular vision", *6th German Conf. on Robotics (ROBOTIK)*, pp. 1-6 (2010).
16. Ning, K. and Wörgötter, F. "Control system development for a novel wire-driven hyper-redundant chain robot, 3d-trunk", *IEEE/ASME Transactions on Mechatronics*, **99**, pp. 1-11 (2012).
17. da Graça Marcos, M., Tenreiro Machado, J. and Azevedo-Perdicoúlis, T.-P. "Trajectory planning of redundant manipulators using genetic algorithms", *Communications in Nonlinear Science and Numerical Simulation*, **14**(7), pp. 2858-2869 (2009).

18. Honarvar, M. "Design and control of a new tendon actuated manipulator with lockable joints", MSc, Sharif University of Technology (2008).

19. Shafaei, M. "Optimization, manufacturing and utilization of a new tendon actuated hyper redundant manipulator", Master of Science (MS), Mechanical Engineering, Sharif University of Technology (2011).

20. Taherifar, A., Alasty, A., Salarieh, H. and Boroushaki, M. "Path planning for a planar hyper-redundant manipulator with lockable joints using particle swarm optimization", *First RSI/ISM International Conference on Robotics and Mechatronics*, Tehran (2013).

21. Taherifar, A., Salarieh, H. and Alasty, A. "Minimum time and minimum switch path planning for a hyper-redundant manipulator with lockable joints", *The Modares Journal of Mechanical Engineering*, **12**(1), pp. 50-65 (2012).

22. Hu, B., Lu, Y., Yu, J.J. and Zhuang, S. "Analyses of inverse kinematics, statics and workspace of a novel 3rps-3spr serial-parallel manipulator", *The Open Mechanical Engineering Journal*, **5**, pp. 65-72 (2012).

23. Gallardo, J., Lesso, R., Rico, J.M. and Alici, G. "The kinematics of modular spatial hyper-redundant manipulators formed from rps-type limbs", *Robotics and Autonomous Systems*, **59**(1), pp. 12-21 (2011).

24. Spong, M.W., Hutchinson, S. and Vidyasagar, M. *Robot Dynamics and Control*, John Wiley & Sons, pp. 190-192 (2004).

25. Chalongrath, P. "Task-based decision making and control of robotic manipulators", Doctoral Dissertation, The University of Texas at Austin (2004).

Biographies

Ali Taherifar was born in Tehran, Iran, in 1986. He received his BS degree in Mechanical Engineering, in 2009, from K.N. Toosi University of Technology, Iran, and his MS degree, in 2011, from Sharif University of Technology, Tehran, Iran, where he is currently studying for his PhD degree. His research interests include human-robot interaction control and parallel mechanisms dynamics and control.

Hassan Salarieh received BS, MS and PhD degrees in Mechanical Engineering in 2002, 2004 and 2008, respectively, from Sharif University of Technology, where he is currently Assistant Professor. His research interests include chaos control, analysis of chaotic systems and fractional systems analysis and control.

Aria Alasty received BS and MS degrees in Mechanical Engineering from Sharif University of Technology, Tehran, Iran, in 1987 and 1989, respectively, and his PhD degree from Carleton University, Canada, in 1996. He is currently Professor at Sharif University of Technology. His research interests include nonlinear systems and control, fuzzy systems and chaos control.