Obstacle Avoidance of Three-DOF Underactuated Manipulator by Using Switching Computed Torque Method

Lanka Udawatta, Keigo Watanabe, Kiyotaka Izumi, and Kazuo Kiguchi

Abstract: Obstacle avoidance of underactuated robot manipulators using switching computed torque method (SCTM) is presented. One fundamental feature of this novel method is to use partly stable controllers (PSCs) in order to fulfill the ultimate control objective. Here, we use genetic algorithms (GAs) to acquire the optimum switching sequence of the control actions for a given time frame with the available set of elemental controllers, depending on which links/variables are controlled. The effectiveness of the concept is illustrated by taking a three-degrees-of-freedom (DOF) manipulator and showing enhanced performance of the proposed control methodology.

Keywords: underactuated manipulators, computed torque method, obstacle avoidance, nonlinear control, genetic algorithms.

I. Introduction

Research on underactuated robot manipulators has received considerable attention in recent years. The class of manipulators with fewer actuators than degrees-of-freedom (DOF) is generally referred as underactuated robot manipulators. However, characteristics such as complex nonlinear dynamics, nonholonomic behavior, and lack of linearizability are often exhibited by this class of nonlinear systems [1]–[4]. The constraints of a dynamical system which can not be represented in the form of $F(\boldsymbol{q}, t) = 0$ are defined as nonholonomic, where $\boldsymbol{q}$ and t denote the coordinates and time respectively. Examples of nonholonomic control systems have been studied in the context of robot manipulators, mobile robots, wheeled vehicles, and space robotics [1],[5]. This type of underactuated robots can play a key role in the areas such as space robots, undersea vehicles, hyper-redundant robots, fault-tolerance robot systems, mobile robots, wheeled vehicles, *etc.* [6]–[8]. The free joints that can rotate freely, or the passive joints that have no actuation but are equipped with passive element like a damper or a breaks, render in the advantages such as reduction of weight, energy consumption and cost of manipulators. Thus, controlling this class of robots is a challenging task and still remains as an open problem. Therefore, further investigation is required to find plausible control approaches to harvest the promising features of underactuated robot manipulators.

Control of underactuated manipulators has been investigated by several researchers in recent studies [2]–[4],[7]–[15]. Stabilization of a PR (i.e., prismatic–revolutionary joints) planar manipulator is carried out via partial feedback linearization and nilpotent approximation [3]. In [2], trajectories for positioning are composed of simple translational and rotational trajectory segments and the trajectory segments are stabilized by nonlinear feedback control. Furthermore, cooperated control for underactuated manipulators has been brought into the research field [8], in which among the many possible control sequences of a robot, the optimal control sequence is selected by dynamic programming. Control of robot manipulators via chaos attractors and fuzzy model-based regulators is focused in [12],[13], where the closed-loop stability analysis was carried out by employing a set of Linear Matrix Inequalities (LMI). Appling fuzzy logic control with two major steps, an underactuated robot with a fuzzy microcontroller is presented in [14]. Here, on the basis of the position of first link and on the position error of the second link, a control action is capable of modifying the position of the first link with suitable voltage to the motor to obtain the right speed. Obstacle avoidance motion planning for a three-axis planar manipulator with a passive revolutionary third joint from one zero velocity state to another was investigated in [11]. This study brings motion planning from one zero velocity state to another for a three-joint robot in a horizontal plane with a passive revolute third joint. Even though various researchers have carried out considerable numbers of studies, there are some limitations, drawbacks and disadvantages.

The main purpose of this paper is to propose a control strategy for underactuated manipulators using switching computed torque method (SCTM), especially with avoiding obstacles. One of the major advantages of this method is the ability of controlling the entire system without using rigorous linearizations or deformation of the original nonlinear system and employing simple computed torque controllers as partly stable controllers (PSCs) in the control system. The obstacle avoidance of underactuated robot manipulators using SCTM is an extended version of our former research works [5],[19],[20]. This genetic algorithms (GAs) based scenario can promote authors to look beyond the classical methods such as chained form [17] while adding intelligent controlling. The rest of the paper is organized as follows: In Section 2, brief introduction to the SCTM and controller selection criteria are presented. Controller selection, design, and results of PSCs of three-DOF robot manipulator are discussed and presented in Section 3. Systematic procedure for obstacle avoidance and results are presented in Section 4 and Section 5 respectively. Finally, concluding remarks and perspectives for future extensions are given in Section 6.

Manuscript received: July. 18, 2002., Accepted: Sept. 27, 2002.

Lanka Udawatta: Dept. of Advanced Systems Control Engineering, Saga University, Japan. (lanka@ieee.org)

Keigo Watanabe: Dept. of Advanced Systems Control Engineering, Saga University, Japan. (watanabe@me.saga-u.ac.jp)

Kiyotaka Izumi: Dept. of Advanced Systems Control Engineering, Saga University, Japan. ((izumi@me.saga-u.ac.jp)

Kazuo Kiguchi: Dept. of Advanced Systems Control Engineering, Saga University, Japan. (kiguchi@me.saga-u.ac.jp)

II. Concept of switching computed torque method

For the purpose of designing the controller, consider the dynamic model of a manipulator given below:

$$M(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{F} \tag{1}$$

where $\boldsymbol{q} \in \Re^n$ is the generalized coordinate vector and $\boldsymbol{F} \in \Re^n$ is the input force/torque vector. $M(\boldsymbol{q}) \in \Re^{n \times n}$ is the symmetric, positive-definite inertia matrix and $\boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \Re^n$ represents Coriolis, centrifugal, gravitational, and friction components. Suppose that an underactuated robot system has m_u number of actuators to control and n-DOF ($1 \leq m_u < n$) links. The equation (1) can be rearranged as follows in order to obtain the desired second-order derivatives ($\ddot{\boldsymbol{q}}$):

$$\begin{aligned} \ddot{\boldsymbol{q}} &= M^{-1}(\boldsymbol{q}) \{-\boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{F}\} \\ &= \frac{1}{D} \hat{M}(\boldsymbol{q}) \{-\boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{F}\} \end{aligned} \tag{2}$$

where $D = \det(M)$ and $\hat{M}$ is the adjoint matrix of M. It is assumed that M is invertible. Here, we represent all the active joint forces as $[F_1 \ F_2 \ \dots \ F_{m_u}]^T \triangleq \boldsymbol{\mathcal{F}} \in \Re^{m_u}$ and it is assumed that m_u-actuators are directly allocated to the roots of m_u-links starting from the first joint. Now, we have ${}_nC_{m_u}$ number of combinations of m_u-dimensional controllers. Moreover, the number of different combinations of n-variables and m_u-actuators at a time, without repetitions, is

$$\begin{pmatrix} n \\ m_u \end{pmatrix} = \frac{n(n-1)\cdots(n-m_u+1)}{m_u!}. \tag{3}$$

One of the above combinations of available controllers should be selected at a given time in order to actuate the robot system (3). For simplicity, we present how to derive an elemental controller for selected m_u number of link coordinates out of n-links as shown below:

$$\ddot{\boldsymbol{s}} = \frac{1}{D} \begin{bmatrix} \hat{S}_{11}(F_1 - h_1) + \cdots + \hat{S}_{1n}(-h_n) \\ \vdots \\ \hat{S}_{m_u 1}(F_1 - h_1) + \cdots + \hat{S}_{m_u n}(-h_n) \end{bmatrix}. \tag{4}$$

This can be rewritten as:

$$\begin{aligned} \ddot{\boldsymbol{s}} &= \begin{bmatrix} \hat{S}_{11} & \cdots & \hat{S}_{1m_u} & \cdots & \hat{S}_{1n} \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ \hat{S}_{m_u 1} & \cdots & \hat{S}_{m_u m_u} & \cdots & \hat{S}_{m_u n} \end{bmatrix} \boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \\ &\quad + \frac{1}{D} \begin{bmatrix} \hat{S}_{11} & \cdots & \hat{S}_{1m_u} \\ \vdots & \cdots & \vdots \\ \hat{S}_{m_u 1} & \cdots & \hat{S}_{m_u m_u} \end{bmatrix} \boldsymbol{\mathcal{F}} \end{aligned}$$

where $\boldsymbol{s} \triangleq [s_1 \ s_2 \ \dots \ s_{m_u}]^T$ denotes one possible combination of m_u-link coordinates out of $\{q_1, q_2, ..., q_n\}$ and $\hat{S}_{ij}$ ($i = 1, ..., m_u; j = 1, ..., n$ or m_u) denotes their associated cofactors consisting of inertia element. However, it should be noted that the choice of desired controllable links/variables using a set of elemental controllers remains open. Define $\mathcal{N}$ and $\mathcal{M}$ matrices as

$$\mathcal{N} = \frac{1}{D} \begin{bmatrix} \hat{S}_{11} & \cdots & \hat{S}_{1m_u} & \cdots & \hat{S}_{1n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \hat{S}_{m_u 1} & \cdots & \hat{S}_{m_u m_u} & \cdots & \hat{S}_{m_u n} \end{bmatrix},$$

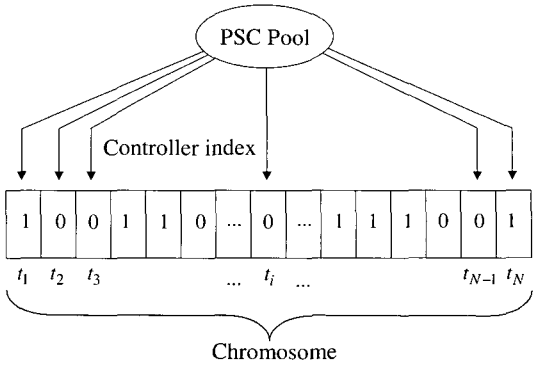


Fig. 1. Coding of genes with control indices

$$\mathcal{M} = \frac{1}{D} \begin{bmatrix} \hat{S}_{11} & \cdots & \hat{S}_{1m_u} \\ \vdots & \ddots & \vdots \\ \hat{S}_{m_u 1} & \cdots & \hat{S}_{m_u m_u} \end{bmatrix}. \tag{5}$$

Since the inertia matrix M is positive definite, $\mathcal{M} \in \Re^{m_u \times m_u}$ is also a full rank matrix for any control law. After simplifying, we have a set of control inputs for any control law as below:

$$\boldsymbol{\mathcal{F}} = \mathcal{M}^{-1} \{\ddot{\boldsymbol{s}}^* + \mathcal{N}\boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}})\} \tag{6}$$

where $\ddot{\boldsymbol{s}}^*$ is the modified acceleration, which can be constructed by using a simple PD servo such as:

$$\ddot{\boldsymbol{s}}^* = \ddot{\boldsymbol{s}}_d + K_v (\dot{\boldsymbol{s}}_d - \dot{\boldsymbol{s}}) + K_p (\boldsymbol{s}_d - \boldsymbol{s}) \tag{7}$$

in which $\boldsymbol{s}_d$, $\dot{\boldsymbol{s}}_d$ and $\ddot{\boldsymbol{s}}_d$ are the desired position, velocity and acceleration vectors of selected link coordinates to be controlled, and $K_v > 0$ and $K_p > 0$ are the derivative and position gain matrices. Thus, we can synthesize all the ${}_nC_{m_u}$ PSCs using the above procedure.

To solve the general control problem with optimum switching of available PSCs, we define the total time span t_N of a trajectory $\Phi(t)$ such that $\{\Phi(t); t \in [0, t_N]\}$. The genes of a chromosome are represented as controller indices. Fitness in this case is assumed to be constructed by the error between the desired values and state variables for the total time span t_N, i.e., from $t = 0$ to $t = t_N$. Thus, the fitness function of the GA optimization process is defined as follows:

$$fitness = \sum_{k=1}^{N} \|\boldsymbol{x}_d(k) - \boldsymbol{x}(k)\|^2_{W(k)} \tag{8}$$

where k is the discrete-time instant, N is the final discrete-time instant, $\boldsymbol{x} \triangleq [q_1 \ q_2 \ \dots \ q_n \ \dots \ \dot{q}_1 \ \dot{q}_2 \ \dots \ \dot{q}_n]^T$ denotes the state variable vector, and $\boldsymbol{x}_d$ is the desired reference vector. Note that the weighting matrix $W(k) = \text{diag}\{w_1(k), w_2(k), ..., w_{2n}(k)\}, k = 1, 2, ..., N$ are selected so that they relax the condition at initial stage while keeping higher weights at the latter part of the time frame. Figure 1 shows a sample chromosome that represents controller indices for two PSCs, where 0 represents the first elemental controller and 1 is for the second elemental controller.

In our past studies [19],[20], it has been shown that one can also select the fitness function (8) to fulfill the sub optimization goals such as energy consumption with introducing constraints [18].

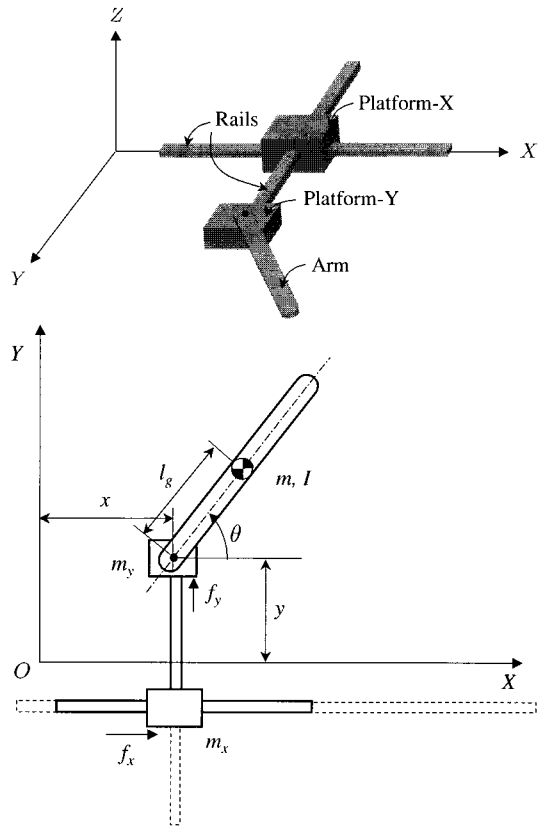


Fig. 2. 3D view of PPR robot system (upper) and its parameters (lower)

III. Three-DOF planar manipulator

In this section, we introduce three-DOF underactuated manipulator in order to illustrate the SCTM. A three-DOF PPR (i.e., prismatic–prismatic–revolutionary joints) planar manipulator shown in Fig. 2 is taken into consideration. Generalized coordinates and inputs of the robot manipulator are defined as $\boldsymbol{q}^T \triangleq [q_1 \;\; q_2 \;\; q_3] = [x \;\; y \;\; \theta]$ and $\boldsymbol{F}^T = [f_x \;\; f_y \;\; 0]$ respectively, in which $\boldsymbol{\mathcal{F}}^T = [f_x \;\; f_y]$. The equation of motion of the robot system is given by (1) and it has the following $M(\boldsymbol{q})$ and $\boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}})$:

$$M(\boldsymbol{q}) = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{12} & M_{22} & M_{23} \\ M_{13} & M_{23} & M_{33} \end{bmatrix} = \begin{bmatrix} m_x + m_y + m & 0 & -ml_g \sin\theta \\ 0 & m_y + m & ml_g \cos\theta \\ -ml_g \sin\theta & ml_g \cos\theta & ml_g^2 + I \end{bmatrix} \tag{9}$$

$$\boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} = \begin{bmatrix} -ml_g\dot{\theta}^2 \sin\theta + d_x\dot{x} \\ -ml_g\dot{\theta}^2 \sin\theta + d_y\dot{y} \\ d_\theta\dot{\theta} \end{bmatrix}. \tag{10}$$

This manipulator has the parameters and variables shown in Table 1.

Taking D as $D = \det(M)$ and $\hat{M}_{ij}$ as cofactors of M gives

$$\ddot{q}_1 = \frac{1}{D}\left\{\hat{M}_{11}(-h_1 + f_x) + \hat{M}_{12}(-h_2 + f_y) - \hat{M}_{13}h_3\right\}$$

$$\ddot{q}_2 = \frac{1}{D}\left\{\hat{M}_{12}(-h_1 + f_x) + \hat{M}_{22}(-h_2 + f_y) - \hat{M}_{23}h_3\right\}$$

$$\ddot{q}_3 = \frac{1}{D}\left\{\hat{M}_{13}(-h_1 + f_x) + \hat{M}_{23}(-h_2 + f_y) - \hat{M}_{33}h_3\right\} \tag{11}$$

where

$$\hat{M} \equiv \begin{bmatrix} \hat{M}_{11} & \hat{M}_{12} & \hat{M}_{13} \\ \hat{M}_{12} & \hat{M}_{22} & \hat{M}_{23} \\ \hat{M}_{13} & \hat{M}_{23} & \hat{M}_{33} \end{bmatrix}. \tag{12}$$

The values of $\hat{M}_{ij}$, $(i = 1, 2, 3;\ j = 1, 2, 3)$ are given below:

$$\hat{M}_{11} = M_{22}M_{33} - M_{23}^2, \quad \hat{M}_{12} = -(M_{12}M_{33} - M_{23}M_{13})$$

$$\hat{M}_{13} = M_{12}M_{23} - M_{13}M_{22}$$

$$\hat{M}_{21} = -(M_{12}M_{33} - M_{23}M_{13}), \quad \hat{M}_{22} = M_{11}M_{33} - M_{13}^2$$

$$\hat{M}_{23} = -(M_{11}M_{23} - M_{12}M_{13})$$

$$\hat{M}_{31} = M_{12}M_{23} - M_{13}M_{22}, \quad \hat{M}_{32} = -(M_{11}M_{23} - M_{12}M_{13})$$

$$\hat{M}_{33} = M_{11}M_{22} - M_{12}^2.$$

Since we have ${}_3C_2 = 3$ number of combinations of $m_u(= 2)$-dimensional controllers, three computed torque controllers can be brought into by two actuators. When considering the first control law for the sub-coordinates $\boldsymbol{s}^T = [q_1 \; q_2]$:

$$\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} = -\begin{bmatrix} \frac{\hat{M}_{11}}{D} & \frac{\hat{M}_{12}}{D} & \frac{\hat{M}_{13}}{D} \\ \frac{\hat{M}_{21}}{D} & \frac{\hat{M}_{22}}{D} & \frac{\hat{M}_{23}}{D} \end{bmatrix}\begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} + \begin{bmatrix} \frac{\hat{M}_{11}}{D} & \frac{\hat{M}_{12}}{D} \\ \frac{\hat{M}_{12}}{D} & \frac{\hat{M}_{22}}{D} \end{bmatrix}\begin{bmatrix} f_x \\ f_y \end{bmatrix} \tag{13}$$

we obtain

- **Control law 1:**

$$\begin{aligned} \begin{bmatrix} f_x \\ f_y \end{bmatrix} &= D\begin{bmatrix} \hat{M}_{11} & \hat{M}_{12} \\ \hat{M}_{12} & \hat{M}_{22} \end{bmatrix}^{-1} \times \left\{ \begin{bmatrix} \ddot{q}_1^* \\ \ddot{q}_2^* \end{bmatrix} + \frac{1}{D}\begin{bmatrix} \hat{M}_{11} & \hat{M}_{12} & \hat{M}_{13} \\ \hat{M}_{21} & \hat{M}_{22} & \hat{M}_{23} \end{bmatrix}\begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} \right\} \\ &= \frac{1}{D_A D}\begin{bmatrix} \hat{M}_{22} & -\hat{M}_{12} \\ -\hat{M}_{12} & \hat{M}_{11} \end{bmatrix} \times \left\{ \begin{bmatrix} \ddot{q}_1^* \\ \ddot{q}_2^* \end{bmatrix} + \frac{1}{D}\begin{bmatrix} \hat{M}_{11} & \hat{M}_{12} & \hat{M}_{13} \\ \hat{M}_{21} & \hat{M}_{22} & \hat{M}_{23} \end{bmatrix}\begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} \right\} \end{aligned} \tag{14}$$

with

$$\mathcal{M} = \frac{1}{D}\begin{bmatrix} \hat{M}_{11} & \hat{M}_{12} \\ \hat{M}_{12} & \hat{M}_{22} \end{bmatrix}, \; \boldsymbol{s}^* = \begin{bmatrix} \ddot{q}_1^* \\ \ddot{q}_2^* \end{bmatrix}, \quad \mathcal{N} = \frac{1}{D}\begin{bmatrix} \hat{M}_{11} & \hat{M}_{12} & \hat{M}_{13} \\ \hat{M}_{21} & \hat{M}_{22} & \hat{M}_{23} \end{bmatrix} \tag{15}$$

$$\mathcal{D}_A = (\hat{M}_{11}\hat{M}_{22} - \hat{M}_{12}^2)/D^2, \quad \hat{\mathcal{M}} = \frac{1}{D}\begin{bmatrix} \hat{M}_{22} & -\hat{M}_{12} \\ -\hat{M}_{12} & \hat{M}_{11} \end{bmatrix} \tag{16}$$

Table 1. Manipulator parameters and details

Parameter (or Variable)	Meaning	Value	Unit
m_x	Mass of platform X	0.5	[kg]
m_y	Mass of platform Y	0.8	[kg]
m	Mass of the arm	0.5	[kg]
l_g	Distance to m from m_y	0.2	[m]
I	Moment of inertia of arm	6.7×10^{-3}	[kgm^2]
x	Distance of m_x along X-axis	–	[m]
y	Distance from m_y along X-axis	–	[m]
θ	Angle from X-axis	–	[rad]
f_x	Force on platform X	–	[N]
f_y	Force on platform Y	–	[N]
d_x	Damping coefficient of X	0.001	[N·s/m]
d_y	Damping coefficient of Y	0.001	[N·s/m]
d_θ	Damping coefficient of arm	0.01	[Nm·s/rad]

where the modified acceleration vector $\ddot{s}^*$ is determined by a simple PD controller:

$$\begin{bmatrix} \ddot{q}_1^* \\ \ddot{q}_2^* \end{bmatrix} = \begin{bmatrix} \ddot{q}_{d1} + K_{v1}(\dot{q}_{d1} - \dot{q}_1) + K_{p1}(q_{d1} - q_1) \\ \ddot{q}_{d2} + K_{v2}(\dot{q}_{d2} - \dot{q}_2) + K_{p2}(q_{d2} - q_2) \end{bmatrix}. \quad (17)$$

Similarly, we can consider the second control law for the sub-coordinates $s^T = [q_2\ q_3]$:

$$\begin{bmatrix} \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} = -\begin{bmatrix} \frac{\hat{M}_{21}}{D} & \frac{\hat{M}_{22}}{D} & \frac{\hat{M}_{23}}{D} \\ \frac{\hat{M}_{31}}{D} & \frac{\hat{M}_{32}}{D} & \frac{\hat{M}_{33}}{D} \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} + \begin{bmatrix} \frac{\hat{M}_{12}}{D} & \frac{\hat{M}_{22}}{D} \\ \frac{\hat{M}_{13}}{D} & \frac{\hat{M}_{23}}{D} \end{bmatrix} \begin{bmatrix} f_x \\ f_y \end{bmatrix}. \quad (18)$$

Then, we have

- **Control law 2:**

$$\begin{bmatrix} f_x \\ f_y \end{bmatrix} = \frac{1}{\mathcal{D}_B}\hat{\mathcal{M}}\left\{ s^* + \mathcal{N}\begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} \right\} \quad (19)$$

with

$$\mathcal{D}_B = (\hat{M}_{12}\hat{M}_{23} - \hat{M}_{13}\hat{M}_{12})/D^2,$$
$$\hat{\mathcal{M}} = \frac{1}{D}\begin{bmatrix} \hat{M}_{23} & -\hat{M}_{22} \\ -\hat{M}_{13} & \hat{M}_{12} \end{bmatrix} \quad (20)$$

$$s^* = \begin{bmatrix} \ddot{q}_2^* \\ \ddot{q}_3^* \end{bmatrix},\ \mathcal{N} = \frac{1}{D}\begin{bmatrix} \hat{M}_{12} & \hat{M}_{22} & \hat{M}_{23} \\ \hat{M}_{13} & \hat{M}_{23} & \hat{M}_{33} \end{bmatrix} \quad (21)$$

where the modified acceleration vector $\ddot{s}^*$ is given by the following PD controller:

$$\begin{bmatrix} \ddot{q}_2^* \\ \ddot{q}_3^* \end{bmatrix} = \begin{bmatrix} \ddot{q}_{d2} + K_{v2}(\dot{q}_{d2} - \dot{q}_2) + K_{p2}(q_{d2} - q_2) \\ \ddot{q}_{d3} + K_{v3}(\dot{q}_{d3} - \dot{q}_3) + K_{p3}(q_{d3} - q_3) \end{bmatrix}. \quad (22)$$

We further consider the third control law for the sub-coordinates $s^T = [q_1\ q_3]$:

$$\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_3 \end{bmatrix} = -\begin{bmatrix} \frac{\hat{M}_{11}}{D} & \frac{\hat{M}_{12}}{D} & \frac{\hat{M}_{13}}{D} \\ \frac{\hat{M}_{13}}{D} & \frac{\hat{M}_{23}}{D} & \frac{\hat{M}_{33}}{D} \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} + \begin{bmatrix} \frac{\hat{M}_{11}}{D} & \frac{\hat{M}_{12}}{D} \\ \frac{\hat{M}_{13}}{D} & \frac{\hat{M}_{23}}{D} \end{bmatrix} \begin{bmatrix} f_x \\ f_y \end{bmatrix}. \quad (23)$$

Then, we obtain

- **Control law 3:**

$$\begin{bmatrix} f_x \\ f_y \end{bmatrix} = \frac{1}{\mathcal{D}_C}\hat{\mathcal{M}}\left\{ s^* + \mathcal{N}\begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} \right\} \quad (24)$$

with

$$\mathcal{D}_C = (\hat{M}_{11}\hat{M}_{23} - \hat{M}_{12}\hat{M}_{13})/D^2,$$
$$\hat{\mathcal{M}} = \frac{1}{D}\begin{bmatrix} \hat{M}_{23} & -\hat{M}_{12} \\ -\hat{M}_{13} & \hat{M}_{11} \end{bmatrix} \quad (25)$$

$$s^* = \begin{bmatrix} \ddot{q}_1^* \\ \ddot{q}_3^* \end{bmatrix},\ \mathcal{N} = \frac{1}{D}\begin{bmatrix} \hat{M}_{11} & \hat{M}_{12} & \hat{M}_{13} \\ \hat{M}_{13} & \hat{M}_{23} & \hat{M}_{33} \end{bmatrix} \quad (26)$$

where the modified acceleration vector $\ddot{s}^*$ is given by the following PD controller:

$$\begin{bmatrix} \ddot{q}_1^* \\ \ddot{q}_3^* \end{bmatrix} = \begin{bmatrix} \ddot{q}_{d1} + K_{v1}(\dot{q}_{d1} - \dot{q}_1) + K_{p1}(q_{d1} - q_1) \\ \ddot{q}_{d3} + K_{v3}(\dot{q}_{d3} - \dot{q}_3) + K_{p3}(q_{d3} - q_3) \end{bmatrix}. \quad (27)$$

Applying the control laws 1, 2 and 3 to the robot manipulator separately, three graphs can be obtained as shown in Fig. 3, respectively. Here, we selected K_p, K_v values such as $K_{p1} = 8.0$, $K_{p2} = 8.0$, $K_{p3} = 15.0$ and $K_{v1} = 4.0$, $K_{v2} = 4.0$, $K_{v3} = 1.0$. Note that the selected values are arbtrary and the designer can select them according to his desire, making sure to keep low gains when underactuated joints are present. For an example, we can obtain a critically damped closed-system by using $K_v = 2\sqrt{K_p}$. This figure shows that each elemental controller can control the associated sub-coordinates, though the total coordinates can not be controlled by using only an elemental controller.

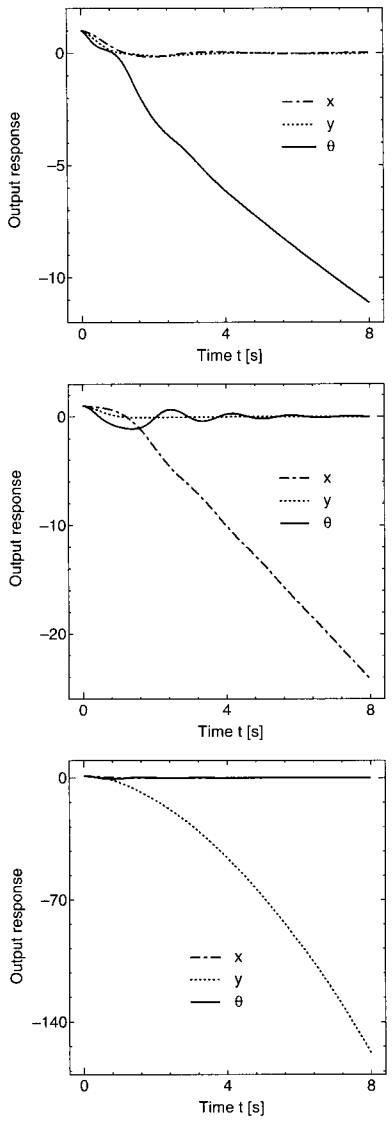


Fig. 3. Time responses of x, y, and θ using each PSC independently: (top) control law 1, (middle) control law 2, and (bottom) control law 3

IV. Obstacle avoidance

In this section, we develop necessary conditions for controlling the above-explained manipulator with avoiding obstacles. The obstacle avoidance problem of manipulator has been formulated in terms of collision avoidance of links rather than points. Link collision avoidance is achieved by continuously controlling the link's closest point to the obstacle. For the collision detection/avoidance using the SCTM, the links of a robot manipulator can be modeled by introducing enclosing ellipsoids [16]. However, in this study, we use an elliptical model rather than ellipsoids because a PPR planar is considered for the simulation. One of the three links for the robot manipulator can be represented and modeled by a 2D ellipse as shown in Fig. 4. Here, lengths of the major and minor axes of the ellipse are denoted by a and b respectively. These values are calculated from the following relationship:

$$a = \hat{a} + \alpha\hat{a}, \quad b = \hat{b} + \beta\hat{b} \tag{28}$$

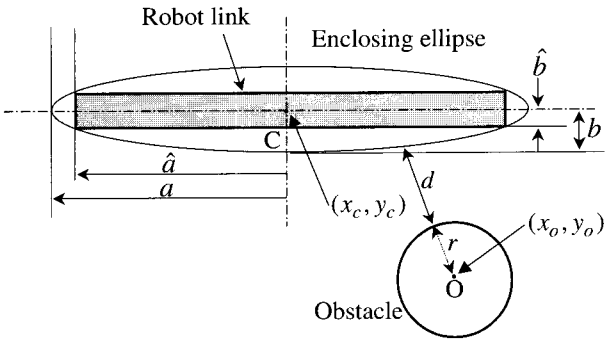


Fig. 4. Elliptical model for modeling the robot links

where $\hat{a}$ and $\hat{b}$ are taken from the manipulator dimensions as shown in Fig. 4 while α and β can be selected by trial and error. Suppose that the modeled ellipse of a selected link has a moving center at $(x_c,\ y_c)$ in $X - Y$ coordinate system. Then the elliptical model can be formulated as below:

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0 \tag{29}$$

where

$$\begin{aligned}
A &= c^2/a^2 + s^2/b^2 \\
B &= 2cs/a^2 - 2cs/b^2 \\
C &= c^2/b^2 + s^2/a^2 \\
D &= -2(x_c c^2 + y_c sc)/a^2 - 2(x_c s^2 - y_c sc)/b^2 \\
E &= -2(y_c s^2 + x_c sc)/a^2 - 2(y_c c^2 - x_c sc)/b^2 \\
F &= ({x_c}^2 c^2 + {y_c}^2 s^2 + 2x_c y_c sc)/a^2 \\
&\quad +({y_c}^2 c^2 + {x_c}^2 s^2 - 2x_c y_c sc)/b^2 - 1
\end{aligned}$$

in which $s = \sin\theta$ and $c = \cos\theta$, where θ is the angle of the ellipse measured counterclockwise from the $X-$axis in $X - Y$ coordinate system.

Now, we formulate the obstacle avoidance problem by using SCTM. In this case, an additional penalty function is brought into the fitness function (8) as below:

$$fitness = \zeta \sum_{k=1}^{N} \mathcal{P}_{Total}(k) + \sum_{k=1}^{N} \|\boldsymbol{x}_d(k) - \boldsymbol{x}(k)\|^2_{W(k)} \tag{30}$$

where ζ is a constant depending on the problem. When the vaule ζ is at low, the effect on the total penalty will be very small. This means the that there is a less effect on avoiding obtacles and still has the capability of controlling the system as in (8). The penalty component $\mathcal{P}_{Total}(k) = \sum_{i=1}^{n} \mathcal{P}_i(k)$. $\mathcal{P}_i(k)$ is given by:

$$\mathcal{P}_i(k) = \begin{cases} 1 & \text{if } f_i(x_o, y_o) < 0 \\ 0 & \text{otherwise} \end{cases} \tag{31}$$

where $f_i(x_o, y_o) = A_i {x_o}^2 + B_i x_o y_o + C_i {y_o}^2 + D_i x_o + E_i y_o + F_i$ and f_i is calculated from the parameters and the current position of ith link ($i = 1, 2, 3$) at a given k. $(x_o,\ y_o)$ is the center of the circular obstacle with radius r. Note that the virtual outer peripheral of the elliptical model is selected such that $a = \hat{a} + \alpha\hat{a} + d$ and $b = \hat{b} + \beta\hat{b} + d$, in order to make sure the value $d > 0$ always.

Table 2. GA parameters used for the simulation

Parameter	Value
Population size	100
Crossover rate	0.6 (uniform)
Bit size	1000
Mutation rate	1/(bit size)
Selection criteria	Best 10

V. Results

1. Case 1

We applied the concept explained in Section 2 to control the three-DOF PPR robot with the obstacle avoidance methodology introduced in Section 4. The three-DOF robot shown in Fig. 2 was taken into consideration, starting from $\boldsymbol{x}(0) = [0.5 \quad 0.5 \quad 1 \quad 0 \quad 0 \quad 0]^T$ towards the desired value $\boldsymbol{x}_d = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T$. A circular obstacle having a radius of $r = 0.08$ [m] is placed, keeping its center at $(0.25, 0.25)$. The K_p and K_v values are kept as they are in Section 3 and set ζ and d values to $\zeta = 10,000$ and $d = 0.1$.

The total time frame t_N was set to 32 [s] with 0.01 [s] sampling. Table 2 shows the GA parameters used in the optimization process.

We also applied the following elements $w_i(k)$ of the $W(k) = \text{diag}\{w_1(k), \ldots, w_6(k)\}$, $k = 1, 2, \ldots, N$ for the fitness function (30):

$$w_i(k) = \begin{cases} w_i(k) & \text{if } 0 \le k < N/2 \\ w_i{}^*(k) & \text{otherwise} \end{cases} \tag{32}$$

$$\begin{aligned} &w_1 = 100, \quad w_2 = w_1, \quad w_3 = w_1, \quad w_4 = 1, \\ &w_5 = w_4, \quad w_6 = w_4 \\ &w_1^* = 10000, \quad w_2^* = w_1^*, \quad w_3^* = w_1^*, \quad w_4^* = 1000, \\ &w_5^* = w_4^*, \quad w_6^* = w_4^*. \end{aligned}$$

Figure 5 shows the simulation results for the time responses of controlled variables x, y, and θ.

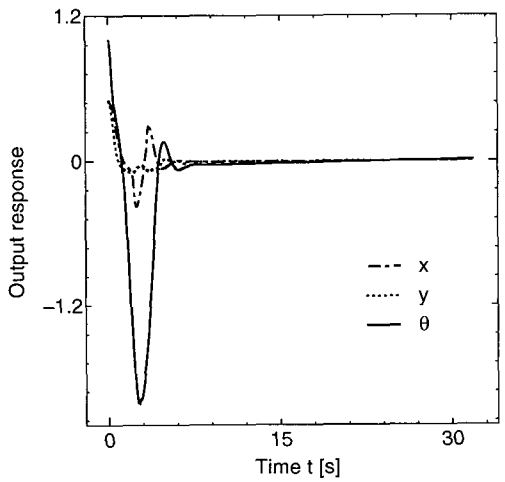


Fig. 5. Time response of controlled variables x, y, and θ

Moreover, the time responses of $\dot{x}, \dot{y}$ and $\dot{\theta}$ are given by Fig. 6 and the path of the evolving elliptical model and the tip of the robot manipulator are given by Fig. 7.

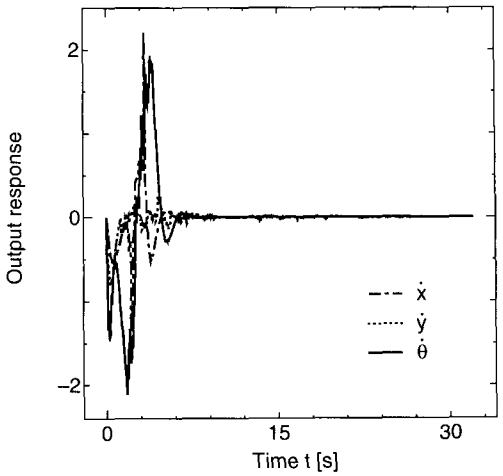


Fig. 6. Time response of controlled variables $\dot{x}, \dot{y}$, and $\dot{\theta}$

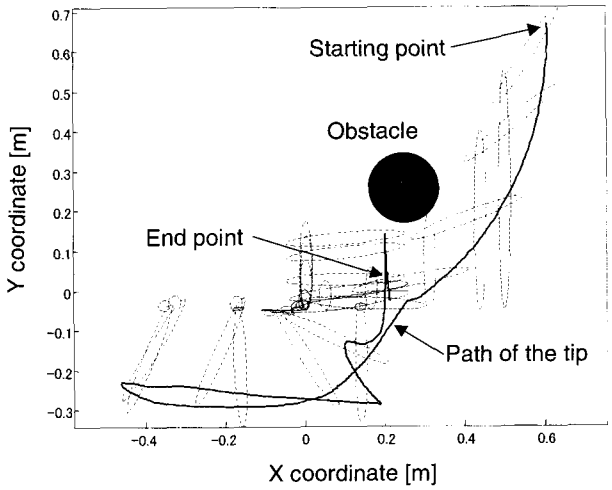


Fig. 7. Path of the evolving elliptical model and the tip of the robot manipulator

Furthermore, Fig. 8 shows the corresponding best fitness values of each generation. In this simulation, we set the GA stopping condition as 350 iterations. If the designer wants to stop the GA evolution at earlier stage it would be better to set the termination condition considering the total error or reqired convergance accuracy of the desired variables.

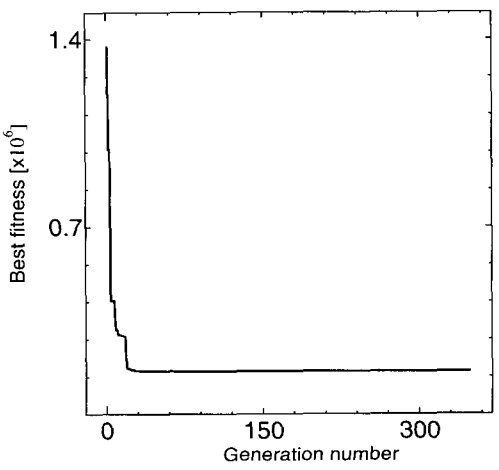


Fig. 8. Evolutionary history

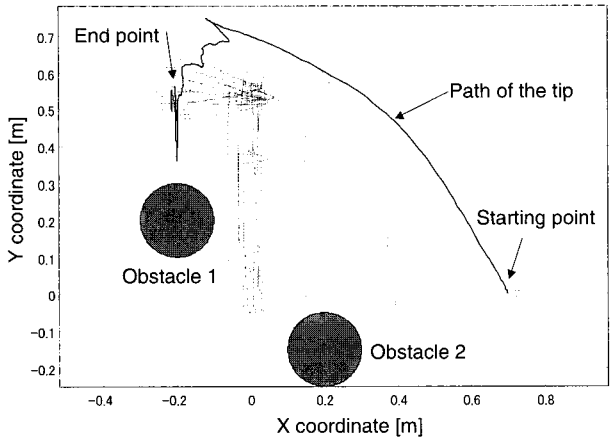


Fig. 9. Path of the evolving elliptical model and the tip of the robot manipulator for two obstacle case

It is found frpm Fig. 5 and Fig. 6 that the system state variables have converged to the desired values. This ensures that the controlled variables are met the desired values within an acceptable level of time frame with avoiding the obstacle. Moreover, the optimum path can be seen in Fig. 7 confirming the evolutionary history of the GA process in Fig. 8.

2. Case 2

Further simulation was carried out for a two obtacle case in order to illustrate the proposed method. Two obstacles were placed in the $X-Y$ plane avoding the movements towards the negative qudrants (see Fig. 9). We tried to dive the manipulator, starting from $\boldsymbol{x}(0) = [0.5\ 0\ 0\ 0\ 0\ 0]^T$ towards the desired value $\boldsymbol{x}_d = [0\ 0.5\ \pi\ 0\ 0\ 0]^T$. In the optimization process, we can add another $\mathcal{P}_{Total}(k)$ term for the fitness function when it introduces the second obtacle.

In this case, the weights were changed in order to illustrate the flexibility of selecting the desired weights in a different manner such that they relax the condition at initial stage while keeping higher weights at the latter part of the time frame:

$$
\begin{aligned}
&w_1 = 10, \quad w_2 = w_1, \quad w_3 = w_1, \quad w_4 = 1,\\
&w_5 = w_4, \quad w_6 = w_4\\
&w_1^* = 10000, \quad w_2^* = w_1^*, \quad w_3^* = w_1^*, \quad w_4^* = 1000,\\
&w_5^* = w_4^*, \quad w_6^* = w_4^*.
\end{aligned}
$$

All the other parameters were set as explained in Case 1. There were no restrictions in Case 1for moving the manipulator on $X-Y$ plane. So that, we could observe the additional movements in the negative quadrants in Case 1, whereas such movements were eliminated in Case 2.

VI. Conclusions and future works

This paper has aimed at the investigation of controlling underactuated robot manipulators with avoiding obstacles. The three-DOF PPR robot, working on a horizontal plane, was taken into consideration so as to illustrate the present method. PSCs were derived using the original dynamic model of the manipulator so that they are employed under a proper switching sequence to achieve the global convergence and stability of the total control system. The optimum switching sequence of PSCs was determined by evolutionary computation, i.e., a simple GA. The results demonstrated the effectiveness of the proposed algorithm. Thus, the proposed control methodology is useful for controlling underactuated robot manipulators with avoiding obstacles.

One of the major advantages of this method is that entire system can be controlled without using rigorous linearizations or deformation of the original nonlinear system and employing simple computed torque controllers as PSCs in the control system. In our past studies, it has been shown that the consumption of energy of robot system and chattering can also be optimized by introducing additional constraints to GA process. Apart from these two cases, it is interesting to see the control performance for various initial conditions, different types of obstacles, and changing the values of fitness fuction parameters (such as ζ and $W(k)$).

Authors are currently investigating a fuzzy rule extraction for selecting PSCs for online operations of underactuated manipulators. Under this, the fuzzy rule base of switching the PSCs is optimized using GA and the optimization is performed off-line. Design parameters of the fuzzy rules are encoded into chromosomes and shapes of the Gaussian functions are evolved to minimize the angular position errors. The angular position errors are used as the inputs to the Gaussian membership functions in the antecedent part and the one index of the PSCs is assigned in the consequent part of the fuzzy reasoning. Then, this trained rule base can be brought into the online operation of the underactuated manipulator. Training for different initial configurations, a robust fuzzy rule base can be extracted for online operations. Details will be available in future publications.

References

[1] I. Kolmanovsky and N. H. McClamroch, "Development in nonholonomic control problems," *IEEE Control Systems Magazine*, vol. 15, no. 6, pp. 20–36, 1995.

[2] H. Arai, K. Tanie, and N. Shiroma, "Nonholonomic control of a three dof planar underactuated manipulator," *IEEE Trans. on Robotics and Automation*, vol. 14, no. 5 pp. 681–695, 1998.

[3] A. D. Luca, S. Iannitti, and G. Oriolo, "Stabilization of a PR planar underactuated robot," in *Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA'2001)*, 2001, pp. 2090–2095.

[4] M. Reyhanoglu, A. V. D. Schaft, N. H. McClamroch, and I. Kolmanovsky, "Dynamics and control of a class of underactuated mechanical systems," *IEEE Trans. on Automatic Control*, vol. 44, no. 9, pp. 1663–1671, 1999.

[5] L. Udawatta, K. Watanabe, K. Kiguchi, and K. Izumi, "Developments in underactuated manipulator control techniques and latest control using AI," in *Proc. of 7th International Symposium on Artificial Life and Robotics, Oita, Japan*, vol. 2, 2002, pp. 425–428.

[6] F. Bullo, N. E. Leonard, and A. D. Lewis, "Controllability and motion algorithms for underactuated Lagrangian systems on Lie group," *IEEE Trans. on Automatic Control*, vol. 45, no. 8 pp. 1437–1453, 2000.

[7] T. Mita and T. K. Nam, "Control of underactuated manipulators using variable period deadbeat control," in *Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA'2001)*, 2001, pp. 2735–2740.

[8] M. Bergerman and Y. Xu, "Optimal control sequence for underactuated manipulators," in *Proc. of IEEE Int.*

Conf. on Robotics and Automation (ICRA'1996), 1996, pp. 3714–3719.

[9] M. D. Berkemeier and R. S. Fearing, "Tracking fast inverted trajectories of the underactuated acrobot," *IEEE Trans. on Robotics and Automation*, vol. 15, no. 4 pp. 740–750, 1999.

[10] Y. Nakamura, T. Suzuki, and M. Koinuma, "Nonlinear behavior and control of a nonholonomic free-joint manipulator," *IEEE Trans. on Robotics and Automation*, vol. 13, no. 6, pp. 853–862, 1997

[11] N. Shiroma, K. M. Lynch, H. Arai, and K. Tanie, "Motion planning for a three-axis planar manipulator with a passive revolute joint," *Trans. on Japan Society of Mechanical Engineers*, vol. 66Cno. 642Cpp. 545–552, 2000.

[12] K. Watanabe, L. Udawatta, K. Kiguchi, and K. Izumi, "Control of chaotic systems using fuzzy model-based regulators," in *Lecture Notes in Artificial Intelligence*, Berlin: Springer-Verlag, 1999, pp. 248–256.

[13] L. Udawatta, K. Watanabe, K. Kiguchi, and K. Izumi, "Fuzzy-chaos hybrid controller for controlling of nonlinear systems," *IEEE Trans. on Fuzzy Systems*, vol. 10Cno. 3Cpp. 401–411, 2002.

[14] G. Muscato, "Fuzzy control of an underactuated robot with a fuzzy microcontroller," *Journal of Microprocessors and Microsystems*, vol. 23, pp. 385–391, 1999.

[15] K. Kobayashi, J. Imura, and T. Yoshikawa, "Nonholonomic control of 3-D.O.F. manipulator with a free joint," *Trans. on SICE*, vol. 33, no. 8, pp. 799–804, 1997.

[16] S. P. Shiang, J. S. Liu, and Y. R. Chien, "Estimate of minimum distance between convex polyhedra based on enclosed ellipsoids," in *Proc. of the 2000 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS 2000)*, Takamatsu, Japan, 2000, pp. 739–744.

[17] W. L. Xu and B. L. Ma, "Stabilization of second-order nonholonomic systems in canonical chained form," *Robotics and Autonomous Systems*, no. 34, pp. 223–233, 2001.

[18] J. Kim and H. Myung, "Evolutionary programming techniques for constrained optimization problems," *IEEE Trans. on Evolutionary Computation*, vol. 1, no. 2, pp. 129–140, 1997.

[19] L. Udawatta, K. Watanabe, K. Izumi, and K. Kiguchi, "Energy optimization of switching torque computed method for underactuated manipulators," in *Proc. of Intelligent Autonomous Vehicle, IAV 2001*, Sapporo, Japan, 2001, pp. 394–399.

[20] L. Udawatta, K. Watanabe, K. Izumi, and K. Kiguchi, "Obstacle avoidance of underactuated robot manipulators using switching computed torque method," in *Proc. of the International Conference on Control, Automation and Systems, ICCAS*, Korea, pp. 321–324, 2001.

Lanka Udawatta

Lanka Udawatta was born in Sri Lanka in 1970. He received the B.Sc. degree in Electrical Engineering from the University of Moratuwa, Sri Lanka in 1996 and M.Eng. degree in Mechanical Engineering from the Saga University, Japan in 2000. Lanka is currently pursuing the doctoral degree at the Department of Advanced Systems Control, Saga University.

His research interests include control of nonlinear systems using genetic algorithms and fuzzy reasoning, especially on controlling nonholonomic systems such as underactuated robot manipulators. Lanka Udawatta is a recipient of the best student paper award for 6th International Conference on Soft Computing and Intelligent Systems Iizuka-2000 in 2000 and he is a student member of IEEE.

Kiyotaka Izumi

Kiyotaka Izumi received the B.E. degree in electrical engineering from Nagasaki Institute of Applied Science, Nagasaki, Japan, in 1991 and the M.E. degree in electrical engineering and the D.E. degree in engineering systems and technology from Saga University, Saga, Japan, in 1993 and 1996, respectively.

From April 1996 to March 2001, he was a Research Associate in the Department of Mechanical Engineering, Saga University. Since April 2001, he has been with the Department of Advanced Systems Control Engineering, Graduate School of Science and Engineering, Saga University. His research interests are in robust control, fuzzy control, behavior-based control, genetic algorithms, evolutionary strategy and their applications to robot control.

Dr. Izumi is a Member of the Society of Instrument and Control Engineers, Japan Society of Mechanical Engineers, Robotics Society of Japan, Japan Society for Fuzzy Theory and Systems, Institute of Electronics, Information and Communication Engineers, Japan Society for Precision Engineering, and IEEE.

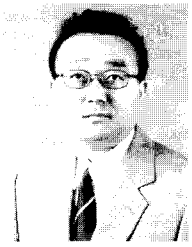
Keigo Watanabe

Keigo Watanabe received the B.E. and M.E. degrees in mechanical engineering from the University of Tokushima, Tokushima, Japan, in 1976 and 1978, respectively, and the D.E. degree in aeronautical engineering from Kyushu University, Fukuoka, Japan, in 1984.

From 1980 to March 1985, he was a Research Associate at Kyushu University. From April 1985 to March 1990, he was an Associate Professor at the College of Engineering, Shizuoka University, Shizuoka, Japan. From April 1990 to March 1993, he was an Associate Professor, and from April 1993 to March 1998, he was a Full Professor in the Department of Mechanical Engineering, Saga University, Saga, Japan. Since April 1998, he has been with the Department of Advanced Systems Control Engineering, Graduate School of Science and Engineering, Saga University. He has published more than 350 technical papers in transactions, journals, and international conference proceedings, and is the author or editor of 18 books, including Adaptive Estimation and Control (Englewood Cliffs: NJ: Prentice-Hall, 1991), Stochastic Large-Scale Engineering Systems (New York: Marcel Dekker, 1992) and Intelligent Control Based on Flexible Neural Networks (Norwell, MA: Kluwer, 1999).

He is an Active Reviewer of many journals and transactions, and an Editor-in-Chief of Machine Intelligence and Robotic Control Journal, and an editorial board members of the Journal of Intelligent and Robotic Systems and the Journal of Knowledge-Based Intelligent Engineering Systems. His research interests are in stochastic adaptive estimation and control, robust control, neural network control, fuzzy control, and genetic algorithms and their applications to machine intelligence and robotic control.

Dr. Watanabe is a Member of the Society of Instrument and Control Engineers, Japan Society of Mechanical Engineers, Japan Society for Precision En-gineering, Institute of Systems, Control and Information Engineers, the Japan Society for Aeronautical and Space Sciences, Robotics Society of Japan, Japan Society for Fuzzy Theory and Systems, and IEEE.



Kazuo Kiguchi

Kazuo Kiguchi received the Bachelor of Engineering degree in mechanical engineering from Niigata University, Japan in 1986, the Master of Applied Science degree in mechanical engineering from the University of Ottawa, Canada in 1993, and the Doctor of Engineering degree from Nagoya University, Japan in 1997. From 1986 to 1989, he was a Research Engineer with Mazda Motor Company. From 1989 to 1991, he was a Research Engineer with MHI Aerospace Systems Company. From 1994 to 1999, he worked for the Department of Industrial and Systems Engineering, Niigata College of Technology, Niigata, Japan. He is currently an Associate Professor in the Dept. of Advanced Systems Control Engineering, Graduate School of Science and Engineering, Saga University, Japan. Dr. Kiguchi received the J.F. Engelberger Best Paper Award at WAC2000. His research interests include biorobotics, intelligent robots, machine learning, application of soft computing for robot control, and application of robotics in medicine.

Dr. Kiguchi is a member of the Robotics Society of Japan, IEEE (SMC, R&A, EMB, IE, and CS Societies), the Japan Society of Mechanical Engineers, the Society of Instrument and Control Engineers, the Japan Society of Computer Aided Surgery, International Neural Network Society, Japan Neuroscience Society, the Virtual Reality Society of Japan, and the Japanese Society for Clinical Biomechanics and Related Research.