SLIDING MODE CONTROL FOR A PLANAR PARALLEL ROBOT DRIVEN BY ELECTRIC MOTORS IN A TASK SPACE

NGUYEN QUANG HOANG[1], VU DUC VUONG[1,2]

[1]*School of Mechanical Engineering, Hanoi University of Science and Technology*
[2]*Faculty of Electronics Engineering, Thai Nguyen University of Technology*
[1]*hoang.nguyenquang@hust.edu.vn*

Abstract. This paper addresses the modeling of a planar parallel robot including electric motors. The dynamic model of the system is derived by applying the substructure method and Lagrangian equations with multipliers in the form of redundant generalized coordinates. These equations are then transformed to the form of minimal coordinates of operational variables. Based on this form, a sliding mode controller (SMC) is designed for trajectory tracking in a task space. Numerical simulations in MATLAB are carried out based on the 3RRR parallel robot in order to show the effectiveness of the proposal approach. The obtained results show a good behavior of the proposed task space tracking controller.

Keywords. parallel robot, electromechanical system, dynamic modeling, sliding mode control.

1. INTRODUCTION

Nowadays, parallel robotic manipulators are used widely in industrial applications due to their advantages such as high stiffness, accuracy, and light links. This kind of robots has attracted numerous researchers. Many of them deal with the modeling of parallel manipulators [4, 8]. In many work such as [1, 2] the parallel manipulators are modeled by a closed loop rigid multibody system driven directly by torques/forces. To establish the dynamic equations of a parallel robot, there are some approaches proposed in literature including Lagrangian formulation with multipliers, Newton-Euler equations, the principle of virtual work, the Jourdain principle, and Kane equation such as in [2, 3, 6-8, 13, 14]. In general, manipulators are driven by electric motors with gear transmission, therefore the dynamic of actuators should be considered. The modeling of electromechanical systems of a parallel robotic manipulator has been examined by only some researchers [19-21].

Besides modeling, the control design of parallel manipulators is also an important problem. Similarly to serial manipulators, many control laws such as PD plus gravity compensation, exact feedback linearization, sliding mode, adaptive, fuzzy and neural network are also exploited to design controllers for parallel manipulators. Several authors have designed controllers for the robot in active joint coordinates [9-11, 17, 18]. This design method required solving the inverse kinematic problem.

In this study, the substructure method is applied in the modeling of a parallel manipulator driven by electric motors. The whole system is divided into three parts: actuators,

massless gear transmissions, and manipulator as a closed loop multiboby system. The dynamic equations of electric actuators are derived, mainly based on the angular momentum law combined with Kirchoff's laws. The equation describing a massless gear transmission is obtained by the power balance at the input and output of the gearbox. For the closed loop multiboby system, the redundant generalized coordinates and the Lagrangian formulation with multipliers are used to establish the equations of motion [1, 2, 4]. Being combined with the constraint equations, the dynamic model of a parallel robot is presented in a form of differential-algebraic equations (DAEs). The obtained equations are then simplified with the assumption that an electric time constant is much smaller than a mechanical time constant. Moreover, to design the controller for the robot the simplified equations in a DAEs form are then converted into the form of minimal coordinates of operational variables.

The rest of this paper is organized as follows: Section 2 presents a dynamical modeling of parallel manipulator actuated by electric motors and the transformation of the differential equations into the minimal form in space of operational coordinates. The controller design in task space is described in Section 3. Some numerical simulations are shown in Section 4. Finally, the conclusion is given in Section 5.

2. DYNAMICAL MODEL

2.1. Equations of motion of a parallel robot driven by electric motors

Let us consider the parallel robot having n degrees of freedom driven by n electric motors as shown in Fig. 1. For governing the equations of motion, the symbols and notations in Table 1 will be used.
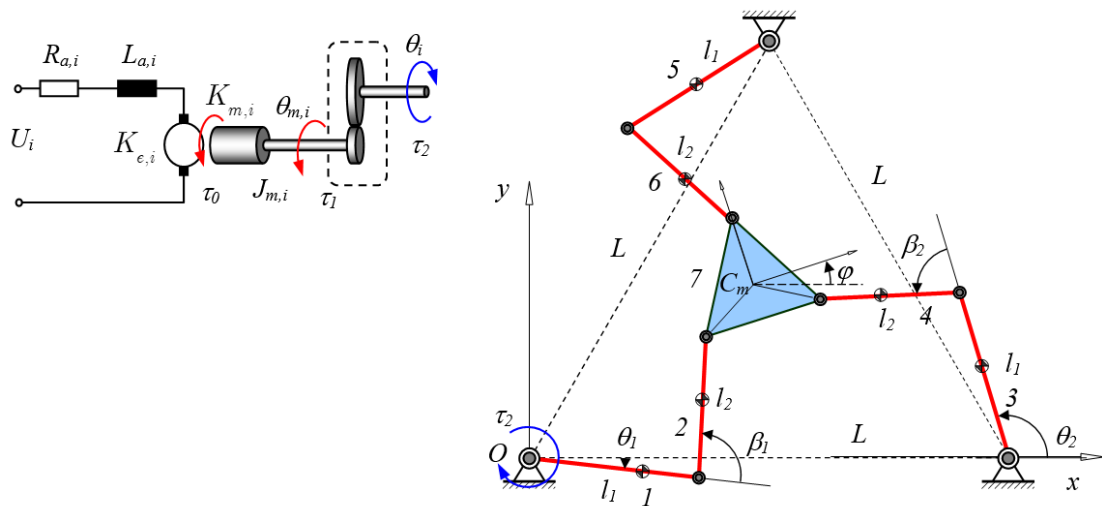

Figure 1. Model of an electric motor with gearbox and a 3RRR parallel robot

The system dynamic model is derived by applying Lagrangian equation with multipliers and the substructure method. The three mentioned substructures are described by the following equations: Dynamics of actuators is described by the mechanical and electric equations as [15, 16].

Table 1. The generalized coordinates and parameters of the system

Order	Notations	Name
1	$\boldsymbol{\theta} = [\theta_1, \theta_2, ..., \theta_n]^T =: \boldsymbol{q_a}$	Active joint variables
2	$\boldsymbol{\beta} = [\beta_1, \beta_2, ..., \beta_l]^T$	Auxiliary variables (unactuated joint)
3	$\boldsymbol{x} = [x_1, x_2, ..., x_n]^T$	Position of moving platform
4	$\boldsymbol{p} = [\boldsymbol{\beta}^T, \boldsymbol{x}^T]^T$	Combination of auxiliary and moving platform variables
5	$\boldsymbol{q} = [\boldsymbol{\theta}^T, \boldsymbol{\beta}^T, \boldsymbol{x}^T]^T$	Redundant coordinates
6	$\boldsymbol{\lambda} = [\lambda_1, \lambda_2, ..., \lambda_r]^T$	Lagrangian multipliers
7	$r = r_i = \theta_{m,i}/\theta_i$	Gear reduction ratio
8	$\boldsymbol{\theta}_m = [\theta_{m1}, \theta_{m2}, ..., \theta_{mn}]^T$	Angle of the motor shaft
9	$\boldsymbol{\tau}_2 = [\tau_{2,1}, \tau_{2,2}, ..., \tau_{2,n}]^T$	Torque/force at the output of transmission
10	$\boldsymbol{\tau}_1 = [\tau_{1,1}, \tau_{1,2}, ..., \tau_{1,n}]^T$	Torque/force at the input of transmission
11	$\boldsymbol{\tau}_0 = [\tau_{0,1}, \tau_{0,2}, ..., \tau_{0,n}]^T$	Torque/force of the motor
12	$\boldsymbol{J}_m = \operatorname{diag}(J_{m,1}, J_{m,2}, ..., J_{m,n})$	Moment of inertia of rotors
13	$\boldsymbol{L}_a = \operatorname{diag}(L_{a,1}, L_{a,2}, ..., L_{a,n})$	Motor coil inductances
14	$\boldsymbol{R}_a = \operatorname{diag}(R_{a,1}, R_{a,2}, ..., R_{a,n})$	Motor coil resistances
15	$\boldsymbol{K}_e = \operatorname{diag}(K_{e,1}, K_{e,2}, ..., K_{e,n})$	Back-emf constants
16	$\boldsymbol{K}_m = \operatorname{diag}(K_{m,1}, K_{m,2}, ..., K_{m,n})$	Torque constants
17	$\boldsymbol{u} = [U_1, U_2, ..., U_n]^T$	Motor input voltages
18	$\boldsymbol{i} = [i_1, i_2, ..., i_n]^T$	Current in motors
19	$\boldsymbol{D}_m = \operatorname{diag}(b_1, b_2, ..., b_n)$	Viscous friction coefficients of motor shafts

$$\boldsymbol{J}_m \ddot{\boldsymbol{\theta}}_m + \boldsymbol{D}_m \dot{\boldsymbol{\theta}}_m = \boldsymbol{\tau}_0 - \boldsymbol{\tau}_1. \tag{1}$$

$$\boldsymbol{L}_a \frac{d\boldsymbol{i}}{dt} + \boldsymbol{R}_a \boldsymbol{i} = \boldsymbol{u} - \boldsymbol{u}_e. \tag{2}$$

The mechanical and electric interaction in DC motors is shown by the relationship between current and motor torque; between motor speed and EMF's voltages as follows

$$\boldsymbol{\tau}_0 = \boldsymbol{K}_m \boldsymbol{i}, \ \boldsymbol{u}_e = \boldsymbol{K}_e \dot{\boldsymbol{\theta}}_m. \tag{3}$$

For the gear transmission, by neglecting the mass and power loss, the constraint equations are written as follows

$$r\dot{\boldsymbol{q}}_a = r\dot{\boldsymbol{\theta}} = \dot{\boldsymbol{\theta}}_m, \tag{4}$$

$$\boldsymbol{\tau}_2 = r\boldsymbol{\tau}_1. \tag{5}$$

For a closed loop multibody system the redundant generalized coordinate will be used and the dynamic equation is described in form of DAEs as follows

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) + \boldsymbol{D}\dot{\boldsymbol{q}} + \boldsymbol{\Phi}^T(\boldsymbol{q})\boldsymbol{\lambda} = \boldsymbol{B}\boldsymbol{\tau}_2, \tag{6}$$

$$\phi(q) = 0, \tag{7}$$

where $M(q)$ is the mass matrix; $g(q)$ the potential generalized force vector; $D\dot{q}$ is the vector of generalized forces of non-conservative forces (linear damping force); $\phi(q) = 0$ is the vector of constraint equations; $\Phi(q) = \partial\phi/\partial q$ is the Jacobian matrix; $B = [E_{n\times n}, \mathbf{0}_{m-n,n}^T]^T$ is the matrix related to the control input arrangement; the coriolis and centrifugal matrix $C(q, \dot{q})$ can be determined from the mass matrix by using the Christoffel formula [2].

Normally, the electric time constant is much smaller than mechanical time constant, so the approximation $L_a d\mathbf{i}/dt \approx 0$ when $t \geq \varepsilon > 0$ can be used to simplify the system of differential equations describing the system. After some calculations one gets

$$\begin{aligned} \left(M(q) + BJ_m r^2 Z\right) \ddot{q} + C(q, \dot{q})\dot{q} + g(q) \\ + \left(D + B(D_m + K_m R_a^{-1} K_e)r^2 Z\right) \dot{q} = BK_m R_a^{-1} r u + \Phi_q^T(q)\lambda \end{aligned} \tag{8}$$

in which the matrix $Z = [E_{n\times n}, \mathbf{0}_{n,m-n}]$ has been used. By defining the following matrices

$$\begin{aligned} B_s(q) &= \left(M(q) + BJ_m r^2 Z\right), \quad C_s(q, \dot{q}) = C(q, \dot{q}) \\ D_s &= \left(D + B(D_m + K_m R_a^{-1} K_e)r^2 Z\right) \\ g_s(q) &= g(q), \quad B_s = BK_m R_a^{-1} r \end{aligned}$$

the equation (8) is rewritten in a compact form as

$$M_s(q)\ddot{q} + C_s(q, \dot{q})\dot{q} + D_s\dot{q} + g_s(q) = B_s(q)u + \Phi_q^T \lambda. \tag{9}$$

Noting that $BJ_m r^2 Z$ is a constant matrix, so the Coriolis matrices $C_s(q, \dot{q})$ or $C(q, \dot{q})$ calculating from mass matrices $M_s(q)$ or $M(q)$ are the same, and the skew-symmetric property of matrix $N = \dot{M}_s(q) - 2C_s(q, \dot{q})$ is remained.

Thus, the dynamic model of a planar parallel robot driven by electric motors is described by a set of differential algebraic equations (9) and (7). These equations show the dynamic relationship between input (voltage u) and output (motion $q(t)$ including the motion of moving platform $x(t)$). These will be used for the inverse and forward dynamic problems of parallel robots.

2.2. Equations of motion in a minimal coordinate form in a task space

The equations of motion in a redundant generalized coordinates form are convenient for deriving it as well as simulation. To design a controller in a task space, the equations need to be transformed to minimal coordinates form of operational variable x with a number of equations equal to the degrees of freedom. By differentiating (7) with respect to time (w.r.t.) one gets

$$\Phi_q(q)\dot{q} = J_y\dot{y} + J_x\dot{x} = 0, \quad \text{with} \quad y = [\theta^T, \beta^T]^T. \tag{10}$$

Assuming that Jacobian matrix $J_y(q) = \partial\phi/\partial y$ is regular. From (10), after some calculations one gets

$$\dot{q} = R_x\dot{x}. \tag{11}$$

Here matrix $R_x(q) \in \mathbf{R}^{m\times n}$ is defined by

$$R_x = \begin{bmatrix} -J_y^{-1}J_x \\ E \end{bmatrix}. \tag{12}$$

By differentiating (12) w.r.t. time one gets

$$\ddot{\boldsymbol{q}} = \boldsymbol{R}_x \ddot{\boldsymbol{x}} + \dot{\boldsymbol{R}}_x \dot{\boldsymbol{x}}. \tag{13}$$

Noting that matrix $\boldsymbol{R}_x(\boldsymbol{q})$ defined by (12) satisfies the following formula

$$\boldsymbol{\Phi}_q \boldsymbol{R}_x = 0 \quad \text{or} \quad \boldsymbol{R}_x^T \boldsymbol{\Phi}_q^T = 0. \tag{14}$$

To eliminate multipliers $\boldsymbol{\lambda}$ in equation (9) one premultiplies it from link with $\boldsymbol{R}_x^T$

$$\boldsymbol{R}_x^T \left[\boldsymbol{M}_s(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}_s(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{D}_s \dot{\boldsymbol{q}} + \boldsymbol{g}_s(\boldsymbol{q}) \right] + \boldsymbol{R}_x^T \boldsymbol{\Phi}_q^T(\boldsymbol{q})\boldsymbol{\lambda} = \boldsymbol{R}_x^T \boldsymbol{B}_s \boldsymbol{u}. \tag{15}$$

In order to get the equation of motion in minimal coordinates, substituting $\dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}}$ defined from (11) and (13) into (15), one obtains

$$\boldsymbol{M}_x(\boldsymbol{x})\ddot{\boldsymbol{x}} + \boldsymbol{C}_x(\boldsymbol{x}, \dot{\boldsymbol{x}})\dot{\boldsymbol{x}} + \boldsymbol{D}_x \dot{\boldsymbol{x}} + \boldsymbol{g}_x(\boldsymbol{x}) = \boldsymbol{R}_x^T \boldsymbol{B}_s \boldsymbol{u} =: \boldsymbol{\tau}_x \tag{16}$$

where

$$\boldsymbol{M}_x(\boldsymbol{x}) = \boldsymbol{R}_x^T \boldsymbol{M}_s(\boldsymbol{q})\boldsymbol{R}_x, \quad \boldsymbol{C}_x(\boldsymbol{x}, \dot{\boldsymbol{x}}) = \boldsymbol{R}_x^T (\boldsymbol{C}_s(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{R}_x + \boldsymbol{M}_s(\boldsymbol{q})\dot{\boldsymbol{R}}_x),$$

$$\boldsymbol{D}_x(\boldsymbol{x}) = \boldsymbol{R}_x^T \boldsymbol{D}_s \boldsymbol{R}_x, \quad \boldsymbol{g}_x(\boldsymbol{x}) = \boldsymbol{R}_x^T \boldsymbol{g}_s(\boldsymbol{q}).$$

In equation (16) the following properties are still guaranteed: $\boldsymbol{M}_x(\boldsymbol{x})$ is a symmetric and positive definite matrix and $\boldsymbol{N}_x = [\dot{\boldsymbol{M}}_x(\boldsymbol{x}) - 2\boldsymbol{C}_x(\boldsymbol{x}, \dot{\boldsymbol{x}})]$ skew-symmetric matrix [5]. These properties are important for control design in the next section.

3. DESIGN OF A SLIDING MODE CONTROLLER IN A TASK SPACE

The objective of the control problem is to find the law of the motor voltage so that the motion of the mobile platform tracks the given trajectory defined by $\boldsymbol{x}_d(t)$. The basis for the design of controller in a task space is the equation (16). To design a controller based on sliding mode, the sliding surface is chosen as follows

$$\boldsymbol{s} = \dot{\boldsymbol{e}}_x(t) + \boldsymbol{\Lambda}\boldsymbol{e}_x(t), \ \ \boldsymbol{\Lambda} = \text{diag}([\lambda_1, \lambda_2, ...\lambda_n]) > 0, \tag{17}$$

with the tracking error $\boldsymbol{e}_x(t) = \boldsymbol{x}(t) - \boldsymbol{x}_d(t)$. Defining

$$\dot{\boldsymbol{x}}_r = \dot{\boldsymbol{x}}_d(t) - \boldsymbol{\Lambda}\boldsymbol{e}_x(t), \ \ \boldsymbol{s} = \dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_r, \ \ \dot{\boldsymbol{s}} = \ddot{\boldsymbol{x}} - \ddot{\boldsymbol{x}}_r. \tag{18}$$

To find a control law, a Lyapunov candidate function is chosen as

$$V = \frac{1}{2}\boldsymbol{s}^T \boldsymbol{M}_x(x)\boldsymbol{s}. \tag{19}$$

Differentiating this function w.r.t. time, one gets

$$\dot{V} = \boldsymbol{s}^T \boldsymbol{M}_x(\boldsymbol{x})\dot{\boldsymbol{s}} + \frac{1}{2}\boldsymbol{s}^T \dot{\boldsymbol{M}}_x(\boldsymbol{x})\boldsymbol{s}. \tag{20}$$

From (16), (18) and the skew-symmetric property of $\dot{\boldsymbol{M}}_x - 2\boldsymbol{C}_x$ ones obtains

$$\dot{V} = \boldsymbol{s}^T [\boldsymbol{\tau}_x - \boldsymbol{C}_x(\boldsymbol{x}, \dot{\boldsymbol{x}})\dot{\boldsymbol{x}}_r - \boldsymbol{D}_x(\boldsymbol{x})\dot{\boldsymbol{x}} - \boldsymbol{g}_x(\boldsymbol{x}) - \boldsymbol{M}_x(\boldsymbol{x})\ddot{\boldsymbol{x}}_r]. \tag{21}$$

The formula (21) suggests to choose a control law as

$$\boldsymbol{\tau}_x = \hat{\boldsymbol{M}}_x(\boldsymbol{x})\ddot{\boldsymbol{x}}_r + \hat{\boldsymbol{C}}_x(\boldsymbol{x}, \dot{\boldsymbol{x}})\dot{\boldsymbol{x}}_r + \hat{\boldsymbol{D}}_x(\boldsymbol{x})\dot{\boldsymbol{x}}_r + \hat{\boldsymbol{g}}_x(\boldsymbol{x}) - \boldsymbol{K}_{pd}\boldsymbol{s} - \boldsymbol{K}_s\,\mathrm{sgn}(\boldsymbol{s}) \qquad (22)$$

where $\hat{\boldsymbol{M}}_x(\boldsymbol{x}), \hat{\boldsymbol{C}}_x(\boldsymbol{x}, \dot{\boldsymbol{x}}), \hat{\boldsymbol{D}}_x(\boldsymbol{x}), \hat{\boldsymbol{g}}_x(\boldsymbol{x})$ are the estimators of $\boldsymbol{M}_x(\boldsymbol{x}), \boldsymbol{C}_x(\boldsymbol{x}, \dot{\boldsymbol{x}}), \boldsymbol{D}_x(\boldsymbol{x}), \boldsymbol{g}_x(\boldsymbol{x})$, $\mathrm{sgn}(\boldsymbol{s}) = [\mathrm{sgn}(s_1), \mathrm{sgn}(s_2), ..., \mathrm{sgn}(s_n)]^T$, $\boldsymbol{K}_{pd}$ and $\boldsymbol{K}_s$ are the symmetric positive definite matrices, $\boldsymbol{K}_{pd} = \boldsymbol{K}_{pd}^T > 0$, $\boldsymbol{K}_s = \boldsymbol{K}_s^T > 0$. For simplicity, these two matrices are selected to be in diagonal form as

$$\boldsymbol{K}_{pd} = \mathrm{diag}\{k_{pd}^{11}, k_{pd}^{22}, ..., k_{pd}^{nn}\}, \quad \boldsymbol{K}_s = \mathrm{diag}\{k_s^{11}, k_s^{22}, ..., k_s^{nn}\}.$$

Substituting control law (22) into (16) one gets

$$\begin{aligned} \boldsymbol{M}_x(\boldsymbol{x})(\ddot{\boldsymbol{x}} - \ddot{\boldsymbol{x}}_r) + \boldsymbol{C}_x(\boldsymbol{x}, \dot{\boldsymbol{x}})(\dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_r) + \boldsymbol{D}_x(\dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_r) = \\ - \boldsymbol{K}_{pd}\boldsymbol{s} - \boldsymbol{K}_s\,\mathrm{sgn}(\boldsymbol{s}) + \tilde{\boldsymbol{M}}_x(\boldsymbol{x})\ddot{\boldsymbol{x}}_r + \tilde{\boldsymbol{C}}_x(\boldsymbol{x}, \dot{\boldsymbol{x}})\dot{\boldsymbol{x}}_r + \tilde{\boldsymbol{D}}_x\dot{\boldsymbol{x}}_r + \tilde{\boldsymbol{g}}_x(\boldsymbol{q}) \end{aligned} \qquad (23)$$

or

$$\boldsymbol{M}_x(\boldsymbol{x})\dot{\boldsymbol{s}} + \boldsymbol{C}_x(\boldsymbol{x}, \dot{\boldsymbol{x}})\boldsymbol{s} + \boldsymbol{D}_x\boldsymbol{s} = -\boldsymbol{K}_{pd}\boldsymbol{s} - \boldsymbol{K}_s\,\mathrm{sgn}(\boldsymbol{s}) + \boldsymbol{d}_x(\boldsymbol{x}, \dot{\boldsymbol{x}}, \dot{\boldsymbol{x}}_r, \ddot{\boldsymbol{x}}_r), \qquad (24)$$

where system uncertainties are

$$\begin{aligned} &\tilde{\boldsymbol{M}}_x(\boldsymbol{x}) = \hat{\boldsymbol{M}}_x(\boldsymbol{x}) - \boldsymbol{M}_x(\boldsymbol{x}), \ \tilde{\boldsymbol{C}}_x(\boldsymbol{x}, \dot{\boldsymbol{x}}) = [\hat{\boldsymbol{C}}_x(\boldsymbol{x}, \dot{\boldsymbol{x}}) - \boldsymbol{C}_x(\boldsymbol{x}, \dot{\boldsymbol{x}})], \\ &\tilde{\boldsymbol{D}}_x = \hat{\boldsymbol{D}}_x - \boldsymbol{D}_x, \ \tilde{\boldsymbol{g}}_x(\boldsymbol{q}) = \hat{\boldsymbol{g}}_x(\boldsymbol{q}) - \boldsymbol{g}_x(\boldsymbol{q}), \\ &\boldsymbol{d} = \tilde{\boldsymbol{M}}_x(\boldsymbol{x})\ddot{\boldsymbol{x}}_r + \tilde{\boldsymbol{C}}_x(\boldsymbol{x}, \dot{\boldsymbol{x}})\dot{\boldsymbol{x}}_r + \tilde{\boldsymbol{D}}_x\dot{\boldsymbol{x}}_r + \tilde{\boldsymbol{g}}_x(\boldsymbol{q}). \end{aligned}$$

From (24) it leads to

$$\boldsymbol{M}_x(\boldsymbol{x})\dot{\boldsymbol{s}} = -\boldsymbol{C}_x(\boldsymbol{x}, \dot{\boldsymbol{x}})\boldsymbol{s} - \boldsymbol{D}_x\boldsymbol{s} - \boldsymbol{K}_{pd}\boldsymbol{s} - \boldsymbol{K}_s\,\mathrm{sgn}(\boldsymbol{s}) + \boldsymbol{d}_x(\boldsymbol{x}, \dot{\boldsymbol{x}}, \dot{\boldsymbol{x}}_r, \ddot{\boldsymbol{x}}_r). \qquad (25)$$

Substituting (25) into (20) one obtains

$$\begin{aligned} \dot{V} &= \boldsymbol{s}^T\boldsymbol{M}_x(\boldsymbol{x})\dot{\boldsymbol{s}} + \tfrac{1}{2}\boldsymbol{s}^T\dot{\boldsymbol{M}}_x(\boldsymbol{x})\boldsymbol{s} \\ &= -\boldsymbol{s}^T[\boldsymbol{D}_x + \boldsymbol{K}_{pd}]\boldsymbol{s} - \boldsymbol{s}^T(\boldsymbol{K}_s\,\mathrm{sgn}(\boldsymbol{s}) - \boldsymbol{d}_x) + \tfrac{1}{2}\boldsymbol{s}^T[\dot{\boldsymbol{M}}_x(\boldsymbol{x}) - 2\boldsymbol{C}_x(\boldsymbol{x}, \dot{\boldsymbol{x}})]\boldsymbol{s} \qquad (26) \\ &= -\boldsymbol{s}^T[\boldsymbol{D}_x + \boldsymbol{K}_{pd}]\boldsymbol{s} - \boldsymbol{s}^T(\boldsymbol{K}_s\,\mathrm{sgn}(\boldsymbol{s}) - \boldsymbol{d}_x). \end{aligned}$$

Assuming that the system uncertainties are bounded, it means $|\boldsymbol{d}| \le d_0$, or $|d_i| \le d_{i,0}$, with the above selection, one gets

$$\begin{aligned} \dot{V} &= -\boldsymbol{s}^T[\boldsymbol{D}_x + \boldsymbol{K}_{pd}]\boldsymbol{s} - \boldsymbol{s}^T\boldsymbol{K}_s\mathrm{sgn}(\boldsymbol{s}) + \boldsymbol{s}^T\boldsymbol{d} \\ &= -\boldsymbol{s}^T[\boldsymbol{D}_x + \boldsymbol{K}_{pd}]\boldsymbol{s} - \sum_{i=1}^n \boldsymbol{K}_s^{(\mathrm{ii})}|\boldsymbol{s}_i| + \sum_{i=1}^n \boldsymbol{s}_i d_i \\ &\le -\boldsymbol{s}^T[\boldsymbol{D}_x + \boldsymbol{K}_{pd}]\boldsymbol{s} - \sum_{i=1}^n (\boldsymbol{K}_s^{(ii)} - |d_i|)\boldsymbol{s}_i. \end{aligned}$$

To be sure $\dot{V} \le 0$, the elements of the matrix $\boldsymbol{K}_s$ will be chosen such that $K_s^{ii} - |d_i| > 0$ or $K_s^{ii} > |d_i| > 0$. In control law (22) the discontinuous terms $\mathrm{sign}(\boldsymbol{s})$ can cause the chattering phenomenon in the system. In order to reduce this effect, the $\mathrm{sign}(\boldsymbol{s})$ function will be replaced by an approximation $(2/\pi)\mathrm{atan}(k\boldsymbol{s}), k >> 1$.

In summary, the control law is determined with the following quantities

$$\begin{aligned} &\boldsymbol{s} = \dot{\boldsymbol{e}}_x(t) + \boldsymbol{\Lambda}\boldsymbol{e}_x(t), \quad \boldsymbol{e}_x(t) = \boldsymbol{x}(t) - \boldsymbol{x}_d(t), \\ &\dot{\boldsymbol{x}}_r = \dot{\boldsymbol{x}}_d(t) - \boldsymbol{\Lambda}\boldsymbol{e}_x(t), \\ &\ddot{\boldsymbol{x}}_r = \ddot{\boldsymbol{x}}_d(t) - \boldsymbol{\Lambda}\dot{\boldsymbol{e}}_x(t), \quad \boldsymbol{s} = \dot{\boldsymbol{x}}(t) - \dot{\boldsymbol{x}}_r(t), \end{aligned}$$

$$\boldsymbol{\tau}_x = \hat{\boldsymbol{M}}_x(\boldsymbol{q})\ddot{\boldsymbol{x}}_r + \hat{\boldsymbol{C}}_x(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{x}}_r + \hat{\boldsymbol{D}}_x\dot{\boldsymbol{x}} + \hat{\boldsymbol{g}}_x(\boldsymbol{q}) - \boldsymbol{K}_{pd}\boldsymbol{s} - \tfrac{2}{\pi}\boldsymbol{K}_s\mathrm{atan}(k\boldsymbol{s}), \qquad (27)$$

$$\boldsymbol{u} = (\boldsymbol{R}_x^T\boldsymbol{B}_s)^{-1}\boldsymbol{\tau}_x. \qquad (28)$$

The control diagram in the task space is given in Fig. 2.
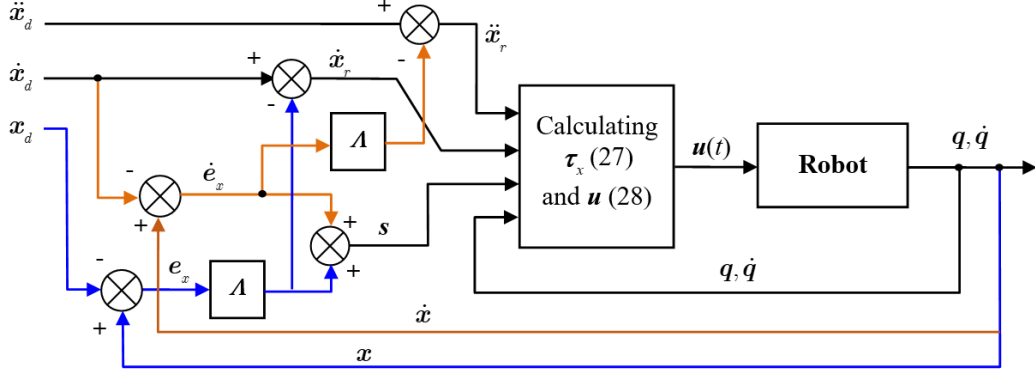
Figure 2. Control diagram

4.   NUMERICAL SIMULATIONS

4.1.   Stabilization technique in simulation

The numerical simulation of a parallel manipulator requires to solve the DAEs (16) and (17). In order to solve the DAEs, normally, the constraint equations are derived twice w.r.t. time to transform the DAEs to ODEs. Because the second order derivative instead of the constrained equation is used in the simulation, the constraint stabilization is needed. Here the Baumgarte's technique is applied in the simulation [12]. The main idea of this approach is that the constraint in acceleration level, $\ddot{\boldsymbol{\phi}}(\boldsymbol{q}) = \boldsymbol{0}$, is replaced by the following equation

$$\ddot{\boldsymbol{\phi}}(\boldsymbol{q}) + 2\delta\omega_0\dot{\boldsymbol{\phi}}(\boldsymbol{q}) + \omega_0^2\boldsymbol{\phi}(\boldsymbol{q}) = 0, \qquad (29)$$

with two parameters of the method $0 < \delta \leq 1$ and $\omega_0 > 0$. Differentiating twice (7) w.r.t. time, yields

$$\boldsymbol{\Phi}_q(\boldsymbol{q})\ddot{\boldsymbol{q}} + \dot{\boldsymbol{\Phi}}_q(\boldsymbol{q})\dot{\boldsymbol{q}} = 0. \qquad (30)$$

So the constraint equation using in simulation derived from (29) will be

$$\boldsymbol{\Phi}_q\ddot{\boldsymbol{q}} + \dot{\boldsymbol{\Phi}}_q\dot{\boldsymbol{q}} + 2\delta\omega_0\boldsymbol{\Phi}_q\dot{\boldsymbol{q}} + \omega_0^2\boldsymbol{\phi} = 0. \qquad (31)$$

Combination with the equation (15), the dynamic model used for simulation of parallel manipulator becomes

$$\boldsymbol{R}_x^T\left(\boldsymbol{M}_s(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}_s(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{D}_s\dot{\boldsymbol{q}} + \boldsymbol{g}_s(\boldsymbol{q})\right) = \boldsymbol{R}_x^T\boldsymbol{B}_s\boldsymbol{u},$$
$$\boldsymbol{\Phi}_q(\boldsymbol{q})\ddot{\boldsymbol{q}} + \dot{\boldsymbol{\Phi}}_q(\boldsymbol{q})\dot{\boldsymbol{q}} + 2\delta\omega_0\boldsymbol{\Phi}_q(\boldsymbol{q})\dot{\boldsymbol{q}} + \omega_0^2\boldsymbol{\phi}(\boldsymbol{q}) = 0,$$

or in the matrix form as follows

$$\begin{bmatrix} \boldsymbol{R}_x^T\boldsymbol{M}_s(\boldsymbol{q}) \\ \boldsymbol{\Phi}_q(\boldsymbol{q}) \end{bmatrix}\ddot{\boldsymbol{q}} + \begin{bmatrix} \boldsymbol{R}_x^T[\boldsymbol{C}_s(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{D}_s] \\ \dot{\boldsymbol{\Phi}}_q + 2\delta\omega_0\boldsymbol{\Phi}_q(\boldsymbol{q}) \end{bmatrix}\dot{\boldsymbol{q}} + \begin{bmatrix} \boldsymbol{R}_x^T\boldsymbol{g}_s(\boldsymbol{q}) \\ \omega_0^2\boldsymbol{\phi}(\boldsymbol{q}) \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}_x^T\boldsymbol{B}_s\boldsymbol{u} \\ 0 \end{bmatrix}. \qquad (32)$$

4.2.   Numerical simulations

Some simulations with a 3-DOF planar parallel manipulator which moves in the horizontal plane driven by 3 actuators are implemented. The numerical simulations are carried out in Matlab. The model of the robot is shown in Fig. 1. The mechanical parameters of the robot are given as follows [22].

Base: $L = 1.2$m.

Two links of the legs:

$$l_{i,1} = 0.581\text{m}, \quad m_{i,1} = 2.072\text{kg}, \quad J_{C1} = 0.13\text{kg.m}^2,$$

$$l_{i,2} = 0.620\text{m}, \quad m_{i,2} = 0.750\text{kg}, \quad J_{C2} = 0.03\text{kg.m}^2,$$

Platform: $a = 0.2$m, $m_7 = 0.978$kg, $J_{C7} = 0.007\text{kg.m}^2$, $\alpha_i = [\frac{7}{6}\pi, -\frac{1}{6}\pi, \frac{1}{2}\pi]$.

Gear trans: $r = 10$ (transmission ratio).

DC motor : $J_m = 0.01\text{kg.m}^2, K_m = 3.00\text{Nm/A}, K_e = 0.10\text{Vs/rad}, R_a = 3.00\text{Ohm}$.

In order to compare the controller proposed in this paper to an augmented PD controller, two simulations are carried out. The first one is implemented with the SMC and the second one is conducted with the augmented PD controller, that is given as

$$\boldsymbol{\tau}_x = \boldsymbol{M}_x(\boldsymbol{x})\ddot{\boldsymbol{x}}_d + \boldsymbol{C}_x(\boldsymbol{x}, \dot{\boldsymbol{x}})\dot{\boldsymbol{x}}_d + \boldsymbol{D}_x(\boldsymbol{x})\dot{\boldsymbol{x}}_d + \boldsymbol{g}_x(\boldsymbol{x}) - \boldsymbol{K}_d\dot{\boldsymbol{e}} - \boldsymbol{K}_p\boldsymbol{e}, \quad \boldsymbol{K}_d, \boldsymbol{K}_p > 0. \quad (33)$$

This control law is driven from a Lyapunov candidate function as

$$V = \frac{1}{2}\dot{\boldsymbol{e}}^T\boldsymbol{M}_x(\boldsymbol{x})\dot{\boldsymbol{e}} + \frac{1}{2}\boldsymbol{e}^T\boldsymbol{K}_p\boldsymbol{e}, \quad \boldsymbol{K}_p = \boldsymbol{K}_p^T > 0, \quad \boldsymbol{e} = \boldsymbol{x} - \boldsymbol{x}_d. \quad (34)$$

In two simulations, the center of the platform will be moved along a circular trajectory, while its orientation is constant, $\varphi = \pi/3$[rad]. The trajectory has a center at $(x_C, y_C) = (0.6, 0.4)$[m] and radius $R = 0.12$[m]. This trajectory is chosen to be in the work space of the mobile platform. The parameters of two controllers are chosen as

SMC: $\boldsymbol{\Lambda} = 30\text{diag}(1, 1, 1), \quad \boldsymbol{K}_{pd} = 200\text{diag}(1, 1, 1), \quad \boldsymbol{K}_s = 0.01\text{diag}(1, 1, 1),$

PD: $\boldsymbol{K}_p = 850\text{diag}(1, 1, 1), \quad \boldsymbol{K}_d = 150\text{diag}(1, 1, 1).$

Additionally, the robot parameters such as masses and moments of inertia used in the control laws are chosen about $15 - 25\%$ different compared to those that are used in the dynamic model.

The results are shown in Fig. 3. to Fig. 12. The results show that with the SMC the motion of the mobile platform tracks the desired trajectory after about 0.22 second (Fig. 7 and 9), while this time is about 1 second with the PD controller (Fig. 8 and 10). The moving platform errors in position are kept about 10^{-5} mm and error in angle is kept about 10^{-5} radian. Fig. 11 and 12 show that the center of mobile platform tracks the desired trajectory better than the one with the augmented PD controller. These results show the robustness of the SMC controller.
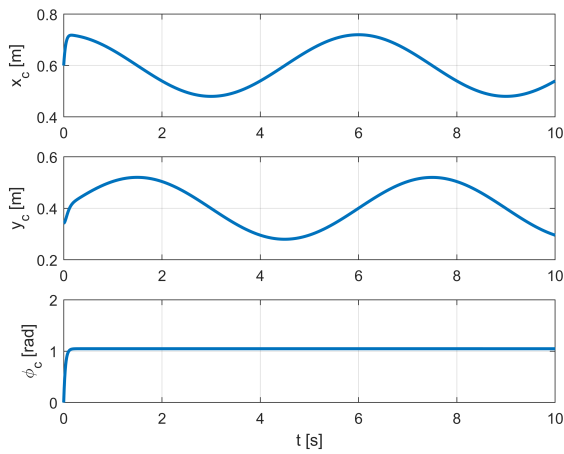
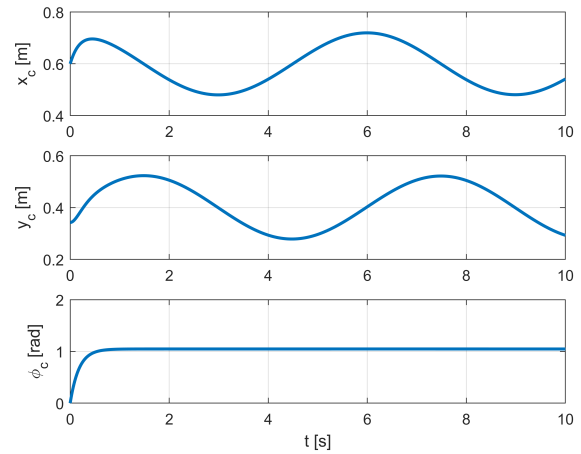*Figure 3.* Trajactory tracking in a task space (SMC)
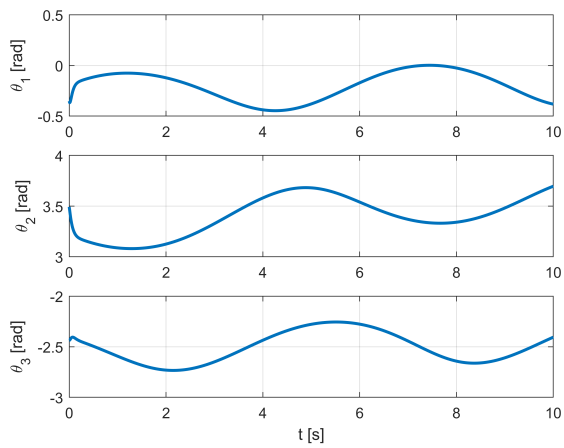

Figure 4. Trajactory tracking in a task space (PD)
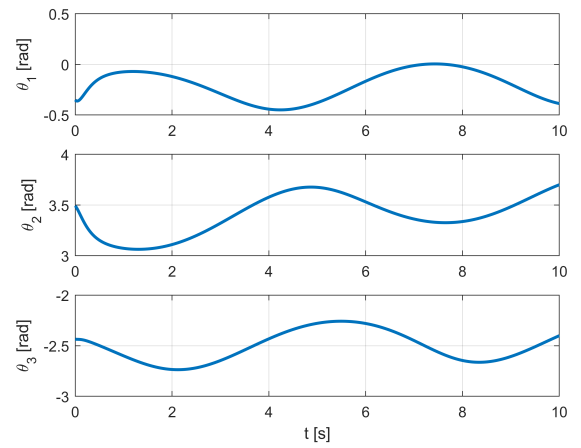

Figure 5. Trajectory tracking of active joint (SMC)

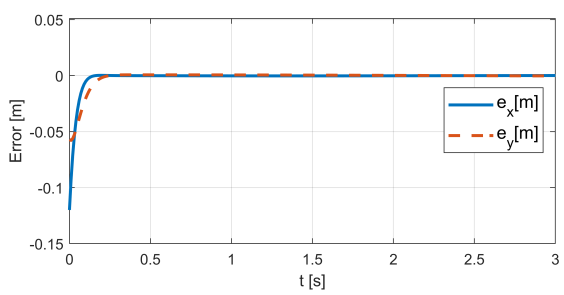*Figure 6.* Trajectory tracking of active joint (PD)

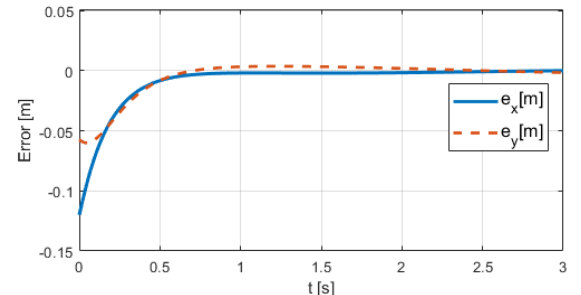*Figure 7.* Tracking errors of x, y coordinates (SMC)

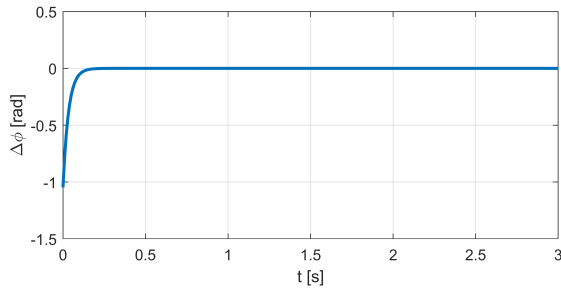*Figure 8.* Tracking errors of x, y coordinates (PD)

Figure 9. Tracking errors of ϕ angle (SMC)
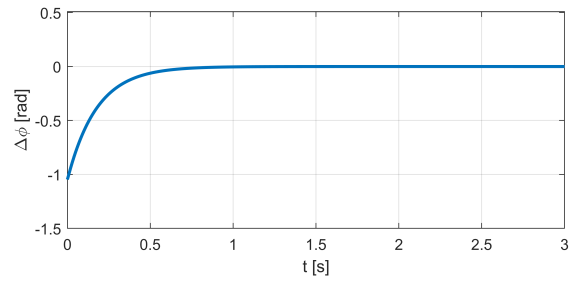
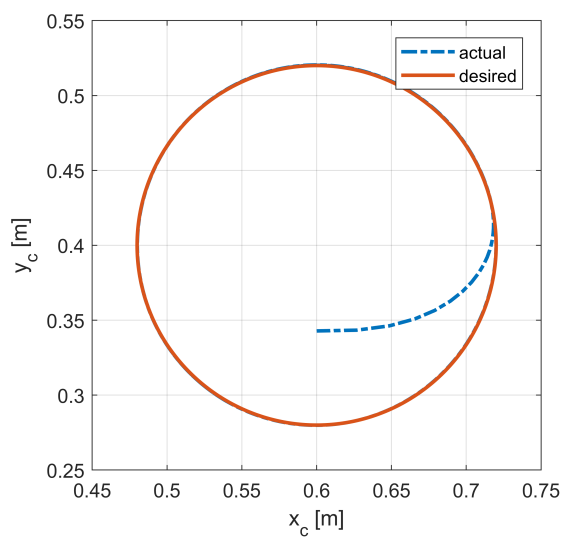*Figure 10.* Tracking errors of ϕ angle (PD)

Figure 11. The desired path vs. the actual path of the platform center (SMC)
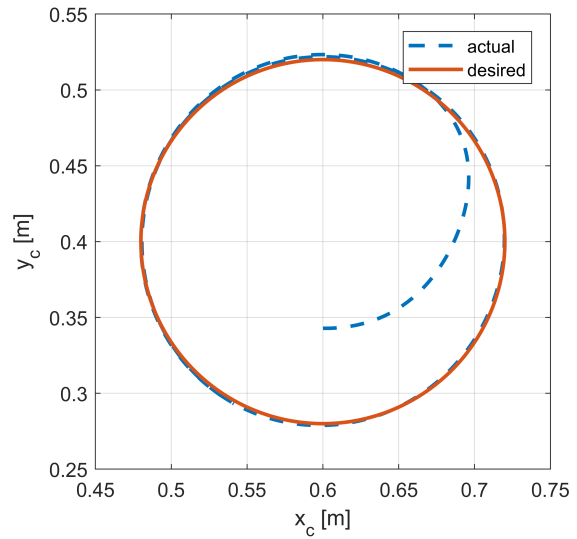
Figure 12. The desired path vs. the actual path of the platform center (PD)

5.   CONCLUSION

This paper presented the modeling of a parallel robot driven by DC motors. The equations of motion for parallel manipulator were derived by using the redundant generalized coordinates and the Lagrangian multipliers. Moreover, the dynamics of DC motors were also integrated in the whole models of the system. The obtained model in this study included electric and mechanical parts, so it described better a mechanical system that is driven by electric motors such as parallel manipulators. The equations of motion in DAEs form were then transformed to ODE form of task space variables for the control design. The sliding mode controllers in the task space has been proposed for the manipulator. Simulation results have shown that the controller is effective for trajectory control of the parallel manipulator.

APPENDIX: THE DYNAMIC MODEL OF A 3R̲RR PARALLEL ROBOT

For this model, the following vectors have been used in Section 2

$$\boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3]^T, \quad \boldsymbol{\beta} = [\beta_1, \beta_2, \beta_3]^T, \quad \boldsymbol{x} = [x_C, y_C, \varphi]^T.$$

Here $\boldsymbol{q} = [\theta_1, \beta_1, \theta_2, \beta_2, \theta_3, \beta_3, x_C, y_C, \varphi]^T$.

The equation of motion is derived with the kinetic and potential energy

$$T(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \frac{1}{2} \sum_{k=1}^{7} (m_k v_{Ck}^2 + J_{Ck} \omega_k^2), \quad \Pi(\boldsymbol{q}) = 0.$$

The constraint equations of the manipulator are given as follows

$$\phi(\boldsymbol{q}) \equiv \phi(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{x}) = 0,$$

$$x_{O1} + l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \beta_1) = x_C + b \cos(\varphi + \alpha_1),$$

$$y_{O1} + l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \beta_1) = y_C + b \sin(\varphi + \alpha_1),$$

$$x_{O2} + l_1 \cos \theta_2 + l_2 \cos(\theta_2 + \beta_2) = x_C + b \cos(\varphi + \alpha_2),$$

$$y_{O2} + l_1 \sin \theta_2 + l_2 \sin(\theta_2 + \beta_2) = y_C + b \sin(\varphi + \alpha_2),$$

$$x_{O3} + l_1 \cos \theta_3 + l_2 \cos(\theta_3 + \beta_3) = x_C + b \cos(\varphi + \alpha_3),$$

$$y_{O3} + l_1 \sin \theta_3 + l_2 \sin(\theta_3 + \beta_3) = y_C + b \sin(\varphi + \alpha_3).$$

The mass matrix $\boldsymbol{M}_s(\boldsymbol{q})$ is obtained as

$$\boldsymbol{M}_s = \begin{bmatrix} m_{11} & m_{12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ m_{21} & m_{22} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & m_{33} & m_{34} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & m_{43} & m_{44} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & m_{55} & m_{56} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & m_{65} & m_{66} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & m_{77} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & m_{88} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & m_{99} \end{bmatrix},$$

$$m_{11} = J_{c1} + J_{c2} + \frac{1}{4} m_1 l_1^2 + m_2 l_1^2 + r^2 J_r, \quad m_{12} = \frac{1}{2} m_2 l_1 l_2 \cos(\theta_1 - \beta_1),$$

$$m_{21} = m_{12}, \ m_{22} = \frac{1}{4} m_2 l_2^2, \quad m_{33} = J_{c_1} + \frac{1}{4} m_1 l_1^2 + m_2 l_1^2 + r^2 J_r,$$

$$m_{34} = \frac{1}{2} m_2 l_1 l_2 \cos(\theta_2 - \beta_2), \quad m_{43} = m_{34},$$

$$m_{44} = J_{c_2} + \frac{1}{4}m_2 l_2^2, \quad m_{55} = J_{c_1} + \frac{1}{4}m_1 l_1^2 + m_2 l_1^2 + r^2 J_r,$$

$$m_{56} = \frac{1}{2}m_2 l_1 l_2 \cos(\theta_3 - \beta_3), \quad m_{65} = m_{56}, \quad m_{66} = J_{c_1} + \frac{1}{4}m_2 l_2^2,$$

$$m_{77} = m_3, \quad m_{88} = m_3, \quad m_{99} = J_{c_3}.$$

The coriolis and centrifugal matrix $\boldsymbol{C}_s(\boldsymbol{q}, \dot{\boldsymbol{q}})$ are given below and other elements are zero

$$c_{21} = -\frac{1}{2}m_2 l_1 l_2 \sin(\theta_1 - \beta_1)\dot{\theta}_1, \quad c_{12} = \frac{1}{2}m_2 l_1 l_2 \sin(\theta_1 - \beta_1)\dot{\beta}_1,$$

$$c_{43} = -\frac{1}{2}m_2 l_1 l_2 \sin(\theta_2 - \beta_2)\dot{\theta}_2, \quad c_{34} = \frac{1}{2}m_2 l_1 l_2 \sin(\theta_2 - \beta_2)\dot{\beta}_2,$$

$$c_{65} = -\frac{1}{2}m_2 l_1 l_2 \sin(\theta_3 - \beta_3)\dot{\theta}_3, \quad c_{56} = \frac{1}{2}m_2 l_1 l_2 \sin(\theta_3 - \beta_3)\dot{\beta}_3.$$

The damping matrix $\boldsymbol{D}_s$ is given with

$$\boldsymbol{D}_s = R_a^{-1} r^2 K_m K_e \mathrm{diag}([1, 0, 1, 0, 1, 0, 0, 0, 0]).$$

The control input matrix $\boldsymbol{B}_s$ is given with zero elements except for

$$B_s(1, 1) = B_s(3, 2) = B_s(5, 3) = R_a^{-1} r K_m.$$

REFERENCES

[1] J.G. Jalon, E. Bayo, *Kinematic and Dynamic Simulation of Multibody Systems The Real-Time Challenge,* Springer Verlag, New York, 1994.

[2] Nguyen Van Khang, *Dynamics of Multibody Systems (in Vietnamese)*, Science and Technics Publishing House, Ha Noi, 2007.

[3] L.-W. Tsai, *Robot Analysis - The Mechanics of Serial and Parallel Manipulators*, John Wiley & Sons, New York, 1999.

[4] J.-P. Merlet, *Parallel Robots (2nd Ed.)*, Springer, Berlin, 2006.

[5] R.M. Murray, Z. Li, and S.S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, Boca Raton, Fla., 1994.

[6] J. Angeles, *Fundamentals of robotics Mechanical Systems (2nd Ed.)*, Springer Verlag, New York, 2003.

[7] Y. Nakamura, *Advanced Robotics/ Redundancy and Optimization*, Addison-Wesley Publishing Company, Reading 1991.

[8] M. Ceccarelli, *Fundaments of Mechanics of Robotic Manipulation*, Springer, Dordrecht, 2004.

[9] A. Muller, T. Hufnagel, "Model-based control of redundantly actuated parallel manipulators in redundant coordinates", *Robotics and Autonomous Systems*, vol. 60, pp. 563–571, 2012.

[10] L. Sciavicco and B. Siciliano, *Modelling and Control of Robot Manipulators, 2nd Ed.*, Springer-Verlag, London, UK, 2000.

[11] Pham Thuong Cat, *Some Modern Control Methods for Industrial Robots (in Vietnamese)*, Thai Nguyen University Publishing, 2009.

[12] J. Baumgarte, "Stabilization of constraints and integrals of motion in dynamical systems, *Computer Methods in Applied Mechanics and Engineering*, vol 1, pp 1–16, 1972.

[13] T.R. Kane and D.A. Levinson, *Dynamics: Theory and applications*, McGraw-Hill, New York, 1985.

[14] A.A. Shabana, *Dynamics of multibody systems, 2nd Ed.*, Cambridge Uni. Press, 1998.

[15] Benjamin C. Kuo and Farid Golnaraghi, *Automatic Control Systems (8th Ed.)*, John Wiley Sons, Inc., New York, NY, USA, 2002.

[16] Katsuhiko Ogata, *Modern Control Engineering (4th Ed.)*, Prentice Hall PTR, Upper Saddle River, NJ, USA, 2001.

[17] A. Fumagalli and P. Masarati, "Real-time inverse dynamics control of parallel manipulators using general-purpose multibody software", *Multibody System Dynamics*, vol. 22, pp. 47-68, 2009.

[18] C. Yang, J. Zhao, L. Li, and S. K. Agrawal, "Design and implementation of a novel modal space active force control concept for spatial multi-DOF parallel robotic manipulators actuated by electrical actuators", *ISA Transactions*, 2017.

[19] C. Roldán-Paraponiaris, F. J. Campa, and O. Altuzarra, "Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics", *Mechatronics*, vol. 47, no. Supplement C, pp. 208–222, 2017.

[20] L. Beji, A. Abichou, P. Joli, and M. Pascal, *Nonlinear Control of a Parallel Robot Including Motor Dynamics*, Springer Vienna, 1997 (pp. 45–52).

[21] M. Jouini, M. Sassi, N. Amara, and A. Sellami, "Modeling and control for a 6-DOF platform manipulator", *2013 Inter. Conf. on Electrical Engineering and Software Applications*, 2013 (pp. 1–5).

[22] Trung Do Thanh, Jens Kotlarski, Bodo Heimann, and Tobias Ortmaier, "Dynamics identification of kinematically redundant parallel robots using the direct search method", *Mechanism and Machine Theory*, vol. 55, pp. 104–121, 2012.