Optimization of trajectory tracking control of 3-DOF translational robot use PSO method based on inverse dynamics control for surgery application

Ngoc Pham Van Bach[1], Quan Dam Hai[2], Thanh Bui Trung[3]
[1, 2]Space Technology Institute, Hanoi, Vietnam
[3]Faculty of Mechanical engineering, Hung Yen University of Technology and Education,
Hung Yen, Vietnam
[1]Corresponding author
E-mail: [1]*pvbngoc@sti.vast.vn*, [2]*damhaiquan@gmail.com*, [3]*buitrungthanh@utehy.edu.vn*

Received 14 April 2021; received in revised form 14 July 2021; accepted 26 July 2021
DOI https://doi.org/10.21595/jve.2021.21997

Abstract. This research presents an optimal trajectory tracking control method for improving the accuracy of 3-DOF translational parallel robots in the surgery field based on the Particle Swarm Optimization (PSO) method. The 3-DOF translational robot has three translational degrees of freedom, which consists of three arms with three revolute joints and twelve spherical joints. Firstly, the kinematics model is established; and the dynamics equation of the Robot is built by applying the Lagrange equation of the first type, and then the dynamics controller of 3-DOF translational robot is designed base on the dynamics equations. Secondly, a trajectory tracking controller model using the Particle Swarm Optimization based on inverse dynamics controlled method for 3-DOF translational Robot is designed. The control performance results of the proposed controller is evaluated by simulation and compared with the other published research results. Finally, the proposed controller can achieve better tracking performance in comparison with other controllers as Proportional Integral Derivative (PID) controller, GPI controller, and adaptive controller.

Keywords: parallel robot, trajectory tracking control, inverse dynamics control, particle swarm optimization.

1. Introduction

The parallel mechanism robot has many good characteristics such as: high stiffness, high performance, high speed, and high accuracy. The Delta-type parallel mechanism robot, which was first proposed by Clavel are commonly used in food industrial, medical surgery and other high speed applications. Up to now, mechanical structure, kinematics, dynamics and intelligent control of Delta-type robot has been studied and developed worldwide. The kinematics of Delta robot was solved by applying vector loop equations and the dynamics equations were built by applying the Lagrangian equation of the first type [1]. Hamid D. Taghirad was discussed about kinematics, dynamics and control of parallel robot. Despite the fact that the controlling of parallel robot was mentioned, however, author have given some briefly block paragraph of controllers. These controller is applied to control planar manipulator and Stewart-Gough platform [2]. The most common control method for trajectory tracking on Delta robot is to use PID controller. To maximize appropriate workspace, Pradya Prempraneerach has researched and proposed the mechanical structure of Delta robot and using the traditional PID controller to tracking trajectory of the robot [3]. Sheng Jian et al. modeled the PMSM motor and used PID to control the motion of PMSM motor at three actuator joints of Delta robot [4], the gains of PID controller were set manual. In [5], L. Angel and J. Viola performed the trajectory tracking using two PID controller. Authors used Integer Order PID (IOPID) and Fractional Order PID (FOPID) base computed torque control technique and proposed an algorithm to find gains of the controller by using Matlab function. The fuzzy method is used by many authors in [6]-[9] to control the motion of Delta robot.

Mohsen Asgari et al. performed dynamics modeling and trajectory tracking control using four methods as PD, PID, fuzzy PD, and fuzzy PID [6]. The comparison between four control method show errors of actuated joints using PID and fuzzy PID are the same (0.008rad) and less than results of PD and fuzzy PD. These errors increased in the presence of disturbance. Similarly, other researches [7]-[9] using fuzzy control show big amount of calculation and large error. Trajectory tracking of Delta robot were implemented by Neural Network based controller [10]-[13]. Similar to fuzzy method, these researches need a big amount of experimental data to train the controller. This defect lead to the increase of time to converge to the reference trajectory. It took more than 1s to reach the reference trajectory [10]. In [14]-[18] the authors applied the robust control method H∞ for motion control of Delta robot. These researches designed the H∞ robust controller and applied to the Delta robot to perform pick-and-place operation. The performance of proposed controller was compared to other controller as Computed Torque Control and PID. Simulation results show that the H∞ controller performed better than others. However, the steady state error of moving platform still exist and be unstable (maximum value more than 1mm). The Computed Torque Control technique is used in many researches [19, 20]. In this technique inverse dynamic is use to compensate nonlinear of the robot's dynamic. This controller also called Inverse Dynamic Controller (IDC). Jianlong Hao et al. established the dynamic equation of Delta robot based on the principal of virtual work and proposed CTC to control trajectory tracking [19]. Extended CTC used extra data from additional sensors at passive joints of the Delta robot [20]. The performance of extended CTC was compared to the traditional CTC. In both researches above, the gains of controller have chosen manual. Luis Angel Castañeda et al. were designed an adaptive controller to solve the trajectory tracking problem of Delta robot with uncertain dynamical model [21]. They have simulated and compared the results with two other controller as PID and GPI. In [22], Joao Fabian presented the implementation of trajectory tracking of Delta robot using PD and LQR controller, where the PD controller parameters were obtained using pole placement. However, authors have not proposed any optimal trajectory tracking control method to manipulate the 3-DOF Translational Delta Robot type in the surrey application yet. In this research, authors focus on optimal mechanism and kinematics of 3-DOF Translational Delta Robot type to enlarge workspace by vary arm lengths. The trajectory tracking was simulated by using PID controller with various payload to demonstrate the pick-and-place operation. To optimize the trajectory of the 3-DOF translational robot, Particle Swarm Optimization method is chosen as a novel optimization trajectory tracking control for 3-DOF translational Delta robot type. PSO is a population-based optimization technique inspired by the motion of bird flocks and schooling fish. PSO shares many similarities with evolutionary computation techniques. The system is initialized with a population of random solutions, and the search for the optimal solution is performed by updating generations. In PSO, the potential solutions of particle swarm optimization for solving various kinds of optimization problems, called particles, move in the problem space by following the current optimum particles. PSO is computationally more efficient in terms of both speed and memory requirements.

This paper organizes like this: presents an optimal trajectory tracking control method for 3-DOF translational robot Delta type. The 3-DOF translational robot with three translational degrees of freedom has chosen. First of all, mechanism, kinematic and dynamic of robot are presented. Secondly, based on the dynamic of 3-DOF translational robot, an inverse dynamic controller (IDC) is designed to track a desired trajectory. To achieve the best performance of trajectory control, the Particle Swarm Optimization (PSO) is used to determine parameters of the controller. In this optimization method, Integral of the Square of the Error (ISE) play an important role as objective conditions. The simulation results are presented and compared with other control methods in the rest of this article.

2. Mathematic of kinematics and dynamics of 3-DOF translational robot

2.1. Kinematic of 3-DOF Translational robot

The 3-DOF translational robot has three arm connect the moving platform and the fixed base is shown in Fig. 1. Each arm have two parts, the upper arm is connected to the active motor using revolute joint. Each lower arm is made up of a four-bar parallelogram which connect with the moving platform and the upper arm using spherical joint. The moving platform can perform translational movement in three directions. In other words, the robot is called three translational degrees of freedom.

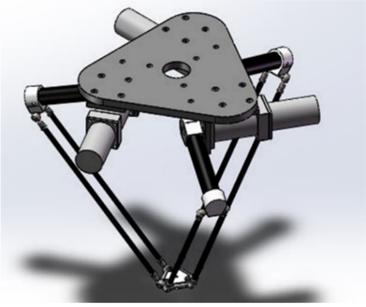


Fig. 1. 3D drawing of 3-DOF translational robot

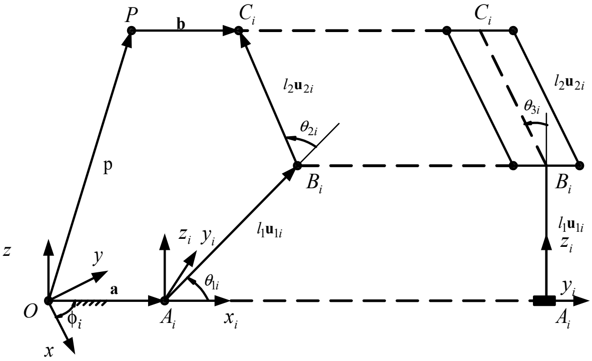


Fig. 2. Vectors description of 3-DOF translational robot

Kinematics of 3-DOF translational robot has discussed by many authors with many methods, in this study authors chose the vector loop equation method to solve this problem. In [1], authors proposed the inverse kinematics of Delta parallel robot base on vectors loop equation. Vectors of 3-DOF translational robot are illustrated in Fig. 2. As shown on Fig. 2, the vectors loop equation of parallel robot is given as follow:

$$l_{1i}u_{1i} + l_{2i}u_{2i} = p + b - a. \tag{1}$$

Expressing Eq. (1) in the (x_i, y_i, z_i) frame with $i = 1,2,3$ yield:

$$\begin{bmatrix} l_1\cos\theta_{1i} + l_2\sin\theta_{3i}\cos(\theta_{1i} + \theta_{2i}) \\ l_2\cos\theta_{3i} \\ l_1\sin\theta_{1i} + l_2\sin\theta_{3i}\sin(\theta_{1i} + \theta_{2i}) \end{bmatrix} = \begin{bmatrix} c_{xi} \\ c_{yi} \\ c_{zi} \end{bmatrix}, \tag{2}$$

where l_1, l_2 are length of lower arm and upper arm; $\theta_{1i}, \theta_{2i}, \theta_{3i}$ with $(i = 1,2,3)$ are actuated joint angles and passive joint angles respectively, c_{xi}, c_{yi}, c_{zi} are position of point C_i. The position of C_i can be obtained as:

$$\begin{bmatrix} c_{xi} \\ c_{yi} \\ c_{zi} \end{bmatrix} = \begin{bmatrix} \cos\varphi_i & \sin\varphi_i & 0 \\ -\sin\varphi_i & \cos\varphi_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} + \begin{bmatrix} a-b \\ 0 \\ 0 \end{bmatrix}, \tag{3}$$

where p_x, p_y, p_z are position of the center point of moving platform in the (x, y, z)coordinate system. The solutions of $\theta_{1i}, \theta_{2i}, \theta_{3i}$ are found by solving Eq. (2).

Solving Eq. (2) obtained:

$$\theta_{3i} = \arccos\left(\frac{c_{yi}}{l_2}\right), \tag{4}$$

$$\theta_{2i} = \arccos(\kappa), \tag{5}$$

where $\kappa = \left(c_{xi}^2 + c_{yi}^2 + c_{zi}^2 - l_1^2 - l_2^2\right)/(2l_1l_2\sin\theta_{3i})$.

Substituting Eq. (4) and Eq. (5) into Eq. (2):

$$l_1\sin\theta_{1i} + l_2\sin\theta_{3i}\sin(\theta_{1i} + \theta_{2i}) = c_{zi}, \tag{6}$$

$$l_1\cos\theta_{1i} + l_2\sin\theta_{3i}\cos(\theta_{1i} + \theta_{2i}) = c_{xi}. \tag{7}$$

Solving two equations above yield:

$$\theta_{1i} = \arctan\frac{-c_{xi}g_{2i} + c_{zi}g_{1i}}{c_{zi}g_{2i} + c_{xi}g_{1i}}, \tag{8}$$

where $g_{2i} = l_2\sin\theta_{3i}\sin\theta_{2i}$, $g_{1i} = (l_1 + l_2\sin\theta_{3i}\cos\theta_{2i})$.

2.2. Dynamics equation of 3-DOF translational robot

In this research author using the Lagrangian formulation of the first type to solve the dynamics problems of 3-DOF translational robot. The Lagrangian function of the first type is written as [1]:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \frac{\partial L}{\partial q_j} = Q_j + \sum_{i=1}^{k} \lambda_i \frac{\partial \Gamma_i}{\partial q_j}, \quad j = 1, \cdots, n, \tag{9}$$

where Γ_i denotes the ith constraint function, k is the number of constraint functions, λ_i is the Lagrangian multiplier. L is the Lagrangian function. Constraint equations Γ_i with $i = 1,2,3$ are obtained from the fact that the distance between joints B and C is always equal to the length of the connecting rod of the upper arm:

$$\begin{aligned} \Gamma_i = \overline{C_iB_i}^2 - l_2^2 &= (p_x + b\cos\varphi_i - a\cos\varphi_i - l_1\cos\varphi_i\cos\theta_{1i})^2 - l_2^2 \\ &+ \left(p_y + b\sin\varphi_i - a\sin\varphi_i - l_1\sin\varphi_i\cos\theta_{1i}\right)^2 + (p_z - l_1\sin\theta_{1i})^2, \quad i = 1,2,3. \end{aligned} \tag{10}$$

Taking the derivatives of contraint functions and Lagrangian function, Lagrangian multipliers, the dynamics of 3-DOF translational robot can be written as:

$$\begin{aligned} \tau_1 = &(I_m + I_1 + m_2l_1^2)\ddot{\theta}_{11} + (m_1l_{1c} + m_2l_1)g_c\cos\theta_{11} \\ &-2l_1\lambda_1\left[\left(p_x\cos\varphi_1 + p_y\sin\varphi_1 + b - a\right)\sin\theta_{11} - p_z\cos\theta_{11}\right], \end{aligned} \tag{11}$$

$$\begin{aligned}\tau_2 &= (I_m + I_1 + m_2 l_1^2)\ddot{\theta}_{12} + (m_1 l_{1c} + m_2 l_1) g_c \cos\theta_{12} \\ &\quad -2 l_1 \lambda_2 \left[(p_x \cos\varphi_2 + p_y \sin\varphi_2 + b - a) \sin\theta_{12} - p_z \cos\theta_{12} \right], \\ \tau_3 &= (I_m + I_1 + m_2 l_1^2)\ddot{\theta}_{13} + (m_1 l_{1c} + m_2 l_1) g_c \cos\theta_{13} \\ &\quad -2 l_1 \lambda_3 \left[(p_x \cos\varphi_3 + p_y \sin\varphi_3 + b - a) \sin\theta_{13} - p_z \cos\theta_{13} \right],\end{aligned}$$

where I_m is axial moment of inertia of the rotor mounted on the ith limb, I_1 is moment of lower arms, m_p, m_1, m_2 are mass of moving platform, lower arm and upper arm respectively Eq. (11) can be rewritten in matrix form as following:

$$M(q)\ddot{q} + G(q) = \tau, \tag{12}$$

where: M is the manipulator inertia matrix, $G(q)$ is the gravitational vector, τ is the vector of generalized forces, q is vector of generalized Lagrange coordinates:

$$M = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & 0 \\ 0 & 0 & m_{33} \end{bmatrix}, \quad G(q) = [G_1 \quad G_2 \quad G_3]^T,$$
$$\tau = [\tau_1 \quad \tau_2 \quad \tau_3]^T, \quad q = [q_1 \quad q_2 \quad q_3]^T.$$

3. Optimization trajectory tracking controller of 3-DOF translational robot

3.1. Controller design

The dynamics Eq. (11) can be rewritten in matrices form as below:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau + \tau_d, \tag{13}$$

where $M(q)$, $C(q,\dot{q})$, $G(q)$ are mass matrix, Coriolis and Centrifugal matrix, gravity vector respectively, τ_d is disturbance.

The Inverse Dynamics control law of 3-DOF transnational robot is written as [19]:

$$\tau = \hat{M}(q)\left(\ddot{q}_d + K_p e + K_d \dot{e}\right) + \hat{C}(q,\dot{q})\dot{q} + \hat{G}(q), \tag{14}$$

where $\hat{M}(q), \hat{C}(q,\dot{q}), \hat{G}(q)$ denote the approximate estimate of the manipulator mass matrix, Coriolis and Centrifugal matrix and gravity vector in the joint space respectively; q_d and $\ddot{q}_d$ are joint angle and joint acceleration desired; K_d, K_p are positive diagonal matrices.

Substituting Eq. (9) into Eq. (8) the dynamics equation of 3-DOF translational robot can be expressed as follow:

$$\begin{aligned} M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) &= \tau + \tau_d \\ &= \hat{M}(q)\left(\ddot{q}_d + K_p e + K_d \dot{e}\right) + \hat{C}(q,\dot{q})\dot{q} + \hat{G}(q) + \tau_d. \end{aligned} \tag{15}$$

In the dynamics equation of 3-DOF translational robot $C(q,\dot{q}) = 0$ thus $\hat{C}(q,\dot{q}) = 0$. In real model, the $M(q)$ and $G(q)$ can calculated, assume that $\hat{M}(q) = M(q)$, $\hat{G}(q) = G(q)$ and there is no disturbance wrench is applied to the manipulator, the error dynamic is simplified to:

$$\ddot{q}_d + K_d \dot{e} + K_p e = 0. \tag{16}$$

With error dynamic equation Eq. (16) above, the tracking error can be designed to convergence toward zero by choosing K_d, K_p appropriate. The Inverse Dynamics controller model in joint space is depicted in Fig. 3.

This section may be divided by subheadings. It should provide a concise and precise description of the experimental results, their interpretation, as well as the experimental conclusions that can be drawn.

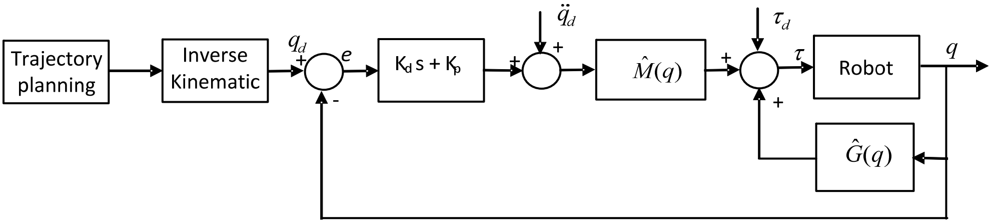


Fig. 3. Trajectory tracking inverse dynamics controller

3.2. Particle Swarm Optimization (PSO) algorithm

This paper is presented a method of choosing the parameters of the Inverse Dynamics Controller. The dynamics error equation reach to zero by choosing K_d, K_p gains. In this research, the authors are using the Particle Swarm Optimization method to find the optimization gains. The advantages of PSO are: simplicity and efficiency, proven in many other parameters training problems. The PSO method was developed by simulation of simplified social model, where each population is called a swarm. Each candidate, called a particle, flies through problem space to look for the optimal position, similar to food searching of bird swarm. Assume that, the search space is n-dimension, each particle i corresponding to an n -dimension position vector $x_i = (x_{i1}, \ x_{i2}, \ \cdots, \ x_{in})$, and velocity is represented by an n-dimension velocity vector, $v_i = (v_{i1}, \ v_{i2}, \ \cdots, \ v_{in})$. The best previously visited position i is called individual best position, $P_i = (p_{i1}, \ p_{i2}, \ \cdots, \ p_{in})$. The position of the best individual of the whole swarm is called the global best position, $G = (g_1, \ g_2, \ \cdots, \ g_n)$. The velocity and position are updated at each step according to the following two equations:

$$v_i(k+1) = w.v_i(k) + c_1 r_1 \big(P_i(k) - x_i(k)\big) + c_2 r_2 \big(G(k) - x_i(k)\big), \tag{17}$$
$$x_i(k+1) = x_i(k) + v_i(k), \tag{18}$$

where w is inertia weight; r_1, r_2 are random variables in the range of $[0 \quad 1]$; c_1, c_2 are positive constant parameters called acceleration coefficients.

In this paper, the PSO was combined with the kinematic and dynamic model of 3-DOF translational robot to find out the gains of controller. The parameters was chosen as $w = 0.9$, $c_1 = 0.12$, $c_2 = 1.2$, $n = 50$, $k = 50$. The fitness function is the measurement of the trajectory's quality. The objective functions applied for this particular problem is the Integral Square Error (ISE). In this case, the best position is determined by the minimum of Integral of the Square of the Error value by using Matlab toolbox. In each iteration, the gains are updated sequentially in individual best position set and global best position set corresponding to ISE value. The gains are taken after 50 iterations show the errors are acceptable as mention in simulation part below.

4. Simulation results and discussion

The simulation of kinematic, dynamic and control of the 3-DOF translational robot was implemented in Matlab/Simulink. Dynamic parameters of the robot are described in Table 1.

As mention above, corresponding to dynamic parameters of the robot, the K_d, K_p of the controller was found by applying the PSO method, we have:

$K_p = [373.3152 \ 0 \ 0; 0 \ 374.47880; 0 \ 0 \ 371.6830]$,
$K_d = [14.0493 \ 0 \ 0; 0 \ 13.9150 \ 0; 0 \ 0 \ 14.1524]$.

Table 1. Parameter of 3-DOF translational

Parameter	Value	Unit
Mass of lower arm m_1	0.426	kg
Mass of upper arm m_2	0.069	kg
Mass of moving platform m_p	0.069	kg
Length of lower arm l_1	171.5	mm
Length of upper arm l_2	396	mm
Radius of fix platform a	100	mm
Radius of moving platform b	22.5	mm

The trajectory form was designed by using the third odder polynomial as below [23]:

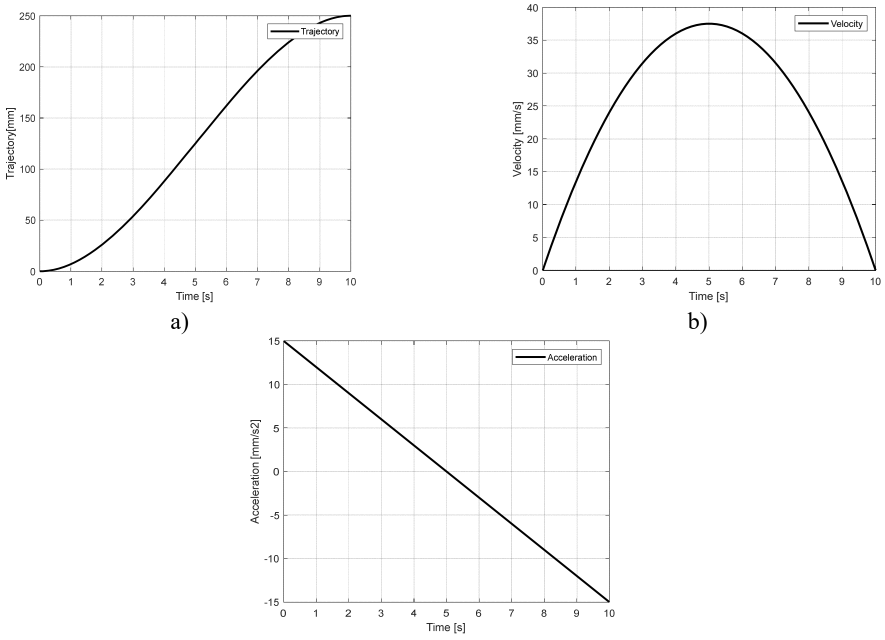
$$q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3. \tag{19}$$

The coefficients of Eq. (14) are obtained by solving set of equations Eq. (15):

$$\begin{bmatrix} 1 & t_0 & t_0^2 & t_0^3 \\ 0 & 1 & 2t_0 & 3t_0^2 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} q_0 \\ v_0 \\ q_f \\ v_f \end{bmatrix}, \tag{20}$$

where $q(t_0) = q_0$, $\dot{q}(t_0) = v_0$, $q(t_f) = q_f$, $\dot{q}(t_f) = v_f$ and t_0, q_0, v_0 are initial time, position and velocity respectively, t_f, q_f, v_f are final time, position and velocity respectively.

In this research, the trajectory of the moving platform is implemented in OX direction which relative to the (x, y, z) coordinate system, where $t_0 = 0$, $t_f = 10$ (sec), $q_0 = 0$, $q_f = 250$ (mm) and $v_0 = 0$, $v_f = 0$. The trajectory, velocity and acceleration are showed in Fig. 4.



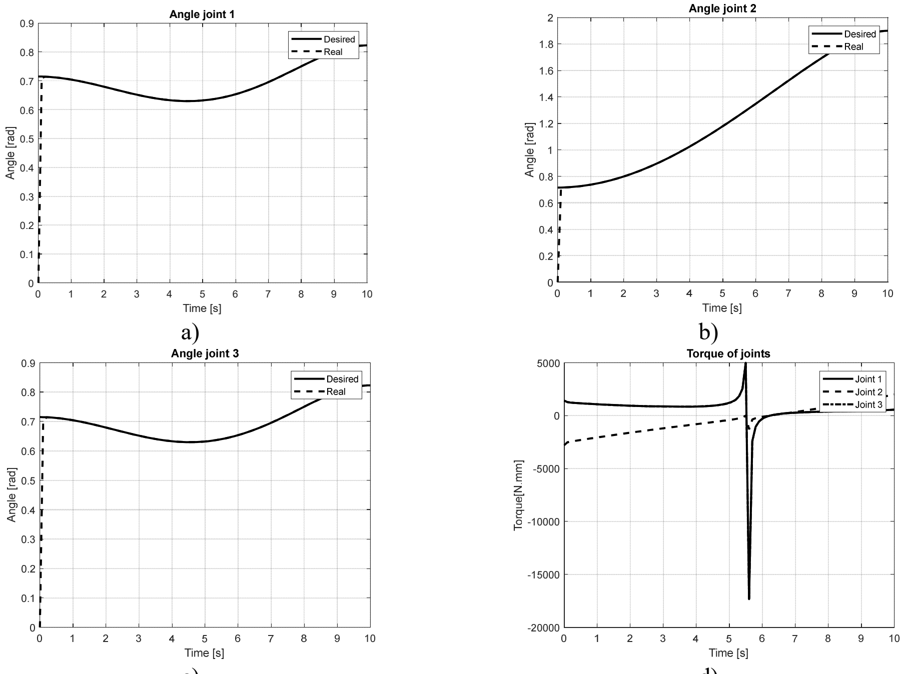
a)

b)

c)

Fig. 4. Trajectory planning

The numerical simulation results of the joint angles of three arm are shown in Figs. 5(a-c). The torque of three actuators joint are presented in Fig. (d).



a)

b)

c)

d)

Fig. 5. Torque of three actuator joint

As mention at the first part, the objective condition of optimization method is integral of the square of the error J_{ISE}. The PSO will apply to find K_d, $K_p$which corresponds to minimum of J_{ISE} [24]:

$$J_{ISE}=\int_{0}^{+\infty} e_{ss}^{2}dt, \tag{21}$$

where e_{ss} is steady state error of actuator joint of the robot. According to parameters and the trajectory planning of the Delta robot, the steady state error of three actuators joints are showed in Fig. 6. Where $J_{ISE1}=0.2555, J_{ISE3}=0.2557$ respectively.

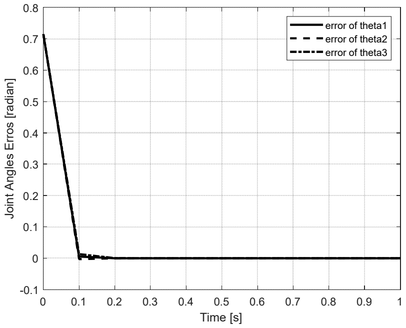
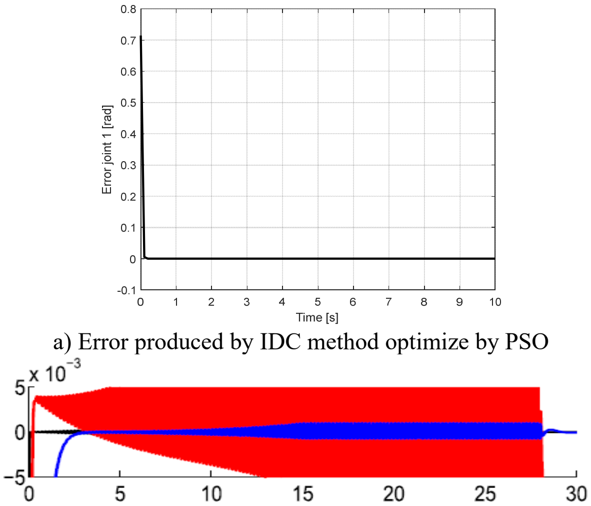


Fig. 6. Steady state error of three actuator joint

The performance of proposed control method could be evaluated by compare with other methods in [21] as shown below in Fig. 7.

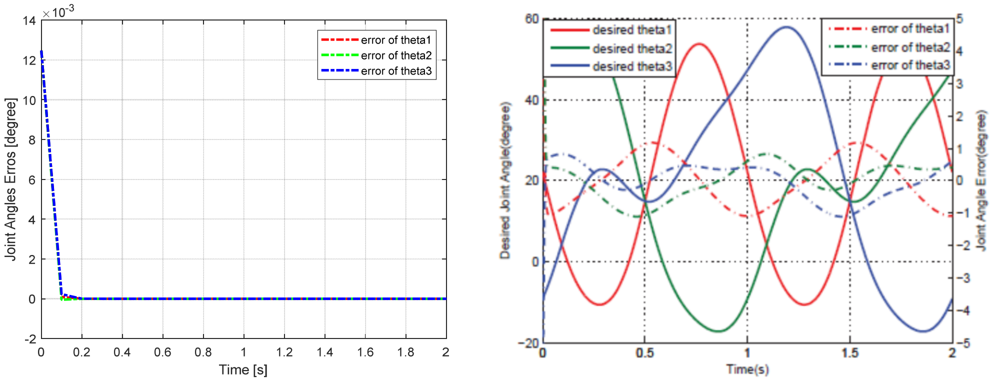


a) Error produced by IDC method optimize by PSO

b) Red line is error produced by PID controller, blue line is error produced by GPI controller (base on extended state observer), black line is error produced by adaptive controller proposed in [21]

Fig. 7. Comparing angle error with other methods

Fig. 7 shows the error of actuator joint 1 of the proposed controller and other controllers implemented in [21]. Fig. 7(b) shows that all results still oscillate around zero points with 0.005 rad deviation. The result of adaptive controller is the best of three methods but still restricted to a vicinity of the origin. Fig. 7(a) shows the error produced by optimization IDC is better than other methods. The graphs show that the error converges to zero after 0.2 s and the steady-state error is maintained no more than 5.10-5 rad. Another comparison can be performed between the proposed controller and the controller in [19], the graph of errors as showed in Fig. 8.



a)

b) The maximum error of joints at 1 degree (0.017 rad) while in a) as mention above, the error is 5.10-5 rad

Fig. 8. Comparing Errors of actuator joints of optimization a) IDC and b) PD-CTC [19]

Fig. 8 show the error of three actuators joint angle of the 3-DOF translational robot. In Fig. 8(a), the error curve converges to zero after 0.1sec, and the steady state error is very small, while in Fig. 8(b), this value cannot converge to zero, even it oscillates around –1 and 1 degree. The comparisons above demonstrates the better performance of this controller in accuracy and stability.

5. Conclusions

In this paper, the trajectory tracking of 3-DOF translational robot problems was implemented and optimized by used the Inverse Dynamics Control (IDC) method and Particle Swarm Optimization (PSO) algorithm. The IDC is used to track the desired trajectory in joint space. The IDC is a novel and effective method for 3-DOF translational robot control systems in the surgery field. PSO algorithm to find out the gains of controller more quick and simple than other methods. The comparison of simulation results showed the steady-state error of the proposed method is smaller and converge to zero faster than other methods. In the future, this control method will be used to combine with other force control to improve the control of the 3-DOF translational robot in the real model.

Acknowledgements

This work is funded by VAST project under grant number VAST01.06/19-20.

References

[1] L. W. Tsai, *Robot Analysis: The Mechanics of Serial and Parallel Manipulators*. New York: Wiley, 1999.

[2] H. D. Taghirad, *Parallel Robots*. Boca Raton: CRC Press, 2013, https://doi.org/10.1201/b16096

[3] P. Prempraneerach, "Delta parallel robot workspace and dynamic trajectory tracking of delta parallel robot," in *2014 International Computer Science and Engineering Conference (ICSEC)*, Jul. 2014, https://doi.org/10.1109/icsec.2014.6978242

[4] S. Jian and Y. Lou, "Application of motion control system for delta parallel robot," in *2017 IEEE International Conference on Information and Automation (ICIA)*, Jul. 2017, https://doi.org/10.1109/icinfa.2017.8079002

[5] L. Angel and J. Viola, "Fractional order PID for tracking control of a parallel robotic manipulator type delta," *ISA Transactions*, Vol. 79, pp. 172–188, Aug. 2018, https://doi.org/10.1016/j.isatra.2018.04.010

[6] M. Asgari, M. A. Ardestani, and M. Asgari, "Dynamics and control of a novel 3-DoF spatial parallel robot," in *2013 First RSI/ISM International Conference on Robotics and Mechatronics (ICRoM 2013)*, Feb. 2013, https://doi.org/10.1109/icrom.2013.6510102

[7] J. Zhang, C. Lian, R. Gao, and L. Shi, "3-Degree-of-freedom parallel robot control based fuzzy theory," in *2010 2nd International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, Aug. 2010, https://doi.org/10.1109/ihmsc.2010.62

[8] O. Linda and M. Manic, "Uncertainty-robust design of interval type-2 fuzzy logic controller for delta parallel robot," *IEEE Transactions on Industrial Informatics*, Vol. 7, No. 4, pp. 661–670, Nov. 2011, https://doi.org/10.1109/tii.2011.2166786

[9] C. E. Boudjedir et al., "Fuzzy logic iterative learning control for trajectory tracking of parallel kinematic manipulators," in *2017 5th International Conference on Electrical Engineering – Boumerdes (ICEE-B)*, Oct. 2017, https://doi.org/10.1109/icee-b.2017.8192041

[10] T. Uzunovic, E. Golubovic, E. A. Baran, and A. Sabanovic, "Configuration space control of a parallel delta robot with a neural network based inverse kinematics," in *2013 8th International Conference on Electrical and Electronics Engineering (ELECO)*, Nov. 2013, https://doi.org/10.1109/eleco.2013.6713892

[11] Y. Huang and Z. Huang, "Neural network based dynamic trajectory tracking of delta parallel robot," in *2015 IEEE International Conference on Mechatronics and Automation (ICMA)*, Aug. 2015, https://doi.org/10.1109/icma.2015.7237782

[12] W. Liao, W. Liang, and A. Wang, "Inverse dynamics control of a parallel robot based on RBF Neural network," in *2017 International Conference on Advanced Mechatronic Systems (ICAMechS)*, Dec. 2017, https://doi.org/10.1109/icamechs.2017.8316501

[13] X. Lu, Y. Zhao, and M. Liu, "Self-learning interval type-2 fuzzy neural network controllers for trajectory control of a delta parallel robot," *Neurocomputing*, Vol. 283, pp. 107–119, Mar. 2018, https://doi.org/10.1016/j.neucom.2017.12.043

[14] M. Rachedi, M. Bouri, and B. Hemici, "Application of an H∞ control strategy to the parallel Delta," in *2012 2nd International Conference on Communications, Computing and Control Applications (CCCA)*, Dec. 2012, https://doi.org/10.1109/ccca.2012.6417912

[15] M. Rachedi, B. Hemici, and M. Bouri, "Application of the mixed sensitivity problem H∞ and H2 to the parallel Delta," in *2013 3rd International Conference on Systems and Control (ICSC)*, Oct. 2013, https://doi.org/10.1109/icosc.2013.6750903

[16] M. Rachedi, M. Bouri, and B. Hemici, "H∞ feedback control for parallel mechanism and application to delta robot," in *2014 22nd Mediterranean Conference of Control and Automation (MED)*, Jun. 2014, https://doi.org/10.1109/med.2014.6961584

[17] M. Rachedi, "Model based control of 3 DOF parallel delta robot using inverse dynamic model," in *2017 IEEE International Conference on Mechatronics and Automation (ICMA)*, Aug. 2017, https://doi.org/10.1109/icma.2017.8015814

[18] M. Rachedi, M. Bouri, and B. Hemici, "Robust control of a parallel robot," in *2015 International Conference on Advanced Robotics (ICAR)*, Jul. 2015, https://doi.org/10.1109/icar.2015.7251491

[19] J. Hao et al., "Dynamic modeling and control simulation of a modified delta manipulator," in *2015 IEEE International Conference on Information and Automation (ICIA)*, Aug. 2015, https://doi.org/10.1109/icinfa.2015.7279537

[20] P. Bengoa, A. Zubizarreta, I. Cabanes, A. Mancisidor, and E. Portillo, "A stable model-based control scheme for parallel robots using additional sensors," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2015, https://doi.org/10.1109/iros.2015.7353816

[21] L. A. Castaneda, A. Luviano-Juarez, and I. Chairez, "Robust trajectory tracking of a delta robot through adaptive active disturbance rejection control," *IEEE Transactions on Control Systems Technology*, Vol. 23, No. 4, pp. 1387–1398, Jul. 2015, https://doi.org/10.1109/tcst.2014.2367313

[22] J. Fabian, C. Monterrey, and R. Canahuire, "Trajectory tracking control of a 3 DOF delta robot: a PD and LQR comparison," in *2016 IEEE XXIII International Congress on Electronics, Electrical Engineering and Computing (INTERCON)*, Aug. 2016, https://doi.org/10.1109/intercon.2016.7815581

Ngoc Pham Van Bach is currently a Doctor in the Space Technology Institute, Vietnam Academy of Science and Technology where he earned the Ph.D. degree in Mechanical and Automation Engineering from the Kyungnam University, Republic of Korea in August 2013. He was a Postdoctoral Associate in the Department of Mechanical and Automation Engineering, Kyungnam University from August to December 2013. His research interests include artificial intelligence, Robotics, Dynamics control of mechanism, Mechanism design and intelligent control. He currently serves as Editorial Board of UTEHY Journal of Science and Technology. He has served as a reviewer of Advances in Science, Technology and Engineering Systems Journal (ASTESJ). He has served as a Chair for some Domestic and International conferences such as ICEMA-5.

Hai Quan Dam received master's degree in electronic engineering from Thai Nguyen University of Technology, Thai Nguyen, Viet Nam, in 2013. Now he works at Thai Nguyen College of Economics and Finance. His current research interests include robot control, electronic engineering.

Trung Thanh-Bui received his bachelor's degree in telecommunication from Hanoi University of Communication and Transport, Vietnam in 2000. He gained both Master and Doctor degree in mechatronics from Asian Institute of Technology, Thailand in 2005 and 2008, respectively. He is now Lecturer at Mechatronics Department, Hung Yen University of Technology and Education. His current researches are control engineering, robotics and optimization.