

Article

Metaheuristic Identification for an Analytic Dynamic Model of a Delta Robot with Experimental Verification

Tae-Hyoung Kim ^{1,*} , Yeongjae Kim ¹ , Taeheon Kwak ¹  and Masaaki Kanno ²

¹ Department of Mechanical Engineering, Chung-Ang University, 84 Heukseok-ro, Dongjak-gu, Seoul 06974, Republic of Korea

² Institute of Science and Technology, Academic Assembly, Niigata University, 8050 Ikarashi 2-Nocho, Nishi-ku, Niigata-shi 950-2181, Japan

* Correspondence: kimth@cau.ac.kr

Abstract: The dynamic-parameter identification process for developing a suitable precise mathematical model for the implementation and operation of parallel-link robots has received attention. In this study, an efficient and reliable system-identification method for a delta robot is proposed. The parallel-link robot's dynamic behavior was mathematically modeled according to the principle of virtual work. The dynamic equations of motion are extended to the system of equations that explicitly characterizes the inertial and centripetal/Coriolis forces, and the frictional effects on the robot's dynamic behavior. Next, the dynamic-parameter identification technique is presented to directly estimate a set of uncertain parameters that are included in the extended dynamic model. In addition, the development of the dynamic model with a generalized inertia matrix for determining the impact of the inertia-coupling characteristic on the robot's dynamic behaviors is examined. Experimental results indicate that the proposed parameter-estimation technique is an extremely useful tool that can achieve the high-quality identification of an analytic dynamic model for a parallel-link robot.

Keywords: delta robots; parallel-link robots; dynamic analytic model; system identification; particle swarm optimization



Citation: Kim, T.-H.; Kim, Y.; Kwak, T.; Kanno, M. Metaheuristic Identification for an Analytic Dynamic Model of a Delta Robot with Experimental Verification. *Actuators* **2022**, *11*, 352. <https://doi.org/10.3390/act11120352>

Academic Editors: Micky Rakotondrabe

Received: 18 October 2022

Accepted: 23 November 2022

Published: 29 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

A delta robot comprises a spatial parallel structure with multiple links that is generally driven by three revolute actuators implemented on actuated links. Such robots comprise multiple closed-loop kinematic chains that connect the moving platform to the fixed base. This mechanism enables load sharing among the kinematic chains, thereby affording benefits such as higher structural rigidity, higher repetitive positioning accuracy, and faster pick-and-place operations compared with serial robots having sequentially connected links [1,2].

Modeling this type of parallel structure of the robot requires a complex dynamic mathematical model consisting of several nonlinear differential equations. The precision of the system parameters describing the robot's dynamic model is critical for model-based controller designs, validation of robotic simulations, and accurate motion-planning tasks [3]. However, the physical data provided by manufacturers for crucial model parameters are usually limited and inaccurate. Furthermore, important information regarding the dynamic parameters related to frictional, inertial, centripetal, and Coriolis forces may be nonexistent. Therefore, the parameters describing the robot dynamic model inevitably contain uncertainties, resulting in the performance deterioration of the controller, as it is sensitive to the parameter values [4]. Therefore, a challenge in operating a parallel-link robot through a real-time model-based controller is to identify the large number of uncertain model parameters contained in its highly complex dynamic model. Accordingly, several studies have been conducted in recent years to identify the set of dynamic parameters included in the analytic model of parallel-link robots.

Angel and Viola [5] recently suggested a parametric identification method for the analytic dynamic model of parallel-link robots using a recursive least-squares (RLS) estimation algorithm. However, their dynamic model, which was derived by introducing the Euler–Lagrange formulation approach, disregarded frictional force terms. Therefore, frictional effects on the dynamic behavior of the robot could not be analyzed explicitly. Further, their RLS algorithm has several critical drawbacks for practical applications, such as the requirement of considerable computational power, high sensitivity to initial conditions, and the potential for the numerical instability of the output, particularly with limited-precision arithmetic. Lastly, their approach was examined using only the Adams–MATLAB cosimulation model and not an actual parallel-link robot. Liu et al. [6] studied a parameter-identification methodology for a class of parallel manipulators that was incorporated with the truncated singular value decomposition (SVD) algorithm. However, their identification scheme was intended for developing only the kinematic model, not the dynamic analytic model, of the robot. Mata et al. [7] proposed parameter-identification processes applicable to parallel manipulators. To perform identification, they first transformed the complex dynamic equations of parallel manipulators into a linear formulation with respect to uncertain system parameters. The resulting system of equations belonged to a mathematically overdetermined case because the dimension of the equations of motion was greater than the number of parameters to be identified. Therefore, the base parameters that were essential for uniquely determining the system dynamics had to be calculated in advance. However, recognizing a set of base parameters is usually a major challenge in dynamic identification. Further, no routine methodology is available for the analytical determination of such a minimal set of base parameters. Because of such difficulties in assessing the identifiability of the base parameters, they adopted a method that uses a simulated manipulator, additionally built using the dynamic simulation program Adams.

The purpose of the present study is to develop an efficient and reliable methodology for identifying multiple parameters involved in the analytic dynamic model of delta robots. The complex dynamic equations of motion are derived according to the virtual work principle and are then extended to a system of equations that explicitly characterize not only the inertial and centripetal/Coriolis forces, but also the frictional effects on the dynamic behavior of the robot. To directly identify a set of uncertain parameters included in such a dynamic model, the proposed scheme employs a particle swarm optimization (PSO) algorithm with a distributed cyclic neighborhood-search mechanism [8] that was modified to be applicable to parametric identification. During the identification process, the swarm of this metaheuristic optimization optimizes the iterative estimation of unknown dynamic parameters so that the torques computed from the analytic dynamic model using the resulting parameter estimation are as close as possible to the real torques actually applied to the parallel-link robot. The prerequisite in most conventional identification methods for a dynamic model of a delta-type robot is the transformation of complex dynamic equations into linear formulation concerning uncertain system parameters [5–7]. The RLS or SVD algorithm can be applied to the unknown parameter identification only after such a transformation task. In contrast, the proposed approach does not require an additional task for modifying the system of equations into a linear form, and the nonlinear dynamic equations are used as-is in the algorithm for parameter identification. This feature can be attributed to the flexibility of the metaheuristic PSO algorithm. This fact also means that the challenging problem related to the overdetermined system of equations does not occur in our identification process. Therefore, our approach can avoid the problem of cumbersome and time-consuming tasks for developing a simulated manipulator model for simply checking the identifiability of the base parameters. An experimental setup, namely, a high-speed pick-and-place parallel-link robot designed at NT Robot, Co. (Moscow, Russian), was established to evaluate the performance of the proposed direct parametric identification method for the analytic dynamic model of a robot. The experimental results clearly verify that our technique provides a simple and efficient approach that can directly achieve high-

quality identification for an analytic dynamic model of a parallel-link robot without relying on the model transformation task or simulated manipulator implementation.

The remainder of this paper is organized as follows. In Section 2, the evaluation of the conventional analytic dynamic model for delta robots is presented. In Section 3, the analytical model structure of the delta robot derived according to the virtual work principle is presented. Then, the method of direct estimation of the parameters included in such a dynamic model using PSO in combination with the distributed cyclic neighborhood-search mechanism is described. Some experimental results are also presented in detail in this section. A discussion of the presented approach is provided in Section 4. Lastly, we present our conclusions in Section 5.

2. Evaluation of the Conventional Dynamic Analytic Model of a Parallel-Link Robot

The parallel-link manipulator shown in Figure 1 is a three-degree-of-freedom robot that usually comprises three closed-loop kinematic chains, a fixed base, and a moving platform. Owing to the small mass of the manipulator and three actuators on the fixed base, this parallel-link robot can achieve high precision and high movement speed for pick-and-place tasks. Figure 2b shows the kinematic chain of this robot.

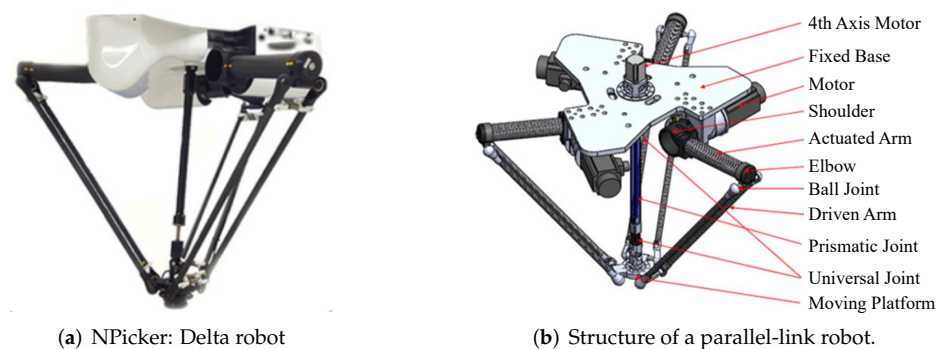


Figure 1. High-speed pick-and-place parallel-link robot designed at NT Robot, Co.

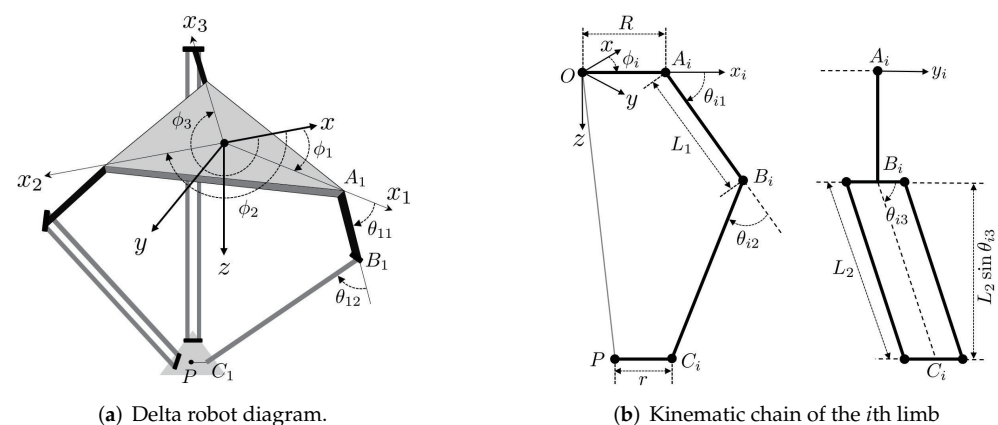


Figure 2. Schematics of the Delta robot.

A global reference coordinate system $O\text{-}xyz$ is located at the center of the fixed base, where the x - and y axis lie on the fixed base and the z axis points vertically upward. Another coordinate system $A_i\text{-}x_iy_iz_i$ ($i = 1, 2, 3$) is attached to the fixed base at a distance R (i.e., fixed base radius) along $O\text{-}xyz$, such that the x_i axis is in line with the extension of $\overrightarrow{OA_i}$, the y_i axis is directed along the revolute joint of the elbow, and the z_i axis is parallel to the z axis of $O\text{-}xyz$. Angle ϕ_i for $i = 1, 2, 3$ is measured from the x axis to the x_i axis, as depicted in Figure 2a, and is a constant parameter of the robot design measured to be $\phi_1 = \pi/3$, $\phi_2 = \pi$, and $\phi_3 = 5\pi/3$. Let $\mathbf{P} = [P_x \ P_y \ P_z]^T$ denote the position vector of the centroid P of the moving platform relative to the $O\text{-}xyz$ coordinate system. The three joint

angles θ_{i1} , θ_{i2} , and θ_{i3} associated with the i th limb are defined as follows: (1) θ_{i1} defines the angle of the actuated arm A_iB_i and is measured from the x_i axis to $\overrightarrow{A_iB_i}$, and (2) θ_{i2} and θ_{i3} represent the rotating angles of the driven arm B_iC_i . The lengths of the actuated and driven arms are denoted as L_1 and L_2 , respectively. Let the radii of the fixed base and moving platform be denoted as R and r , respectively. The masses of the actuated arm, driven arm, and moving platform are denoted as m_a , m_d , and m_p , respectively.

The equations of motion of the considered delta robot can be derived using different methods such as Lagrangian equations, virtual work principle, and Newton–Euler methods. Let the vector of the actuator torques to be applied to the active joints at the shoulders in Figure 1b be denoted as $\tau = [\tau_1 \ \tau_2 \ \tau_3]^T$. Then, from the Euler–Lagrange formulation based on the energy considerations of robotic systems, the dynamic analytic model can be derived and expressed as follows [9–11]: for $i = 1, 2, 3$,

$$\tau_i = \left(\gamma^2 I_m + I_{L_1} + L_1^2 m_d \right) \ddot{\theta}_{i1} - (m_a L_{1c} + m_d L_1) g \cos \theta_{i1} - 2L_1 \lambda_i [P_x \cos \phi_i + P_y \sin \phi_i + r - R] \sin \theta_{i1} - P_z \cos \theta_{i1}, \quad (1)$$

where Lagrangian multipliers λ_i ($i = 1, 2, 3$) are given as follows:

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} = \begin{bmatrix} P_x + rc(\phi_1) - Rc(\phi_1) - L_1c(\phi_1)c(\theta_{11}) & P_x + rc(\phi_2) - Rc(\phi_2) - L_1c(\phi_2)c(\theta_{21}) & P_x + rc(\phi_3) - Rc(\phi_3) - L_1c(\phi_3)c(\theta_{31}) \\ P_y + rs(\phi_1) - Rs(\phi_1) - L_1s(\phi_1)c(\theta_{11}) & P_y + rs(\phi_2) - Rs(\phi_2) - L_1s(\phi_2)c(\theta_{21}) & P_y + rs(\phi_3) - Rs(\phi_3) - L_1s(\phi_3)c(\theta_{31}) \\ P_z - L_1s(\theta_{11}) & P_z - L_1s(\theta_{21}) & P_z - L_1s(\theta_{31}) \end{bmatrix}^{-1} \begin{bmatrix} (m_p + 3m_d)\ddot{P}_x/2 \\ (m_p + 3m_d)\ddot{P}_y/2 \\ (m_p + 3m_d)(\ddot{P}_z - g)/2 \end{bmatrix}. \quad (2)$$

where s denotes $\sin$, and c denotes $\cos$ in the above expressions. Table 1 lists other parameter values provided by the manufacturer.

Table 1. Parameter values provided by the manufacturer for the dynamic analytic model in (1).

Parameter	Value	Unit
Mass of actuated arm (m_a)	3.977	kg
Mass of driven arm (m_d)	0.402	kg
Mass of moving platform (m_p)	2.033	kg
Length of actuated arm (L_1)	0.3423	m
Length of mass center of actuated arm (L_{1c})	0.1563	m
Radius of moving platform (r)	0.045	m
Radius of fixed base (R)	0.2	m
Mass moment of inertia of actuated arm (I_{L_1})	191.9×10^{-4}	kg·m ²
Motor inertia with gear ratio ($\gamma^2 I_m$)	$21^2 \times 2.25 \times 10^{-4}$	kg·m ²

Once the path planning of the center of the moving platform determines $P = [P_x \ P_y \ P_z]^T$, $\dot{P}$, and $\ddot{P}$, the command signal τ_i ($i = 1, 2, 3$), that is, the amount of torque that the motor should exert on each parallel link to achieve the trajectory-tracking performance, can be determined as follows:

1. From the inverse kinematics analysis given in Appendix B, three active joint angles θ_{11} , θ_{21} , and θ_{31} are numerically obtained.
2. The actuated joint velocity $\dot{\theta}_1$ for $\theta_1 = [\theta_{11} \ \theta_{21} \ \theta_{31}]^T$ can be obtained from $\dot{P} = J\dot{\theta}_1$, where J denotes the Jacobian matrix described subsequently (see (16)). Further, angular acceleration $\ddot{\theta}_1$ can be determined from $\ddot{P} = J\ddot{\theta}_1 + \dot{J}\dot{\theta}_1$ (see (23) and (24)).
3. By using P and θ_1 and their time derivatives $\dot{P}$, $\ddot{P}$, $\dot{\theta}_1$, and $\ddot{\theta}_1$, Lagrangian multipliers λ_i for $i = 1, 2, 3$ can be obtained from (2).
4. Command signals τ_1 , τ_2 , and τ_3 for accomplishing the torque control of motors mounted on the fixed base, as shown in Figure 1b, can be determined from (1).

Command signals τ_i ($i = 1, 2, 3$) that are numerically calculated using (1) by the aforementioned procedure should perfectly match the corresponding actual torque τ_{actual}^i for guaranteeing that the centroid P of the moving platform perfectly tracks the three-dimensional trajectory P . In other words, an imprecise τ_i value can never provide high

tracking performance for the predesigned trajectory of P . Therefore, a thorough similarity evaluation between τ_i and τ_{actual}^i is an essential prerequisite for the practical application of delta robots. In this respect, the accuracy of τ_i obtained using (1) with the parameter values provided by the manufacturer, as listed in Table 1, is evaluated as follows. The actual torque τ_{actual}^i that is delivered by the implemented motor to the actuated arm cannot be directly determined. Thus, the well-known fact that the torque is proportional to the square of the operating voltage (i.e., $\tau_{\text{actual}}^i \approx (v_{\text{actual}}^i)^2$) was introduced in the following experiments. First, the user-selected v_{actual}^i ($i = 1, 2, 3$) was supplied to each motor to activate the parallel-link robot. The dashed line in Figure 3a indicates such a v_{actual}^1 value that is approximately equal to the square root of τ_{actual}^1 . Second, the active joint angle θ_{i1} was directly measured by sensors during robot operation. Angular velocity $\dot{\theta}_{i1}$ and acceleration $\ddot{\theta}_{i1}$ were calculated from the joint angle data by using central finite difference methods. Lastly, the computed command signal τ_i in (1) was obtained by using the parameter values listed in Table 1 and the θ_{i1} , $\dot{\theta}_{i1}$, and $\ddot{\theta}_{i1}$ values. The solid line in Figure 3a denotes the computed voltage that is equivalent to the square root of the computed torque τ_1 . The discrepancy between τ_{actual}^1 and τ_1 in Figure 3a implies that the amount of torque exerted by the motor with a computed voltage input $v_{\text{computed}}^i (= \sqrt{\tau_i})$ cannot achieve the high-precision tracking of the predetermined reference trajectory P . By contrast, the reference-tracking errors may become small upon the implementation of a suitably designed controller. For example, inverse dynamic control or computed torque control is one of the most effective, well-accepted, and widely used control schemes for rigid robotic manipulator systems to drive the manipulator along the given trajectory as precisely as possible [12]. The key idea of such a control scheme is to introduce feedback linearization to cancel the nonlinear robot dynamics under the assumption that all parameters of the system model are exactly known. Therefore, the inexactness of the model parameters, which can be verified from the discrepancy between τ_{actual}^1 and τ_1 in Figure 3a, results in the degradation of robot performance, even when a model-based motion controller is implemented. Accordingly, the development and identification of a dynamic analytic model of the delta robot are indispensable for calculating actuator torques with high precision. In addition, such an exact dynamic model can contribute to reducing the control effort of model-based controllers required for the trajectory-tracking behavior of robot manipulators.

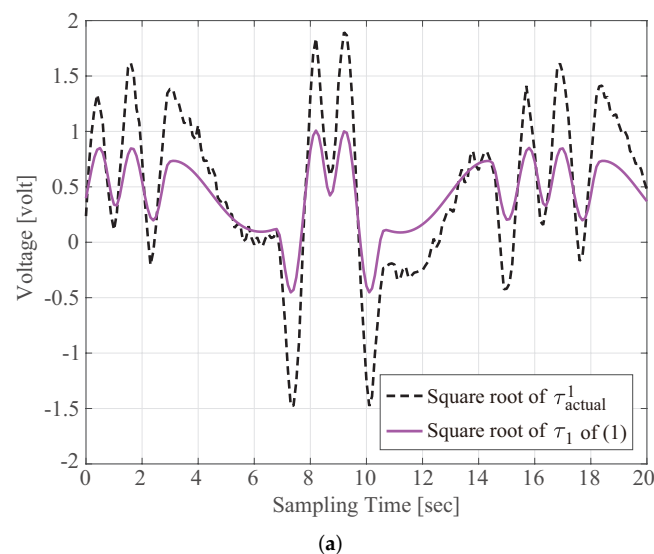


Figure 3. Cont.

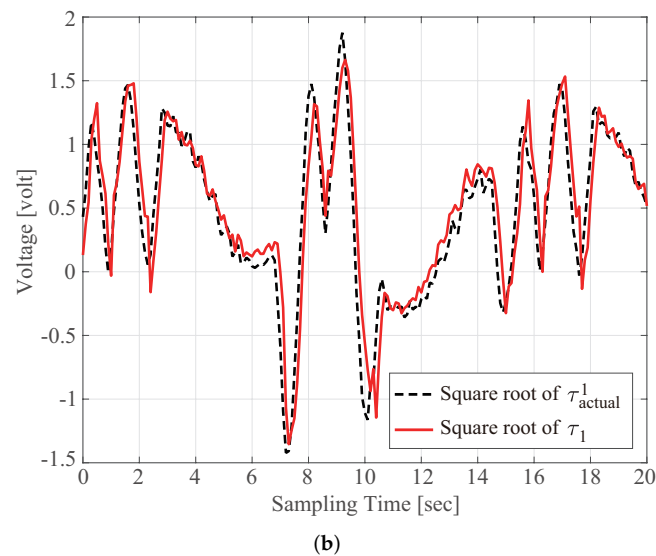


Figure 3. Measured actual torque τ_{actual}^1 is compared with (a) torque $\tau(1)$ calculated using the parameters listed in Table 1 and (b) torque $\tau_1(k)$ computed using (26) with the estimated parameters in (36). (a) The behavior of $\tau_1(k)$ computed using (1) with the parameters listed in Table 1. (b) The behavior of $\tau_1(k)$ computed using (26) with the parameters listed in (36).

3. Construction of an Identification-Based Dynamic Model of the Delta Robot

The analytic model structure of the delta robot shown in Figures 1 and 2b was derived using the virtual work principle as follows. Vector F_{GP} for the gravitational force of the moving platform is given as follows:

$$F_{GP} = \tilde{m}_P [0 \quad 0 \quad g]^T, \quad (3)$$

where g denotes the gravitational acceleration. $\tilde{m}_P$ denotes the equivalent mass of the moving platform and is defined as $\tilde{m}_P \approx m_p + 3m_d$, where m_p and m_d are the masses of the moving platform and the driven arm, respectively. Vector F_{IP} for the inertia force of the moving platform is obtained as follows:

$$F_{IP} = \tilde{m}_P \ddot{\mathbf{P}} = \tilde{m}_P [\ddot{p}_x \quad \ddot{p}_y \quad \ddot{p}_z]^T. \quad (4)$$

Further, vector M_{GA} for the gravity torque of the actuated arm is given as follows:

$$M_{GA} = \tilde{m}_A g L_1 [\cos \theta_{11} \quad \cos \theta_{21} \quad \cos \theta_{31}]^T, \quad (5)$$

where L_1 denotes the length of the actuated arm. $\tilde{m}_A$ denotes the equivalent mass of the actuated arm and is defined as $\tilde{m}_A \approx m_a + 0.5m_d$, where m_a is the mass of the actuated arm. Vector M_{IA} for the inertia torque of the actuated arm is

$$M_{IA} = I_A \ddot{\boldsymbol{\theta}}_1 = I_A [\ddot{\theta}_{11} \quad \ddot{\theta}_{21} \quad \ddot{\theta}_{31}]^T, \quad (6)$$

where I_A denotes the inertia matrix of the actuated arm with respect to the fixed frame and is approximately given as follows:

$$I_A = \begin{bmatrix} I_1 & 0 & 0 \\ 0 & I_2 & 0 \\ 0 & 0 & I_3 \end{bmatrix} \approx \tilde{m}_A L_1^2 \mathbf{I}, \quad (7)$$

where $\mathbf{I}$ is an identity matrix of order 3. Let the vector of the virtual angular displacements of joints A_1 , A_2 , and A_3 be denoted by $\delta\boldsymbol{\theta}_1 = [\delta\theta_{11} \quad \delta\theta_{21} \quad \delta\theta_{31}]^T$. The vector of the virtual

linear displacement of the moving platform is denoted by $\delta \mathbf{P} = [\delta P_x \ \delta P_y \ \delta P_z]^T$. Then, the following dynamic model can be derived by applying the virtual work principle [13]:

$$\boldsymbol{\tau}^T \delta \boldsymbol{\theta}_1 + \mathbf{M}_{GA}^T \delta \boldsymbol{\theta}_1 + \mathbf{F}_{GP}^T \delta \mathbf{P} - \mathbf{M}_{IA}^T \delta \boldsymbol{\theta}_1 - \mathbf{F}_{IP}^T \delta \mathbf{P} = 0, \quad (8)$$

where $\boldsymbol{\tau} = [\tau_1 \ \tau_2 \ \tau_3]^T$. Next, $\delta \mathbf{P}$ is defined in terms of $\delta \boldsymbol{\theta}_1$ as follows. The relationship between the velocity of the centroid of the moving platform, $\dot{\mathbf{P}}$, and the actuated joint velocity, $\dot{\boldsymbol{\theta}}_1$, can be written as follows:

$$\dot{\mathbf{P}} = \mathbf{J} \dot{\boldsymbol{\theta}}_1, \quad (9)$$

where $\mathbf{J}$ denotes the Jacobian matrix, which is derived as follows. Consider the following constraint condition that can be confirmed easily from Figure 2b [14]: for $i = 1, 2$, and 3,

$$\|\overrightarrow{B_i C_i}\|^2 - L_2^2 = 0, \quad (10)$$

where L_2 denotes the length of the driven arm. Let $\mathbf{s}_i$ be the vector $\overrightarrow{B_i C_i}$ written as

$$\mathbf{s}_i = \overrightarrow{OP} - (\overrightarrow{OA_i} - \overrightarrow{PC_i} + \overrightarrow{A_i B_i}) = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} - \begin{bmatrix} c(\phi_i) & -s(\phi_i) & 0 \\ s(\phi_i) & c(\phi_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \left(\begin{bmatrix} R-r \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} L_1 c(\theta_{i1}) \\ 0 \\ L_1 s(\theta_{i1}) \end{bmatrix} \right), \quad (11)$$

where R and r are the radii of the fixed base and moving platform, respectively. Then, (10) can be rewritten as

$$\mathbf{s}_i^T \mathbf{s}_i - L_2^2 = 0 \quad (12)$$

The time derivative of (12) becomes

$$\mathbf{s}_i^T \dot{\mathbf{s}}_i + \dot{\mathbf{s}}_i^T \mathbf{s}_i = 0, \quad (13)$$

where

$$\dot{\mathbf{s}}_i = \underbrace{\begin{bmatrix} \dot{P}_x \\ \dot{P}_y \\ \dot{P}_z \end{bmatrix}}_{\dot{\mathbf{P}}} - \underbrace{\begin{bmatrix} c(\phi_i) & -s(\phi_i) & 0 \\ s(\phi_i) & c(\phi_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -L_1 s(\theta_{i1}) \\ 0 \\ L_1 c(\theta_{i1}) \end{bmatrix}}_{\boldsymbol{\Pi}_i} \dot{\theta}_{i1} = \dot{\mathbf{P}} - \boldsymbol{\Pi}_i \dot{\theta}_{i1} \quad (14)$$

Then, from (14) and the commutative property for vector inner products, (13) becomes equivalent to

$$\mathbf{s}_i^T \dot{\mathbf{s}}_i = \mathbf{s}_i^T (\dot{\mathbf{P}} - \boldsymbol{\Pi}_i \dot{\theta}_{i1}) = \mathbf{s}_i^T \dot{\mathbf{P}} - \mathbf{s}_i^T \boldsymbol{\Pi}_i \dot{\theta}_{i1} = 0 \quad (15)$$

Therefore, (9) can be derived from (15) as follows:

$$\dot{\mathbf{P}} = \underbrace{\begin{bmatrix} \mathbf{s}_1^T \\ \mathbf{s}_2^T \\ \mathbf{s}_3^T \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{s}_1^T \boldsymbol{\Pi}_1 & 0 & 0 \\ 0 & \mathbf{s}_2^T \boldsymbol{\Pi}_2 & 0 \\ 0 & 0 & \mathbf{s}_3^T \boldsymbol{\Pi}_3 \end{bmatrix}}_{\mathbf{J}} \dot{\boldsymbol{\theta}}_1 = \mathbf{J} \dot{\boldsymbol{\theta}}_1 \quad (16)$$

$\mathbf{s}_i$, defined as (11), depends not only on ϕ_i and θ_{i1} , but also on the three-dimensional coordinate $\mathbf{P} = [P_x \ P_y \ P_z]^T$ of the terminal moving platform. Such a positional coordinate can be obtained by measuring the actuated joint angles θ_{i1} ($i = 1, 2, 3$), as described in Appendix A. The derivative of $\mathbf{P}$ and $\boldsymbol{\theta}_1$ with respect to time t can be represented as the ratio of an infinitesimal change in time-varying variables. From this fact, it follows that (16) can be expressed as

$$\delta \mathbf{P} = \mathbf{J} \delta \boldsymbol{\theta}_1 \quad (17)$$

Thus, substituting (17) into (8) gives

$$\boldsymbol{\tau}^T \delta \boldsymbol{\theta}_1 + \mathbf{M}_{GA}^T \delta \boldsymbol{\theta}_1 + \mathbf{F}_{GP}^T \mathbf{J} \delta \boldsymbol{\theta}_1 - \mathbf{M}_{IA}^T \delta \boldsymbol{\theta}_1 - \mathbf{F}_{IP}^T \mathbf{J} \delta \boldsymbol{\theta}_1 = (\boldsymbol{\tau}^T + \mathbf{M}_{GA}^T + \mathbf{F}_{GP}^T \mathbf{J} - \mathbf{M}_{IA}^T - \mathbf{F}_{IP}^T \mathbf{J}) \delta \boldsymbol{\theta}_1 = 0 \quad (18)$$

Because (18) holds for any virtual displacement $\delta\theta_1$,

$$\tau = M_{IA} + J^T F_{IP} - M_{GA} - J^T F_{GP} \quad (19)$$

By contrast, the dynamic model in (19) does not cover all the forces acting on the parallel-link robot; that is, (19) includes only some forces that can be readily analyzed from the rigid-body mechanics [15]. The important forces that are not considered in the aforementioned derivation are those due to friction. Therefore, the aforementioned dynamic model should be extended to at least roughly include the frictional forces in order to render such a model reflect the reality of the physical behavior of the robot and thereby improve the model quality. First, the mathematical model of viscous friction in which each torque in τ_v caused by frictional force is proportional to the corresponding actuated joint velocity in $\dot{\theta}_1$, is given as follows:

$$\tau_v = \underbrace{\begin{bmatrix} c_{v1} & 0 & 0 \\ 0 & c_{v2} & 0 \\ 0 & 0 & c_{v3} \end{bmatrix}}_{C_v} \begin{bmatrix} \dot{\theta}_{11} \\ \dot{\theta}_{21} \\ \dot{\theta}_{31} \end{bmatrix} = C_v \dot{\theta}_1, \quad (20)$$

where c_{v_i} ($i = 1, 2, 3$) are viscous friction coefficients. Second, the Coulomb friction model is represented as

$$\tau_c = \underbrace{\begin{bmatrix} c_{c1} & 0 & 0 \\ 0 & c_{c2} & 0 \\ 0 & 0 & c_{c3} \end{bmatrix}}_{C_c} \begin{bmatrix} \text{sgn}(\dot{\theta}_{11}) \\ \text{sgn}(\dot{\theta}_{21}) \\ \text{sgn}(\dot{\theta}_{31}) \end{bmatrix} = C_c \text{sgn}(\dot{\theta}_1), \quad (21)$$

where c_{c_i} ($i = 1, 2, 3$) are Coulomb friction coefficients, and $\text{sgn}(\cdot)$ denotes a signum function. This model shows that each torque in τ_c due to Coulomb friction is constant except for the sign, which depends on the actuated joint velocity. Therefore, the extended formula for joint torques that considers the forces due to friction can be derived from (19), (20), and (21) as follows:

$$\tau = M_{IA} + J^T F_{IP} - M_{GA} - J^T F_{GP} - \tau_v - \tau_c \quad (22)$$

Because $\dot{P} = J\dot{\theta}_1$, as shown in (16), it holds that

$$\ddot{P} = J\ddot{\theta}_1 + \dot{J}\dot{\theta}_1, \quad (23)$$

where

$$\dot{J} = \begin{bmatrix} s_1^T \\ s_2^T \\ s_3^T \end{bmatrix}^{-1} \left(\begin{bmatrix} \dot{s}_1^T \Pi_1 + s_1^T \dot{\Pi}_1 & 0 & 0 \\ 0 & \dot{s}_2^T \Pi_2 + s_2^T \dot{\Pi}_2 & 0 \\ 0 & 0 & \dot{s}_3^T \Pi_3 + s_3^T \dot{\Pi}_3 \end{bmatrix} - \begin{bmatrix} \dot{s}_1^T \\ \dot{s}_2^T \\ \dot{s}_3^T \end{bmatrix} J \right) \quad (24)$$

Then, from (3)–(6), (20), (21), and (23), (22) can be rewritten as follows:

$$\begin{aligned} \tau &= I_A \ddot{\theta}_1 + J^T \ddot{m}_P \ddot{P} - \underbrace{\ddot{m}_A g L_1}_{T_g} \begin{bmatrix} c(\theta_{11}) \\ c(\theta_{21}) \\ c(\theta_{31}) \end{bmatrix} - J^T \ddot{m}_P \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} - C_v \dot{\theta}_1 - C_c \text{sgn}(\dot{\theta}_1) \\ &= I_A \ddot{\theta}_1 + J^T \ddot{m}_P J \ddot{\theta}_1 + J^T \ddot{m}_P \dot{J} \dot{\theta}_1 - T_g \begin{bmatrix} c(\theta_{11}) \\ c(\theta_{21}) \\ c(\theta_{31}) \end{bmatrix} - J^T \ddot{m}_P \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} - C_v \dot{\theta}_1 - C_c \text{sgn}(\dot{\theta}_1), \end{aligned} \quad (25)$$

where $T_g := \ddot{m}_A g L_1$. In conclusion, (25) results in the following form of the joint space dynamic model:

$$\tau = D(\theta) \ddot{\theta}_1 + C(\theta, \dot{\theta}) \dot{\theta}_1 + N(\theta, \dot{\theta}), \quad (26)$$

where

$$D(\theta) = \begin{bmatrix} I_1 & 0 & 0 \\ 0 & I_2 & 0 \\ 0 & 0 & I_3 \end{bmatrix} + J^T \tilde{m}_P J, \quad (27)$$

$$C(\theta, \dot{\theta}) = J^T \tilde{m}_P \dot{J} - \begin{bmatrix} c_{v1} & 0 & 0 \\ 0 & c_{v2} & 0 \\ 0 & 0 & c_{v3} \end{bmatrix}, \quad (28)$$

$$N(\theta, \dot{\theta}) = -J^T \tilde{m}_P \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} - T_g \begin{bmatrix} c(\theta_{11}) \\ c(\theta_{21}) \\ c(\theta_{31}) \end{bmatrix} - \begin{bmatrix} c_{c1} & 0 & 0 \\ 0 & c_{c2} & 0 \\ 0 & 0 & c_{c3} \end{bmatrix} \begin{bmatrix} \text{sgn}(\dot{\theta}_{11}) \\ \text{sgn}(\dot{\theta}_{21}) \\ \text{sgn}(\dot{\theta}_{31}) \end{bmatrix}, \quad (29)$$

where $D(\theta) \in \mathbb{R}^{3 \times 3}$ is the inertia matrix, $C(\theta, \dot{\theta}) \in \mathbb{R}^{3 \times 3}$ is the matrix of viscous friction coefficients and Coriolis forces, and $N(\theta, \dot{\theta}) \in \mathbb{R}^3$ is the vector of gravity forces and Coulomb friction forces.

The PSO algorithm was inspired by the social and biological behaviors of bird flocks searching for food sources. In this nature-based algorithm, individuals are referred to as particles and fly through the search space seeking the best global position that minimizes (or maximizes) a given problem. The classical PSO algorithm is summarized in Appendix C. Next, the direct estimation of parameters in the derived dynamic model (26) with (27)–(29) using the PSO algorithm in combination with a distributed cyclic neighborhood search, which is a new variant of the aforementioned classical PSO algorithm, is described. Let the vector of the uncertain model parameters to be identified be defined as

$$\mathbf{M}_{\text{PAR}} := [\tilde{m}_p \quad T_g \quad I_1 \quad I_2 \quad I_3 \quad c_{v1} \quad c_{v2} \quad c_{v3} \quad c_{c1} \quad c_{c2} \quad c_{c3}]^T \in \mathbb{R}^{11} \quad (30)$$

Let $\mathbf{M}_{\text{PAR}_i}^\ell \in \mathbb{R}^{11}$ represent the position vector of each particle in the swarm at the iteration step $\ell (= 0, 1, 2, \dots, \ell_{\max})$, where $\ell_{\max}$ denotes the maximum number of iterations and $i (= 1, 2, \dots, n_p)$ denotes the particle index. The vector of the sampled measurement of the supplied torques, $\tau_{\text{actual}}(t) = [\tau_{\text{actual}}^1(t) \quad \tau_{\text{actual}}^2(t) \quad \tau_{\text{actual}}^3(t)]^T$, is denoted by $\tau_{\text{actual}}(k) = [\tau_{\text{actual}}^1(k) \quad \tau_{\text{actual}}^2(k) \quad \tau_{\text{actual}}^3(k)]^T$, where $k = 1, 2, \dots, N_k$. Further, the sampled measurements of $\theta_1(t)$, $\dot{\theta}_1(t)$, and $\ddot{\theta}_1(t)$ are denoted by $\theta_1(k)$, $\dot{\theta}_1(k)$, and $\ddot{\theta}_1(k)$, respectively. Then, the estimation of $\tau_{\text{actual}}(k)$ is denoted by $\hat{\tau}(k) = [\hat{\tau}_1(k) \quad \hat{\tau}_2(k) \quad \hat{\tau}_3(k)]^T$. The identification process for developing the dynamic model of a delta robot involves the following steps.

Step 0. Initialize the iteration number as $\ell = 0$. The position vectors of N_p particles are initialized with randomly chosen $\mathbf{M}_{\text{PAR}_i}^0 \in \mathbb{R}^{11}$. Then, $\mathbf{M}_{\text{PAR}_{\text{pbest},i}}^0$ and $\mathbf{M}_{\text{PAR}_{\text{sbest},i}}^0$ are, respectively, set as follows:

$$\mathbf{M}_{\text{PAR}_{\text{pbest},i}}^0 \leftarrow \mathbf{M}_{\text{PAR}_i}^0 \quad (31)$$

$$\mathbf{M}_{\text{PAR}_{\text{sbest},i}}^0 \leftarrow \arg \min_{\mathbf{M}_{\text{PAR}} \in \{\mathbf{M}_{\text{PAR}_j}^0 | j=i-\frac{N_s}{2}, \dots, i+\frac{N_s}{2}\}} \mathcal{F}(\mathbf{M}_{\text{PAR}}) \quad (32)$$

where the predefined even-valued $N_s (< N_p)$ is the number of neighbors of the i th particle, $\mathbf{M}_{\text{PAR}_j}^\ell := \mathbf{M}_{\text{PAR}_{(j-1 \bmod N_p)+1}}^\ell$ for $j < 1$ or $N_p + 1 \leq j$. Further, $\mathcal{F}(\mathbf{M}_{\text{PAR}})$ is the objective function that is defined as follows:

$$\mathcal{F}(\mathbf{M}_{\text{PAR}_i}^\ell) := \sum_{\zeta=1}^3 \sum_{k=0}^{N_k} |\tau_{\text{actual}}^\zeta(k) - \hat{\tau}_\zeta(\mathbf{M}_{\text{PAR}_i}^\ell; k)| \quad (33)$$

where the ζ th torque estimation denoted by $\hat{\tau}_\zeta(\mathbf{M}_{\text{PAR}_i}^\ell; k)$ is calculated from (26)–(29) using the identified parameters in $\mathbf{M}_{\text{PAR}_i}^\ell$ and the discrete measurement data of

$\theta_1(k)$, $\dot{\theta}_1(k)$, and $\ddot{\theta}_1(k)$. The aim of the objective function in (33) is to minimize the deviation between the measurement of the supplied torques and their estimation by employing the joint space dynamic model in (26) with (27)–(29) at each sampling time.

Step 1. Once the termination criterion, set as the total number of iterations performed, is satisfied, the identification process is terminated with the optimal parameters given by

$$\mathbf{M}_{\text{PAR}}^* := \arg \min_{\mathbf{M}_{\text{PAR}} \in \{\mathbf{M}_{\text{PAR}_i}^j | i=1, \dots, N_p; j=0, \dots, \ell\}} \mathcal{F}(\mathbf{M}_{\text{PAR}})$$

Otherwise, the optimization process continues until the termination criterion is satisfied.

Step 2. Each particle in the swarm evolves according to the following equation:

$$\hat{\mathbf{M}}_{\text{PAR}_i}^{\ell+1} \leftarrow c_0 \hat{\mathbf{M}}_{\text{PAR}_i}^{\ell} + c_1 r_{1,i}^{\ell} (\mathbf{M}_{\text{PAR}_{\text{pbest},i}}^{\ell} - \mathbf{M}_{\text{PAR}_i}^{\ell}) + c_2 r_{2,i}^{\ell} (\mathbf{M}_{\text{PAR}_{\text{sbest},i}}^{\ell} - \mathbf{M}_{\text{PAR}_i}^{\ell}) \quad (34)$$

where c_0 , c_1 , and c_2 are the inertial factor, cognitive-scaling factor, and social-scaling factor, respectively; $r_{1,i}^{\ell}$ and $r_{2,i}^{\ell}$ are uniformly distributed random numbers generated separately in the unit interval $[0, 1]$ at each iteration step ℓ ; and the initial $\hat{\mathbf{M}}_{\text{PAR}_i}^{\ell}$ equals $\hat{\mathbf{M}}_{\text{PAR}_i}^0 = \mathbf{0} \in \mathbb{R}^{11}$. Then, $\mathbf{M}_{\text{PAR}_i}^{\ell+1}$ is updated as follows:

$$\mathbf{M}_{\text{PAR}_i}^{\ell+1} \leftarrow \mathbf{M}_{\text{PAR}_i}^{\ell} + \hat{\mathbf{M}}_{\text{PAR}_i}^{\ell+1} \quad (35)$$

Lastly, $\mathbf{M}_{\text{PAR}_{\text{pbest},i}}^{\ell+1}$ and $\mathbf{M}_{\text{PAR}_{\text{sbest},i}}^{\ell+1}$ are, respectively, updated as follows:

$$\begin{aligned} \mathbf{M}_{\text{PAR}_{\text{pbest},i}}^{\ell+1} &\leftarrow \arg \min_{\mathbf{M}_{\text{PAR}_i} \in \{\mathbf{M}_{\text{PAR}_i}^j | j=0, 1, \dots, \ell\}} \mathcal{F}(\mathbf{M}_{\text{PAR}}) \\ \mathbf{M}_{\text{PAR}_{\text{sbest},i}}^{\ell+1} &\leftarrow \arg \min_{\mathbf{M}_{\text{PAR}_i} \in \{\mathbf{M}_{\text{PAR}_i}^j | j=i-\frac{n_s}{2}, \dots, i+\frac{n_s}{2}\}} \mathcal{F}(\mathbf{M}_{\text{PAR}}) \end{aligned}$$

Next, go to Step 1 with $\ell \leftarrow \ell + 1$.

The task is now to determine the optimal parameter vector $\mathbf{M}_{\text{PAR}}^*$ of $\mathbf{M}_{\text{PAR}}$ in (30) that renders the torque τ computed from (26) with $\mathbf{M}_{\text{PAR}}^*$ as close as possible to the applied actual torque τ_{actual} obtained from N_k measurements from $(\theta(1), \dot{\theta}(1), \ddot{\theta}(1))$ to $(\theta(N_k), \dot{\theta}(N_k), \ddot{\theta}(N_k))$. Such a parameter estimation is performed by using the aforementioned PSO-based optimization algorithm and the dataset of $[\theta(k), \dot{\theta}(k), \ddot{\theta}(k)]$ used for the comparison illustrated in Figure 3a. Figure 4 shows a schematic overview of the experimental setup designed at NT Robot, Co., to measure the actual torque $\tau_{\text{actual}}(k)$ and $[\theta(k), \dot{\theta}(k), \ddot{\theta}(k)]$ and update the dynamic model parameters via the proposed identification procedure. These data are acquired when the moving platform follows the trajectory shown in Figure 5a. The optimization task for parameter estimation is performed with $c_0 = 0.72984$, $c_1 = c_2 = 1.49618$, $N_p = 300$, $N_s = 60$, and the maximum iteration number of PSO $\ell_{\text{max}} = 1000$. Then, the identified parameters of the dynamic model (26) with (27)–(29) are obtained as follows:

$$\begin{aligned} \mathbf{M}_{\text{PAR}}^* &= [\tilde{m}_p^*, T_g^*, I_1^*, I_2^*, I_3^*, c_{v_1}^*, c_{v_2}^*, c_{v_3}^*, c_{c_1}^*, c_{c_2}^*, c_{c_3}^*]^T \\ &\approx [4.3963, 5.3480, 0.3823, 0.3237, 0.3047, 3.2111, 1.1688, 3.1343, -9.1037, -7.6127, -9.2628]^T \end{aligned} \quad (36)$$

The solid line in Figure 3b shows the characteristics of the computed torque determined from (26) with the estimated parameters in (36). A comparison with the measured actual torque $\tau_{\text{actual}}^i(k)$ indicates the good fidelity of the derived dynamic model providing $\tau_i(k)$. This high model fidelity is attributed to the PSO-based identification scheme introduced for nonlinear dynamic modeling (26) that includes the forces due to viscous and Coulomb frictions.

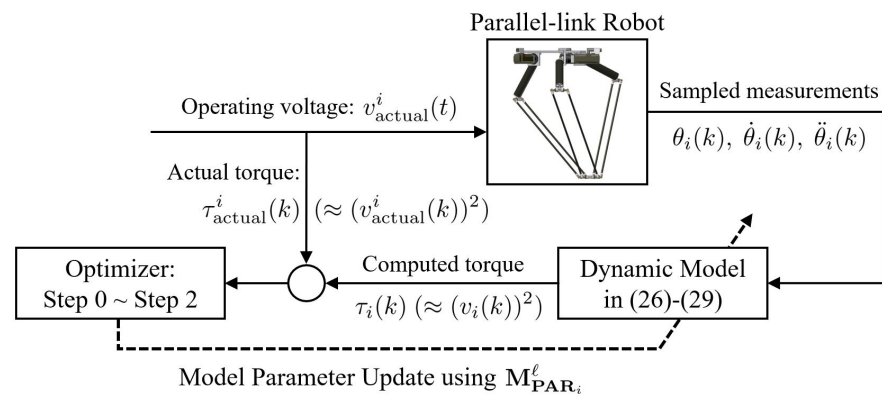


Figure 4. Schematic overview of the system structure for identification of the dynamic analytic model.

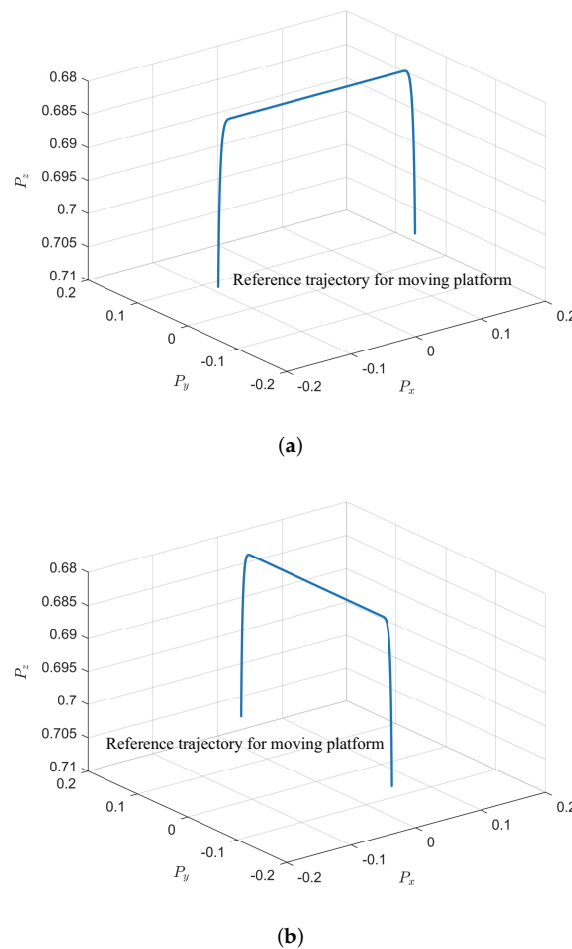


Figure 5. Reference trajectories of moving platform used for identification of the dynamic analytic model and model validation. (a) Reference trajectory for model parameter identification. (b) Reference trajectory for model validation.

In summary, our study presented a simple and efficient approach that can directly achieve high-quality identification of an analytic dynamic model for a parallel-link robot. The phrase “simple and efficient” can be characterized from two aspects. First, PSO itself has the advantages of fast search speed, memory, limited parameters, and simple structure. In addition, it is easier to implement at the validation stage than other optimization algorithms are. In addition, the use of PSO facilitates generalized cost functions, which enables the direct identification of a large number of model parameters involved in the analytic dynamic model of delta robots. Second, our identification procedure does not require

cumbersome and time-consuming tasks, such as model transformation or development of simulated manipulators used in conventional studies, which can be mainly attributed to the aforementioned flexibility of the PSO algorithm. Thus, our approach is simple and efficient and can directly identify the model parameters effectively. By contrast, the phrase, “high-quality identification”, can be explained as follows. Most robot-related studies that are not based on identification approaches use the physical data provided by manufacturers as model parameters. These physical parameter values often lead to a large discrepancy between the calculated and measured signals, as depicted in Figure 3a. However, with the application of our proposed approach for estimating model parameters, the model can be obtained with high accuracy. This result is experimentally demonstrated in Figure 6, wherein the discrepancy between the calculated and measured signals is considerably smaller than that in Figure 3a. This experimentally demonstrates that our approach enables the achievement of high-quality identification performance.

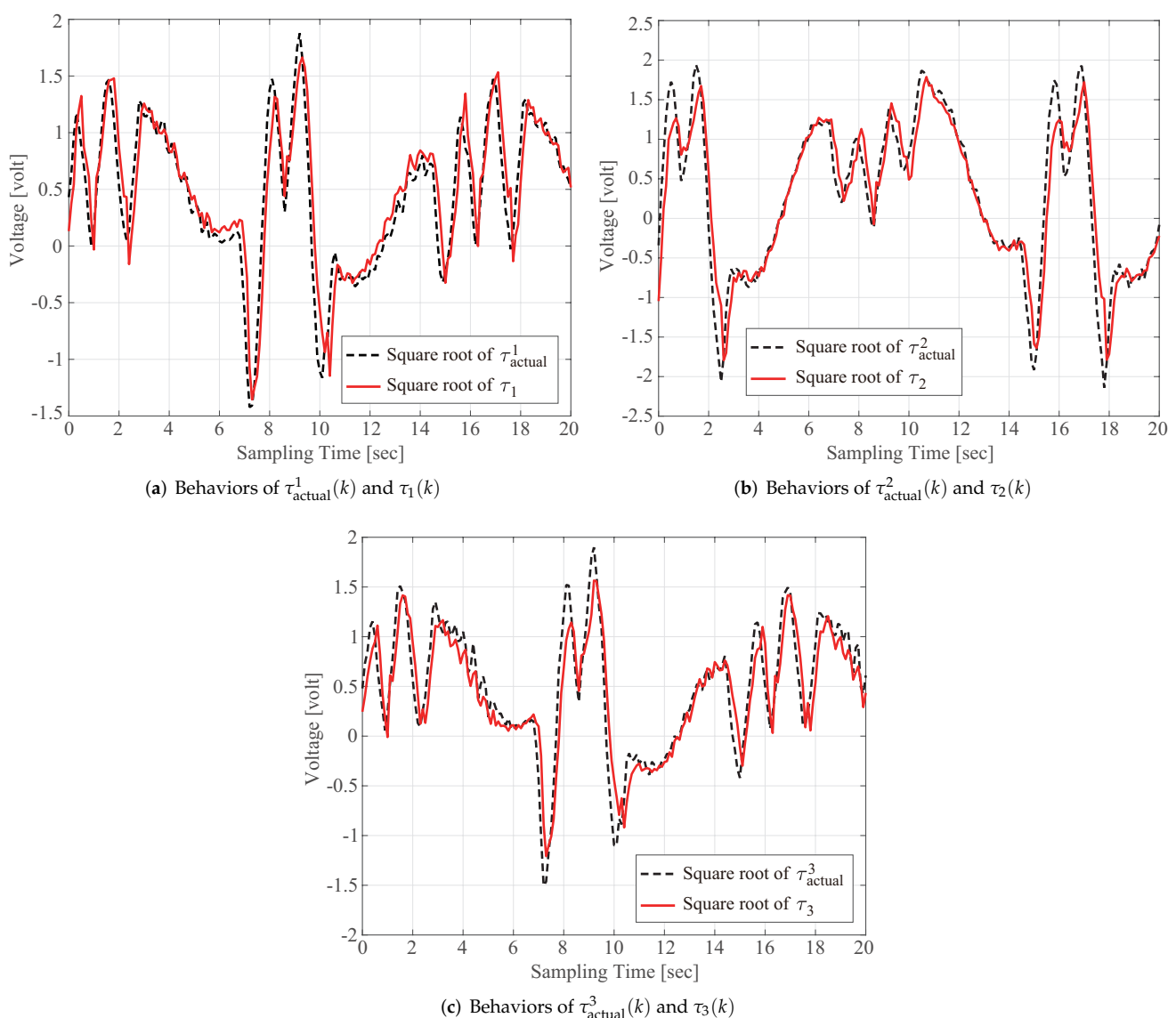


Figure 6. Comparison of the measured actual torque τ_{actual} and the torque τ computed using (26) with the identified parameter values in (36).

Remark 1. The performance degradation of the classical PSO algorithm may mainly be due to the poor particle diversification characteristics, which, in practical implementation, often leads to the potential problem of premature convergence of the swarm to local optima. Diversification enables

the optimizer to efficiently explore different regions, possibly resulting in a global search for a good optimum. However, most population-based evolutionary computation techniques, including PSO, depend on the principle of reducing the search space in the progression toward the global optima. Because of the swarm-movement pattern, after a certain number of evolutionary iterations, offspring that outperform their parents can no longer be produced; thus, all the particles remain trapped in a region that may not even contain the local optima. In such cases, if population diversity is not enhanced, particles evolving according to the conventional PSO mechanism may find reaching the true global optimal position challenging. On the basis of this background, the distributed cyclic neighborhood-search mechanism serves as a diversity-boosting tool in our PSO algorithm. This mechanism enables a particle to share information through a “nearby-neighbor” interaction with a series of successively numbered particles, with the particle as the center. In this structure, a particle’s nearby neighbors are not necessarily particles that are close to each other in the hyperdimensional search space. Instead, nearby neighbors are particles that share information on individual fitness values. Thus, the key improvement in the particles’ exploration abilities (or population diversity) in our PSO method can be attributed to their local social learning from their respective neighborhoods, rather than their learning from only one global best particle in the swarm, as in the canonical star topology.

Remark 2. Although the total number of particles, N_p , is selected through an empirical approach, a systematic method may be required for our PSO algorithm to be more successful. Recently, Kononova et al. [16] revealed and quantified structural bias that, if present, would predispose the algorithm toward limiting its search to specific regions of the solution space. Their result would be useful in determining the suitable swarm size in this PSO scheme. However, the system-identification problem for a delta robot is solved via a one-run optimization process, which implies that the dynamic model is obtained through off-line system identification. Therefore, compared to online identification requiring computational efficiency, the computational complexity may not be a critical issue in this approach. Accordingly, the maximal number of iterations was set as a termination criterion in our study.

4. Discussion

To demonstrate the superiority of the proposed identification scheme, the conventional identification of a parallel robot proposed by Angel and Viola [5] is introduced. Their techniques for the parametric identification of the analytical dynamic model were mainly based on the RLS algorithm. With the aforementioned aim, the nonlinear dynamic model for a delta robot is formulated in the form of the linear regression model. From (1), we have

$$\begin{aligned}\tau_1 &= \hat{\theta}_1 \ddot{\theta}_{11} - \hat{\theta}_2 P_x s(\theta_{11}) - \hat{\theta}_3 P_y s(\theta_{11}) + \hat{\theta}_4 P_z c(\theta_{11}) - \hat{\theta}_5 c(\theta_{11}) + \hat{\theta}_6 s(\theta_{11}), \\ \tau_2 &= \hat{\theta}_1 \ddot{\theta}_{21} - \hat{\theta}_7 P_x s(\theta_{21}) - \hat{\theta}_8 P_y s(\theta_{21}) + \hat{\theta}_9 P_z c(\theta_{21}) - \hat{\theta}_5 c(\theta_{21}) + \hat{\theta}_{10} s(\theta_{21}), \\ \tau_3 &= \hat{\theta}_1 \ddot{\theta}_{31} - \hat{\theta}_{11} P_x s(\theta_{31}) - \hat{\theta}_{12} P_y s(\theta_{31}) + \hat{\theta}_{13} P_z c(\theta_{31}) - \hat{\theta}_5 c(\theta_{31}) + \hat{\theta}_{14} s(\theta_{31}),\end{aligned}\quad (37)$$

where

$$\begin{aligned}\hat{\theta}_1 &:= \gamma^2 I_m + I_{L_1} + L_1^2 m_d, \quad \hat{\theta}_2 := 2L_1 \lambda_1 \cos \phi_1, \quad \hat{\theta}_3 := 2L_1 \lambda_1 \sin \phi_1, \quad \hat{\theta}_4 := 2L_1 \lambda_1, \\ \hat{\theta}_5 &:= m_d L_{1c} + m_d L_1, \quad \hat{\theta}_6 := 2L_1 \lambda_1 (R - r), \quad \hat{\theta}_7 := 2L_1 \lambda_2 \cos \phi_2, \quad \hat{\theta}_8 := 2L_1 \lambda_2 \sin \phi_2, \\ \hat{\theta}_9 &:= 2L_1 \lambda_2, \quad \hat{\theta}_{10} := 2L_1 \lambda_2 (R - r), \quad \hat{\theta}_{11} := 2L_1 \lambda_3 \cos \phi_3, \quad \hat{\theta}_{12} := 2L_1 \lambda_3 \sin \phi_3, \\ \hat{\theta}_{13} &:= 2L_1 \lambda_3, \quad \hat{\theta}_{14} := 2L_1 \lambda_3 (R - r).\end{aligned}\quad (38)$$

Then, the following linear regression model is derived from (37):

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} =: \boldsymbol{\tau} = \boldsymbol{\Psi} \hat{\boldsymbol{\theta}} =: \begin{bmatrix} \boldsymbol{\psi}_1 \\ \boldsymbol{\psi}_2 \\ \boldsymbol{\psi}_3 \end{bmatrix} \begin{bmatrix} \hat{\theta}_1 \\ \vdots \\ \hat{\theta}_{14} \end{bmatrix}\quad (39)$$

where $\psi_j^T \in \mathbb{R}^{14}$ and

$$\psi = \begin{bmatrix} \ddot{\theta}_{11} & -P_x s(\theta_{11}) & -P_y s(\theta_{11}) & P_z c(\theta_{11}) & -c(\theta_{11}) & s(\theta_{11}) & 0 & 0 \\ \ddot{\theta}_{21} & 0 & 0 & 0 & -c(\theta_{21}) & 0 & -P_x s(\theta_{21}) & -P_y s(\theta_{21}) \\ \ddot{\theta}_{31} & 0 & 0 & 0 & -c(\theta_{31}) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ P_z c(\theta_{21}) & s(\theta_{21}) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -P_x s(\theta_{31}) & -P_y s(\theta_{31}) & P_z c(\theta_{31}) & s(\theta_{31}) & 0 & 0 \end{bmatrix} \quad (40)$$

For the above linear regression model, the RLS algorithm presented in Islam and Bernstein [17] is applied, which yields the computed torque $\tau_{\text{RLS}}^i(k)$ ($i = 1, 2, 3$) shown in Figure 7, where τ_{PSO}^i denotes the torque computed using our identification results. These experimental results clearly demonstrate that our identification scheme outperforms the conventional RLS-based method.

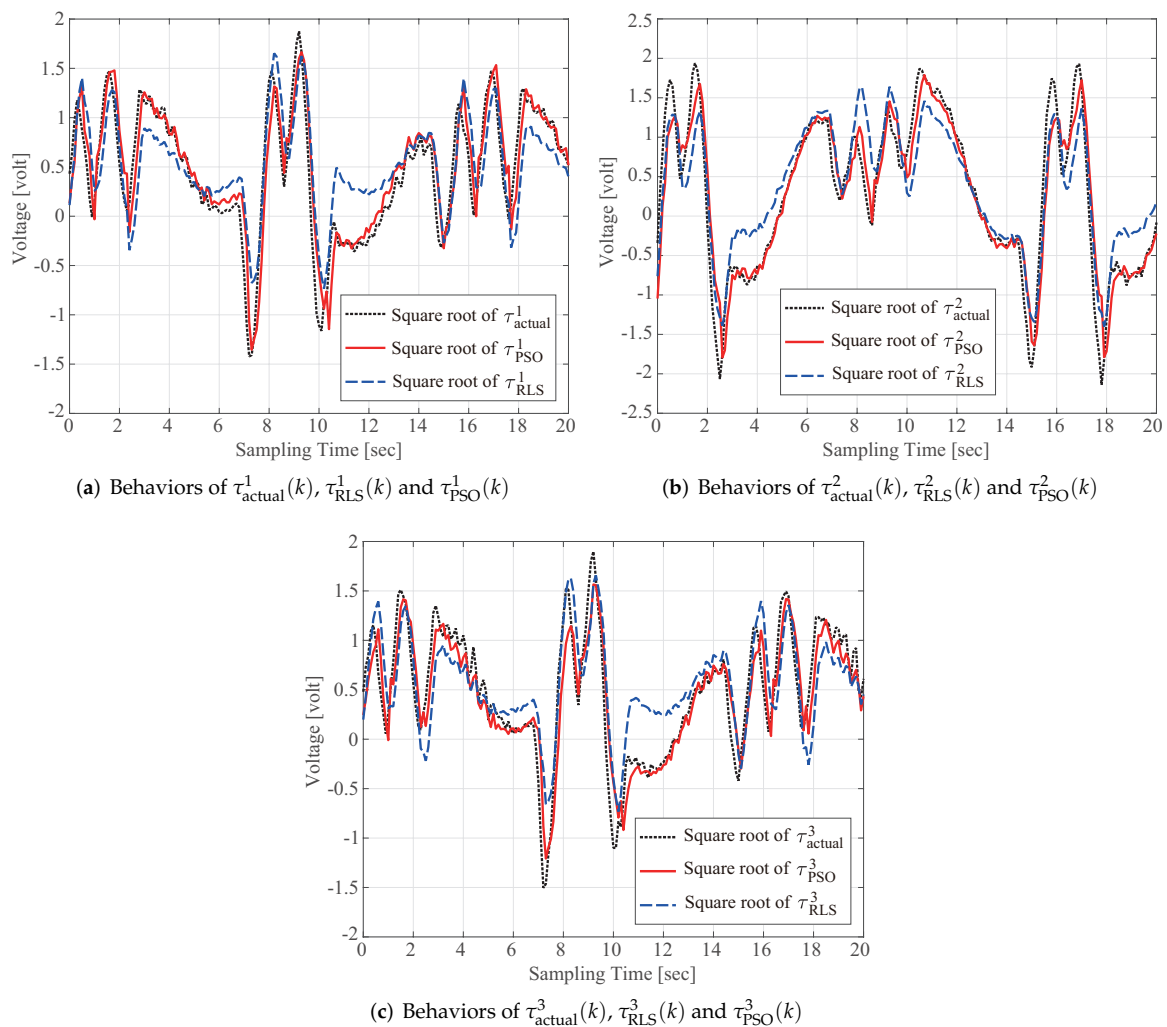
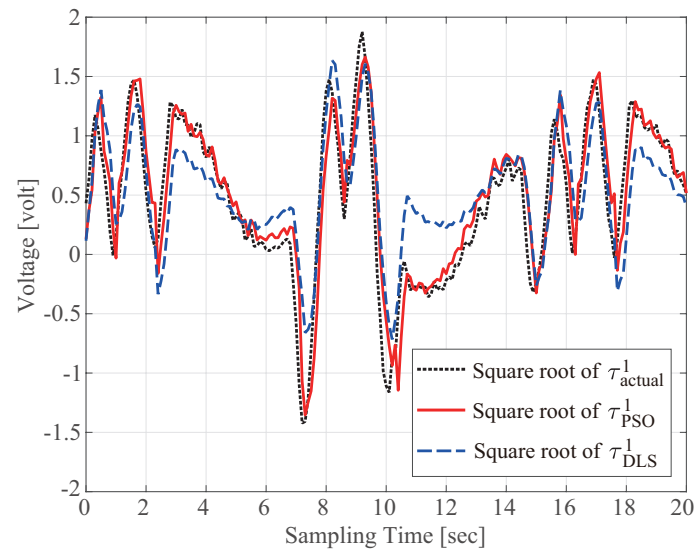


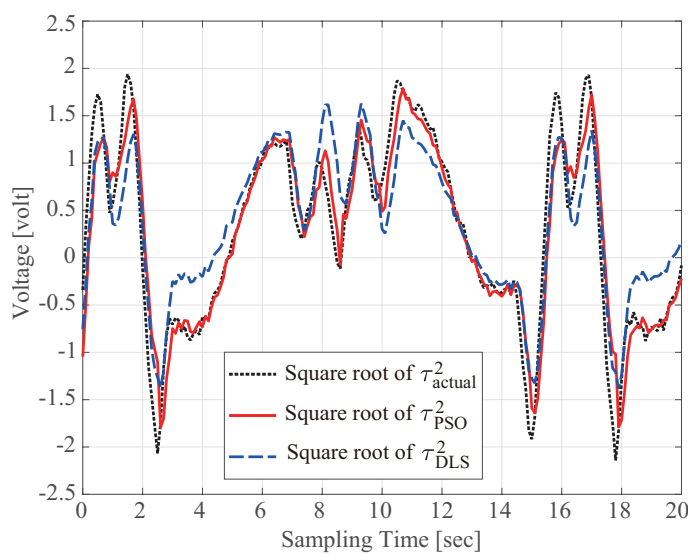
Figure 7. Comparison of the measured actual torque τ_{actual} , the torque $\tau_{\text{RLS}}^i(k)$ obtained from RLS-based identification, and the torque $\tau_{\text{PSO}}^i(k)$ computed using (26) with the identified parameter values in (36).

To further validate the high-quality identification performance of the proposed scheme, two other identification methods are examined. The first is the damped least-squares (DLS) method-based identification [18], and the second is the SVD method for least-squares identification [19]. Figures 8 and 9 present the computed torques $\tau_{\text{DLS}}^i(k)$ and $\tau_{\text{SVD}}^i(k)$, respec-

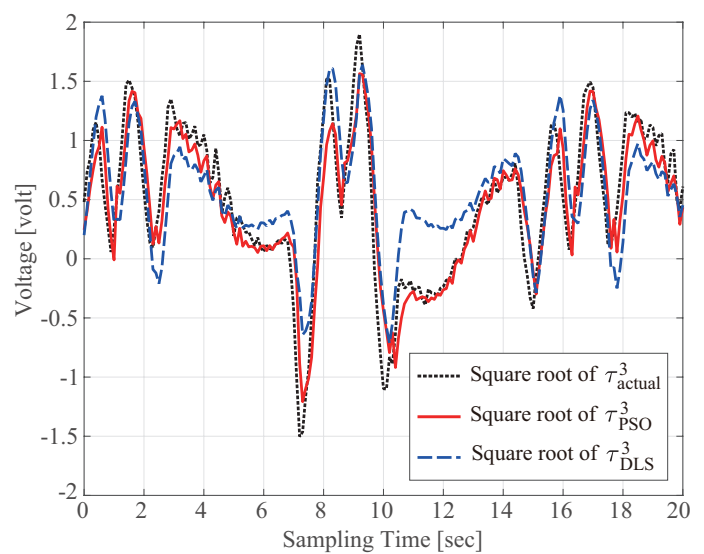
tively, for $i = 1, 2, 3$. The above examinations clearly verify that the proposed identification scheme for a delta robot achieved better performance compared with the identification results obtained via conventional DLS- and SVD-based identification methods.



(a) Behaviors of $\tau_{actual}^1(k)$, $\tau_{DLS}^1(k)$ and $\tau_{PSO}^1(k)$



(b) Behaviors of $\tau_{actual}^2(k)$, $\tau_{DLS}^2(k)$ and $\tau_{PSO}^2(k)$



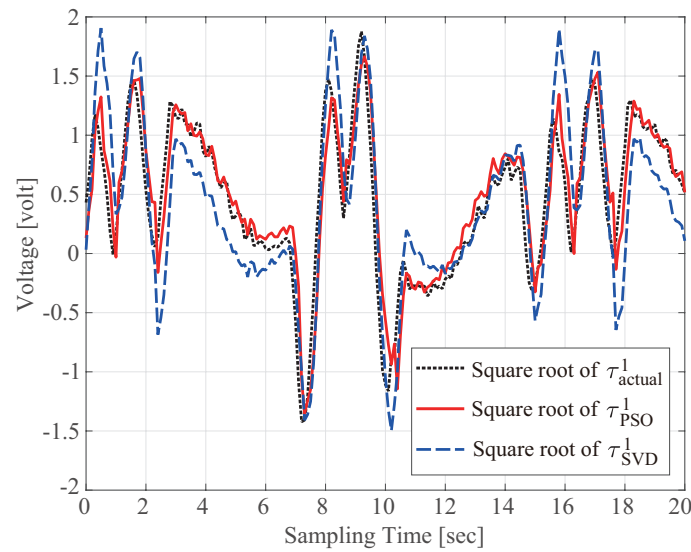
(c) Behaviors of $\tau_{actual}^3(k)$, $\tau_{DLS}^3(k)$ and $\tau_{PSO}^3(k)$

Figure 8. Comparison of the measured actual torque τ_{actual} , the torque $\tau_{DLS}^i(k)$ obtained from damped least-squares (DLS)-based identification, and the torque $\tau_{PSO}^i(k)$ computed using (26) with the identified parameter values in (36).

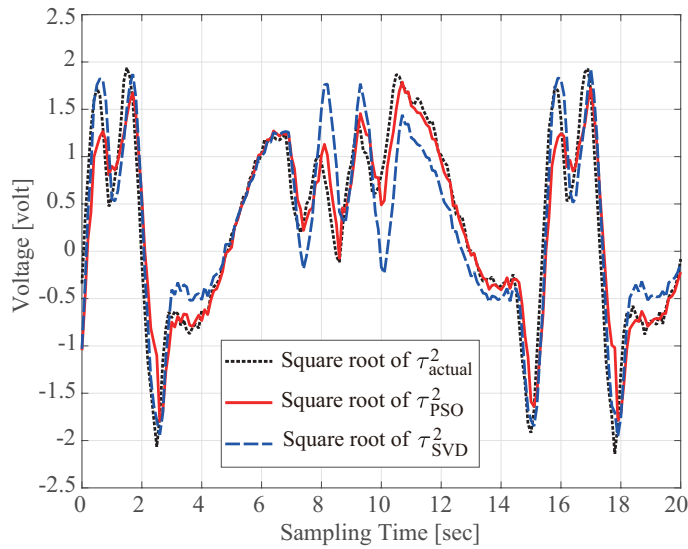
Lastly, to further illustrate the superiority of the presented identification scheme, the model validation is performed by using the actual torque $\tau_{actual}(k)$ and $[\theta(k), \dot{\theta}(k), \ddot{\theta}(k)]$, which are measured when the moving platform follows the reference trajectory given in Figure 5b. Figure 10 summarizes the model validation results of the PSO-, RLS-, DLS-, and SVD-based identification methods, which demonstrates the validity of our PSO-based identification scheme. The normalized root mean square error (NRMSE) index is introduced to quantitatively evaluate the model fitness, and results were calculated on the basis of the following equation for $i = 1, 2, 3$ and are listed in Table 2:

$$\text{Model fitness} = 1 - \sqrt{\frac{\sum_{k=1}^{N_k} (\tau_*^i(k) - \tau_{\text{actual}}^i(k))^2}{N_k}} \quad (41)$$

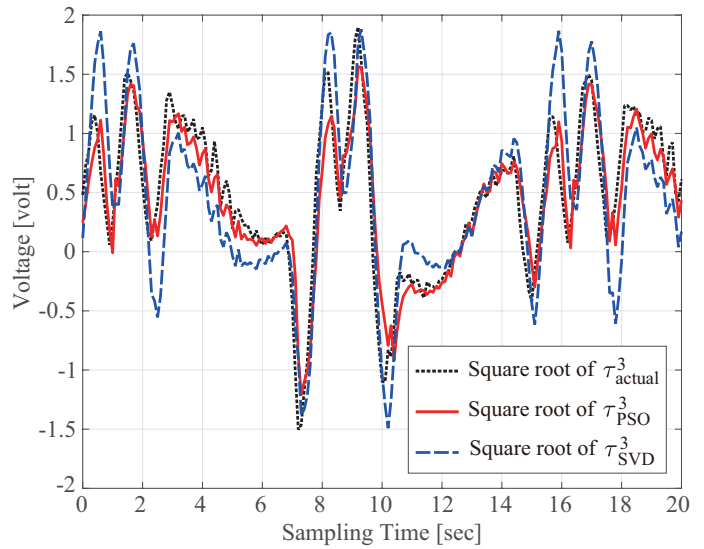
where the notation “*” denotes PSO, RLS, DLS, or SVD, and $\tau_{*,\max}^i$ and $\tau_{*,\min}^i$ are the maximal and minimal elements of the sampled data from the computed torque $\tau_*^i(k)$. A value for (41) close to 1 indicates less variance between the actual torque and model-based computed torque; thus, Table 2 verifies that the proposed scheme outperforms the conventional identification method.



(a) Behaviors of $\tau_{\text{actual}}^1(k)$, $\tau_{\text{SVD}}^1(k)$ and $\tau_{\text{PSO}}^1(k)$



(b) Behaviors of $\tau_{\text{actual}}^2(k)$, $\tau_{\text{SVD}}^2(k)$ and $\tau_{\text{PSO}}^2(k)$



(c) Behaviors of $\tau_{\text{actual}}^3(k)$, $\tau_{\text{SVD}}^3(k)$ and $\tau_{\text{PSO}}^3(k)$

Figure 9. Comparison of the measured actual torque τ_{actual} , the torque $\tau_{\text{SVD}}^i(k)$ obtained from SVD-based identification, and the torque $\tau_{\text{PSO}}^i(k)$ computed using (26) with the identified parameter values in (36).

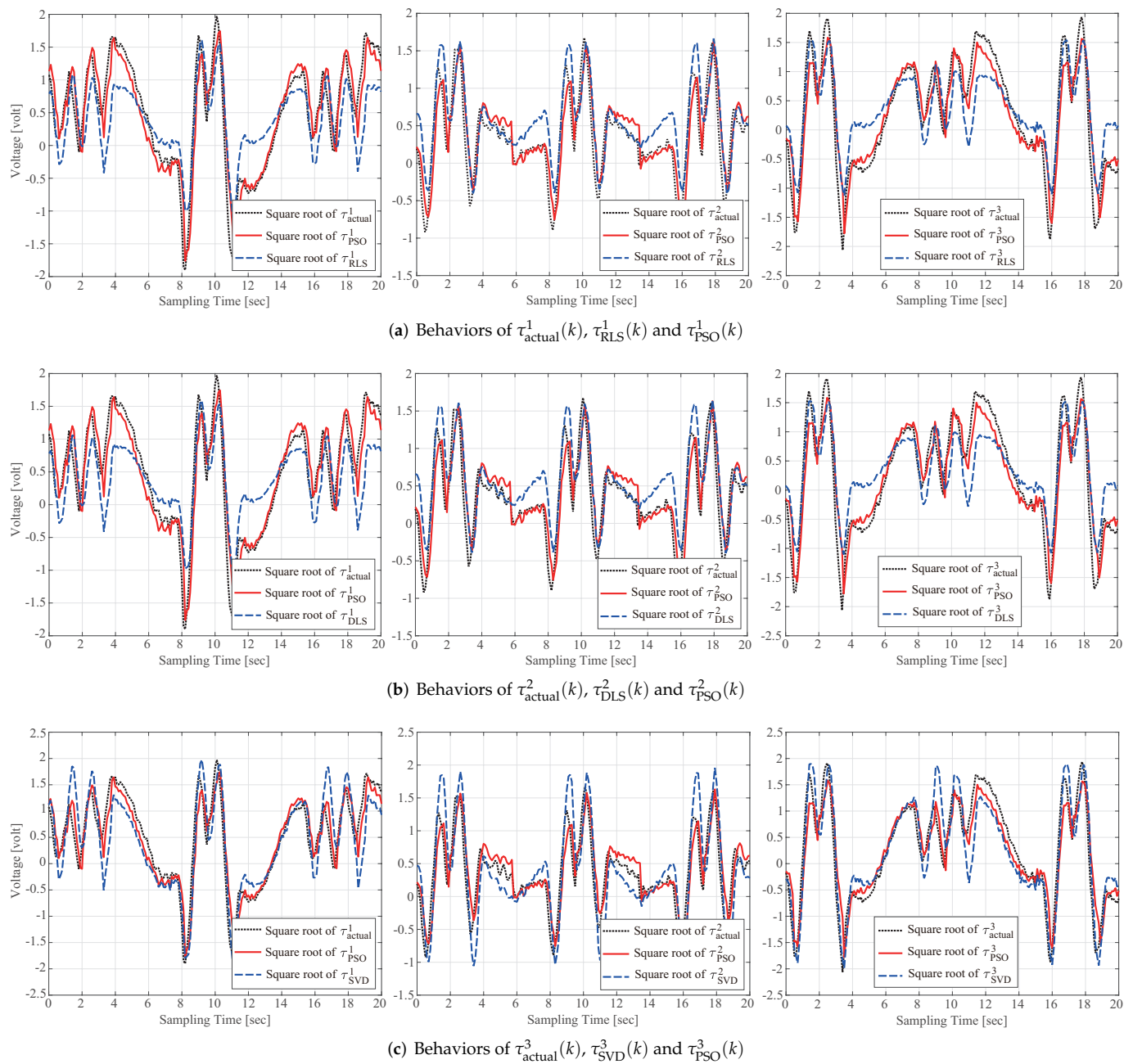


Figure 10. Comparison of the actual torque τ_{actual} , the torque $\tau_{\text{RLS}}^i(k)$ obtained from RLS-based identification, the torque $\tau_{\text{DLS}}^i(k)$ obtained from DLS-based identification, the torque $\tau_{\text{SVD}}^i(k)$ obtained from SVD-based identification, and the torque $\tau_{\text{PSO}}^i(k)$ computed using (26) with the identified parameter values in (36).

Table 2. Identified model fitness evaluated for each actuator torque.

Identification Methods	This Study	RLS-Based Method	DLS-Based Method	SVD-Based Method
Model fitness for τ_1	0.9240	0.8724	0.8714	0.8545
Model fitness for τ_2	0.9053	0.8679	0.8684	0.8001
Model fitness for τ_3	0.9228	0.8761	0.8751	0.8604

5. Conclusions

In this study, an efficient and reliable parameter-identification method was proposed for the analytic dynamic model of a delta robot. The complex dynamic equations of motion were derived according to the virtual work principle and then extended to characterize the inertial and centripetal/Coriolis forces and the frictional effects on the robot's dynamic behavior. To directly identify the set of uncertain parameters included in such a dynamic model, the PSO algorithm with a distributed cyclic neighborhood-search mechanism was employed. Compared to conventional methods, our identification technique exhibits the following distinctive features: (i) Owing to the flexibility of the PSO algorithm, the task of transforming the system of equations into a linear formulation with respect to the uncertain system parameters is not necessary. (ii) No cumbersome and time-consuming tasks, such as developing an additional robot-simulation model solely for identifying the parameters in an overdetermined set of equations, are required. The experimental results indicate that the proposed technique can directly achieve high-quality identification of an analytic dynamic model for a parallel-link robot.

Author Contributions: Conceptualization and methodology, T.-H.K. and M.K.; software and experiments, T.-H.K., Y.K. and T.K.; validation and formal analysis, M.K., Y.K. and T.K.; writing—original draft preparation, T.-H.K. and M.K.; writing—review and editing, T.-H.K.; funding acquisition, T.-H.K. All authors have read and agreed to the published version of the manuscript.

Funding: this research was supported by National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (no. 2021R1A2C1008686) and the Chung-Ang University Graduate Research Scholarship in 2021.

Conflicts of Interest: the authors declare no conflict of interest.

Appendix A. Forward Kinematics of the Delta Robot

In this section, the forward kinematics of the delta robot is derived. The vector loop-closure equation of the kinematic chain for the i th limb is expressed as follows:

$$\overrightarrow{OP} + \overrightarrow{PC_i} = \overrightarrow{OA_i} + \overrightarrow{A_iB_i} + \overrightarrow{B_iC_i} \quad (\text{A1})$$

Rewriting the vector components of (A1) in the coordinate system A_i - x_i - y_i - z_i gives

$$\begin{bmatrix} c(\phi_i) & s(\phi_i) & 0 \\ -s(\phi_i) & c(\phi_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} + \begin{bmatrix} r \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} R \\ 0 \\ 0 \end{bmatrix} + L_1 \begin{bmatrix} c(\theta_{i1}) \\ 0 \\ s(\theta_{i1}) \end{bmatrix} + L_2 \begin{bmatrix} s(\theta_{i3})c(\theta_{i2} + \theta_{i1}) \\ c(\theta_{i3}) \\ s(\theta_{i3})s(\theta_{i2} + \theta_{i1}) \end{bmatrix} \quad (\text{A2})$$

Three equations in (A2) are squared and summed to eliminate the passive joint angle θ_{i2} as follows:

$$(P_x + P_{x_i})^2 + (P_y + P_{y_i})^2 + (P_z + P_{z_i})^2 = L_2^2 \quad (\text{A3})$$

where

$$P_{x_i} := (r - R - L_1 \cos \theta_{i1}) \cos \phi_i, P_{y_i} := (r - R - L_1 \cos \theta_{i1}) \sin \phi_i, P_{z_i} := -L_1 \sin \theta_{i1} \quad (\text{A4})$$

Then, subtracting Equation (A3) of the first kinematic chain (i.e., $i = 1$) from those of the second ($i = 2$) and third ($i = 3$) kinematic chains gives

$$P_{x_{12}}P_x + P_{y_{12}}P_y + P_{z_{12}}P_z = \Delta P_{x_{12}}, \quad (\text{A5})$$

$$P_{x_{13}}P_x + P_{y_{13}}P_y + P_{z_{13}}P_z = \Delta P_{x_{13}}, \quad (\text{A6})$$

respectively, where $P_{x_{1\ell}} = 2(P_{x_1} - P_{x_\ell})$, $P_{y_{1\ell}} = 2(P_{y_1} - P_{y_\ell})$, $P_{z_{1\ell}} = 2(P_{z_1} - P_{z_\ell})$, and $\Delta P_{x_{1\ell}} = P_{x_\ell}^2 + P_{y_\ell}^2 + P_{z_\ell}^2 - P_{x_1}^2 - P_{y_1}^2 - P_{z_1}^2$ for $\ell = 2, 3$. Then, Equations (A5) and (A6) can be written in matrix form as follows:

$$\begin{bmatrix} P_{x12} & P_{y12} \\ P_{x13} & P_{y13} \end{bmatrix} \begin{bmatrix} P_x \\ P_y \end{bmatrix} = \begin{bmatrix} \Delta P_{x12} - P_{z12} P_z \\ \Delta P_{x13} - P_{z13} P_z \end{bmatrix} \quad (\text{A7})$$

Therefore, P_x and P_y can be obtained in terms of P_z as follows:

$$\begin{bmatrix} P_x \\ P_y \end{bmatrix} = \frac{1}{P_{x12} P_{y13} - P_{y12} P_{x13}} \begin{bmatrix} P_{y13} \Delta P_{x12} - P_{y12} \Delta P_{x13} + (-P_{y13} P_{z12} + P_{y12} P_{z13}) P_z \\ -P_{x13} \Delta P_{x12} + P_{x12} \Delta P_{x13} + (P_{x13} P_{z12} - P_{x12} P_{z13}) P_z \end{bmatrix} \quad (\text{A8})$$

Lastly, the three-dimensional position coordinate $\mathbf{P} = [P_x \ P_y \ P_z]^T$ can be derived from (A3) and (A8).

Appendix B. Inverse Kinematics of the Delta Robot

In this section, the derivation of the inverse kinematics of a delta robot is presented. Consider Equation (A2). From the second row of (A2), angle θ_{i3} is calculated as

$$\theta_{i3} = \cos^{-1} \left(\frac{-P_x \sin \phi_i + P_y \cos \phi_i}{L_2} \right) \quad (\text{A9})$$

By contrast, rewriting (A2) gives

$$\begin{bmatrix} L_1 c(\theta_{i1}) + L_2 s(\theta_{i3}) c(\theta_{i2} + \theta_{i1}) \\ L_2 c(\theta_{i3}) \\ L_1 s(\theta_{i1}) + L_2 s(\theta_{i3}) s(\theta_{i2} + \theta_{i1}) \end{bmatrix} = \begin{bmatrix} \tilde{P}_x \\ \tilde{P}_y \\ \tilde{P}_z \end{bmatrix} := \begin{bmatrix} c(\phi_i) & s(\phi_i) & 0 \\ -s(\phi_i) & c(\phi_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} + \begin{bmatrix} r - R \\ 0 \\ 0 \end{bmatrix} \quad (\text{A10})$$

Then, three equations in (A10) are squared and summed to obtain

$$L_1^2 + L_2^2 + 2L_1 L_2 \sin \theta_{i3} \cos \theta_{i2} = \tilde{P}_x^2 + \tilde{P}_y^2 + \tilde{P}_z^2 \quad (\text{A11})$$

Because θ_{i3} is determined from (A9) and θ_{i2} is the only unknown parameter in (A11), the joint angle θ_{i2} can be obtained as follows:

$$\theta_{i2} = \cos^{-1} \left(\frac{\tilde{P}_x^2 + \tilde{P}_y^2 + \tilde{P}_z^2 - L_1^2 - L_2^2}{2L_1 L_2 \sin \theta_{i3}} \right) \quad (\text{A12})$$

Once θ_{i3} and θ_{i2} are determined from (A9) and (A12), respectively, two equations with θ_{i1} as the only unknown parameter can be derived from (A10). Thus, the joint angle θ_{i1} can be obtained numerically.

Appendix C. Classical Particle Swarm Optimization Algorithm

The classical PSO algorithm is a population-based stochastic computational technique with a powerful global-search capability for resolving the following form of the optimization problem without constraints:

$$\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}), \quad f(\mathbf{x}) : \mathbb{R}^n \mapsto \mathbb{R} \quad (\text{A13})$$

where the linear or nonlinear objective function $f(\mathbf{x})$ is minimized with respect to the vector of the design variables $\mathbf{x} \in \mathbb{R}^n$. Therefore, a particle in the PSO algorithm represents a potential solution $\mathbf{x}$ in (A13). Let $\mathbb{D}$ denote the limited subregion of an entire n -dimensional Euclidean space and be assumed to contain the optimal solutions. The conventional evolutionary PSO searching mechanism is initiated with a swarm randomly generated over the space $\mathbb{D}$. Thereafter, each particle moves in a coordinated manner through the $\mathbb{D}$ -dimensional search space. The behavior of each particle is mainly influenced by both its own best previous experience and a social compulsion to move toward a single best particle from the entire swarm as follows:

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \mathbf{v}_i^{k+1}, \quad (\text{A14})$$

$$\mathbf{v}_i^{k+1} = c_0 \mathbf{v}_i^k + c_1 r_{1,i}^k (\mathbf{x}_{\text{pbest},i} - \mathbf{x}_i^k) + c_2 r_{2,i}^k (\mathbf{x}_{\text{gbest}} - \mathbf{x}_i^k) \quad (\text{A15})$$

where $i = 1, 2, \dots, n_p$ denotes the index of the particle; $k = 1, 2, \dots, k_{\max}$ represents the iteration number; $\mathbf{x}_i^k \in \mathbb{R}^n$ and $\mathbf{v}_i^k \in \mathbb{R}^n$ denote the position and velocity vectors, respectively, for the i th particle at the k th iteration; $\mathbf{x}_{\text{pbest},i}$ denotes the vector of the best previous position yielding the minimum fitness value $f(\cdot)$ for the i th particle; $\mathbf{x}_{\text{gbest}}$ denotes the vector of the global best position determined by the entire swarm; c_0 denotes the inertia weight, with c_1 and c_2 representing the cognitive and social-scaling parameters, respectively; and $r_{1,i}^k$ and $r_{2,i}^k$ are the random parameters generated uniformly in the range of $[0, 1]$.

References

1. Liu, C.; Cao, G.; Qu, Y. Safety analysis via forward kinematics of delta parallel robot using machine learning. *Saf. Sci.* **2019**, *117*, 243–249. [\[CrossRef\]](#)
2. Coronado, E.; Maya, M.; Cardenas, A.; Guarneros, O.; Piovesan, D. Vision-based control of a delta parallel robot via linear camera-space manipulation. *J. Intell. Robot. Syst.* **2017**, *85*, 93–106. [\[CrossRef\]](#)
3. Khalil, W.; Dombre, E. *Modeling Identification and Control of Robots*; Hermes Penton Ltd.: London, UK, 2002.
4. Wu, J.; Wang, J.; You, Z. An overview of dynamic parameter identification of robots. *Robot. Comput. Integr. Manuf.* **2010**, *26*, 414–419. [\[CrossRef\]](#)
5. Angel, L.; Viola, J. Parametric identification of a delta type parallel robot. In Proceedings of the 2016 IEEE Colombian Conference on Robotics and Automation (CCRA), Bogota, Colombia, 29–30 September 2016; pp. 1–6.
6. Liu, Y.; Wu, J.; Wang, L.; Wang, J. Parameter identification algorithm of kinematic calibration in parallel manipulators. *Adv. Mech. Eng.* **2016**, *8*, 1687814016667908. [\[CrossRef\]](#)
7. Mata, V.; Farhat, N.; Díaz-Rodríguez, M.; Manipulators, P. Dynamic parameter identification for parallel manipulators. In *Parallel Manipulators, Towards New Applications*; IntechOpen: London, UK, 2009; pp. 21–43.
8. Kim, T.-H.; Byun, J.-I. Truss sizing optimization with a diversity-enhanced cyclic neighborhood network topology particle swarm optimizer. *Mathematics* **2020**, *8*, 1087. [\[CrossRef\]](#)
9. Kuo, Y.-L. Mathematical modeling and analysis of the Delta robot with flexible links. *Comput. Math. Appl.* **2016**, *71*, 1973–1989. [\[CrossRef\]](#)
10. Angel, L.; Viola, J. Fractional order PID for tracking control of a parallel robotic manipulator type delta. *ISA Trans.* **2018**, *79*, 172–188. [\[CrossRef\]](#) [\[PubMed\]](#)
11. Park, S.B.; Kim, H.S.; Song, C.; Kim, K. Dynamics modeling of a Delta-type parallel robot (ISR 2013). In Proceedings of the ISR 2013—44th International Symposium on Robotics, Seoul, Korea, 24–26 October 2013; pp. 1–5.
12. Ruderman, M.; Iwasaki, M.; Song, C.; Chen, W. Motion-control techniques of today and tomorrow: A review and discussion of the challenges of controlled motion. *IEEE Ind. Electron. Mag.* **2020**, *14*, 41–55. [\[CrossRef\]](#)
13. Li, Y.; Xu, Q. Dynamic analysis of a modified DELTA parallel robot for cardiopulmonary resuscitation. In Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, AB, Canada, 2–6 August 2005; pp. 233–238.
14. Codourey, A. Dynamic modelling and mass matrix evaluation of the DELTA parallel robot for axes decoupling control. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS '96, Osaka, Japan, 4–8 November 1996; pp. 1211–1218.
15. Craig, J. *Introduction to Robotics: Mechanics and Control*, 3rd ed.; Pearson Prentice Hall: Hoboken, NJ, USA, 2005.
16. Kononova, A.V.; Corne, D.W.; De Wilde, P.; Shneer, V.; Caraffini, F. Structural bias in population-based algorithms. *Inf. Sci.* **2015**, *298*, 468–490. [\[CrossRef\]](#)
17. Islam, S.A.U.; Bernstein, D.S. Recursive least squares for real-time implementation [Lecture Notes]. *IEEE Control Syst. Mag.* **2019**, *39*, 82–85. [\[CrossRef\]](#)
18. Argha, A.; Ye, L.; Cao, K.; Su, S.W.; Celler, B.G. Real-time identification of heart rate responses via auxiliary-model-based damped RLS scheme. In Proceedings of the 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Jeju Island, Korea, 11–15 July 2017; pp. 1312–1315.
19. Yaacoub, F.; Abche, A.; Karam, E.; Hamam, Y. MRI reconstruction using SVD in the least square sense. In Proceedings of the 2008 21st IEEE International Symposium on Computer-Based Medical Systems, Jyväskylä, Finland, 17–19 June 2008; pp. 47–49.