Institution of MECHANICAL ENGINEERS

Journal of MULTI-BODY DYNAMICS

Original Article

Analytical dynamic modeling of Delta robot with experimental verification

Proc IMechE Part K: J Multi-body Dynamics
2020, Vol. 234(3) 623–630

DOI: 10.1177/1464419320929160
journals.sagepub.com/home/pik
SAGE

Farshid Asadi and Ali Heydari

Abstract

In this paper, an explicit dynamic model of Delta robot is obtained analytically. The main contribution of this work is that, unlike existing prior work, the final dynamics model is given directly in the form of $M\ddot{X} + C\dot{X} + G = J^T\tau$, with explicit expressions for *M, C* and *G*. This is of great importance, since many advanced control techniques like *Optimal Control* need dynamic model in an explicit form, i.e. time derivative of state vector given explicitly in terms of the states and control vectors. To this goal, first, velocity and acceleration analysis is done by differentiating robot's geometrical loops directly. Then, Jacobian matrices are calculated to have kinematic relations in a more compact form. After that, principle of virtual work is implemented to derive the dynamic equations. In this part, Jacobian matrices are substituted into dynamic model. This is unlike other referenced works on Delta robot dynamics that need to continue the derivation in symbolic software or derive the model implicitly. Using Jacobians, dramatically simplifies the final explicit dynamic model. Therefore, the final dynamic equations are calculated in a straightforward manner without any use of symbolic calculation software. After all, the presented model is verified with an experimental setup. The model shows good accuracy in terms of torque prediction.

Keywords

Delta robot, explicit dynamic model, analytical model, experimental

Date received: 31 October 2019; accepted: 2 May 2020

Introduction

Delta robot is a parallel manipulator that is mainly designed as a pick and place robot.[1] However, because of its high force to weight ratio and simple design, it can also be used for other purposes, e.g. machining, haptic interface, surgery.[2,3] In model-based control techniques, one needs a dynamic model of the system, even though there are similarities in dynamic modeling procedures of parallel robots, due to their different structure, each case should be resolved separately. Thus, several methods including Newton–Euler,[4] Lagrange equation,[5–11] Hamilton equation,[12] principle of virtual work[10,13,14] and recursive methods[15,16] are used to derive dynamics equations of this manipulator. In all the mentioned works, friction in spherical joints are neglected, which as discussed in Clavel,[17] is reasonable because of using low friction spherical joints. Also in some of the mentioned works, intermediate legs are modeled as point masses distributed over two ends of the intermediary legs.[4,6,7,9,12] Effects of this simplification are generally case dependent, but it is studied in Brinker et al.[18,19] These two works compared accuracy and computational cost of different dynamic modeling approaches. Both of the studies are based on multi-body dynamics software. Based on these two studies, there is around 10% inaccuracy in predicting torques for the simplified model in software simulations; however, it is 38% more efficient in terms of computational cost.

To the best of our knowledge with the existing models, one needs using symbolic calculation software to calculate the dynamic model explicitly. While this is possible, the resultant equations are too long and hard to track. Having an explicit model is a requirement of some modern control techniques. This motivated us to develop an approach to derive explicit dynamic equations of Delta robot analytically, while keeping it neat and compact without using computer symbolic calculation. The calculated model can be used directly for all control methods which require the model of the dynamics to be in the form of

Department of Mechanical Engineering, Southern Methodist University, Dallas, USA

Corresponding author:
Farshid Asadi, Southern Methodist University, Dallas, TX 75205, USA.
Email: farshidasadi47@yahoo.com

$\dot{x} = f(x, u)$, where x and u represent the states and control vectors.

In what follows, first, velocity and acceleration analysis is done based on Codourey.[4,10] Then, Jacobian matrices of Tsai[20] are utilised to have velocity and accelerations in a more compact form. After that, virtual work principle is employed to derive the dynamic equations. Next, Jacobian matrices are substituted in dynamic model to simplify it in an organised and traceable way. Furthermore, experimental verification of the model is done and presented in the last part.

Kinematics

3-Revolute, spherical, spherical (3-RSS) Delta robot is constructed by three serial chains connecting the fixed platform to the moving platform as depicted in Figure 1. To describe kinematics of the robot, two frames are assigned for each serial chain. First, a frame defined by a rotation about Z-axis of the world frame (that is perpendicular to fixed platform plane), whose rotation matrix is called R_i. Second, a frame is assigned to motorised leg as a consecutive body fixed rotation about the x-axis with rotation matrix O_i (see Figures 1 and 2). The mentioned rotation matrices can be written as

$$R_i = \begin{bmatrix} \cos(\varphi_i) & -\sin(\varphi_i) & 0 \\ \sin(\varphi_i) & \cos(\varphi_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$O_i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(q_i) & -\sin(q_i) \\ 0 & \sin(q_i) & \cos(q_i) \end{bmatrix} \quad (2)$$

One can write the following equations

$$\begin{cases} X + R_{ai} = R_{bi} + \overrightarrow{B_iC_i} + \overrightarrow{C_iA_i} \\ \overrightarrow{C_iA_i}^{\mathrm{T}} \overrightarrow{C_iA_i} = l_K^2 \end{cases} \quad (3)$$

where all vectors are stated in world frame. $X, R_{bi}, R_{ai}, \overrightarrow{B_iC_i}, \overrightarrow{C_iA_i}$, and l_K are moving platform centre of mass position with respect to world frame origin, fixed platform joint position with respect to world frame origin, moving platform joint position with respect to its centre of mass, motorised leg vector, intermediary leg vector, and intermediary leg length. The first equation of (3) is called *Loop Closure*. The second equation of (3) denotes the constant length of the intermediary legs.

Note that vectors R_{bi} and R_{ai} are constant vectors in their corresponding $x'_iy'_iz'_i$ frame. Therefore, one can write $R_{bi} = R_iR_b$ and $R_{ai} = R_iR_a$, where R_b and R_a are fixed platform joint position and moving platform joint position stated in $x'_iy'_iz'_i$ frames. Also the same is true about motorised leg vector

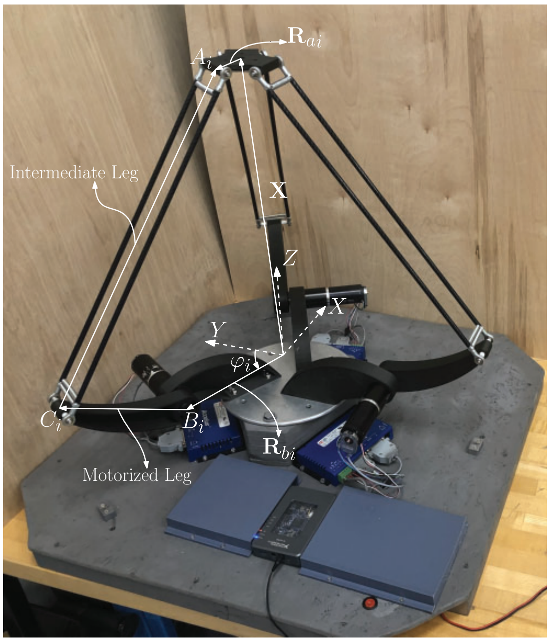

Figure 1. Delta robot structure.

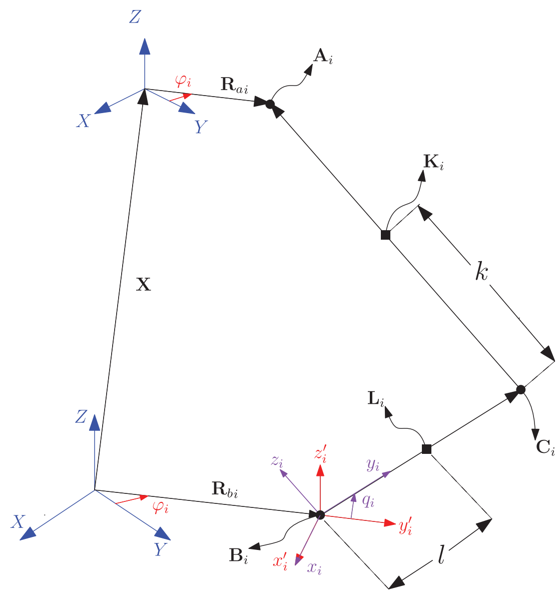

Figure 2. Delta robot frame assignments for *i*th leg chain.

$\overrightarrow{B_iC_i}$ in the $x_iy_iz_i$ frame. So, one can write $\overrightarrow{B_iC_i} = R_iO_iL$, where L is motorised leg vector stated in its own $x_iy_iz_i$ frame. Using these in equation (3) and after rearrangement, one can write

$$\begin{cases} U_i = X - R_i((R_b - R_a) + O_iL) \\ U_i^{\mathrm{T}} U_i = l_K^2 \end{cases} \quad (4)$$

where $\overrightarrow{C_iA_i}$ is renamed to U_i for ease of notation. This equation should be solved to obtain forward or inverse kinematic of the manipulator, that is, to calculate X based on q (where $q = [q_1, q_2, q_3]^{\mathrm{T}}$) and vice versa. While equation (4) is used for obtaining the dynamical model in the next section, calculating the mentioned kinematics is not the objective in this work and therefore is skipped. See Williams[21] and Hsu et al.[22] for detailed solution and further analysis of the kinematics of the robot.

Velocity, Acceleration and Jacobian

In order to obtain a dynamic model of the manipulator, velocity and acceleration analysis should be done. Moreover, auxiliary Jacobians are helpful to make the final equations simpler. First, the main Jacobians of the system should be calculated. This is done here, by implementing the approach of Codourey.[4,10] Differentiating the second equation of (4) results into

$$U_i^{\mathrm{T}} \dot{U}_i = 0 \tag{5}$$

Calculation of $\dot{U}_i$ is straightforward through chain rule. The terms R_i, R_b, R_a, and L are constant and have zero derivatives. Therefore, one only needs to calculate the term $R_i \dot{O}_i L$. Since O_i is a rotation matrix for an elementary rotation around x_i' axis, its derivative can be calculated as Ginsberg[23]

$$\dot{O}_i = S_1 O_i \dot{q}_i \tag{6}$$

where S_1 is the skew symmetric matrix related to the derivative of rotation around a general x-axis, which is defined as

$$S_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

Using above notes, one can write

$$\dot{U}_i = \dot{X} - T_i \dot{q}_i \tag{7}$$

where $T_i = R_i S_1 O_i L$. By substituting equations (4) and (7) in equation (5), the result can be combined as

$$\begin{bmatrix} U_1^{\mathrm{T}} \\ U_2^{\mathrm{T}} \\ U_3^{\mathrm{T}} \end{bmatrix} \dot{X} = \begin{bmatrix} U_1^{\mathrm{T}} T_1 & 0 & 0 \\ 0 & U_2^{\mathrm{T}} T_2 & 0 \\ 0 & 0 & U_3^{\mathrm{T}} T_3 \end{bmatrix} \dot{q} \tag{8}$$

The above equation can be rewritten as $\dot{q} = J\dot{X}$, where J is the Jacobian of the system and is defined as

$$J(X, q) = \begin{bmatrix} U_1^{\mathrm{T}} T_1 & 0 & 0 \\ 0 & U_2^{\mathrm{T}} T_2 & 0 \\ 0 & 0 & U_3^{\mathrm{T}} T_3 \end{bmatrix}^{-1} \begin{bmatrix} U_1^{\mathrm{T}} \\ U_2^{\mathrm{T}} \\ U_3^{\mathrm{T}} \end{bmatrix} \tag{9}$$

By differentiating each row of equation (8) and rearranging it one has

$$\dot{U}_i^{\mathrm{T}} \dot{X} + U_i^{\mathrm{T}} \ddot{X} = (\dot{U}_i^{\mathrm{T}} T_i + U_i^{\mathrm{T}} \dot{T}_i)\dot{q}_i + U_i^{\mathrm{T}} T_i^{\mathrm{T}} \ddot{q}_i \tag{10}$$

where $\dot{T}_i = R_i S_1 S_1 O_i L \dot{q}_i$. Equation (10) can be rearranged and written in matrix form as

$$\ddot{q} = J\ddot{X} + \dot{J}\dot{X} \tag{11}$$

where $\dot{J}$ in the above equation can be written as

$$\begin{aligned} \dot{J}(X, q, \dot{X}, \dot{q}) = & \begin{bmatrix} U_1^{\mathrm{T}} T_1 & 0 & 0 \\ 0 & U_2^{\mathrm{T}} T_2 & 0 \\ 0 & 0 & U_3^{\mathrm{T}} T_3 \end{bmatrix}^{-1} \\ & \times \left(\begin{bmatrix} \dot{U}_1^{\mathrm{T}} \\ \dot{U}_2^{\mathrm{T}} \\ \dot{U}_3^{\mathrm{T}} \end{bmatrix} - \begin{bmatrix} (\dot{U}_1^{\mathrm{T}} T_1 + U_1^{\mathrm{T}} \dot{T}_1) J_1 \\ (\dot{U}_2^{\mathrm{T}} T_2 + U_2^{\mathrm{T}} \dot{T}_2) J_2 \\ (\dot{U}_3^{\mathrm{T}} T_3 + U_3^{\mathrm{T}} \dot{T}_3) J_3 \end{bmatrix} \right) \end{aligned} \tag{12}$$

where J_i denotes ith row of J.

Now that the main Jacobians are calculated, leg velocities and accelerations are easy to compute. The process done in Tsai[20] is adopted here for this purpose.

Angular velocity of motorised leg can be written in its own frame, that is $x_i y_i z_i$, as

$$^{O}\omega_{L_i} = \begin{bmatrix} J_i \\ 0_{1\times 3} \\ 0_{1\times 3} \end{bmatrix} \dot{X} \tag{13}$$

For linear velocity of the motor leg COM one has

$$^{O}v_{L_i} = {}^{O}\omega_{L_i} \times (r_L \hat{j}) = r_L l_L \begin{bmatrix} 0_{1\times 3} \\ 0_{1\times 3} \\ J_i \end{bmatrix} \dot{X} \tag{14}$$

where $\hat{j}$ is defined as $[0, 1, 0]^{\mathrm{T}}$, l_L is motor leg length and r_L is the ratio of motor leg length that shows the distance between motor leg COM and corresponding fixed platform joint, that is $\frac{l}{l_L}$. For angular acceleration of motor leg COM one can write

$$^{O}\dot{\omega}_{L_i} = \begin{bmatrix} J_i \\ 0_{1\times 3} \\ 0_{1\times 3} \end{bmatrix} \ddot{X} + \begin{bmatrix} \dot{J}_i \\ 0_{1\times 3} \\ 0_{1\times 3} \end{bmatrix} \dot{X} \tag{15}$$

Also, linear acceleration of motor leg COM can be calculated as

$$\begin{aligned} ^{O}\dot{v}_{L_i} &= {}^{O}\omega_{L_i} \times ({}^{O}\omega_{L_i} \times r_L l_L \hat{j}) + {}^{O}\dot{\omega}_{L_i} \times r_L l_L \hat{j} \\ &= r_L l_L \begin{bmatrix} 0_{1\times 3} \\ 0_{1\times 3} \\ J_i \end{bmatrix} \ddot{X} + r_L l_L \begin{bmatrix} 0_{1\times 3} \\ -\dot{X}^{\mathrm{T}} J_i^{\mathrm{T}} J_i \\ \dot{J}_i \end{bmatrix} \dot{X} \end{aligned} \tag{16}$$

Note that replacing r_L with one in equation (14) and equation (16) gives linear velocity and acceleration of point C_i in the corresponding motor leg frame.

Having velocity and acceleration of moving platform and point C_i, motion properties of intermediary legs can be determined easily in motor leg frames. For linear velocities and accelerations, one can see that the following equations hold. Note that every point of moving platform moves with the same linear velocity and acceleration. Therefore, one can write

$$\begin{aligned} {}^{O}v_{K_i} &= {}^{O}v_{C_i} + r_K({}^{O}v_{A_i} - {}^{O}v_{C_i}) \\ &= \left((1-r_K)l_L \begin{bmatrix} 0_{1\times3} \\ 0_{1\times3} \\ J_i \end{bmatrix} + r_K O_i^{\mathrm{T}} R_i^{\mathrm{T}} \right) \dot{X} \end{aligned} \tag{17}$$

$$\begin{aligned} {}^{O}\dot{v}_{K_i} &= {}^{O}\dot{v}_{C_i} + r_k({}^{O}\dot{v}_{A_i} - {}^{O}\dot{v}_{C_i}) \\ &= \left((1-r_K)l_L \begin{bmatrix} 0_{1\times3} \\ 0_{1\times3} \\ J_i \end{bmatrix} + r_K O_i^{\mathrm{T}} R_i^{\mathrm{T}} \right) \ddot{X} \\ &\quad + (1-r_K)l_L \begin{bmatrix} 0_{1\times3} \\ -\dot{X}^{\mathrm{T}} J_i^{\mathrm{T}} J_i \\ \dot{J}_i \end{bmatrix} \dot{X} \end{aligned} \tag{18}$$

where r_K is the intermediary leg length parameter that denotes the distance between point C_i and intermediary leg COM, that is equal to $\frac{k}{l_K}$.

Finally, auxiliary Jacobians can be defined from the calculated velocities for our points of interest, including motor leg and intermediary leg COMs and point C_i, as

$$J_{L_i} = r_L l_L \begin{bmatrix} 0_{1\times3} \\ 0_{1\times3} \\ J_i \end{bmatrix} \tag{19}$$

$$J_{K_i} = \left((1-r_K)l_L \begin{bmatrix} 0_{1\times3} \\ 0_{1\times3} \\ J_i \end{bmatrix} + r_K O_i^{\mathrm{T}} R_i^{\mathrm{T}} \right) \tag{20}$$

$$J_{C_i} = l_L \begin{bmatrix} 0_{1\times3} \\ 0_{1\times3} \\ J_i \end{bmatrix} \tag{21}$$

Dynamics

Before proceeding to the main dynamic model, a useful modeling simplification can be done. Because of the structure of intermediary legs in Delta robot, product of inertias of intermediary legs is negligible. Also, moment of inertia around its longitudinal axis is much less in comparison with other two principal moments of inertia. Furthermore, the dominant moment of inertias is roughly equal. Therefore, longitudinal moment of inertia is neglected here and then the intermediary leg is replaced with three masses, located at points C, A and intermediary leg COM, so that the result is dynamically equivalent to the intermediary leg (considering that longitudinal moment of inertia is neglected). To guarantee dynamic equivalence, mass distribution between three points can be calculated through solving the following linear system of equation

$$\begin{bmatrix} 1 & 1 & 1 \\ r_K & (r_K-1) & 0 \\ r_K^2 l_K^2 & (r_K-1)^2 l_K^2 & 0 \end{bmatrix} \begin{bmatrix} m_C \\ m_A \\ m_K \end{bmatrix} = \begin{bmatrix} M_K \\ 0 \\ I_K \end{bmatrix} \tag{22}$$

where m_C, m_A and m_K are masses distributed over points A, C, and intermediary leg COM. M_K and I_K are also intermediary leg mass and dominant moment of inertia with respect to its COM. In equation (22), the first row shows mass equivalence, the second row represents COM equivalence, and the third row denotes inertial equivalence. This process results in the same equivalence that is derived in Codourey[10]; however, it is in a form more suitable for applying virtual work method. Note that this simplification is different than the mentioned simplification in literature.[4,6,7,9,12] In the referenced simplified models, mass of intermediary leg is distributed over point C and A; thus, there is no such dynamic equivalence as discussed here.

To construct equation of motion of the mechanism by the principle of virtual work, one needs to first calculate applied and inertial active wrenches (forces and momentums). This basically needs to be calculated at motor, motor leg COM, point C_i, intermediary leg COM, point A_i and moving platform COM.

For the moving platform, there is no rotational term. So, one has

$$W_P = F_P + m_P g - m_P \ddot{X} \tag{23}$$

where F_P, m_P and g are external force applied to moving platform, moving platform mass and gravity vector, respectively. Motors are fixed and there is no translational term; also note that motors only have active wrenches along their own rotational axis direction. Therefore, one can combine motors active wrenches and write

$$W_{mo} = -I_{mo}\ddot{q} - \tau_{fric} + \tau \tag{24}$$

where I_{mo}, τ_{fric} and τ are motor inertia, friction in motor joint and motor torques, respectively. Active translational wrenches of motor leg can be written as the following

$$W_{L_i} = m_L O_i^{\mathrm{T}} R_i^{\mathrm{T}} g - m_L {}^{O}\dot{v}_{L_i} \tag{25}$$

where m_L is motor leg mass. Motor legs have active rotational wrenches only along their x-axis (they rotate around a fixed axis); thus, rotational wrenches of motor legs can be collected in vector form and written as

$$W_L = -I_L \ddot{q} \tag{26}$$

where I_L is moment of inertia of motor leg around x-axis of motor leg frame, passing through its COM. Finally, active wrenches of point C_i, A_i and COM of intermediary leg can be computed as

$$W_{C_i} = m_C O_i^{\mathrm{T}} R_i^{\mathrm{T}} g - m_C {}^{O}\dot{v}_{C_i} \tag{27}$$

$$W_{A_i} = m_A g - m_A {}^{O}\dot{v}_{A_i} \tag{28}$$

$$W_{K_i} = m_K O_i^{\mathrm{T}} R_i^{\mathrm{T}} g - m_k {}^{O}\dot{v}_{K_i} \tag{29}$$

By substituting acceleration terms (equations (11), (16) and (18)) into equations (23) to (29), active wrenches can be rewritten in terms of generalised coordinate X and its derivatives. The summary is as follows

$$W_P = -m_P \ddot{X} + F_P + m_P g \tag{30}$$

$$W_{mo} = -I_{mo} J \ddot{X} - I_{mo} \dot{J} \dot{X} - \tau_{fric} + \tau \tag{31}$$

$$\begin{aligned} W_{L_i} = & -m_L r_L l_L \begin{bmatrix} 0_{1\times3} \\ 0_{1\times3} \\ J_i \end{bmatrix} \ddot{X} \\ & -m_L r_L l_L \begin{bmatrix} 0_{1\times3} \\ -\dot{X}^{\mathrm{T}} J_i^{\mathrm{T}} J_i \\ \dot{J}_i \end{bmatrix} \dot{X} + m_L O_i^{\mathrm{T}} R_i^{\mathrm{T}} g \end{aligned} \tag{32}$$

$$W_L = -I_L J \ddot{X} - I_L \dot{J} \dot{X} \tag{33}$$

$$\begin{aligned} W_{C_i} = & -m_C l_L \begin{bmatrix} 0_{1\times3} \\ 0_{1\times3} \\ J_i \end{bmatrix} \ddot{X} \\ & -m_C l_L \begin{bmatrix} 0_{1\times3} \\ -\dot{X}^{\mathrm{T}} J_i^{\mathrm{T}} J_i \\ \dot{J}_i \end{bmatrix} \dot{X} + m_C O_i^{\mathrm{T}} R_i^{\mathrm{T}} g \end{aligned} \tag{34}$$

$$W_{A_i} = -m_A \ddot{X} + m_A g \tag{35}$$

$$\begin{aligned} W_{K_i} = & -m_k \left((1-r_K) l_L \begin{bmatrix} 0_{1\times3} \\ 0_{1\times3} \\ J_i \end{bmatrix} + r_K O_i^{\mathrm{T}} R_i^{\mathrm{T}} \right) \ddot{X} \\ & -m_K (1-r_K) l_L \begin{bmatrix} 0_{1\times3} \\ -\dot{X}_i^{\mathrm{T}} J_i^{\mathrm{T}} J_i \\ \dot{J}_i \end{bmatrix} \dot{X} + m_K O_i^{\mathrm{T}} R_i^{\mathrm{T}} g \end{aligned} \tag{36}$$

where F_P is the external force exerted on moving platform.

Principle of virtual work as used in Tsai[20] can be written for the mechanism as

$$\begin{aligned} \sum_{i=1}^{3} & (\delta x_{C_i}^{\mathrm{T}} W_{C_i} + \delta X W_{A_i} + \delta x_{L_i}^{\mathrm{T}} W_{L_i} + \delta x_{K_i}^{\mathrm{T}} W_{K_i}) \\ & + \delta q^{\mathrm{T}} (W_{mo} + W_L) + \delta X W_P = 0 \end{aligned} \tag{37}$$

where δ terms denote virtual displacements. All virtual displacements can be stated in terms of moving platform virtual displacement, i.e. δX, as

$$\begin{cases} \delta q = J \delta X \\ \delta x_{L_i} = J_{L_i} \delta X \\ \delta x_{C_i} = J_{C_i} \delta X \\ \delta x_{K_i} = J_{K_i} \delta X \end{cases} \tag{38}$$

Finally, by substituting acceleration terms (equations (11), (16) and (18)) and Jacobians (equations (9) and (19) to (20)) into equation (37), factoring out δX and rearranging the equation, one can write the dynamic equation of the mechanism in the form of

$$\begin{aligned} & M(X,q)\ddot{X} + C(X,q,\dot{X},\dot{q})\dot{X} \\ & \quad + G(X,q) - F_P = J(X,q)^{\mathrm{T}} (\tau - \tau_{fric}) \end{aligned} \tag{39}$$

$$\begin{aligned} M(X,q) = & (I_{mo} + I_L) J^{\mathrm{T}} J + (m_P + 3m_B) I_3 \\ & + \sum_{i=1}^{3} (m_K r_K J_{K_i}^{\mathrm{T}} O_i^{\mathrm{T}} R_i^{\mathrm{T}}) \\ & + \sum_{i=1}^{3} \left(\begin{pmatrix} m_C l_L J_{C_i}^{\mathrm{T}} + m_L r_L l_L J_{L_i}^{\mathrm{T}} \\ + m_K (1-r_K) l_L J_{K_i}^{\mathrm{T}} \end{pmatrix} \begin{bmatrix} 0_{1\times3} \\ 0_{1\times3} \\ J_i \end{bmatrix} \right) \end{aligned} \tag{40}$$

$$\begin{aligned} C(X,q,\dot{X},\dot{q}) = & (I_{mo} + I_L) J^{\mathrm{T}} \dot{J} \\ & + \sum_{i=1}^{3} \left(\begin{pmatrix} m_C l_L J_{C_i}^{\mathrm{T}} + m_L r_L l_L J_{L_i}^{\mathrm{T}} \\ + m_K (1-r_K) l_L J_{K_i}^{\mathrm{T}} \end{pmatrix} \begin{bmatrix} 0_{1\times3} \\ -\dot{X}^{\mathrm{T}} J_i^{\mathrm{T}} J_i \\ \dot{J}_i \end{bmatrix} \right) \end{aligned} \tag{41}$$

$$\begin{aligned} G(X,q) = & -(m_P + 3m_A) g \\ & - \sum_{i=1}^{3} \left((m_C J_{C_i}^{\mathrm{T}} + m_L J_{L_i}^{\mathrm{T}} + m_K J_{K_i}^{\mathrm{T}}) O_i^{\mathrm{T}} R_i^{\mathrm{T}} \right) g \end{aligned} \tag{42}$$

where matrices M, C and G are summarised in equations (40) to (42). I_3 in these equations denotes to

3×3 identity matrix. Also keep in mind that equation (39) can be easily written in state-space form as

$$\frac{d}{dt}\left(\begin{bmatrix} X \\ \dot{X} \end{bmatrix}\right) = \begin{bmatrix} \dot{X} \\ -M^{-1}\left(C\dot{X} + G - F_P\right) \end{bmatrix} + \begin{bmatrix} 0_{3\times3} \\ M^{-1}J^{\mathrm{T}} \end{bmatrix}(\tau - \tau_{fric}). \tag{43}$$

The procedure for calculating model parameters at every instance is summarised here:

1. Calculate X and q from kinematics.
2. Calculate $J(X, q)$ from equation (9).
3. Calculate $\dot{X}$ and $\dot{q}$ from Jacobian J.
4. Calculate $\dot{J}$ from equation (11).
5. Calculate J_{L_i}, J_{K_i} and J_{C_i} from equations (19) to (21).
6. Calculate matrices M, C and G from equations (40) to (42).
7. Use matrices M, C, G and J in equation (39).

Evaluation

Verification of the model is an important task to be done. This has been done experimentally in Codourey,[4,10] where the verification is done through checking tracking error of closed loop control (which cannot explicitly show the accuracy of the model itself), and in Miller[12] through examining the predicted torque explicitly. In the rest of the cited prior works, there is no verification or it has been done only on multi-body dynamics software. In the present work, to verify the accuracy of the dynamic model, torques required for performing a typical trajectory are compared on both the proposed model and the experimental setup (see Figure 4); this is motivated by Miller.[12] Here, the robot end effector is controlled (with a simple joint space PD control[7]) to move along a circle in $z = 450$ (mm) plane, with $x = 250\cos(t)$ (mm) and $y = 250\sin(t)$ (mm). The real trajectory travelled by robot can be seen in Figure 3. For torque measurement, it is assumed that the current–torque relation of the motors is linear. It is also assumed that the current commanded to motors is provided instantly, which is a reasonable assumption comparing time constant of the current control loop (around 0.5 ms) to the time constant of the robot control loop (around 120 ms). Furthermore, motor friction is identified through Stribeck model[24] and subtracted from experimental results (Figure 4 (solid line)). Predicted torques of the model (Figure 4 (dashed line with star marks)) are also calculated by giving the real trajectory travelled by robot (figure 3) to the proposed dynamic model in equation (39). As it can be seen in Figure 4, the dynamic model prediction closely matches experimental results. Also note that there are discontinuous bumps in experimental torques when it reaches to its extremum values. This happens when motors change from acceleration to deceleration and vice versa, and is attributed to motor-gearbox backlash.

Other than accuracy of the model, its computational cost is another important aspect. To analyse

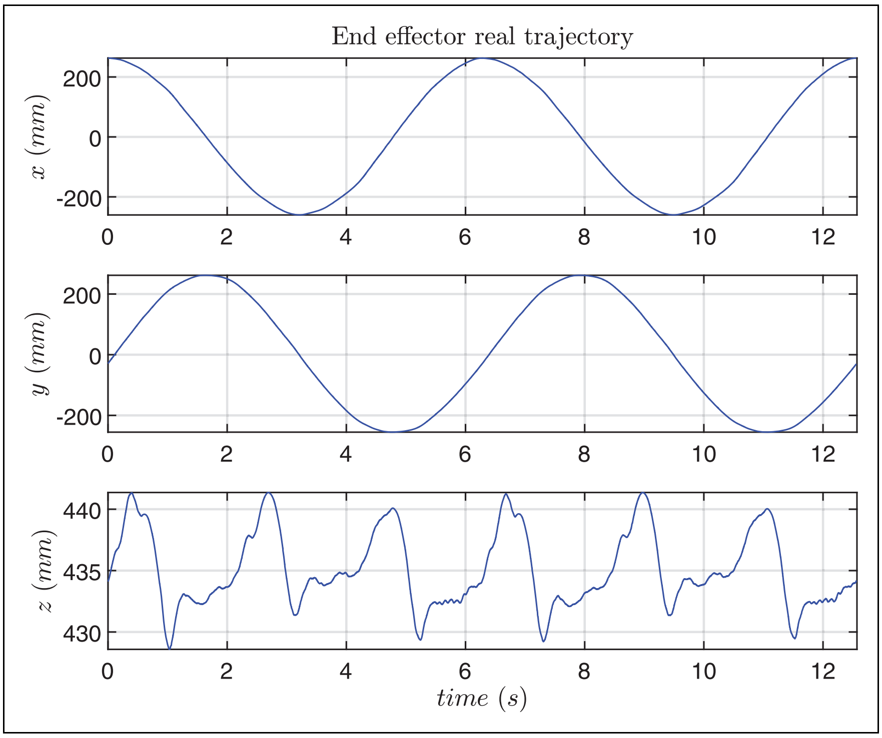

Figure 3. Trajectory passed by end effector.

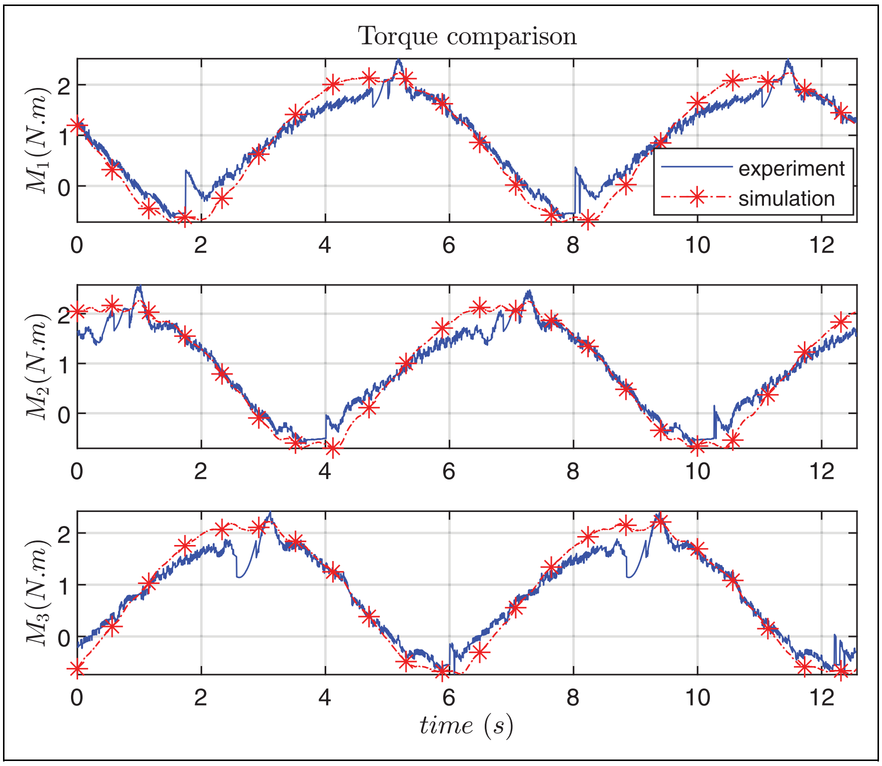

Figure 4. Torques.

this, average calculation time of the presented model is compared to the same property of the simplified implicit dynamics of the Delta robot based on Lagrange equations, given in Brinker et al.[18] On average, the proposed method takes 105% longer to be calculated. This matches the comparison done in Brinker et al.[18] and shows that the current explicit derivation does not add much computational cost to existing implicit virtual work-based models. It is also good to mention that the proposed model has minimum level of simplification in its derivation.

Properties of the Delta robot used in experiments are given in Appendix.

Conclusion

In this paper, an explicit dynamic model of the Delta robot is calculated analytically in a very compact form in comparison to the models in the referenced works. Friction in spherical joints and longitudinal moments of inertia of intermediary leg are neglected in the model. The model is obtained without any need for computer-based symbolic calculation. This is done using auxiliary Jacobians and implementing principle of virtual work. Using the auxiliary Jacobians simplified the final model effectively. The resulted model is derived explicitly and this makes it suitable for applying modern model-based control techniques. Also because of the well-organised structure of the model, contribution of each part of the robot to the dynamics can be understood easily. Furthermore, the proposed model is verified experimentally and good accuracy in experimental evaluation is observed.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research was partially supported by the United States National Science Foundation through Grant 1745212.

ORCID iD

Farshid Asadi https://orcid.org/0000-0003-2553-6808

References

1. Clavel R. Device for the movement and positioning of an element in space. Patent 4,976,582, USA, 1990.
2. Grange S, Conti F, Rouiller P et al. The delta haptic device. Technical report, Ecole Polytechnique Fdrale de Lausanne.
3. Bonev I. Delta parallel Robot - the story of success. *Newsletter*, www.parallelmic.org (2001, accessed April 2019).
4. Codourey A. Dynamic modelling and mass matrix evaluation of the delta parallel robot for axes decoupling control. In *Proceedings of IEEE/RSJ international conference on intelligent robots and systems. IROS'96*, Osaka, Japan, vol. 3. pp.1211–1218. Piscataway: IEEE.

5. Miller K and Clavel R. The Lagrange-based model of Delta-4 robot dynamics. *Robotersysteme* 1992; 8: 49–54.
6. Stamper RE. *A three degree of freedom parallel manipulator with only translational degrees of freedom*. Dissertation, University of Maryland, USA, 1997.
7. Tsai LW. The mechanics of serial and parallel manipulators. *Robot Analysis* 1999; 505.
8. Park SB, Kim HS, Song C et al. Dynamics modeling of a delta-type parallel robot (isr 2013). In: *IEEE ISR 2013*, Seoul, South Korea, pp.1–5. Piscataway: IEEE.
9. Tho TP and Thinh NT. Analysis of kinematics and dynamics of 4-dof delta parallel robot. In: *Robot intelligence technology and applications 2: results from the 2nd international conference on robot intelligence technology and applications*. New York: Springer International Publishing, 2014, pp.901–910.
10. Codourey A. Dynamic modeling of parallel robots for computed-torque control implementation. *Int J Robotics Res* 1998; 17: 1325–1336.
11. Kuo YL. Mathematical modeling and analysis of the delta robot with flexible links. *Comp Math Appl* 2016; 71: 1973–1989.
12. Miller K. Experimental verification of modeling of delta robot dynamics by direct application of Hamilton's principle. In *Proceedings of 1995 IEEE international conference on robotics and automation*, Nagoya, Japan, vol. 1, pp.532–537. Piscataway: IEEE.
13. Codourey A and Burdet E. A body-oriented method for finding a linear form of the dynamic equation of fully parallel robots. In: *Proceedings of international conference on robotics and automation*, Albuquerque, NM, USA, vol. 2, pp.1612–1618. Piscataway: IEEE.
14. Wang J and Gosselin CM. A new approach for the dynamic analysis of parallel manipulators. *Multibody Syst Dyn* 1998; 2: 317–334.
15. Staicu S. Recursive modelling in dynamics of delta parallel robot. *Robotica* 2009; 27: 199–207.
16. Carp-Ciocardia D et al. Dynamic analysis of clavel's delta parallel robot. In: *2003 IEEE international conference on robotics and automation (Cat. No. 03CH37422)*, vol. 3, pp.4116–4121. Piscataway: IEEE.
17. Clavel R. Conception d'un robot parallèle rapide à 4 degrés de liberté. Technical report, EPFL, 1991.
18. Brinker J, Corves B and Wahle M. A comparative study of inverse dynamics based on Clavels Delta robot. In *Proceedings of the 14th world congress in mechanism and machine science*, Taipei, Taiwan. pp.25–30, Germany: RWTH Publications.
19. Brinker J, Ingenlath P and Corves B. A study on simplified dynamic modeling approaches of Delta parallel robots. In: Jadran L and Jean PM (eds) *Advances in robot kinematics 2016*. New York: Springer, 2018. pp.119–128.
20. Tsai LW. Solving the inverse dynamics of a Stewart-Gough manipulator by the principle of virtual work. *J Mech Des* 2000; 122: 3–9.
21. Williams II RL. The delta parallel robot: kinematics solutions. www.ohio.edu/people/williar4/html/pdf/DeltaKin.pdf (2016, accessed April 2019).
22. Hsu K, Karkoub M, Tsai MC, et al. Modelling and index analysis of a delta-type mechanism. *Proc IMechE, Part K: J Multi-body Dynamics* 2004; 218: 121–132.
23. Ginsberg J. *Engineering dynamics*, vol. 10. Cambridge: Cambridge University Press, 2008.
24. Olsson H, Åström KJ, De Wit CC, et al. Friction models and friction compensation. *Eur J Control* 1998; 4: 176–195.

Appendix

Notation

Geometrical parameters

$r_b = 0.2$; m	fixed platform radius
$r_a = 0.2$; m	moving platform radius
$\varphi_1 = \pi$	1st leg position angle
$\varphi_2 = -\frac{\pi}{3}$	2nd leg position angle
$\varphi_3 = \frac{\pi}{3}$	3rd leg position angle
$l_L = 0.2$;m	motor leg length
$r_L = 0.3933$	motor leg COM length ratio
$l_K = .52$; m	intermediary leg length
$r_K = .5$	intermediary leg COM length ratio

$$R_b = \begin{bmatrix} 0 & r_b & 0 \end{bmatrix}^T$$

$$R_a = \begin{bmatrix} 0 & r_a & 0 \end{bmatrix}^T$$

Inertial parameters

$d_K = 0.08$; m	intermediary leg distance from each other at every joint
$I_K = .5.74769459 \times 10^{-3}$; kg.m^2	intermediary leg pairs dominant inertia

$$I_{Kmatrix} = \begin{bmatrix} 2840617 & 0 & 0 \\ 0 & 2840617 & 0 \\ 0 & 0 & 1016 \end{bmatrix} \times 10^{-9}; \text{kg.m}^2$$

single intermediary leg inertia matrix
$I_L = 6.4345319 \times 10^{-4}$; kg.m^2 motor leg inertia

$I_{mo} = 0.0465475$; kg.m^2	motor inertia
$m_K = 2 \times 0.05788$; kg	intermediary leg mass
$m_L = 0.116$; kg	motor leg mass
$m_P = 1.055$; kg	moving platform mass