Discrete Time Delay Feedback Control of Stewart Platform with Intelligent Optimizer Weight Tuner

Farzam Tajdari[1], Mahsa Tajdari[2], and Amin Rezaei[3]

Abstract—In the presence of complicated kinematic and dynamic, we present a generalizable robust control technique for the 6-Degree of Freedom (6DoF) Stewart integrated platform with revolving, time-delayed torque control actuators to achieve faster, and reliable efficiency for parallel control manipulators. The suggested optimal solution involves the construction of a time-delay Linear Quadratic Integral (LQI) controller integrated with an on-line Artificial Neural Network (ANN) as the cost function gain tuner. The controller is formulated to robustly mitigate the nonlinear system's real-time tracking error with large time-delay, which is implemented via ADAMS software. The method is validated through simulation experiments to demonstrate that the developed methodology is practical, optimum, and zero-error convergence.

Index Terms—Time-Delay, Integral Control, Stewart Platform, Artificial Neural Network, Optimal Solution.

I. INTRODUCTION

Parallel systems are recommended over serial manipulators for applications such as surgeries, scanning, and 3D printing that require higher speeds and acceleration, greater accuracy, and the lightest weight [1]. The invention of anatomically adaptable recovery equipment [2] and the upcoming generation of Ultra Personalized Products and Services (UPPS) [3], [4] are of particular concern due to the intense importance of robot-influenced well-being and human life. Rapid automatic breast scanning [5] is an interesting scanning scenario for the study's potential implementation, considering the fact that it is a very complex procedure due to the flexibility and deformity of breast tissue, particularly in motion scanning. The extremely high precision necessitates a thorough knowledge of parallel robotics, including sensor or actuator time delays and implementation objections [6]. This examination necessitates a thorough understanding of the vector, complex, and complicated control approaches. Well-known parallel robots with widespread use are derived from the mechanism suggested by Stewart platform [7] as the basis of recent parallel robots in the applications mentioned above. The complex approach explains the interaction between internal or external forces and torques, as well as the state of the end-effector controller, in terms of the dynamic equations' main problems. With respect to dynamics, many approaches have been used, including the momentum-based approach [8], the Newton-Euler methodology [9] and the Lagrangian approach [10]. The key aim of the parallel robot control system architecture is to ensure an exact continuation for the target position and orientation of dynamic and static variables in the moving end-effector of the robot [11]. Due to their extreme complexity, the robot's dynamics have lower control methods for the rotary Stewart robot while a wide range of control mechanisms are available, for example, optimum feedback robotic control [12]; backstage adaptive control [13]–[15]; and backstage adaptive control [16], [17]. In order to directly exploit the movement of the end-effector, reverse dynamic controls are extremely necessary. In limited studies [18], however, rotary torque controls are explored and provide the possibility of further use of recognition technologies, non-linear testing methods and impedance control systems. Moreover, few of them studied time delays (particularly in the operational file of robots) as an undeniable part of robotic controllers in the studies on rotary control Stewart, which highlighted high accuracy. Time-Delay Control (TDC) is generally considered an optimal way to solve this problem. The TDC scheme is also used for large applications such as robot manipulators [19], and flexible robots [20]. Time-delay estimation (TDE) errors will eventually be estimated by using time-related controls in functional management approaches. There are also increased TDE errors, which allow the trust of the closed-loop system regulations to be extremely complicated and non-linear (e.g. saturation.) As a general solution, TDC is the main benefit of its model-free functionality.

In the face of a large time lag with the actuators to maximized torques and reduce tracking errors, the objective of this paper is to establish and test a modern, stable torque management strategy for the complex parallel system. In addition, the controller is evaluated by a valid non-linear test bed using the ADAMS model proposed for the intelligent implementation of [21], [22] and experimentally analyzed using a robot manufactured.

II. EQUATIONS OF MOTION

A. Kinematics equation

The power torques of the motors and their rotational angles are controllable variables on a Stewart base powered by rotat-

[1]Mechatronic Design Engineering, Faculty of Industrial Design Engineering, Delft University of Technology, Delft, Netherlands `f.tajdari@tudelft.nl`

[2]Department of Mechanical Engineering, Northwestern University, Evanston, IL, USA `mahsa@u.northwestern.edu`

[3]Department of Computer Engineering and Computer Science, California State University Long Beach, Long Beach, CA, USA `me@aminrezaei.com`
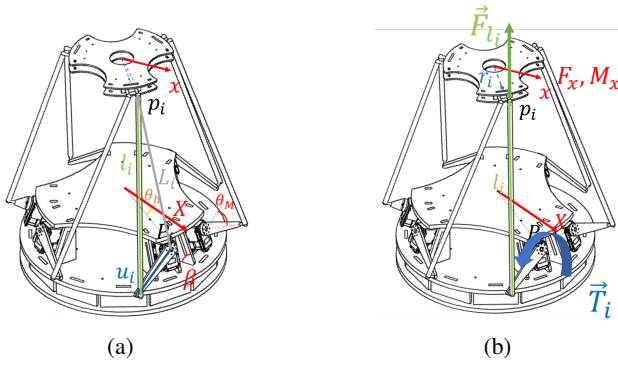
Fig. 1: Schematic of a Stewart platform. (a) Defined variables and vectors. (b) Dynamic force-torque diagram.

ing engines. The relationship between the motor variables and the end-effector position is stated in the following equations:

$$\bar{L} = f(\bar{X}), \ \theta_M = g(\bar{L}), \tag{1}$$

where vector $\bar{X} = [\phi \ \theta \ \psi \ x \ y \ z]^T$ represents the end-effector's variables which are the rotation angles ϕ, ψ, and θ of the moving plate about the X, Y and Z axis respectively, and the position of the end-effector center of mass as x, y, and z with respect to the fixed coordinate XYZ, according to Fig. 1(a). Moreover, the vector $\bar{L} = [L_1 \ L_2 \ L_3 \ L_4 \ L_5 \ L_6]^T$ representing the distance between the joints on the fixed base ($\vec{P}_i$) and the corresponding joints on the movable frame ($\vec{p}_i$), and eventually $\theta_M = [\theta_{M_1} \ \theta_{M_2} \ \theta_{M_3} \ \theta_{M_4} \ \theta_{M_5} \ \theta_{M_6}]^T$ is the motor angular shown in Fig. 1(a). We've used the findings of the same model in [21] to find the relationship between Li and θMi, then

$$\theta_{M_i} = -(asind\left(\frac{\mathcal{C}_i}{\sqrt{\mathcal{A}_i^2 + \mathcal{B}_i^2}}\right)$$
$$- acosd\left(\frac{\mathcal{B}_i}{\sqrt{\mathcal{A}_i^2 + \mathcal{B}_i^2}}\right) + 180), \tag{2}$$

where $\mathcal{A}_i, \mathcal{B}_i, \mathcal{C}_i \in U(\bar{X})$ and

$$\mathcal{A}_i = 2|\vec{u}_i| \left(|\vec{L}_{i_x}|Sin(\theta_h) - |\vec{L}_{i_y}|Cos(\theta_h)\right), \tag{3}$$

$$\mathcal{B}_i = 2|\vec{u}_i|(|\vec{L}_{i_x}|Cos(\beta)Cos(\theta_h)$$
$$+ |\vec{L}_{i_y}|Cos(\beta)Sin(\theta_h) - |\vec{L}_{i_z}|Sin(\beta)), \tag{4}$$

$$\mathcal{C}_i = |\vec{l}_i|^2 - |\vec{L}_i|^2 \tag{5}$$

in which θ_h is the angle between $\vec{P}_i$ and $\vec{X}$ in global axis, and β is the installation angle between the motor and horizon. The equation explained in (2) specifies that with a $\bar{X}$ on the end-effector sensor, the angle value of each motor is defined. What helps to monitor these robots location.

B. Kinetics equations

In this section, the dynamic equations of the Stewart platform powered by six rotary motors are obtained using the Newton-Euler method. The derivations are derived only in order to illustrate the different dynamic properties of the systems and the effects. The Stewart system with six rotary

motors as actuators studied in this paper is shown in Fig. 1 (a). The frame comprises, as can be seen, of a revolving plate as an end-actuator, of a fixed plate as a base and, as an end-actuator manipulator, of six legs connected to six rotary actuators. The circular joints are used to connect the end-effector and the spine to the six thighs. The kinetics equations for end-effector can be written as:

$$\Sigma \vec{M} = \bar{I}\vec{\alpha}, \ \Sigma \vec{F} = \bar{m}\vec{a} \tag{6}$$

where $\vec{\alpha}$ is the angular acceleration vector, and $\vec{a}$ is the linear acceleration vector of the end-effector. In addition, $\bar{m}$ contains the mass value (m) of the end-effector, and $\bar{I}$ includes the end-effector moment of inertia around x, y, and z axis, then

$$\begin{bmatrix} \bar{I} & 0_{3\times3} \\ 0_{3\times3} & \bar{m} \end{bmatrix} X = \begin{bmatrix} M \\ F \end{bmatrix} \tag{7}$$

$$\bar{I} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}, \quad \bar{m} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & m \end{bmatrix} \tag{8}$$

and,

$$M = [M_x, \ M_y, \ M_z]^T, \quad F = [F_x, \ F_y, \ F_z]^T. \tag{9}$$

In Fig. 1(b), M and F represents the torques and the exerted forces on the end-effector, respectively. The complex equation should be driven in the following way, when discussing the torque-manipulation of the rotary motors

$$\begin{bmatrix} \bar{I} & 0_{3\times3} \\ 0_{3\times3} & \bar{m} \end{bmatrix} \ddot{X} = \begin{bmatrix} M \\ F \end{bmatrix} = \tau_{6\times6}T \tag{10}$$

where $T = [T_1 \ T_2 \ T_3 \ T_4 \ T_5 \ T_6]^T$, and each component of T as T_i is the manipulated torque with motor i. The kinetics equations can also be formed using their unit vectors. We can write:

$$\Sigma \vec{M} = \Sigma \vec{r}_i \times \vec{F}_{l_i} = \Sigma \vec{p}_i \times \vec{e}_{l_i}|\vec{F}_{l_i}| = \Sigma \vec{e}_{M_i}|\vec{F}_{l_i}| \tag{11}$$

in which, $\vec{e}_{M_i} = \vec{p}_i \times \vec{e}_{l_i}$ (i.e., $\vec{e}$ denotes the unit vector of its corresponding vector). Then, the summation of the exact forces can be obtained by:

$$\Sigma \vec{F} = \Sigma \vec{e_{l_i}}|\vec{F}_{l_i}|, \tag{12}$$

consequently,

$$\begin{bmatrix} M \\ F \end{bmatrix} = \begin{bmatrix} e_{M_1} & \cdots & e_{M_6} \\ e_{l_1} & \cdots & e_{l_6} \end{bmatrix} \begin{bmatrix} F_{l_1} \\ \vdots \\ F_{l_6} \end{bmatrix} \tag{13}$$

where,

$$\tau_1 = \begin{bmatrix} e_{M_1} & \cdots & e_{M_6} \\ e_{l_1} & \cdots & e_{l_6} \\ . \end{bmatrix}. \tag{14}$$

As $\vec{F}_{l_i} = \vec{e}_{l_i} \cdot \vec{e}_{N_i} \frac{|\vec{T}_i|}{|\vec{u}_i|}$ where $\vec{e}_{N_i}$ is the unit vector of $\vec{T}_i$, then

$$\begin{bmatrix} F_{l_1} \\ \vdots \\ F_{l_6} \end{bmatrix} = \begin{bmatrix} \frac{\vec{e}_{l_1} \cdot \vec{e}_{N_1}}{|\vec{u}_1|} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{\vec{e}_{l_6} \cdot \vec{e}_{N_6}}{|\vec{u}_6|} \end{bmatrix} \begin{bmatrix} T_1 \\ \vdots \\ T_6 \end{bmatrix}. \tag{15}$$
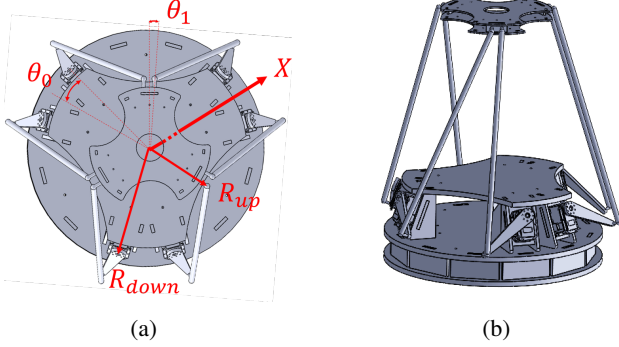
Fig. 2: 3D model of the Stewart platform designed with Solidworks software. (a) Top view. (b) Perspective view.

TABLE I: Dynamical features of the robot's components.

Component	Dimension (m)	Inertia $[I_x, I_y, I_z]$ $(\times 10^{-4} \text{kg.m}^2)$	Mass (kg)	Quantity
l_i	0.404	[16, 16, 0.01]	0.08	6
u_i	0.11	[0.01, 0.58, 0.6]	0.04	6
End-effector	Circle (R_{up}=0.12)	[30, 30, 60]	4	1
Base	Circle (R_{down}=0.22)	[270, 270, 540]	7	1

Considering,

$$\tau_2 = \begin{bmatrix} \frac{\vec{e}_{l_1} \cdot \vec{e}_{N_1}}{|\vec{u}_1|} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{\vec{e}_{l_6} \cdot \vec{e}_{N_6}}{|\vec{u}_6|} \end{bmatrix} \quad (16)$$

one can obtain

$$\begin{bmatrix} \bar{I} & 0_{3\times3} \\ 0_{3\times3} & \bar{m} \end{bmatrix} \ddot{\bar{X}} = \begin{bmatrix} M \\ F \end{bmatrix} = \tau_{6\times6} T = \tau_1 \tau_2 \, T. \quad (17)$$

Therefore, the ultimate dynamic transfer matrix, from the end-effector to the base is $\tau = \tau_1 \tau_2$, which defines the equations of motion based on the end-effector states and the applied torques by the actuators, which enables us to directly implement the torque control method as below:

$$\ddot{\bar{X}} = \tau_{tot} T \quad (18)$$

where,

$$\tau_{tot} = \begin{bmatrix} \bar{I} & 0_{3\times3} \\ 0_{3\times3} & \bar{m} \end{bmatrix}^{-1} \tau. \quad (19)$$

C. Non-linear model

In order to evaluate the controller design, it is vital to have a reasonable non-linear model to characterize the real device. While certain physical restrictions, such as collision of objects, stiffness and elasticity of rigid bodies, and the friction between hard surfaces, cannot not be considered in the MATLAB software, the ADAMS model can be efficiently used [21] to examine these physical constrains. Then, the 3D machine model that is shown in Fig. 2 can be integrated into ADAMS and the basic constrains can be implemented on the robot model. The given parameters in Fig. 2(a) and Fig. 1(a) are as follows: R_{up} =0.12 mm, R_{down} =0.22 mm, θ_0 =5°, and θ_1 =15°. In addition, the dynamic parameters are reported in Table I in which, the system consists of fourteen components of mass and moment of inertia.

III. CONTROLLER DESIGN

Given the fact that the dynamic system in the ADAMS software is developed based on the nonlinear parameters of the real system, it can be seen as a nonlinear Stewart platform; thus, in order to proficiently monitor the system, the proposed controller has to minimize the tracking errors in the presence of nonlinearities [23]. Besides robustness, input signals optimization of the parallel manipulator is another problem to be tackled in the control design step. In this respect, an optimal control unit for the Linear Quadratic Integral (LQI) is a strong and suitable candidate.

A. State-space equations

The state space and dynamic equations of the Stewart robot are formulated as follows:

$$\dot{\bar{x}}(t) = \bar{A}_c \bar{x}(t) + \bar{B}_c(t) T(t) + \bar{d}_c(t) \quad (20)$$

where $\bar{x}$ represents the main states in $\bar{X}$, and the speed of the states. Consequently, the error dynamic is defined is the following fashion:

$$\dot{\bar{e}}(t) = \bar{A}_c \bar{e}(t) + \bar{B}_c(t) T(t) + \bar{d}_c(t) \quad (21)$$

where, vector $\bar{d}_c$ depicts the time variant disturbance. $\bar{e}$ is the error matrix of the states and is defined as $\bar{e} = \bar{x} - x_d$, where x_d denotes the desired values of each states. In addition, matrix $\bar{A}_c$ includes all the $a_{i,j}$ elements in (22) which represents the system dynamics. Finally, matrix $\bar{B}_c$, carries $b_{i,j}$ elements in (23) which produces the interconnections between the applied torques and corresponding states as follows:

$$a_{i,j} = \begin{cases} 1 & \text{if } i = 2n - 1, \text{and } j = i + 1 \\ 0 & \text{otherwise} \end{cases} \quad (22)$$

$$b_{i,j} = \begin{cases} \tau_{tot}(\frac{i}{2}, j) & \text{if } i = 2n, \\ 0 & \text{otherwise} \end{cases} \quad (23)$$

where $n = 1, \cdots, 6$. Since in any real system the presence of disturbances (i.e., uncertainties in dynamic parameters,) is inevitable, an integral controller is used to eradicate constant disturbances [23]. The key control issue resulting from (21) can therefore be reformulated by taking into account integral states (z) of S (i.e., as many as the number of links,) described by:

$$\dot{z}(t) = \bar{C}_c \bar{e}(t) \quad (24)$$

where $\bar{C}_c$ consists of $\bar{c}_{i,j}$ denoted in the following way:

$$\bar{c}_{i,j} = \begin{cases} 1 & \text{if } i = n, \text{and } j = 2i - 1 \\ 0 & \text{otherwise} \end{cases} \quad (25)$$

hence, by adding the integral states, the resulting dynamic error of the system can be written as bellow:

$$\dot{e} = Ae(t) + B(t)T(t) + d(t) \tag{26}$$

$$e = \begin{bmatrix} \bar{e} \\ z \end{bmatrix}, \quad d = \begin{bmatrix} \bar{d}_c \\ 0_{S \times 1} \end{bmatrix} \tag{27}$$

$$A = \begin{bmatrix} \bar{A}_c & 0_{H \times S} \\ \bar{C}_c & 0_{S \times S} \end{bmatrix}, B = \begin{bmatrix} \bar{B}_c \\ 0_{S \times F} \end{bmatrix}, C = \begin{bmatrix} \bar{C}_c & 0_{n \times S} \\ 0_{S \times 2n} & I_{S \times S} \end{bmatrix}. \tag{28}$$

B. Time-delay control scheme

If we consider a time-delay of η for the system in (26) (η is the summation of all the delays including actuator and sensor), then the time variant form of the system is written as:

$$\dot{e}(t) = Ae(t) + BT(t - \eta)$$
$$y(t) = Ce(t). \tag{29}$$

Accordingly, the solution for the well-known differential equation in (29) is presented in [24]:

$$\dot{e}(t) = e^{A_t(t-t_0)}e(t_0) + \int_{t_0}^{t} e^{A(t-\zeta)}BT(\zeta)d\zeta \tag{30}$$

In the discrete time domain and considering only one step delay, t can be considered as

$$t_0 = k\,\Delta t, \quad t = k\,\Delta t + \Delta t. \tag{31}$$

Generally, it may happen that the resolution of the time step (Δt) cannot cover the delay of η, then $\eta = n\Delta t + \eta_0$, and

$$e(k+1) = e^{A\Delta t}e(k) + \int_0^{\eta_0} e^{A(\Delta t-\nu)}Bd\nu\ T(k-1)$$
$$+ \int_{\eta_0}^{\Delta t} e^{A(\Delta t-\nu)}Bd\nu\ T(k) \tag{32}$$

where $k\Delta t - \zeta = -\nu$, and $T(k-1)$ is the delayed output of $T(k)$ for one time step. From now on, the μ number of steps delayed output is shown as $T_{k-\mu}$ Which is considered as extra states to be controller as following,

$$e_{1_{dis}} = \begin{bmatrix} e \\ T_{k-1} \end{bmatrix}. \tag{33}$$

Then, the state space in (32) is formulated as

$$e_{1_{dis}}(k+1) = A_{1_{dis}}e_{1_{dis}}(k) + B_{1_{dis}}T(k) \tag{34}$$

where,

$$A_{1_{dis}} = \begin{bmatrix} e^{A\Delta t} & \int_0^{\eta_0} e^{A(\Delta t-\nu)}Bd\nu \\ 0_{n \times (N+S)} & 0_{n \times n} \end{bmatrix}$$

$$B_{1_{dis}} = \begin{bmatrix} \int_{\eta_0}^{\Delta t} e^{A(\Delta t-\nu)}Bd\nu \\ ------- \\ I_{n \times n} \end{bmatrix}, C_{1_{dis}} = \begin{bmatrix} C & 0_{S \times n} \end{bmatrix} \tag{35}$$

where, n is the number of the system inputs, S is the number of integral states, N denotes the number of elements in matrix

$\dot{e}$ in (21), and I is the identity matrix. Now, we explain the discrete system with μ steps delay which can be written as:

$$e_{1_{dis}}(k+1) = A_{1_{dis}}e_{1_{dis}}(k) + B_{1_{dis}}T_{k-\mu}$$
$$y(k) = C_{1_{dis}}e_{1_{dis}}(k). \tag{36}$$

Considering each delay line of the control signal as a single state of the discrete system and the total number of variable in $e_{1_{dis}}$ as $H = N + S + n$, then, the state variable matrix of the discrete system with the new states is

$$e_{dis(H+\mu n) \times 1} = \begin{bmatrix} e_{1_{dis}} \\ e_{dis_{H+1:H+n}} \\ \vdots \\ e_{dis_{H+(\mu-1)n+1:H+\mu n}} \end{bmatrix} = \begin{bmatrix} e_{1_{dis}} \\ T_{k-\mu} \\ \vdots \\ T_{k-1} \end{bmatrix}. \tag{37}$$

And, the system with μ-step delay is

$$e_{dis}(k+1) = A_{dis}e_{dis}(k) + B_{dis}T_k$$
$$y(k) = C_{dis}e_{dis}(k) \tag{38}$$

where,

$$A_{dis} = \begin{bmatrix} A_{1_{dis}} & B_{1_{dis}} & 0_{H \times (\mu-1)n} \\ 0_{(\mu-1)n \times H} & 0_{(\mu-1)n \times n} & I_{(\mu-1)n \times (\mu-1)n} \\ 0_{n \times H} & 0_{n \times n} & 0_{n \times (\mu-1)n} \end{bmatrix}$$

$$B_{dis} = \begin{bmatrix} 0_{(H+(\mu-1)n) \times n} \\ I_{n \times n} \end{bmatrix}, C_{dis} = \begin{bmatrix} C_{1_{dis}} & 0_{S \times (H+\mu n)} \end{bmatrix}. \tag{39}$$

Now, we can define the following quadratic cost function over an infinite time horizon, which is defined for minimization of all states and control inputs:

$$\min J = \sum_{k=0}^{\infty} [e_{dis}(k)^T C_{dis}^T Q C_{dis} e_{dis}(k) + T^T(k)RT(k)] \tag{40}$$

where,

$$Q = \omega_Q I_{(H+\mu n) \times (H+\mu n)}, R = \omega_R I_{n \times n}. \tag{41}$$

The Q and R matrices are weight matrices that tend to minimize the error of all states and the control signals. And, $\omega_Q > 0$ and $\omega_R > 0$ are formulated for weighing matrices.

A Linear Quadratic Regulator (LQR) is used to solve the resultant optimal control problem (40), (41), in which the goal is to stabilize feedback gains via the assumption. Although the stability and detectability parameters are fulfilled according to Hautus-test in [25]–[27] extended to [21] by the initial resulting method.

C. Controller design

In order to solve the LQI problem, a linear feedback control law is proposed as follows:

$$T(k) = -Ke_{dis}(k) \tag{42}$$

where

$$K = (R + B_{dis}^T P B_{dis})^{-1} B_{dis}^T P A_{dis}, \tag{43}$$

$$P = C_{dis}^T Q C_{dis} + A_{dis}^T P A_{dis}$$
$$- A_{dis}^T P B_{dis} \left(R + B_{dis}^T P B_{dis} \right)^{-1}. \quad (44)$$

Resulting K in (43) as optimal gain, and the Algebraic Riccati Equation (44) are investigated in [28].

D. Intelligent estimator design

As a function of time variables in the dynamic equation (20), $B_{dis}(k)$ is the time variable in (38) according to (23), which leads to the conclusion that the process behaves differently in various paths. Thus, using intelligent approaches is an achievement in solving non-linearity and maximum efficiency [29]–[34]. As the cost efficiency can be maximized by well-defining the w_Q and w_R in (40) it's a path dependence approach to approximate true variables by time.

Where w_Q indicates the degree of state error minimization and w_R optimizes the torque values. An ANN-based estimator will be defined where the input variables are the density of the functions of the state error and the torque density of the function and the outputs of the estimator are w_Q and w_R.

The density functions are derived from [21] which describes the density function for any arbitrary ϵ parameter by the following equation:

$$\rho_\epsilon(k) = \Omega_\epsilon(\epsilon(k)) = (w_{in} + 1) \frac{\epsilon(k)}{\sum_{i=k-w_{in}}^{k} \epsilon(i)} \quad (45)$$

where ρ_ϵ denotes the parameter's density function. When the average of the parameter with window size $w_{in} + 1$ is significantly less than $\epsilon(k)$, the ρ_ϵ value becomes highly sensitive. As a consequence, the ρ_ϵ that is appropriate is the one that has the least divergence from the average values. This gives the a parameter changes a smooth and rational instigation, according to [21]. Therefore, the inputs of the intelligent estimator are $\rho_{\bar{e}_{1 \times 6}}$ and $\rho_{T_{1 \times 6}}$, and the outputs are $w_{Q_{1 \times 1}}$, and $w_{R_{1 \times 1}}$ as shown in Fig. 3.

We used controller output, for $w_{in} = 30$, and w_Q and w_R as 0.1, 1, 4 and various quantities of delay (μ) as 0, 5, 10, 20 and 30 intervals including 45 experimental intervals in all. We have repeated the above measurements with frequencies of 0.5 Hz, 2 Hz, 4 Hz, and 6 Hz to consider the frequency effects of the desired values, so that the final datasets are composed of 180, and the sampling time during these tests is 0.1s and the time spent is 10 s. That ends the data set with 18000 samples of the input-output.

Regarding the Fig. 3, the input layer consists of 18 neurons, the hidden layer contains 10 neurons, and the output layer includes 2 neurons. Also, the employed activation functions are linear and sigmoid for the input and the output, and the hidden layer respectively. In addition, the training algorithm of the ANN is a Levenberg-Marquardt back-propagation method based on [35]. A collection of data, including the inputs-outputs, is needed to design an ANN-based estimator. As a result, the estimator's controller (42) is used as a master. Thus the randomly chosen 70% data set for training is considered, while the remaining 30% is used to test the model collected.
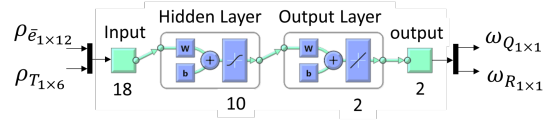

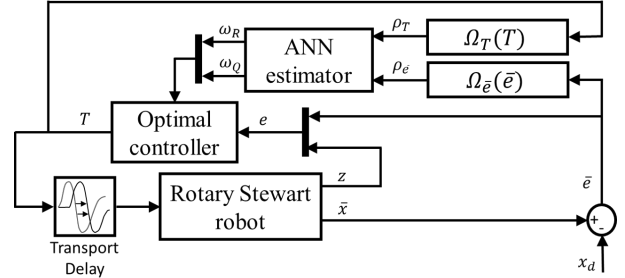Fig. 3: Structure of the ANN-based intelligent estimator.

Fig. 4: Closed-loop time-delay control diagram.

The closed-loop optimum feedback management diagram of the proposed device displays in Fig. 4 where the ANN estimator changes the controller gains of w_Q and w_R at each step, depending on error states and torque values.

IV. SIMULATION RESULTS

The results of the simulation are produced by implementing $w_{in} = 30$, $\mu = 8$, and $\Delta t = 0.1$ s in ADAMS, while the desired variables ($\bar{X}_d$) are considered to be following:

$$\bar{X}_d(t) = \left[\phi_d(t), \theta_d(t), \psi_d(t), x_d(t), y_d(t), z_d(t) \right]^T =$$
$$\left[\tfrac{\pi}{12}, \tfrac{\pi}{12}, \tfrac{\pi}{12}, 0.02, 0.02, .08 \right]^T Sin(t) + \left[0, 0, 0, 0, 0, .38 \right]^T \quad (46)$$

where t is time. The corresponding simulation results are obtained by implementing a proposed controller (42). Two different control methodologies (controllers without integral states, integral controllers, intelligent time-delay controller without integral state, and intelligent time-delay integral controller) are compared, according to Fig. 5(a). While the first two controllers are technically stable without any delay in the actuator, as discussed in [21], they cannot monitor the robot in the presence of a time-delay (cyan and green line) due to the unique dynamics of the robots (resulting from pair instability ($\bar{A}_c, \bar{B}_c$)). In Section III-C, the other two controllers have a stable output, however, with finite errors. The intelligent integrated time-delay controller ensures flawless monitoring (looking at Fig. 5(a) on the blue lines and refuses errors as seen in Fig. 5 (b).

The reduction in error was completed by an integral status zero error function and with well-specified w_Q and w_R controller time variant parameters recorded in Fig. 6 (a). The smart time-delay controller without any detailed states does, however, cause a substantial error, especially in the crucial points of the required path (i.e. the output of the controller with respect to a x, y and z in Fig. 5(b)). This is a discrepancy even though the controller attempts to generate a periodic motion close to the desired value.
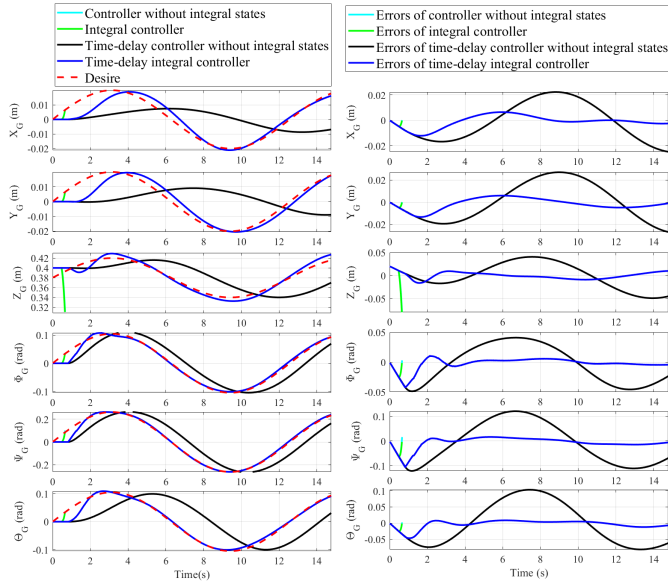
Fig. 5: Controlled case performance via the ADAMS model. (a) Main states ($\bar{X}$). (b) Error of Main states ($\bar{X} - \bar{X}_d$)

TABLE II: Accumulative absolute percentage error (E_{ac}) report.

Controller	E_{ac}
Controller without integral states	NaN
Integral controller	NaN
Time-delay controller without integral states	225
Time-delay integral controller	80

To compare the experiments with regard to the monitoring error, a criterion with the accumulated absolute percentage error E_{ac} is added

$$E_{ac} = \frac{2}{H} \sum_{i=1}^{\frac{H}{2}} \int_0^t \left(\frac{|\bar{X}_i(t) - \bar{X}_{d_i}(t)|}{A_{m_i}} \times 100 \right) dt \qquad (47)$$

where E_{ac} is the sum of all absolute amounts, from the beginning to the end of the simulation. Where $i = 1, \ldots, \frac{H}{2}$, $t_{end} = 10$ s, and A_{m_i} represents the peak-to-peak magnitude of the desired values ($\bar{X}_{d_i}$) of the relative vector ($\bar{X}_i$) (where the desire is constant, A_{m_i} is presumed to be 1). The output of the controllers is then defined separately on the basis of the parameters in TABLE II. As for the table, all controllers without the time-delay scheme (the controller with no integral states and the integral controller) failed as expected, at around $t = 0.8$ s looking at the Figs. 5(a) and 5(b). While time-delay controllers kept the system in the allowed work-space, the time-delay integral controller had a significant advantage over the time-delay controller without integral states, with a 64% improvement. This reflects integral states' ability to reject unknown shocks and uncertainties. In comparison, Fig. 7 displays the torques that have been applied on the device for the two time-delay management scenarios.
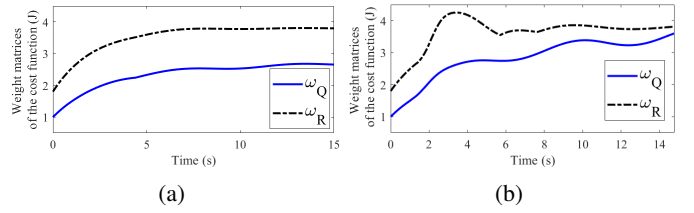

Fig. 6: Online estimated of w_Q, and w_R. (a) Time-delay controller without integral states. (b) Time-delay integral controller.

As the torques in Fig. 7(b) are compared to the torques in Fig. 7(a), the control effort produced by the intelligent time-delay integral controller decreases over time as the errors decrease. In addition, the Integral controller's torques have more trembling (from $t = 0$ s to $t = 5$ s) than the torques of the controller without integral states, which is attributable to the integral states' increased sensitivity to the tracking error due to (24). Due to the significant tracking error, the amplitude of the control signals produced by the intelligent time-delay controller without integral states is increasing.
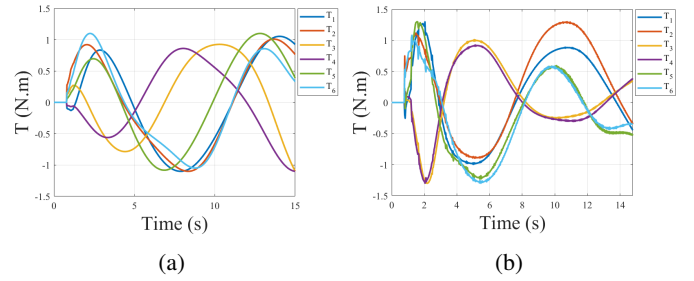

Fig. 7: Motor torques. (a) Time-delay controller without integral states. (b) Time-delay integral controller.

V. CONCLUSION

In this paper, we investigated an innovative control methodology for a Stewart platform parallel robot with delayed rotary actuator. We derived the equations of motion for the system, and optimally designed the dimensions of the platform. We also implemented a 3D model of the system in ADAMS software. The current dynamic is controlled online via the intelligent time-delay tuning of the LQI controller. The robustness, as well as the adaptive component, i.e., the ANN estimator, enables us for an easy practical implementation, without the necessity of lengthy and costly measurements of the dynamic parameters. Finally, the simulation results showed that the ideal time-delay LQI controller with an intelligent estimator was able to monitor the dynamics of the device and eradicate the error of monitoring given the presence of actual dynamic forces and the enormous delay of the actuators in the ADAMS model. Numerically, we showed that the time-delay intelligent integral controller achieved 64% improvement in terms of accumulative absolute percentage error (E_{ac}). Further developments include the study of the controller parameters sensitivity, robustness of the method in presence of noise and actuator saturation which will appear in our future scholar.

REFERENCES

[1] H. D. Taghirad, *Parallel robots: mechanics and control*. CRC press, 2013.

[2] K. Nomura, T. Yonezawa, H. Takemura, and H. Mizoguchi, "Development of six-dof human ankle motion control device using stewart platform structure for fall prevention," *Journal of Robotics and Mechatronics*, vol. 28, no. 5, pp. 654–663, 2016.

[3] Y. Yang, T. Yuan, T. Huysmans, W. S. Elkhuizen, F. Tajdari, and Y. Song, "Posture-invariant 3d human hand statistical shape model," *Journal of Computing and Information Science in Engineering*, vol. 21, no. 3, p. 031006, 2021.

[4] M. Tajdari, A. Pawar, H. Li, F. Tajdari, A. Maqsood, E. Cleary, S. Saha, Y. J. Zhang, J. F. Sarwark, and W. K. Liu, "Image-based modelling for adolescent idiopathic scoliosis: Mechanistic machine learning analysis and prediction," *Computer methods in applied mechanics and engineering*, vol. 374, p. 113590, 2020.

[5] Y. Sun, Z. Jiang, X. Qi, Y. Hu, B. Li, and J. Zhang, "Robot-assisted decompressive laminectomy planning based on 3d medical image," *IEEE Access*, vol. 6, pp. 22 557–22 569, 2018.

[6] M. Shoham, M. Burman, E. Zehavi, L. Joskowicz, E. Batkilin, and Y. Kunicher, "Bone-mounted miniature robot for surgical procedures: Concept and clinical applications," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 5, pp. 893–901, 2003.

[7] D. Stewart, "A platform with six degrees of freedom," *Proceedings of the institution of mechanical engineers*, vol. 180, no. 1, pp. 371–386, 1965.

[8] A. M. Lopes, "Dynamic modeling of a stewart platform using the generalized momentum approach," *Communications in Nonlinear Science and Numerical Simulation*, vol. 14, no. 8, pp. 3389–3401, 2009.

[9] B. Dasgupta and T. Mruthyunjaya, "A newton-euler formulation for the inverse dynamics of the stewart platform manipulator," *Mechanism and machine theory*, vol. 33, no. 8, pp. 1135–1152, 1998.

[10] Z. Bingul and O. Karahan, *Dynamic modeling and simulation of Stewart platform*. INTECH Open Access Publisher, 2012.

[11] J.-P. Merlet, *Parallel robots*. Springer Science & Business Media, 2006, vol. 128.

[12] H. Tourajizadeh, M. Yousefzadeh, and A. Tajik, "Closed loop optimal control of a stewart platform using an optimal feedback linearization method," *International Journal of Advanced Robotic Systems*, vol. 13, no. 3, p. 134, 2016.

[13] C.-I. Huang and L.-C. Fu, "Adaptive backstepping tracking control of the stewart platform," in *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, vol. 5. IEEE, 2004, pp. 5228–5233.

[14] F. Tajdari, M. Kabganian, E. Khodabakhshi, and A. Golgouneh, "Design, implementation and control of a two-link fully-actuated robot capable of online identification of unknown dynamical parameters using adaptive sliding mode controller," in *2017 Artificial Intelligence and Robotics (IRANOPEN)*. IEEE, 2017, pp. 91–96.

[15] F. Tajdari, N. E. Toulkani, and N. Zhilakzadeh, "Semi-real evaluation, and adaptive control of a 6dof surgical robot," in *2020 11th Power Electronics, Drive Systems, and Technologies Conference (PEDSTC)*. IEEE, 2020, pp. 1–6.

[16] J. Velasco Pascua, I. Calvo Gordillo, O. Barambones Caramazana, P. Venegas, and C. Napole, "Experimental validation of a sliding mode control for a stewart platform used in aerospace inspection applications," 2020.

[17] F. Tajdari, M. Kabganian, N. F. Rad, and E. Khodabakhshi, "Robust control of a 3-dof parallel cable robot using an adaptive neuro-fuzzy inference system," in *2017 Artificial Intelligence and Robotics (IRANOPEN)*. IEEE, 2017, pp. 97–101.

[18] X. Yang, H. Wu, B. Chen, S. Kang, and S. Cheng, "Dynamic modeling and decoupled control of a flexible stewart platform for vibration isolation," *Journal of Sound and Vibration*, vol. 439, pp. 398–412, 2019.

[19] M. Jin, S. H. Kang, P. H. Chang, and J. Lee, "Robust control of robot manipulators using inclusive and enhanced time delay control," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 5, pp. 2141–2152, 2017.

[20] S. Roy, I. N. Kar, J. Lee, N. G. Tsagarakis, and D. G. Caldwell, "Adaptive-robust control of a class of el systems with parametric variations using artificially delayed input and position feedback," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 2, pp. 603–615, 2017.

[21] F. Tajdari, N. E. Toulkani, and N. Zhilakzadeh, "Intelligent optimal feedback torque control of a 6dof surgical rotary robot," in *2020 11th Power Electronics, Drive Systems, and Technologies Conference (PEDSTC)*. IEEE, 2020, pp. 1–6.

[22] F. Tajdari and N. E. Toulkani, "Implementation, and intelligent gain tuning feed-back based optimal torque control of a rotary parallel robot," *Journal of Vibration and Control*, 2021, In Press.

[23] K. J. Åström and T. Hägglund, *PID controllers: theory, design, and tuning*. Instrument society of America Research Triangle Park, NC, 1995, vol. 2.

[24] I. Karafyllis and M. Krstic, "Stabilization of nonlinear delay systems using approximate predictors and high-gain observers," *Automatica*, vol. 49, no. 12, pp. 3623–3631, 2013.

[25] F. Tajdari, C. Roncoli, N. Bekiaris-Liberis, and M. Papageorgiou, "Integrated ramp metering and lane-changing feedback control at motorway bottlenecks," in *2019 18th European Control Conference (ECC)*. IEEE, 2019, pp. 3179–3184.

[26] F. Tajdari, C. Roncoli, and M. Papageorgiou, "Feedback-based ramp metering and lane-changing control with connected and automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[27] F. Tajdari and C. Roncoli, "Adaptive traffic control at motorway bottlenecks with time-varying fundamental diagram," in *CTS2021: 16th IFAC Symposium on Control in Transportation Systems, June, Lille, France*. Elsevier, 2021, In Press.

[28] H. Navvabi and A. Markazi, "New afsmc method for nonlinear system with state-dependent uncertainty: Application to hexapod robot position control," *Journal of Intelligent & Robotic Systems*, vol. 95, no. 1, pp. 61–75, 2019.

[29] A. Ghaffari, A. Khodayari, A. Kamali, F. Tajdari, and N. Hosseinkhani, "New fuzzy solution for determining anticipation and evaluation behavior during car-following maneuvers," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of automobile engineering*, vol. 232, no. 7, pp. 936–945, 2018.

[30] B. Tarvirdizadeh, A. Golgouneh, E. Khodabakhshi, and F. Tajdari, "An assessment of a similarity between the right and left hand photoplethysmography signals, using time and frequency features of heart-rate-variability signal," in *2017 IEEE 4th international conference on knowledge-based engineering and innovation (KBEI)*. IEEE, 2017, pp. 0588–0594.

[31] A. Khodayari *et al.*, "A new model of car following behavior based on lane change effects using anticipation and evaluation idea," *Iranian Journal of Mechanical Engineering Transactions of the ISME*, vol. 16, no. 2, pp. 26–38, 2015.

[32] F. Tajdari, A. Ghaffari, A. Khodayari, A. Kamali, N. Zhilakzadeh, and N. Ebrahimi, "Fuzzy control of anticipation and evaluation behaviour in real traffic flow," in *2019 7th International Conference on Robotics and Mechatronics (ICRoM)*. IEEE, 2019, pp. 248–253.

[33] F. Tajdari, N. E. Toulkani, and M. Nourimand, "Intelligent architecture for car-following behaviour observing lane-changer: Modeling and control," in *2020 10th International Conference on Computer and Knowledge Engineering (ICCKE)*. IEEE, 2020, pp. 579–584.

[34] H. Li, O. L. Kafka, J. Gao, C. Yu, Y. Nie, L. Zhang, M. Tajdari, S. Tang, X. Guo, G. Li *et al.*, "Clustering discretization methods for generation of material performance databases in machine learning and design optimization," *Computational Mechanics*, vol. 64, no. 2, pp. 281–305, 2019.

[35] B. Kosko and J. C. Burgess, "Neural networks and fuzzy systems," 1998.