



## Original Article

# More Realistic Dynamical Modeling of a Humanoid Robotic Arm to Reduce Control Effort with Uncertainty Using Self-Impact Joint Constrained

Seyed Iman Hosseini<sup>1</sup>, Yousef Bazargan Lari<sup>1\*</sup>

<sup>1</sup>Department of Mechanical Engineering, Shiraz Branch, Islamic University, Shiraz, Iran

## ARTICLE INFO

### Article History:

Received: 11/11/2018

Revised: 29/01/2019

Accepted: 28/04/2019

### Keywords:

Self-impact joint constraint

Uncertainty

More realistic

Power consumption

Please cite this article as:

Hosseini SI, Bazargan Lari Y. More Realistic Dynamical Modeling of a Humanoid Robotic Arm to Reduce Control Effort with Uncertainty Using Self-Impact Joint Constrained. JRSR. 2019;6(2):68-79. doi: 10.30476/JRSR.2019.75400.

## ABSTRACT

**Background:** The main challenge of modeling humanoid robots is establishing a compromise between the simplicity of the model and accuracy of the system. One of the realities of movement which is important about humans is moving their hands to keep balance and reduce energy consumption while they are walking.

**Methods:** In this context, the role of elbow joint and the limitation that the joint exerts in terms of movement on the forearm and arm as self-impact joint constraint is undeniable. This paper deals with modeling and control of humanoid robot's hand as double-pendulum will consider mentioned constraint while normal walking and also in throwing darts.

**Results:** The presence of the self-impact joint constraint contributed to about a 26% saving in power consumption of robot motors within an impact range of 0.6346 to 0.6896 during normal human walking.

Since this control has a high power, 10 to 30% of the uncertainty was added to the length and mass parameters. As was observed, this controller routed the desired curves in the least possible time.

**Conclusion:** As mentioned earlier, consideration of this constraint in elbow joint of the humanoid robot will help in approaching the reality of system in comparison with past models previously designed. As constraint causes addition of severe nonlinear terms to dynamic system equation, the control of systems with this type of constraint faces a great deal of complexity. For adaptive-neural controller to control of the system of humanoid robot's hand will be used. Also, to display the ability of control system, the uncertainty of length and mass for this system will be considered. The existence of self-impact joint constraint will cause saving in consumption power of robot engines within the impact range.

2019© The Authors. Published by JRSR. All rights reserved.

## Introduction

Nowadays, several humanoid robots have been designed and modeled, but dynamicity and modeling issues have been neglected in the actual behavior of the system. Clearly, one of the abilities and daily necessities

of humans is to move from one place to another. All humans move their hands in a reverse direction of their feet, while walking. In general, four models have been presented for moving hands while walking, as shown in Figure 1 [1]. Scientifically, the particular reason for this behavior is preserving the body balance. After boundless investigations, the researchers discovered that holding hands completely immobilized would consume 12%, and anti-normal moving will make it 26% more energy consuming than the normal state of the body. The above

\*Corresponding author: Yousef Bazargan Lari, Department of Mechanical Engineering, Shiraz Branch, Islamic University, P. O. Box: 71993-3, Shiraz, Iran  
Tel: +98 9177115105; E-mail: [bazarganlari@iaushiraz.ac.ir](mailto:bazarganlari@iaushiraz.ac.ir)

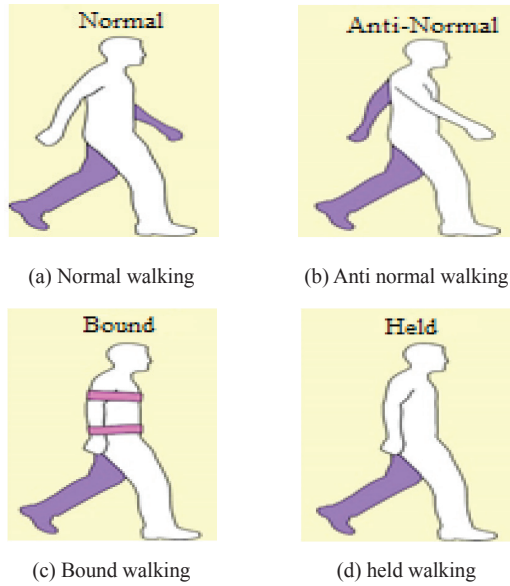


Figure 1: Different types of human’s hand in walking [1]

research indicates the importance of hand movement and modeling to achieve optimal energy consumption and longer walking cycle durability [1]. In 2011, Arellano and Karam [2] focused their research on the impact of step width and movement of human hands while running to reduce energy consumption and more stability of humans. In 2013, Means et al. [3] examined the performance of hand movement and its effect on human balance and stability while walking. In 2015, Pant et al. [4] investigated the swinging movement of human hands while walking to maintain balance and its importance in prevention of collapse. Kato et al. [5] reviewed the mechanical arm robot with two degrees of freedom during throwing along with its controlling process. According to the studies, most of the modeling performed on the human arm have not been in line with the actual model, and the phenomenon called a joint self-impact has not been considered. The arm movement modeling should be similarly close to reality otherwise malfunctions and complexities will emerge in sustainability and energy issues. The particular phenomenon of joint self-impact is one of the most recently introduced issues. However, due to the complexity of the system in terms of controlling and the existence of nonlinear parameters, this issue has not been studied extensively. It has not been discussed accurately and completely either in many modeling and studies, where the control methods and modeling systems have been implemented regardless of these constraints. In this paper, a constrained double pendulum, as a hand of a moving robot - an instance of systems in which these constraints occur in their movement – is investigated dynamically through a controlled condition.

Bazargan Lari et al. modeled this self-impact joint constraint for the human leg, at the first time. Initially, self-impact joint constraint was considered on the robot foot with the ability to rotate and afterward they monitored the shin and thigh during a walking cycle through an exact feedback linearization method [6].

Since it is inevitable to address the problem of parametric uncertainties such as mass and length of shin and thigh, they investigated this issue in their other research [7]. They also examined both human feet simultaneously, while walking, taking into account the constraints [8]. Given the importance of modeling the self-impact joint constraint according Figure 1 and its role in sustaining the stability while walking, the impact joint constraint is of paramount importance. Further, the presence, importance, and applicability of the constraint in sports such as dart throwing, tennis, weightlifting, golf and etc. are even more evident. In this paper, we investigate the presence of a self-impact joint constraint in the hands of humans while walking. Dart throw is also considered as an example of these sports. It will also be demonstrated that the presence of this constraint saves more power in a robot engine.

Dynamic Model Analysis

Hand Movement without Constraint

The scheme for double pendulum with no constraint is illustrated in Figure 2. In this form,  $\theta_1$  and  $\theta_2$  represent the angles of rotation of the elbow and shoulder relative to the vertical axis, respectively.  $\tau_1$  and  $\tau_2$  denotes the external torques that move the forearm and arm, respectively. As presented in the following equations, the dynamic equation of this system is modeled based on Hamilton Formula:

$$H = \sum_{i=1}^2 \theta_i P_{\theta_i} - L \tag{1}$$

$$P_{\theta_i} = \frac{\partial L}{\partial \dot{\theta}_i} \tag{2}$$

$$\theta_i = \frac{\partial H}{\partial P_{\theta_i}} \tag{3}$$

$$\theta_2 = \frac{-m_2 L_2 P_{\theta_1} \cos(\theta_1 - \theta_2) + (m_1 + m_2) L_1 P_{\theta_1}}{m_2 L_1 L_2^2 [m_1 + m_2 \sin^2(\theta_1 - \theta_2)]} \tag{4}$$

$$\dot{P}_{\theta_1} = -(m_1 + m_2) g L_1 \sin \theta_1 - h_1 + h_2 \sin [2(\theta_1 - \theta_2)] \tag{5}$$

$$\dot{P}_{\theta_2} = -m_2 g L_2 \sin \theta_2 + h_1 - h_2 \sin [2(\theta_1 - \theta_2)] \tag{6}$$

$$h_1 = \frac{P_{\theta_1} P_{\theta_2} \sin(\theta_1 - \theta_2)}{L_1 L_2 [m_1 + m_2 \sin^2(\theta_1 - \theta_2)]} \tag{7}$$

$$h_2 = \frac{m_2 L_2^2 P_{\theta_1}^2 + (m_1 + m_2) L_1^2 P_{\theta_2}^2 - 2 m_2 L_1 L_2 P_{\theta_1} P_{\theta_2} \cos(\theta_1 - \theta_2)}{2 L_1^2 L_2^2 [m_1 + m_2 \sin^2(\theta_1 - \theta_2)]^2} \tag{8}$$

$m_1 m_1$  and  $m_2 m_2$  indicate the mass and  $l_1 l_1$  and  $l_2 l_2$  represent the length of forearm and arm, respectively.

Movement of arm with consideration of constraint

In an unconstrained double pendulum, the rotation angle of the elbow and arm in joints is assumed without boundaries and limits (Figure 2). Nevertheless, this assumption is not possible for human hands in reality as the angle of the arm cannot be larger than the forearm angle (Figure 1). The weakness of this type of modeling, which does not represent the reality of the system

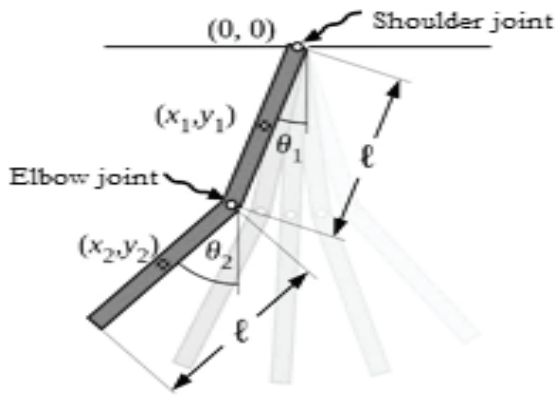


Figure 2: Schematics of an unconstrained double pendulum

of human hands, becomes further important as the controller tries to prevent the system from exceeding the permissible angles. Consequently, controlling effort is considerably increased compared to the condition where the constraint is mechanically considered in dynamic modeling (Figures 3 and 4).

*Self-Impact Constrained Joint*

In joints, the self-impact constraints are considered as inhibitors which are stated as follows:

$$F_n = c\delta + k\delta \tag{9}$$

Where, c and k [9] are torsional damper and stiffness factor, while  $\delta$  and  $\delta$  represent the rotational angle and angular velocity, respectively [10].

Additionally, the phenomenon of self-impact joint should be modeled by forces that are frequently applied in execution during the activation period. The dynamic equation in the double pendulum of the self-impact with the damper and torsion spring model is described as follows:

$$\dot{P}_{\theta_1} = -(m_1 + m_2)gL_1\sin\theta_1 - h_1 + h_2\sin[2(\theta_1 - \theta_2)] + Q_{nc} \tag{10}$$

$$\dot{P}_{\theta_2} = -m_2gL_2\sin\theta_2 + h_1 - h_2\sin[2(\theta_1 - \theta_2)] + Q_n \tag{11}$$

$$h_1 = \frac{P_{\theta_1}P_{\theta_2}\sin(\theta_1 - \theta_2)}{L_1L_2[m_1 + m_2\sin^2(\theta_1 - \theta_2)]} \tag{12}$$

$$h_2 = \frac{m_2L_1^2P_{\theta_1}^2 + (m_1 + m_2)L_1^2P_{\theta_2}^2 - 2m_2L_1L_2P_{\theta_1}P_{\theta_2}\cos(\theta_1 - \theta_2)}{2L_1^2L_2^2[m_1 + m_2\sin^2(\theta_1 - \theta_2)]^2} \tag{13}$$

$$Q_{nc} = U(\theta_2 - \theta_1)(k(\theta_1 - \theta_2) + c_1(\dot{\theta}_1 - \dot{\theta}_2)) \tag{14}$$

To solve the equations of the system, we adopt the approximation of the heavy side step function, which is a sigmoid function group. Below is an instance of a function which is utilized to approximate the heavy side function:

$$U(\theta_2 - \theta_1) \approx \frac{1}{2}(1 + \tanh[r(\theta_2 - \theta_1)]) = \frac{1}{1 + e^{-2r(\theta_2 - \theta_1)}} \tag{15}$$

Note that its approximation can be adjusted through increasing or decreasing r.

*Dynamic Equations Matrix*

The dynamic equations of the hand as a double-constrained pendulum are described as follows:

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) + \tau_f(q, \dot{q}) = \tau \tag{16}$$

Where, q,  $\dot{q}$ , and  $\ddot{q}$  are angular vectors, angular velocity, and angular acceleration of the joints, respectively. M(q) denotes symmetric positive inertia matrix, C(q,  $\dot{q}$ ) shows Coriolis vector,  $\dot{M}(q) - 2C(q, \dot{q})$  is unbalanced symmetric centrifugal torque matrix, G(q) indicates gravitational torque vector,  $\tau_f(q, \dot{q})$  reflects the approximate vector of the self-impact joint, and  $\tau$  is a joint actuator torque vector.

$$M(q) = \begin{bmatrix} m_1d_1^2 + I_1 + m_2l_1^2 & m_2l_1d_2\cos(\theta_1 - \theta_2) \\ m_2l_1d_2\cos(\theta_1 - \theta_2) & m_2d_2^2 + I_2 \end{bmatrix} \tag{17}$$

$$C(q, \dot{q}) = \begin{bmatrix} 0 & m_2l_1d_2\dot{\theta}_2\sin(\theta_1 - \theta_2) \\ m_2l_1d_2\dot{\theta}_1\sin(\theta_1 - \theta_2) & 0 \end{bmatrix} \tag{18}$$

$$G(q) = \begin{bmatrix} m_2gd_2\sin\theta_1 \\ -g\sin\theta_1(m_1d_1 - m_2l_1) \end{bmatrix} \tag{19}$$

*Adaptive Neural Network Controller*

According to our search, there are a few articles about the common problems of double pendulum systems with regard to the constraint. No publication has been found about routing control of a self-impact joint system in human hands. Since dynamic equations of dual self-impact pendulum have complicated nonlinear equations, a nonlinear intelligent controller designed with a comparative neural network control is recommended; this controller requires estimation of the reverse dynamic model and the training process. To achieve accurate tracing and high controlling performance, there are few available control methods. The torque estimation method is one of the most intelligent ways that relies on the precise

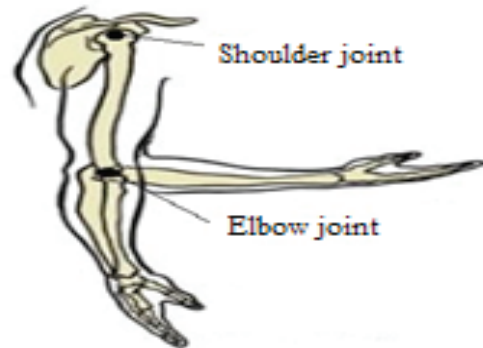


Figure 3: Schematics of constrained double pendulum in reality

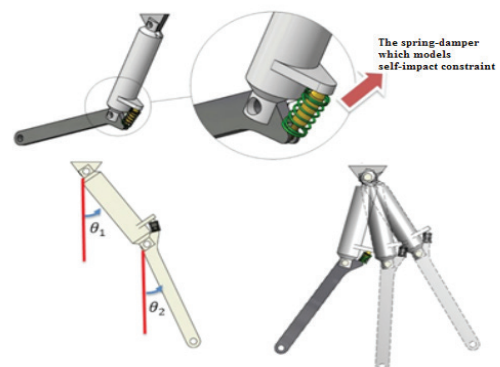


Figure 4: Schematics of constrained double pendulum



elimination of the nonlinear dynamics of the system. This method requires a precise dynamic model, which is its main weakness. The extended adaptive controlling methods, which are popular for many researchers, are implemented in order to overcome this limitation. The advantage of comparative control methods is that the no prior knowledge of unknown parameters is required.

*Gauss Radial Base Function*

Learning control plans have been developed to improve the performance of systems with operations and duplicate moves. Being suitable only for repetitive operations is one of the drawbacks of this method.

Some improvements have been made in using neural networks to control the target. In general, the design of a neural network control has two stages. The first stage is estimated using the dynamic model of the neural network. Then, when this estimation, which is usually done offline, is accurate enough, in the second stage, the control movement can be implemented. It is crucial to note that this approach cannot manage the system variations. Accordingly, the utilization of comparative control is beneficial. Some successful studies have been done using a comparative neural network strategy. They have managed to regulate the control rule directly by means of an appropriate neural network. This closed-circuit system has progressed with the proper sustainability. The other advantage of this approach is the reverse dynamic model estimation and no need to dedicate a considerable amount of time for the training process. Neural Networks can be set to zero at no cost with the prior knowledge of the system. Therefore, an adaptive neural network controller is effective for real-time implications.

In the mechanical engineering field, the neural network has been used frequently to estimate a nonlinear function given for a small tolerance error. In this paper, the Gaussian function neural network is investigated. This is a special network method that uses a number of Gaussian function formulas.

$$a_i(y) = \exp\left(-\frac{(y-\mu_i)^T(y-\mu_i)}{\sigma^2}\right) \quad (20)$$

Where,  $\mu_i \in R^n$  is the central vector and  $\sigma^2 \in R$  denotes the variance. As observed in Figure 5, each Gauss's basic function has three layers: input layer, hidden layer which consists of the Gauss's basic function, and an output layer. Network output  $\hat{f}(W, y)$  could be expressed as the following relations:

$$\hat{f}(W, y) = W^T a(y) \quad (21)$$

Where,  $a(y) = [a_1(y), a_2(y), \dots, a_l(y)]$  is the basic function vector. Note that only the connections between hidden and output layers are evaluated.

Gauss's basic function network has been very successful in demonstrating the complex nonlinear function. This suggests that any continuous function, which is not necessarily indefinitely smooth, can be estimated uniformly by a Gaussian linear combination.

*Neural Network Model of Self-Impact Joint System*

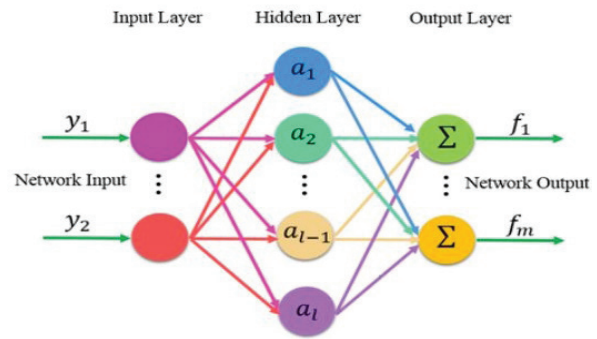


Figure 5: Gaussian radial basis function

The dynamic matrix equation of the n freedom degree of the robot is expressed in terms of the spacing coordinates in the following relations:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + \tau_f(q, \dot{q}) = \tau \quad (22)$$

As can be seen, M (q) and G (q) are the only functions among the q investigated samples. Therefore, static Neural Network is suitable for the modeling. Assume  $m_{kj}(q)$  and  $g_k(q)$  could be expressed as below:

$$m_{kj}(q) = \sum_i \theta_{kji} \xi_{kji}(q) + \varepsilon_{dkj}(q) = \theta_{kj}^T \xi_{kj}(q) + \varepsilon_{dkj}(q) \quad (23)$$

$$g_k(q) = \sum_i \beta_{ki} \eta_{ki}(q) + \varepsilon_{gk}(q) = \beta_k^T \eta_k(q) + \varepsilon_{gk}(q) \quad (24)$$

Where,  $\theta_{kji} \text{ \& } \beta_{ki} \in R$  are the Neural Network weights.  $\xi_{kji}(q) \text{ \& } \eta_{ki}(q) \in R$  represent the Gaussian radial base functions with input vector of q and  $\varepsilon_{dkj}(q) \text{ \& } \varepsilon_{gk}(q) \in R$  show the  $d_{kj}(q)$  and  $g_k(q)$  model's error. Assume that both of them are limited. For  $C(q, \dot{q})$ , a dynamic neural network from  $\dot{q}$  and q is required to be modelled by it. Assume that  $c_{jk}(q, \dot{q})$  can be modeled as the following relations:

$$c_{jk}(q, \dot{q}) = \sum_i \alpha_{kji} \zeta_{kji}(z) + \varepsilon_{ckj}(z) = \alpha_{kj}^T \zeta_{kj}(z) + \varepsilon_{ckj}(z) \quad (25)$$

Where,  $z = [q^T, \dot{q}^T]^T \in R$  and  $\alpha_{kji} \in R$  are the weight functions,  $\zeta_{kji}(z) \in R$  indicate Gauss basic functions with z as input vector, and  $\varepsilon_{ckj}(z)$  is the error of  $c_{jk}(q, \dot{q})$  which is limited. Hence, Equation 11 can be defined as:

$$m_{kj}(q) = \theta_{kj}^T \xi_{kj}(q) + \varepsilon_{dkj}(q) \quad (26)$$

$$c_{kj}(q, \dot{q}) = \alpha_{kj}^T \zeta_{kj}(z) + \varepsilon_{ckj}(z) \quad (27)$$

$$g_k(q, \dot{q}) = \beta_k^T \eta_k(z) + \varepsilon_{gk}(z) \quad (28)$$

M(q),  $C(q, \dot{q})$ ,  $C(q, \dot{q})$  and G(q) relations, utilizing GL matrix, will be:

$$M(q) = \{[\theta]^T \cdot \{\Xi(q)\}\} + E_D(q) = \begin{bmatrix} \theta_{11}^T \xi_{11}(q) & \theta_{12}^T \xi_{12}(q) & \dots & \theta_{1n}^T \xi_{1n}(q) \\ \theta_{21}^T \xi_{21}(q) & \theta_{22}^T \xi_{22}(q) & \dots & \theta_{2n}^T \xi_{2n}(q) \\ \vdots & \vdots & \ddots & \vdots \\ \theta_{n1}^T \xi_{n1}(q) & \theta_{n2}^T \xi_{n2}(q) & \dots & \theta_{nn}^T \xi_{nn}(q) \end{bmatrix} + E_M(q) \quad (29)$$

$$(q) = \{[\beta]^T \cdot \{H(q)\}\} + E_G(q) = \begin{bmatrix} \beta_1^T \eta_1(q) \\ \beta_2^T \eta_2(q) \\ \vdots \\ \beta_n^T \eta_n(q) \end{bmatrix} + E_G(q) \quad (30)$$

Where,  $\{H(q)\}$ ,  $\{B\}$ ,  $\{Z(z)\}$ ,  $\{A\}$ ,  $\{E(q)\}$ ,  $\{\theta\}$  are GL matrices and  $\alpha_{kji} \text{ \& } \xi_{kji}, \theta_{kji}, \alpha_{kj}$  represent its elements.  $E_D(q) \in R^{n \times n}$ ,  $E_C(z) \in R^{n \times n}$ , and  $E_G(q) \in R^n$  are model error matrices whose elements consist of  $\varepsilon_{dkj}(q)$ ,  $\varepsilon_{ckj}(z)$ , and  $\varepsilon_{gk}(q)$ .

**Control System Designing**

If  $q_d(t)$  is the optimal path in the joint space and  $\dot{q}(t)$  and  $\ddot{q}_d(t)$  are the desired speed and acceleration, respectively, they are expressed according to the following relations:

$$e(t) = q_d(t) - q(t) \tag{31}$$

$$\dot{q}_r(t) = \dot{q}_d(t) + q_v + \Lambda e(t) \tag{32}$$

$$r(t) = \dot{r}_1(t) - \dot{q}(t) = \dot{e}(t) + \Lambda e(t) \tag{33}$$

Where,  $\Lambda$  is a definite positive matrix. The control rule can be proposed as below [4]:

$$\tau = [[\hat{\theta}]^T \cdot \{E(q)\}] \dot{q}_r + [[\hat{A}]^T \cdot \{Z(z)\}] \dot{q}_r + [[\hat{B}]^T \cdot \{H(q)\}] + K_r r + K_v \text{sgn}(r) \tag{34}$$

Where,  $K \in R^{n \times n}$ ,  $ks > ||E||$  and  $E = E_D(q)\dot{q}_r + E_C(z)\dot{q}_r + E_G(q) + \tau_f(q, \dot{q})$ . The first three terms of the control rule are model-based controls where the term  $K_r$  gives a proportional derivative of it. Note that proportional derivative control necessarily calls for a control rule expressed through the definition given in Equation 22. The latest term in the control rule has been added to overcome the neural network modeling errors and the parameter control rule is updated to prove convergence and sustainability needs.

$$\hat{\theta} = \Gamma_k \cdot \{\xi_k(q)\} \dot{q}_r \eta_k \tag{35}$$

$$\hat{\alpha} = Q_k \cdot \{\zeta_k(z)\} \dot{q}_r \eta_k \tag{36}$$

$$\hat{\beta} = N_k \eta_k(q) \eta_k \tag{37}$$

Where,  $\Gamma_k \cdot Q_k$  and  $N_k$  are positive, definite, and symmetric.  $\hat{\theta}, \hat{\alpha}$  represent the column vectors respectively with  $\hat{\theta}_{kj}, \hat{\alpha}_{kj}$  as elements.

**Results and Discussion**

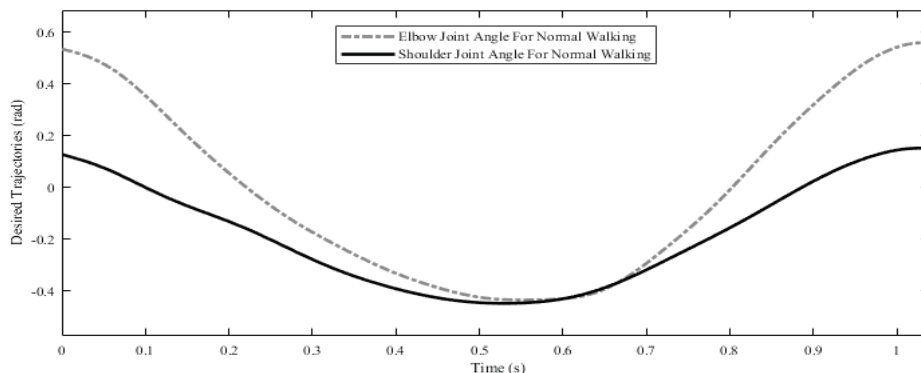
Table 1 presents the anthropometric parameters of the human hand [11] which are employed to simulate the movement of the hand as a self-impact double pendulum. For the tracing issue, the required paths from the elbow and shoulder joint of the self-impact double pendulum indicate movement of the hand during normal walking. It is based on the information available in the normal human walking cycle, which is repeated through the imaging of human hand movement and processing Images. The curves are shown in Figure 6, and the range within which the self-impact joint occurs is illustrated. According to this form, the self-impact joint phenomenon for double constrained pendulum systems occurs when the rotation angle of the elbow is larger than that of the shoulder.

*Proposed Controller Block Diagram*

In this paper, a comparative neural network approach has been proposed for the purpose of tracking the path of

**Table 1:** Parameters used for simulation of arm locomotion as a self-impact double pendulum.

Parameter	Information	Value
$m_1$ (kg)	Mass of arm link	0.2
$m_2$ (kg)	Mass of forearm	0.2
$l_1$ (m)	Length of arm link	0.32
$l_2$ (m)	Length of forearm link	0.38
$c$ (N.s/rad)	Damping coefficient	1.6
$k$ (N/rad)	Stiffness coefficient	34
$r$ (10e5)	Accuracy of the approximating function	10
$n_m$ (Nodes)	Number of static neural network nodes	12
$n_c$ (Nodes)	Number of dynamic neural network nodes	24
$n_s$ (Nodes)	Number of static neural network nodes	12
$\mu$	The center vector of the Gaussian RBF	0
$\sigma^2$	The variance of the Gaussian RBF	10



**Figure 6:** Human forearm and upper-arm normal gait cycles

the constraint dual pendulum. The controller structure is illustrated in Figure 7.

Figure 8 and Figure 9 show the desirable traced paths of shoulder and elbow joints, respectively. As can be observed, the desired values for the shoulder and elbow joint are closely matched. The angular errors of the

shoulder and elbow joints are shown in Figures 10 and 11, while the angular velocity errors, are demonstrated in Figures 12 and 13, respectively.

*Dart Throwing*

As already mentioned, one of the exercises that can

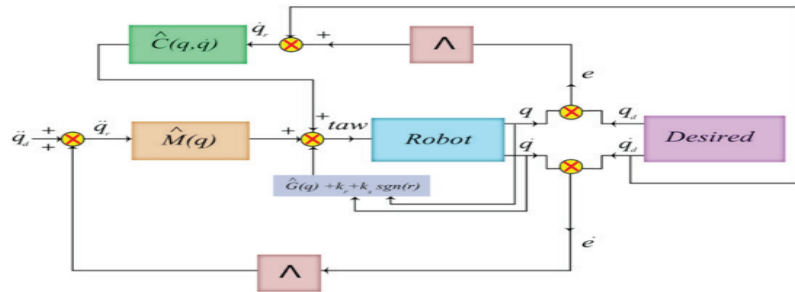


Figure 7: Structure of the adaptive neural network controller

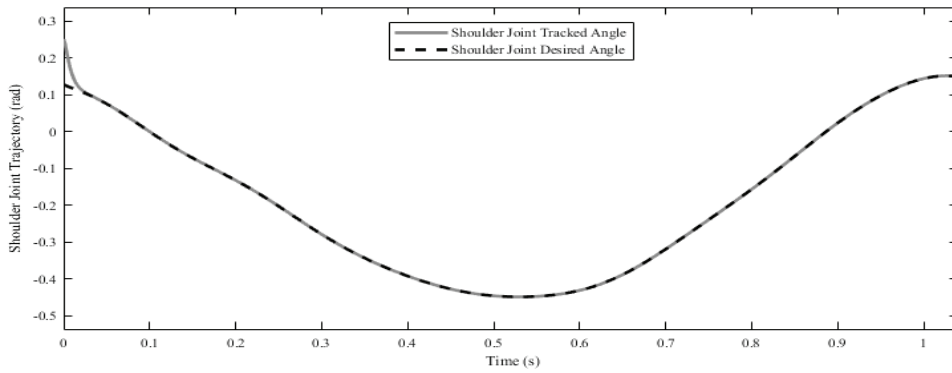


Figure 8: Desired and tracked trajectories of shoulder joint

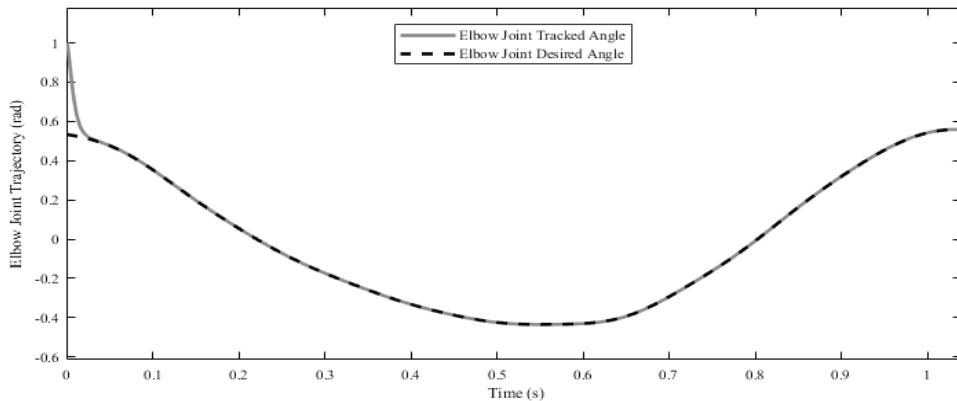


Figure 9: Desired and tracked trajectories of elbow joint

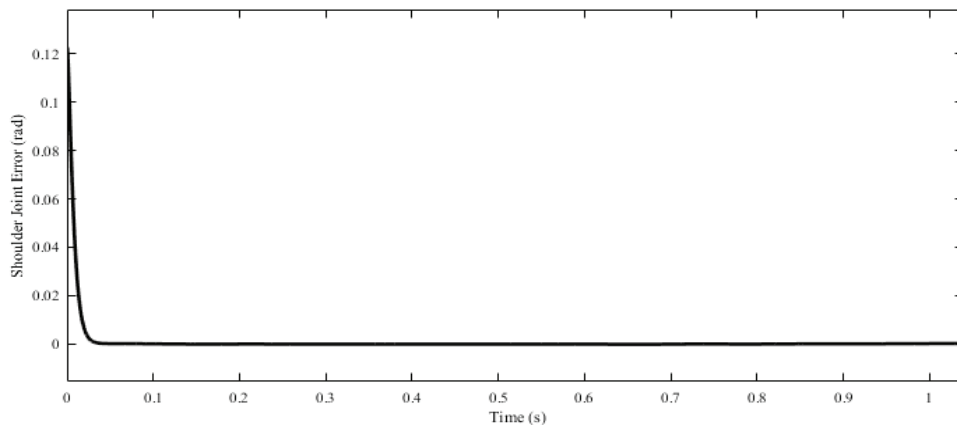


Figure 10: Time history of shoulder joint error

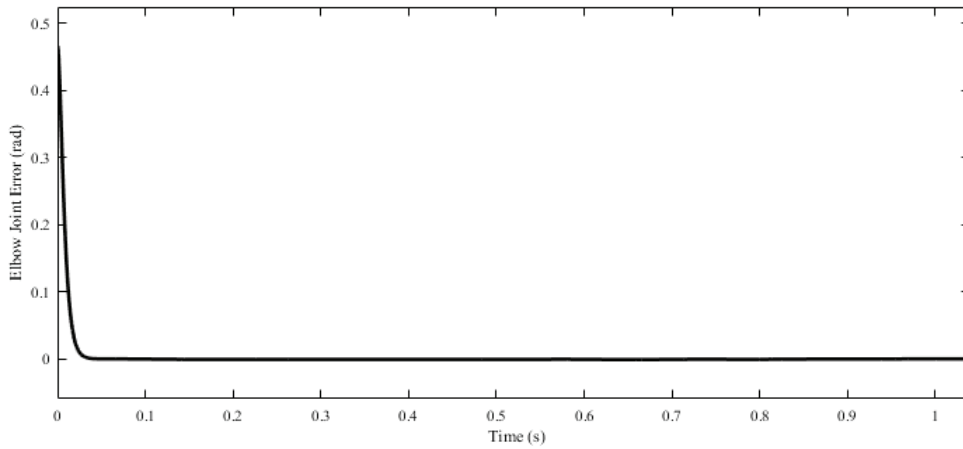


Figure 11: Time history of elbow joint error

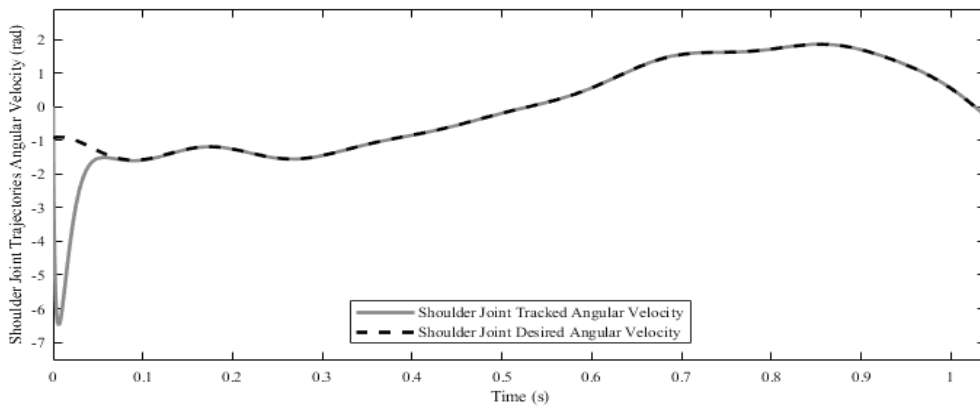


Figure 12: Time history of shoulder joint angular velocity error

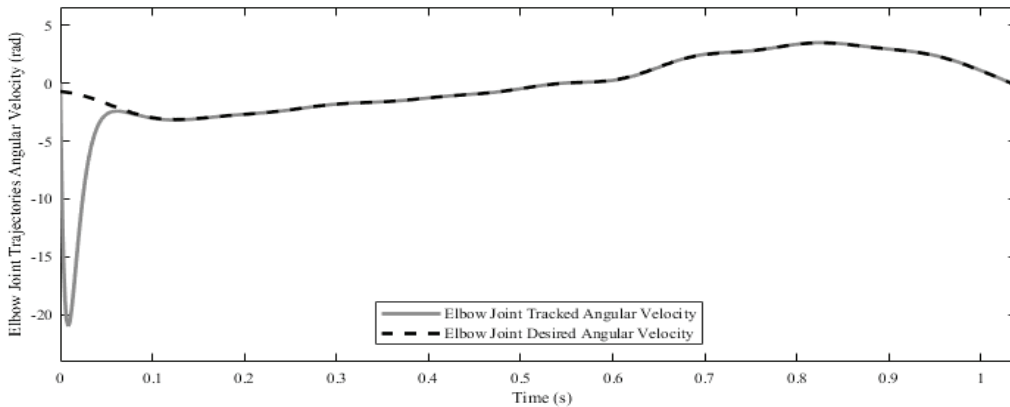


Figure 13: Time history of elbow joint angular velocity error

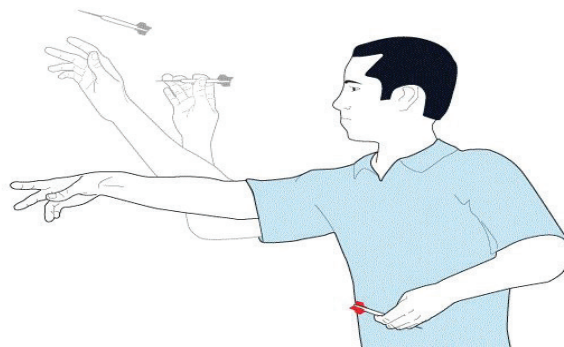


Figure 14: Schematics of human arm in throwing dart

clearly represent the self-impact joint constraint is throwing darts (Figure 14). As with the normal walking cycle, the curves of the shoulder and elbow joints in the dart throwing were obtained via image processing and used as desirable values.

Figures 15, 16 and 17 illustrate the traced and desired paths of elbow and shoulder joint, respectively. The combined shoulder and elbow errors are displayed in Figures 18 and 19, while the angular velocity errors are depicted in Figures 20 and 21, respectively.

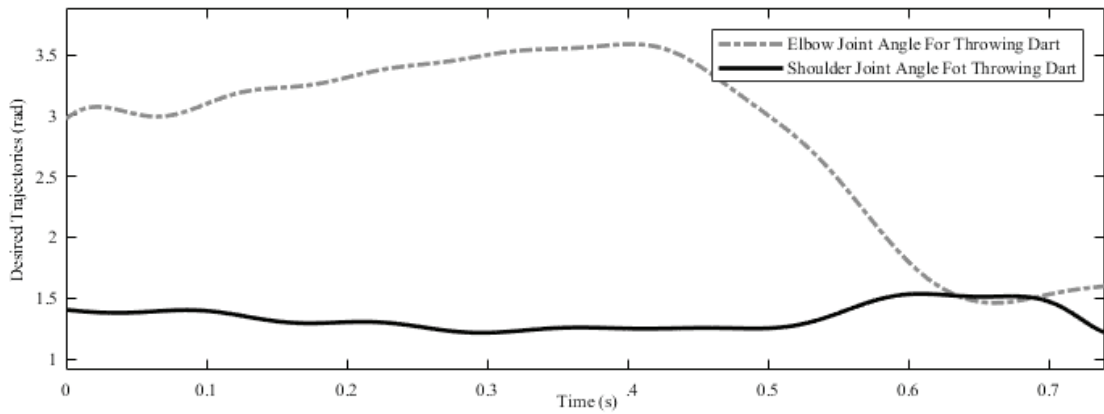


Figure 15: Human forearm and upper-arm throwing dart

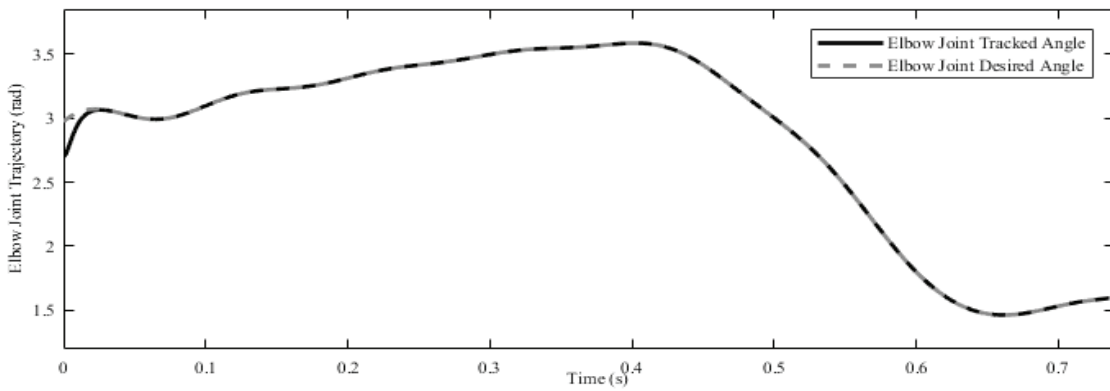


Figure 16: Desired and tracked trajectory of elbow joint in throwing dart

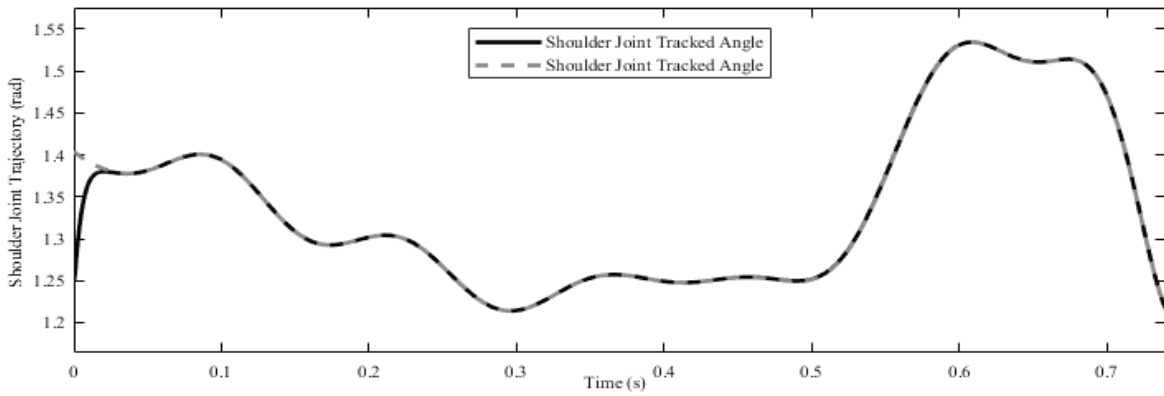


Figure 17: Desired and tracked trajectory shoulder joint in throwing dart

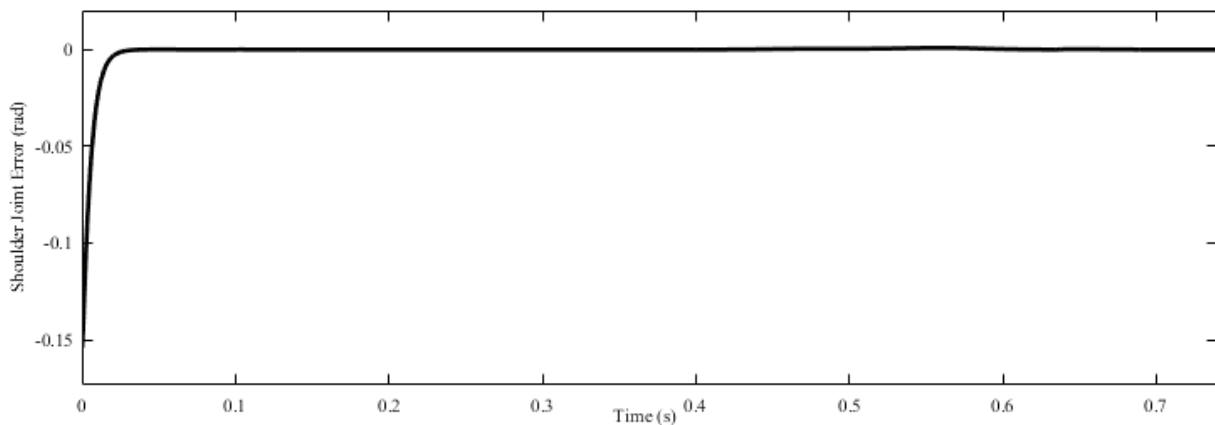


Figure 18: Time history of shoulder joint error in throwing dart



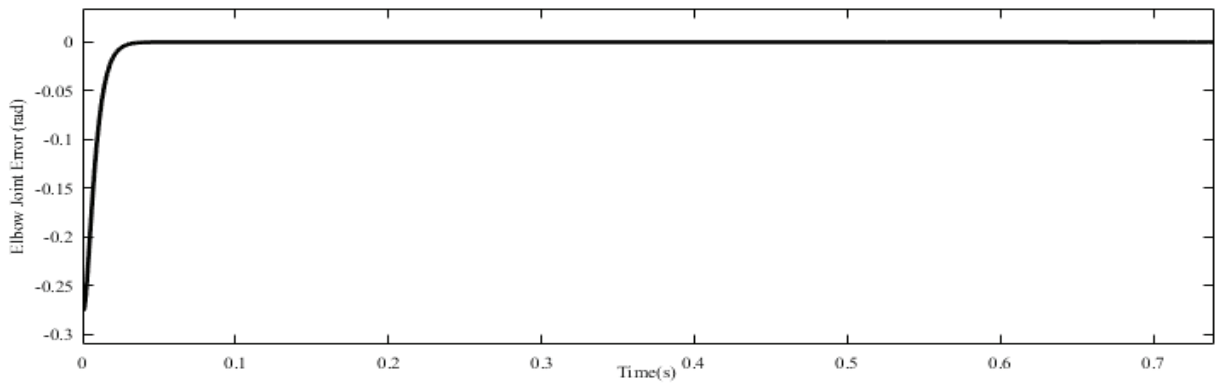


Figure 19: Time history of elbow joint error in throwing dart

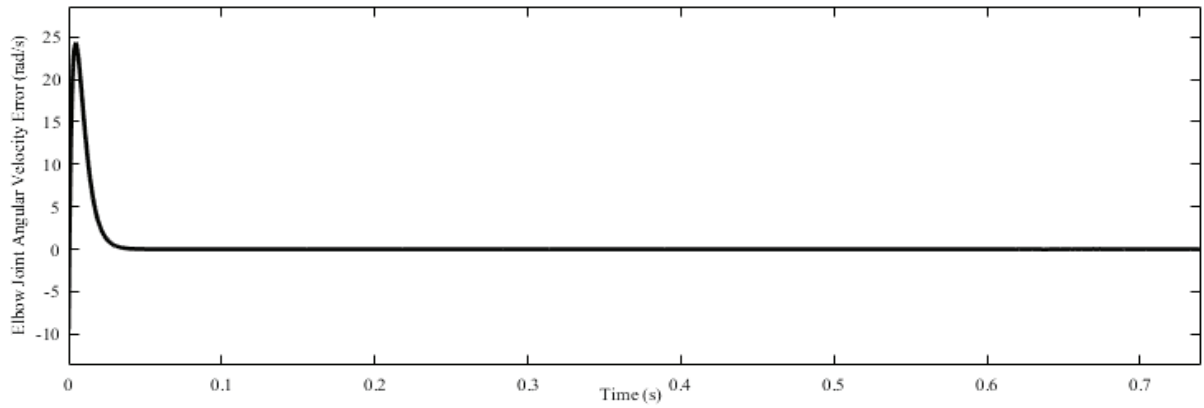


Figure 20: Time history of elbow joint angular velocity error in throwing dart

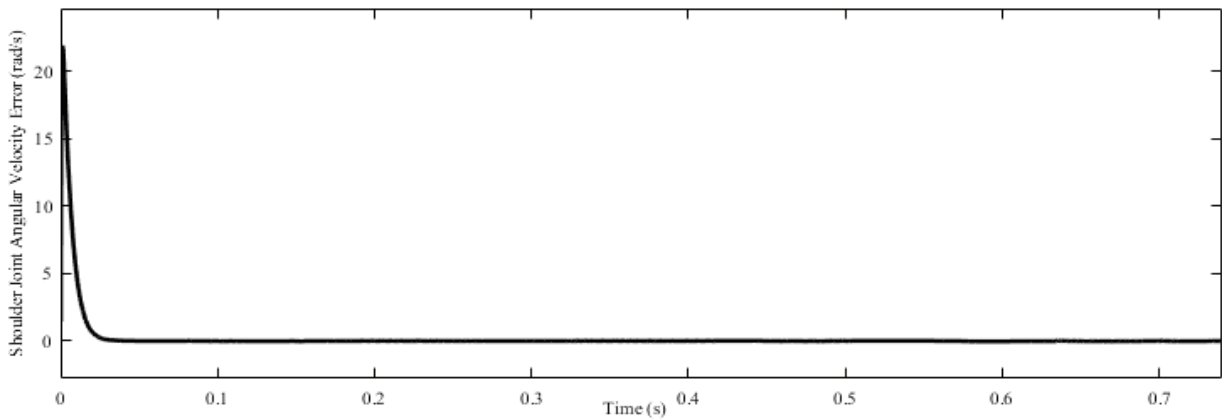


Figure 21: Time history of shoulder joint angular velocity error in throwing dart

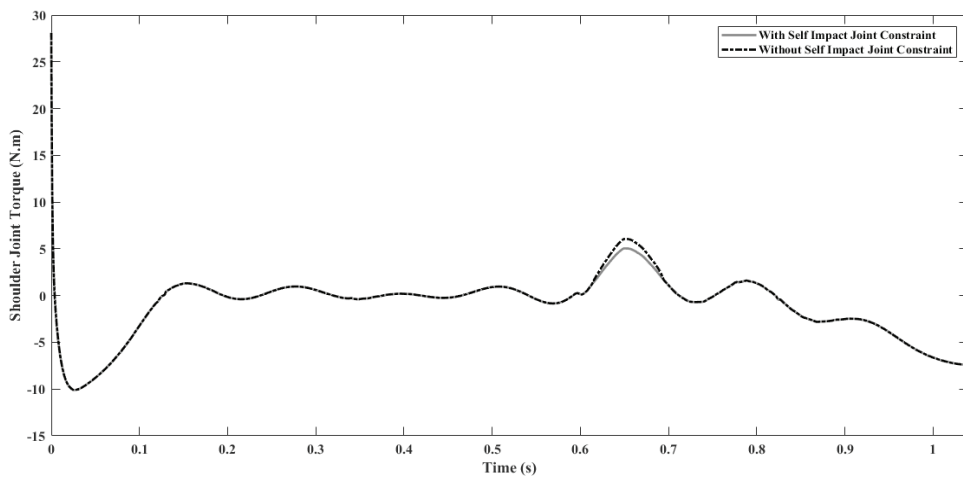


Figure 22. Elbow torque during normal gait cycle

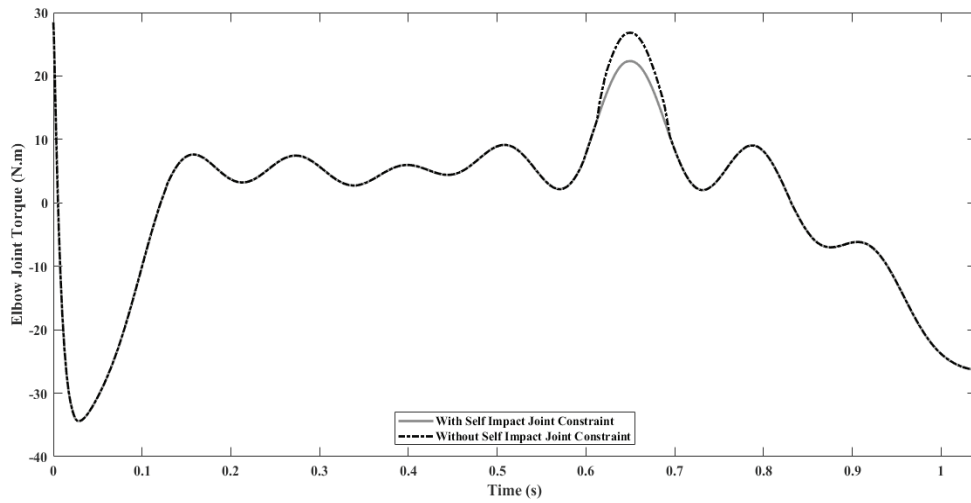


Figure 23: Shoulder torque during normal gait cycle

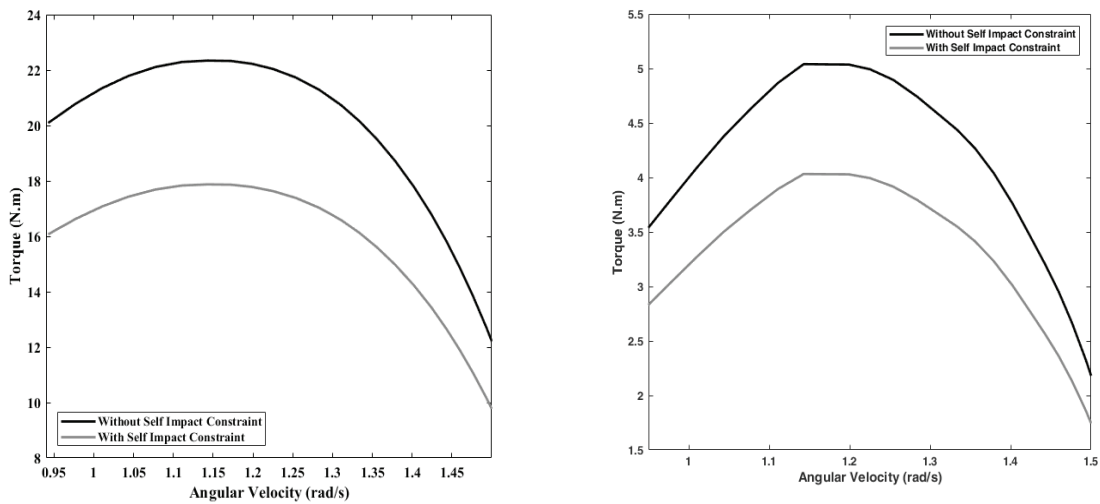


Figure 24: The comparison of the power of elbow and shoulder joint within the impact range

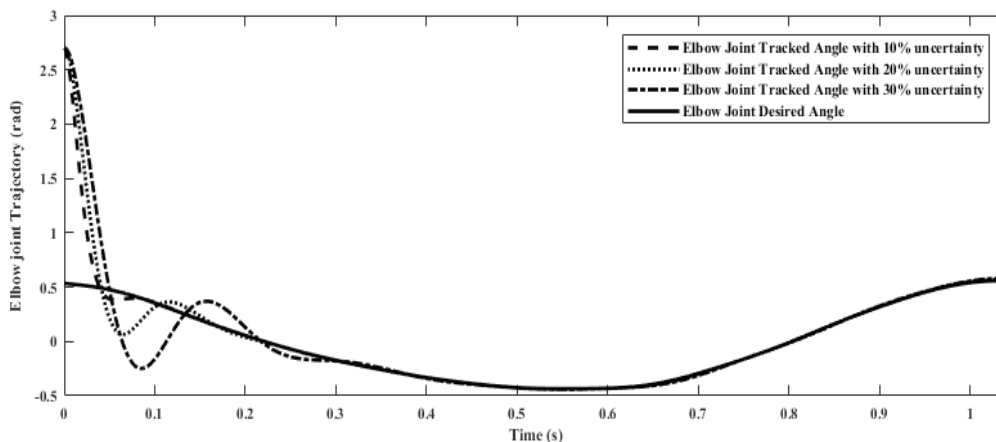


Figure 25: Desired and tracked trajectory of elbow joint with uncertainty

Figures 22 and 23 represent the torque diagram of the shoulder and elbow joints when self-impact joint constraints are considered and ignored. The constraint interval has been magnified in both figures, as there are differences in systems with and without constraints within the constraint interval. As can be seen, lower

torque values have been employed when constraints are considered for the drive motor of the robot joint.

Figure 24 represents the torque variations with angle velocity of each joint. The area under this curve represents the power consumption of the robot. Because of the importance of the self-impact joint constraint

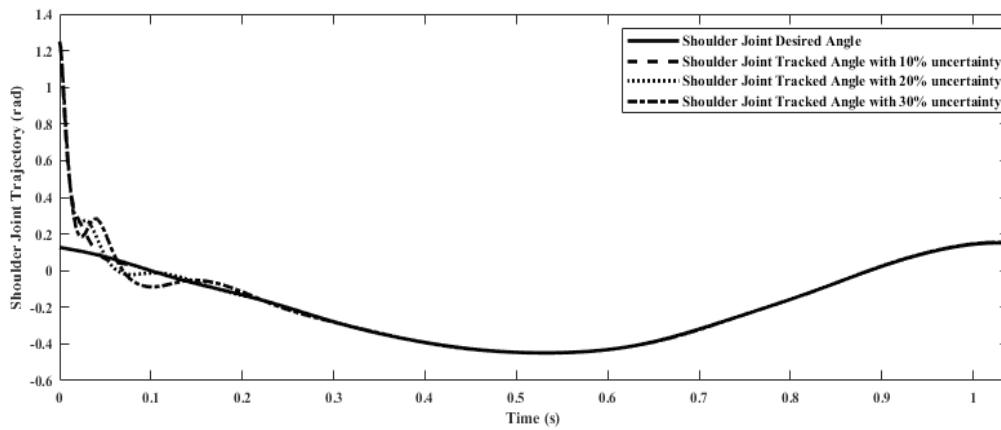


Figure 26: Desired and tracked trajectory of shoulder joint with uncertainty

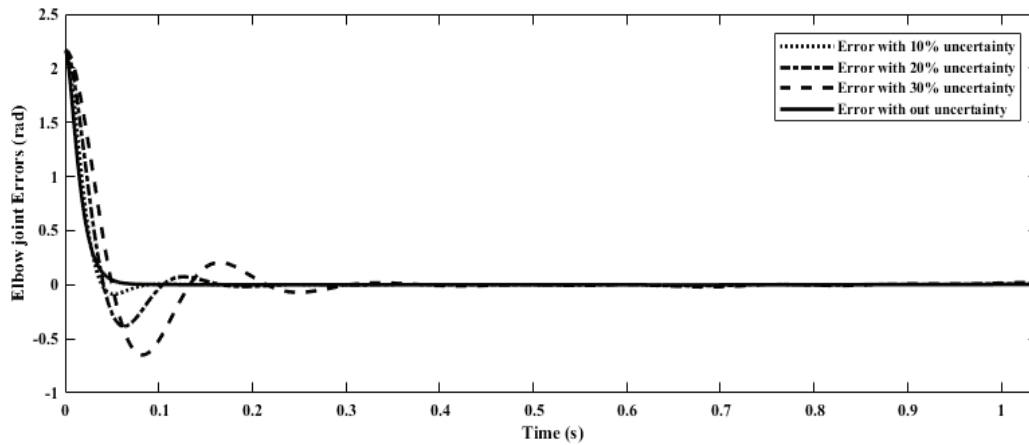


Figure 27: Time history of elbow joint error with uncertainty

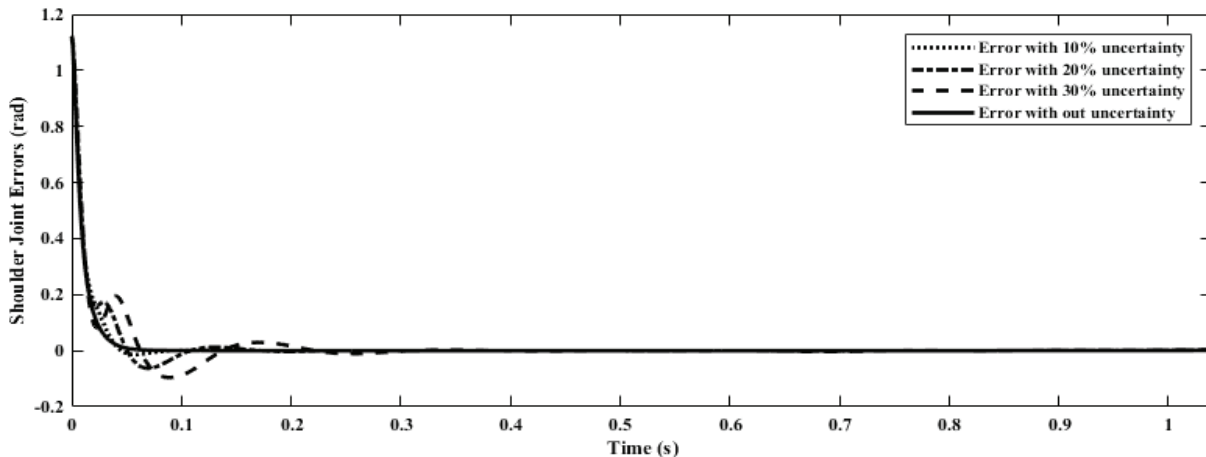


Figure 28: Time history of shoulder joint error with uncertainty

on reducing the power consumption of the robot, the corresponding figures have been given for both cases in this interval.

To illustrate the comparative control power, we examine the uncertainty of length and mass. In this case, we extend them from 10 to 30%, relative to the initial value. According to Figures 25 and 26, the uncertainty of the parameters has caused no impairment in the control system, and the controller tracks the desired path within the shortest time. The elbow and shoulder joint errors are shown in Figures 27 and 28, with regard to uncertainty.

**Conclusion**

In the present paper, a humanoid robot hand as a

double pendulum was model, taking into account the self-impact of the joint constrained during normal walking and in dart throwing. According to the results, consideration of this constraint in the elbow joint of the humanoid robot made the system closer to reality. In the double pendulum, taking into account the constraint, the angle of rotation of the elbow and arm in the joints was assumed to be bound and constrained; thus, relative to the condition where the constraint is omitted, the controlling effort is considerably reduced. On the other hand, since this constraint adds nonlinear parameters to the dynamic equations of the system, the adaptive- nervous control was utilized to manage the humanoid robot hand system.

The presence of the self-impact joint constraint contributed to about 26% saving in power consumption

of robot motors within an impact range of 0.6346 to 0.6896 during normal human walking.

Since this control has a high power, 10 to 30% of the uncertainty was added to the length and mass parameters. As was observed, this controller routed the desired curves within the shortest possible time.

**Conflict of interest:** None declared.

## References

- Collins, Steven H., Peter G. Adamczyk, and Arthur D. Kuo. "Dynamic arm swinging in human walking." *Proceedings of the Royal Society of London B: Biological Sciences* 276.1673 (2009): 3679-3688.
- Arellano, Christopher J., and Rodger Kram. "The effects of step width and arm swing on energetic cost and lateral balance during running." *Journal of biomechanics* 44.7 (2011): 1291-1295.
- Meyns, Pieter, Sjoerd M. Bruijn, and Jacques Duysens. "The how and why of arm swing during human walking." *Gait & posture* 38.4 (2013): 555-562.
- Punt, Michiel, et al. "Effect of arm swing strategy on local dynamic stability of human gait." *Gait & posture* 41.2 (2015): 504-509.
- Kato, Norihiko, Kenji Matsuda, and Tatsuya Nakamura. "Adaptive control for a throwing motion of a 2 DOF robot." *Advanced Motion Control, 1996. AMC'96-MIE. Proceedings., 1996 4th International Workshop on*. Vol. 1. IEEE, 1996.
- Bazargan-Lari, Yousef, et al. "Tracking Control of A Human Swing Leg as a Double-Pendulum Considering Self-Impact Joint Constraint by Feedback Linearization Method." *Journal of Control Engineering and Applied Informatics* 17.1 (2015): 99-110.
- Bazargan-Lari, Yousef, et al. "Adaptive neural network control of a human swing leg as a double-pendulum considering self-impact joint constraint." *Transactions of the Canadian Society for Mechanical Engineering* 39.2 (2015): 201-219.
- Bazargan-Lari, Yousef, et al. "Tracking control of a planar five-link bipedal walking system with point contact, considering self-impact joint constraint by adaptive neural network method." *Latin American Journal of Solids and Structures* 12.6 (2015): 1074-1101.
- Piovesan, Davide, et al. "Measuring multi-joint stiffness during single movements: numerical validation of a novel time-frequency approach." *PloS one* 7.3 (2012): e33086.
- Lin, Chou-Ching K., Ming-Shaung Ju, and Han-Wei Huang. "Gender and age effects on elbow joint stiffness in healthy subjects." *Archives of physical medicine and rehabilitation* 86.1 (2005): 82-85.
- McCaw, Steven T. "Introduction to BiomechCInICS." *Foundations of Exercise Science* (2001): 155.