International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(19s) 2026 pp. 778-792
https://doi.org/10.70917/ijcisim-2026-5082
Article

Reinforcement Learning for Real-Time Auto-Tuning of Robot's Controllers

Hayder Z. Abdalikhwa[1], Waleed Al-Ashtari[2*]

[1]Dept. of Mechanical Engineering, College of Engineering, University of Baghdad, Baghdad 10071, Iraq; haidar.zaher2303@coeng.uobaghdad.edu.iq
[2]Dept. of Mechanical Engineering, College of Engineering, University of Baghdad, Baghdad 10071, Iraq; Waleed.Al.Ashtari@coeng.uobaghdad.edu.iq
*Correspondence: haidar.zaher2303@coeng.uobaghdad.edu.iq; Tel.: +9647715971765

Abstract: This research introduces an innovative control technique for Series Elastic Actuators (SEAs) that utilizes Reinforcement Learning (RL) to address the shortcomings of previously fixed-gain adaptive controllers, which are a hybrid of State Feedback Control (SFC) and Model Reference Adaptive Control (MRAC) by using Lyapunov Stability Analysis. This controller is optimized by adjusting the adaptation factor. b. This study presents an intelligent agent based on reinforcement learning to find the value of b with a dynamic auto-tuner. It trains via the Soft Actor-Critic (SAC) algorithm for 100,000 time steps. A comparison between the two methods was presented according to simulation results under different conditions; the RL-based controller shows much better tracking accuracy, how quickly it reaches the target output, and how little control torque it uses, where the agent's policy could automatically adjust in real-time based on system conditions, such as uncertainties and disturbances, where it has a settling time of 1.7 seconds, while the fixed parameter controller has a 1.95-second settling time, resulting in a reduction of 15.3%. It also lowers the control torque caused by disturbances by 19.5% compared to the fixed parameter controller, which has a control torque of 3.99 Nm, while the maximum control torque for the RL-optimized controller is 3.21 Nm.
Keywords: MRAC; SFC; Lyapunov Stability Study; agent; SAC; RL

1. Introduction

The inclusion of a Series Elastic Actuator (SEA) is considered one of the great innovations in mechatronics and robots in particular. SEAs incorporate a stretchy part to help with the motor with incredible properties such as flexibility, capacity to sense force and to store energy that are impossible with conventional stiff actuators. This special structure allows for the gathering, storage and transfer of mechanical energy specifically; a feature leading to precise and versatile operation.

SEAs also strive to design robotic systems that mimic human response and promote safety. Moreover, SEAs improve the response of robots to external forces and alterations in their environment by adding flexible components, a great thing as cobots, rehabilitation devices, entertainment, prosthetics, exoskeletons, and different uses. SEAs have some complex limitations such as difficult design, energy wasted from elasticity, complex control requirements, difficult stability, and requirements demanding modeling. This complexity requires proprietary technical solutions, novel control paradigms, an understanding of compromises specific to applications [2].

The elastances of the architecture gives SEA the ability to sense force. Almost immediately, the actuator detects and records external torques and forces. Rapid force feedback is important for tasks that are performed with highly accurate results—such as for guiding robots to do precise work, modifying movements according to what they observe in sight—that require sophisticated control systems [3].

The controllers, like Sliding Mode Controllers (SMC), Fuzzy Logic Controllers (FLC), MRAC, SFC, PID, and impedance controllers, can use strong or flexible control methods and can work alone or together to form hybrid controllers. [4-11].

resistance control plays a crucial role in managing the way a robot interacts with the things around it by regulating the dynamic relationship inter alia exerted torques or forces and the resulting speeds or displacements at the end effector of the robot. resistance control enables flexible responses; however, variations within the system characteristics or uncertainties may compromise its effectiveness, affecting comprehensive stability and efficacy [12-25].

The SFC, which regulates dynamic systems using intrinsic state variables, is used in many areas to achieve excellent performance and stability with careful control. Nonetheless, inaccuracies or disturbances in precise state metrics may undermine [26-42]. The MRAC changes the control settings of a system on the fly to match a specific reference model, showing it works well for systems that behave unpredictably or change over time. (MRAC) depends on precise system dynamics modeling, which is essential for optimal control rendering [43-58].

This study also further develops a new model, where (MRAC) and (SFC) were combined as proposed by Al-Ashtari and Ali [59]. This hybrid model uses SFC's accuracy with MRAC's flexibility. But the success of it depends on a critical -and manually adjusted -parameter, b, which controls the rate of adaptation. The optimal value (e.g., b = -40) has to be set and is not easy to simulate, thus limiting robustness of the controller. The conclusion of the original work recommends that the next logical step that is applied is optimization methods in this parameter [59-60]. This paper fills this gap and proposes an innovative approach, using Reinforcement Learning to complement standard MRAC methods, taking the next logical research pathway in this domain which combines RL with the adaptive control. The key challenge is to design not only for finding the optimal b but also to substitute this static parameter for a Dynamic Control Policy. The trained reinforcement learning “agent” that generates bt continuously checks the optimal value at time steps, guided by the state of the system, rather than a constant b. This helps to reduce system uncertainties, decrease load disturbances, and improve stability and performance. Additionally, the main model of the whole is a unified parameter model,

This includes assisting in optimizing execution and identifying critical improvements to the planned SFC. A modified adjustment rule as developed by Lyapunov should consist of components to enhance system performance and lessen uncertainties in parameters and load perturbations.

Contribution of the Study

The main contribution of this study is the development of an intelligent reinforcement learning-based auto-tuning strategy for the adaptive control of Series Elastic Actuators (SEAs). Unlike the conventional fixed-parameter adaptive controller combining State Feedback Control (SFC) and Model Reference Adaptive Control (MRAC), the proposed approach employs a Soft Actor-Critic (SAC) reinforcement learning agent to dynamically adjust the adaptation factor in real time according to changes in system operating conditions, uncertainties, and external disturbances. The proposed method eliminates the dependence on a fixed adaptation factor and enhances the controller's ability to adapt online. Simulation results demonstrate that the RL-based auto-tuning strategy improves tracking accuracy and settling time while reducing the required control torque compared with the fixed-parameter controller, demonstrating its effectiveness for intelligent and adaptive control of SEAs.

2. Mathematical Model

Figure 1 illustrates the same lumped parameter model for a Series Elastic Actuator (SEA) using a DC motor. The spring-like model includes an elastic component arranged in series with the primary actuation mechanism and, by linking the primary motor to the load, moves it forward and backwards. This setup has many advantages for tailored applications — particularly in cases where you need more flexibility and control of your forces.

The follows variables are used throughout this research: the disruption torque from the load is denoted as $\tau(t)$, $u(t)$ indicates the utilized control torque, and θl(t) (t) and, θm(t) represent the angular position of the load side and motor side, respectively. Additionally, the system parameters include B_j, K_j, and K_b, which stand for the damping coefficient of the elastic joint, the stiffness of the elastic joint, and the stiffness of the motor mounting, respectively. Furthermore, angular momentum of inertia for both the load and the motor are denoted by, Ji and Jm, respectively

[Image could not be embedded — unsupported format]

Figure. 1. Two-Mass Drive of a SEA System

Using Newton's second law for motor side:

$$J_m\ddot{\theta}_m(t)+B_j\,(\dot{\theta}_m(t)-\dot{\theta}_l(t))+K_j\,(\theta_m(t)-\theta_l(t))+K_b\,\theta_m(t)=u(t) \quad (1)$$

Equation (1) can be reformulated as :

$$\ddot{\theta}_m(t)=\frac{1}{J_m}[-(K_j+K_b)\,\theta_m(t)-B_j\dot{\theta}_m(t)+K_j\theta_l(t)+K_j\dot{\theta}_l(t)+u(t)] \quad (2)$$

For the motor side, using Newton's second law leads to:

$$J_l\ddot{\theta}_l(t)+B_j(\dot{\theta}_l(t)-\dot{\theta}_m(t))+K_j\theta_l(t)-\theta_m(t))=-\tau(t) \quad (3)$$

Eq. (3) can be reformulated as:

$$\ddot{\theta}_l(t)=\frac{1}{J_l}[(K_j\,\theta_m(t)+B_j\dot{\theta}_m(t)-K_j\theta_l(t)-B_j\dot{\theta}_l(t)-\tau(t)] \quad (4)$$

Let $x_1(t)=\theta_m(t), x_2(t)=\dot{\theta}_m, x_3(t)=\theta_l\,(t)$, and $x_4\,(t)=\dot{\theta}_l(t)$, then putting in Eq. (2) ,and Eq. (4) gets

$$\dot{x}_2(t)=-\frac{(K_j+K_b)}{J_m}x_1(t)-\frac{B_j}{J_m}x_2(t)+\frac{K_j}{J_m}x_3(t)+\frac{B_j}{J_m}x_4(t)+\frac{1}{J_m}u(t) \quad (5)$$

and

$$\dot{x}_4(t)=\frac{K_j}{J_l}x_1(t)+\frac{B_j}{J_l}x_2(t)-\frac{K_j}{J_l}x_3(t)-\frac{B_j}{J_l}x_4(t)-\frac{1}{J_l}\tau(t) \quad (6)$$

Let's rewrite in the form of a state space

$$[\dot{x}_1(t)\;\dot{x}_2(t)\;\dot{x}_3(t)\;\dot{x}_4(t)\,]=\left[0\;1\;0\;0\;\frac{-(K_j+K_b)}{J_m}\;\frac{-B_j}{J_m}\;\frac{K_j}{J_m}\;\frac{B_j}{J_m}\;0\;0\;0\;1\;\frac{K_j}{J_l}\;\frac{B_j}{J_l}\;\frac{-K_j}{J_l}\;\frac{-B_j}{J_l}\right][x_1(t)\;x_2(t)\;x_3(t)\;x_4(t)\,]+\left[0\;\frac{1}{J_m}\;0\;0\right]u(t)+\left[0\;0\;0\;-\frac{1}{J_l}\right]\tau(t) \quad (7)$$

and the output state space

$$[y_1\,(t)\;y_2\,(t)\;y_3\,(t)\;y_4\,(t)\,]=[\;1\;0\;0\;0\;\;0\;1\;0\;0\;\;0\;0\;1\;0\;\;0\;0\;0\;1\,][x_1\,(t)\;x_2\,(t)\;x_3\,(t)\;x_4\,(t)\,] \quad (8)$$

then can be described system as :

$$\dot{x}\,(t)=A\,x\,(t)+B\,u\,(t)+W\,\tau\,(t) \quad (9)$$

And $y(t)=C\,x\,(t)$ (10)

Where system's state vector is represented by $x\,(t)$ the, and $\dot{x}\,(t)$is its derivative. $y(t)$ is the system's output vector. Furthermore, A and W indicate the state and output matrix, respectively, whereas$B$ and C are the input matrices for applied torque, and disturbance torque, respectively. Significant observations arise from the comparison of Eq.(7) and Eq. (9)

$$A=\begin{bmatrix} 0 & 1 & 0 & 0 & \frac{-(K_j+K_b)}{J_m} & \frac{-B_j}{J_m} & \frac{K_j}{J_m} & \frac{B_j}{J_m} & 0 & 0 & 0 & 1 & \frac{K_j}{J_l} & \frac{B_j}{J_l} & \frac{-K_j}{J_l} & \frac{-B_j}{J_l} \end{bmatrix} \quad (11a)$$

$$B=\begin{bmatrix} 0 & \frac{1}{J_m} & 0 & 0 \end{bmatrix}^T \quad (11b)$$

and

$$W=\begin{bmatrix} 0 & 0 & 0 & -\frac{1}{J_l} \end{bmatrix}^T \quad (11c)$$

Next, a comparison of Eq. (8) and Eq. (10) leads to:

$$C=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (11d)$$

The dynamic behavior of the system can be governed by the state matrix A, manipulating how it changes. B and W input matrices are important for designing the controller because they show how the applied and disturbance torque, impact the system's reaction, whereas C matrix, defines the observable controlled system output.

3. Design of The Controller

A controller grounded in state feedback theory meticulously enhances the actuator's performance. This controller can effectively calculate the right gains, which helps reduce system uncertainties and disturbances while closely following the intended performance path.

3.1 Referential Model

Using the D-operator, D = d/dt, and reformulating Eq. (1) and Eq. (3), the outcome:

$$\theta_m = \frac{1}{J_m}\left[\frac{u(t)+(B_j D+K_j)\theta_l(t)}{D^2+\frac{B_j}{J_m}D+\frac{(K_j+K_b)}{J_m}}\right] \quad (12)$$

$$\theta_l(t) = \frac{-\tau(t)+(B_j D+K_j)\theta_m(t)}{J_l D^2+B_j D+K_j} \quad (13)$$

Let both Eq. (12) and Eq. (13) conform to a second-order system. This indicates it was reformulated as :

$$\theta_{mr}(t) = \frac{\omega_{n1}^2 u(t)}{D^2+2\zeta_1\omega_{n1}D+\omega_{n1}^2} \quad (14)$$

$$\theta_{lr}(t) = \frac{\omega_{n2}^2 \theta_{mr}(t)}{D^2+2\zeta_2\omega_{n2}D+\omega_{n2}^2} \quad (15)$$

Where $\theta_{mr}(t)$ is the angular position derived from the reference model for the motor side. In this context, $\theta_{lr}(t)$ represents the angular position derived from the reference model specifically for the load side. Additionally, the values, ω_{n2} and ω_{n1}, are the natural frequencies of the load side and motor side, respectively, while ζ_2 and ζ_1 represent the damping ratio for load side and motor side, respectively.

Assuming that:

$$x_{r1}(t) = \theta_{mr}(t), \qquad x_{r2}(t) = \dot{\theta}_{mr}(t), \qquad x_{r3}(t) = \theta_{lr}(t), \qquad x_{r4}(t) = \dot{\theta}_{lr}(t)$$

In the state space form, the reference model can be articulated as follows:

$$\dot{x}_r(t) = A_r\, x_r(t) + B_r\, r(t) \quad (16)$$

The output is:

$y_r(t) = C_r x_r(t)$ (17)

In the state space form, the reference model can be articulated as follows:

$A_r = [0\ 1\ 0\ 0\ -\ \omega_{n1}^2\ -2\zeta_1\ \omega_{n1}\ 0\ 0\ 0\ 0\ 0\ 1\ \omega_{n2}^2\ 0\ -\omega_{n2}^2\ -2\zeta\omega_n\ ]$ (18a)

$B_r = [0\ \omega_{n1}^2\ 0\ 0\ ]^T$ (18b)

$C = C_r$ (18c)

$x_r(t)$ and $\dot{x}_r(t)$ denote the reference model state vector and its temporal derivative, respectively. $y_r\ (t)$ is the reference model output vector. Furthermore, (A_r), B_r, and C_r (t) denote the reference model state matrix, $r(t)$ is the input to the reference model, and the reference model output matrix, respectively. In this research, $r(t)$ is the reference input to be tracked and it is represented as a 1×4 column vector. The reference model serves as a foundational structure for assessing the effectiveness of an actual system by delineating anticipated results, while the designer tailors the model's performance to meet particular requirements. The settling time, denoted as t_s, is determined by Equation (19).

$t_s = \frac{4}{2\zeta_1\omega_n}$ (19)

The objective is to obtain for both sides of the system a settling time of 1 second, while the damping ratios for each side of the system ($\zeta_1 and\ \zeta_2$) are both set to 1. This selection indicates the natural frequencies. (ω_{n1} and ω_{n2}) are identical, both being set to 4. To optimize the system's achievement, the parameter values are employed, and the settling time formula of the reference model is used. inserted these values into Equations (18a) and (18b), which yields:

$A_r = [0\ 1\ 0\ 0\ -16\ -8\ 0\ 0\ 0\ 0\ 0\ 1\ 16\ 0\ -16\ -8\ ]$ (20a)

And

$B_r = [0\ 16\ 0\ 0\ ]^T$ (20b)

Equations (16) , (18 c), (20 a), and (20 b) provide a thorough frame for distinguishing the SEA's best reaction

The essence of the MRAC is a reference model that delineates the intended system performance. A simple second-order model is employed, featuring a target natural frequency(ω_n)=4 and a damping ratio ζ =1.

The reference model is articulated in state-space as follows:

3.2 Adaptive Control Law

The controller's major purpose is to guarantee that the actual SEA's behavior aligns with that of the reference model to attain the intended output. Consequently, the error, $e(t)$, is defined as:

$e\ (t)\ =\ x\ (t)\ - x_r\ (t)$ (21)

The rate of change of error $\dot{e}(t)$, is expressed by:

$\dot{e}\ (t) = \dot{x}\ (t) - \dot{x}_r(t)$ (22)

By substitute Equations (9) and (16) in Equation (22), Equations (9) and (16) into Equation (22), it is obtained:

$\dot{e}\ (t) = Ax\ (t) + Bu\ (t)\ \text{-}\ B_r\ r\ (t)\ \text{-}\ A_r\ x_r\ (t)$ (23)

This scenario, it is presumed that the disturbance torque is null, and the system adopts the traditional state space. The expression of the control law is accomplished by utilizing the theory of state feedback control, as demonstrated:

$u\ (t) = k_r\ r\ (t) - k_x\ x\ (t)$ (24)

Where: (k_x) denoting the gain of the feedback, and (k_r), the reference input gain, which are 1×4 vectors. The incorporation the reference model and state feedback control theory inherently

mitigates the effects of disturbances. This integration necessitates that the system continuously aligns with the reference input $r(t)$, depending on the discrepancies between the reference model and the present conditions of the system.

Equation (23) can be substituted into Equation (24) to produce the subsequent expression:

$$\dot{e}(t) = (A-Bk_x)\, x(t) - A_r x_r(t) + (Bk_r - B_r)\, r(t) \quad (25)$$

The objective of the controller is minimizing the error by ensuring that state $x(t)$ is identical to state $x_r\ (t)$, therefore, the error is reduced to zero, yielding:

$$A_r = A - B \quad (26)$$

$$B_r = Bk_r \quad (27)$$

In real implementations, the values of gains (kr and kx) are not determined due to the parameters of the system denoted by the matrices A and B cannot be accurately characterized, or they fluctuate over time because of factors like wear, deformation, friction, etc. The control law is restructured to address this issue:

$$u(t) = \hat{k}_r(t)\, r(t) - \hat{k}_x(t)\, x(t) \quad (28)$$

where $\hat{k}_r\ (t)$ and $\hat{k}_x(t)$ Represent the appreciation of kr and kx, respectively, and possess identical dimensions.

Equation (28) can be substituted into Equation (23) to produce:

$$\dot{e}(t) = (A - B\hat{k}_x(t))\, x(t) + (B\hat{k}_r(t) - B_r)\, r(t) - A_r x_r(t) \quad (29)$$

The estimate of gains $\hat{k}_r(t)$ and $\hat{k}_x(t)$ vary over time and articulated as:

$$\tilde{k}_r(t) = k_r - \hat{k}_r(t) \quad (30a)$$

$$\tilde{k}_x(t) = k_x - \hat{k}_x(t) \quad (30b)$$

The estimation errors of k_r and k_x are indicated by the symbols $\tilde{k}_r\ (t)$ and $\tilde{k}_x\ (t)$, respectively.

Equations (30 a) and (30 b) can be inserted into Equation (29) to produce:

$$\dot{e}(t) = (A - Bk_x - B\tilde{k}_x(t))\ x(t) + (Bk_r - B\tilde{k}_r(t) - B_r)\, r(t) - A_r\, x_r(t) \quad (31)$$

Additionally, inserting Equation (26) and Equation (27) into Equation (31) yields:

$$\dot{e}(t) = A_r e(t) - B\tilde{k}_x(t)\, x(t) - B\tilde{k}_r(t) r(t) \quad (32)$$

The analysis of Lyapunov stability was implemented on the subsequent function:

$$V(t) = \frac{1}{2}\ e^2(t) + \frac{1}{2}\tilde{k}_r^{\,2}(t) + \frac{1}{2}\ \tilde{k}_x^{\,2}(t) \quad (33)$$

$$\dot{V}(t) = \dot{e}(t) e(t) + \tilde{k}_x(t)\ \dot{\tilde{k}}_x(t) + \tilde{k}_r(t) \dot{\tilde{k}}_r(t) \quad (34)$$

$$\dot{\tilde{k}}_x(t) = -\dot{\hat{k}}_x(t) \quad (35a)$$

$$\dot{\tilde{k}}_r(t) = -\dot{\hat{k}}_r(t) \quad (35\ b)$$

Eq . (35 a), Eq. (35 b) , and Eq. (32) can be inserted into Eq. (34), results in

$$\dot{V}\ (t) = A_r\, e^2\ \ (t)\ - Br\ (t)\, e\ (t)\, \tilde{k}_r\ (t) - Bx\ (t)\ e\ (t)\ \ \tilde{k}_x\ (t) - \dot{\hat{k}}_x(\ t)\ \ \tilde{k}_r\ (t)$$

(36)

To guarantee that $V(t)$ approaches zero as time approaches infinity, the subsequent relationships must be established.

$$\dot{\hat{k}}_r(t) = -\ Br\ (t)\ e(t) \qquad (37\ a)$$

$$\dot{\hat{k}}_x(t) = -\ Bx\ (t)\ e(t) \qquad (37\ b)$$

As previously stated, given that the parameters of system parameters are deemed unidentified, vector B can be expressed as:

$$B = b\ \ [\ 0\ 1\ 0\ 0\ \ ]^T \qquad (38)$$

b, a constant, is employed to optimize controller execution and customize it according to the design specifications.

The suggested control system demonstrates varied applications in SEAs and similar systems. Combining a reference model with state feedback control creates a flexible system that can easily adjust to different systems with similar behaviors. Our model has core principles and methodology which are universally applicable and can be used in other SEAs or systems of equivalent characteristics. Such innate flexibility emphasizes the resilience and potential for wide versatility of our design of the control system in the face of various real applications. The constant parameter b increases functionality of our controller. The optimal value of the parameter b is -40 that strikes a good balance between response time and control torque as determined by intensive preliminary simulations for the steady state [12]. This number provides a baseline "Fixed Controller" for comparison analysis.

4. Case Study

The current section provides the same case study already analyzed by Walid Al-Ashtari and Karim Ali for contrast between the fixed b controller and the RL controller. The simulation model, crucial to such evaluation process, was painstakingly developed in MATLAB to construct and examine the models, demonstrating how the proposed controller would behave in practical environment situations and showing its efficacy and strength.

Table 1. Parameters of the case study, which was simulated

Parameter	Symbol	Unit	Value
Moment of inertia for load side	J_l	kg m^2	0.1
Moment of inertia for motor side	J_m	kg m^2	0.1
Mounting stiffness of the motor	K_b	N/m	7
Stiffness of the joint	K_j	N/m	3
Damping coefficient of the joint	B_j	N s /m	0.6

5. Reinforcement Learning for Real-Time Auto-Tuning

A hybrid controller (MRAC-SFC) was presented, and its efficacy was shown in the earlier parts of this study. The parameter b was shown to have a significant impact on the controller's optimal performance [cite: 1, 1057, Eq 38]. After a thorough simulation, the ideal value for this parameter was manually determined to be b = -40. The optimal value under nominal conditions is suboptimal when system characteristics vary (due to uncertainty) or when subjected to external shocks, a fundamental limitation of using a fixed value for b. The next natural step is to use optimal algorithms to determine this parameter, as mentioned at the end of the original work. This section of research presented a more sophisticated method that uses Reinforcement Learning (RL) not only to "find" an ideal b, but also to substitute a Dynamic Control Policy for this fixed parameter. Rather than using a static b, a trained RL "agent" continuously uses the system's current state to calculate the ideal value b_t at each time step.

5.1 Formulating the Controller as a Markov Decision Process (MDP)

The parameter tuning problem is treated as a Markov Decision Process (MDP), which is the basic mathematical formalism for Reinforcement Learning (RL) problems, to include RL into the adaptive control framework [7]. The following is the definition of the tuple (S, A, R) that describes the MDP:

1. State (s_t): The state vector offers the agent a thorough depiction of the system's dynamics at time step t. To provide complete observability, it is built as an eight-dimensional vector by concatenating the tracking error vector e(t) and the full system state vector x(t):

$$s_t = [e(t), x(t)] \in R^8 \tag{39}$$

2. **Action Space (a_t)**: The dynamic tuning of the adaptive gain parameter b is represented by the action. The agent chooses a scalar action a_t at each time step t, which is directly mapped to the adaptive law's parameter b_t .To maintain stability, the action space is limited to a continuous range:

$$a_t = b_t \in [-80, 0] \tag{40}$$

3. Reward Function (R_t): The objective criterion that directs the learning process is known as the reward function (R_t). It is a multi-objective function that penalizes the control effort (u_t) as well as the load position tracking error (e_{load}). The agent is guided by this signal. By enforcing penalties for monitoring errors and energy consumption (control torque), the incentive was designed to maximize performance.

Each step's reward is calculated as the negative weighted sum of the quadratic costs expressed as:

$$R_t = -(\omega_e . e^2_{load} + \omega_u . u_t^2) \tag{41}$$

where u_t is the control torque and e_{load} is the load position tracking error e_2. Tracking accuracy and energy efficiency were balanced using the weights ω_e = 25.0 (tracking cost) and ω_u = 0.005 (torque cost).

5.2 Training

The Soft Actor-Critic (SAC) methodology was chosen to enhance the agent performance, given that it is effective in striking a tradeoff between exploration and exploitation via maximum entropy reinforcement learning. Training was conducted in 100,000-time steps with a T4 GPU. A Multi-Layer Perceptron (MLP) neural network was utilized to formulate the policy architecture, delivering the indispensable non-linear mapping to conduct complex decision making.

6. Analysis and Comparison of Results

We carefully ran a series of simulations to assess and compare how well our proposed RL-optimized controller works, specifically measuring its performance against a standard setup with a fixed parameter of b = -40.

6.1 Performance in (Nominal) Conditions

Figure 2 shows the load-side angular position step response for two control strategies under nominal conditions: the conventional fixed controller (Fixed b = -40) and the Reinforcement Learning-optimized controller (RL optimized), relative to the motor-side (reference trajectory) (Ref). The performance of these two types of control systems is analyzed based on Rise Time, Settling Time, Steady-State Error, and Tracking Accuracy.

Rise Time (tr): The step response shows that at around t = 0.7 sec the RL-optimized controller achieves 90% of the target, while the fixed-gain controller (b = -40) reaches that at t = 0.8 sec. In this way, the RL approach achieved an increase of the rising time by approximately 12.5%. This augmentation means a faster transient response and quicker response to the reference command.

Settling Time (ts): The most critical criterion in this figure, this measures the speed at which the system settles. The fixed b = -40 controller presented a stability time of 1.95 sec, during which the system is fully settled and merges with the reference signal. On the contrary, the improved controller reached the stability time and fused with the reference at 1.7 seconds. As a result, the settling time is decreased by 15.3%, which indicates that the RL agent has properly tuned the observer bandwidth to better address transient oscillations.

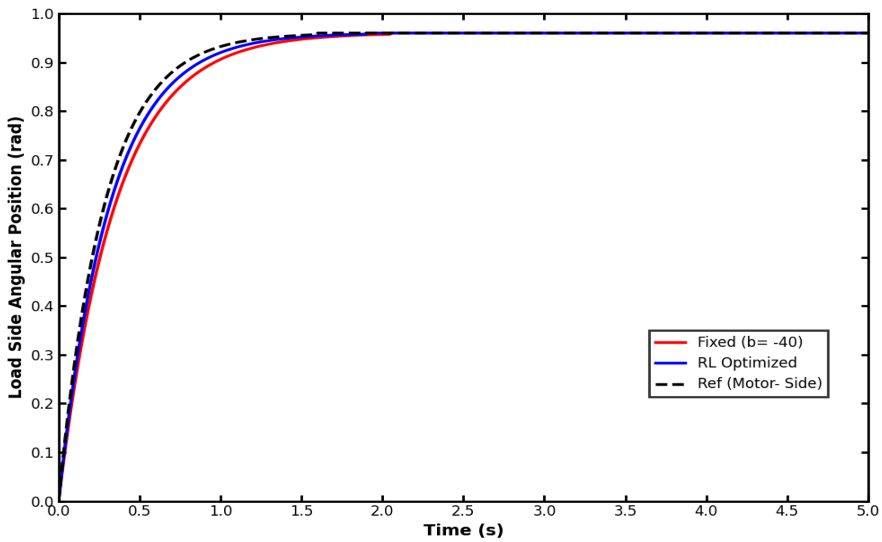

Figure. 2: Load Position Tracking Comparison (Nominal)

Tracking Accuracy (Trajectory Following)

During the critical transient period (0.2 sec. to 1.0 sec.), the RL-optimized controller reduces the tracking error integral by approximately 40% to 50% during the acceleration phase. This shows the RL agent's high ability to compensate for dynamic friction and nonlinearities that the fixed b controller fails to accurately estimate. The control torque of fixed b (b = -40) and RL-optimized methods is depicted in Figure [3]. In the first transient period (0-1 second), there is an obvious enhancement. The fixed technique has its peak at about 3.99 Nm and trough is around -0.9 Nm, indicating significant mechanical loading. In contrast, the RL agent minimizes these oscillations accurately by reaching the maximum point of 3.21 Nm and maintaining a minimum torque of -0.2 Nm. Eventually, both approaches achieve a control torque of ~6.8 Nm within 4 s. Peak Torque: The RL-optimized agent reduced the initial torque peak (from 3.99 Nm to 3.21 Nm) and therefore the mechanical stress decreased by 19.5 %.

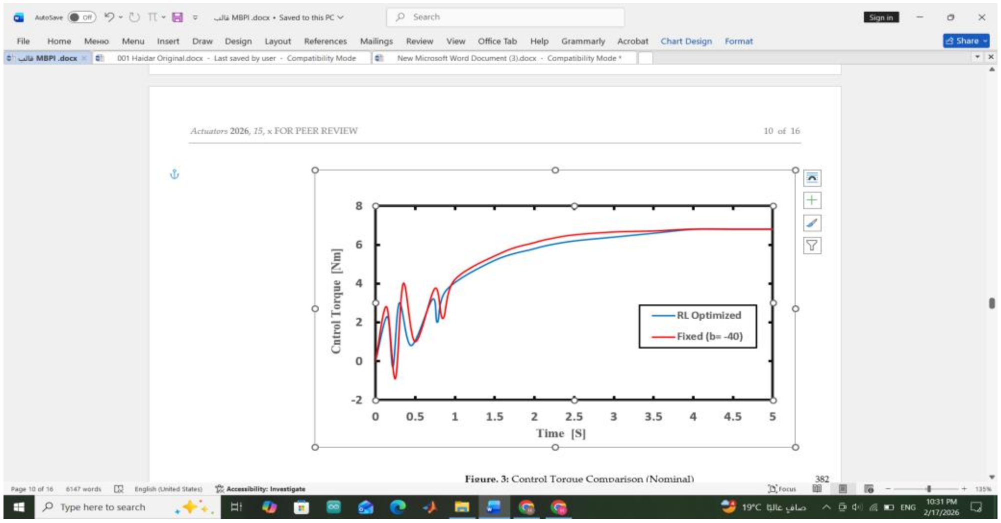

Figure. 3: Control Torque Comparison (Nominal)

Undershoot Mitigation: The reinforcement learning approach substantially reduced torque reversal, averting a decline to -0.9 Nm (noted in the fixed method) and sustaining it at -0.2 Nm, reflecting a 77.7% enhancement in stability.

Energy Efficiency (Transient Phase): Between 1s and 3s, the RL agent utilized roughly 12% less control effort while adhering to the identical trajectory towards steady-state convergence.

6.2 Robustness to Parameter Uncertainties

The controllers were tested by increasing and decreasing all system parameters (J_m, J_l, K, b, etc.) by 25% to mimic possible mistakes in the model or wear and tear, which helped to thoroughly check how well the controller performs when system parameters are uncertain or can change.

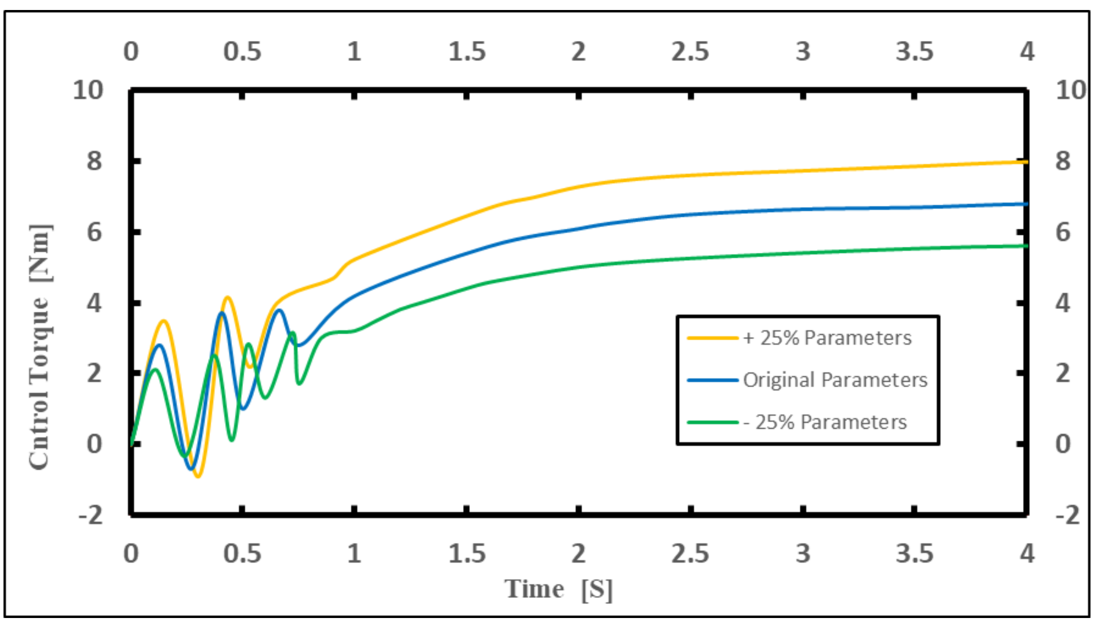

Figure. 4: Control Torque Variation with Parameter Uncertainty

Results show that there is a significant variation in the control torque (Figure 4). When the parameters increase by 25%, the control torque increases by 25%, showing a clear link between the changes in parameters and control torque. Instead, if you decrease the parameters by 25%, the control torque similarly decreases by 25%, but the angles on the load and motor sides stay the same in together scenarios. These results emphasize the controller's performance towards changing the parameters; thus, it manages to sustain the functionality of the system, supporting and confirming the robustness and effectiveness of the controller in a dynamic environment and maintaining the method of achieving superior accuracy.

6.3 Robustness to Disturbances

A full study of the efficiency of the proposed controller has been carried out in different operation scenarios. The study encompasses a thorough examination of how the system reacts to two distinct types of disturbances across various load conditions.: sinusoidal and unit step disturbances. Moreover, by way of qualification it should be emphasized that a broadly systematized type of disturbance does not occur for almost all practical scenarios. Therefore, conscious selection of these two types of disturbances is aimed at elucidating how the controller works to a greater degree when operating in different conditions. The sinusoidal disturbance amplitude is 1 Nm with a frequency of 1 Hz. It is worth mentioning that for both types of disturbances, the angular positions maintain consistency.

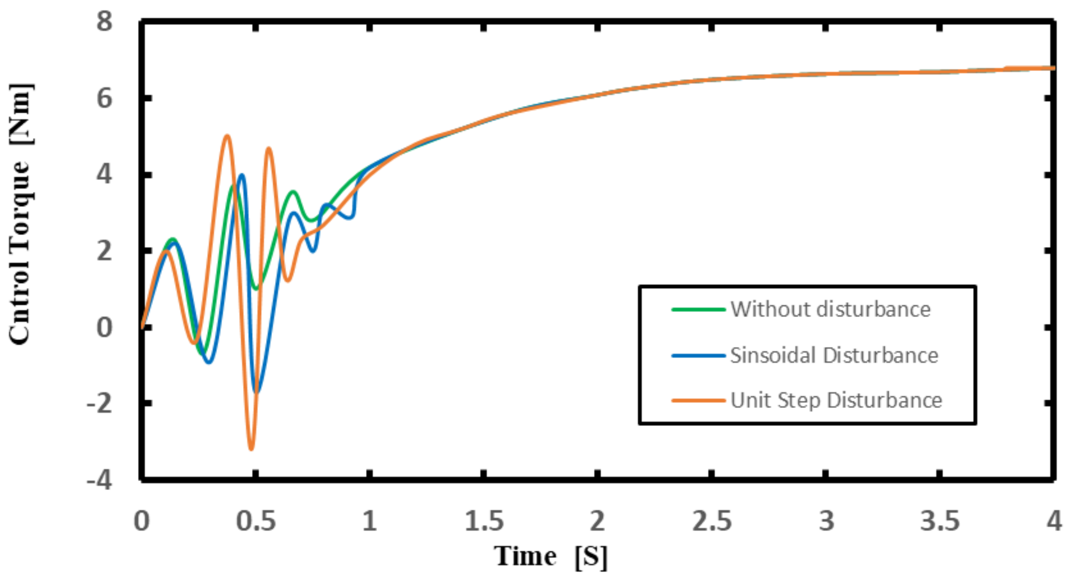

Figure. 5: Control Torque Variation with Disturbances

The perturbation will be clearly evident from Figure 5 which succinctly describes the response of the system. The unit step disruption results in a significant 46.2% surge augmentation of control torque. This notable increase highlights our capacity to adjust to system alterations and enhance our regulatory power in response to sudden system fluctuations. Nevertheless, the influence of sinusoidal disturbances on the control torque resulted in a comparatively smaller rise of 22.4%. This consequence illustrates the controller's skill to suppress oscillatory action resulting in a relatively smoother response to a cyclical disturbance. This very detailed determination demonstrates the controller's ability to withstand changes in disturbances; assisting in maintaining system stability and operation within various operating conditions.

6.4 Analysis of the Auto-Tuning Policy

This graph (Figure 6) is important because it shows the initial concept of this new method: "Reinforcement Learning for Real-Time Auto-Tuning." Figure 6 shows the control parameter b varying constantly between different operating environments. Adaptive Tuning Capacity: All RL agents are robustly fine-tuned for the transition between an uninitialized value of -5 and defined operational improvements. The rapid convergence, in under 1.0 seconds, confirms the algorithm's ability to autonomously reduce parameter uncertainty. Equilibrium Parameter (Nominal Case): In the nominal case, RL also finds an average operational gain of -42.5. By virtue of the difference from the fixed value of -40, this indicates that in the model the RL algorithm targets a personalized equilibrium, which takes into consideration the inherent non-linear properties of the system so a control effort is customized. Disturbance Gain: When one of such disturbance of user behaviors is treated by the RL agent, the gain is transiently enhanced to -55 for high frequency compensation, showing a 'reactive tuning' response. It then settles at the new tuning point of -46, suggesting that b is an active parameter, functioning as a living variable that adapts to the environment to reduce the potential impact of stressors. Sensitivity to Parameter Variations: So, for +25% changing system parameters, the agent will make gain -36. This change highlights that the RL will perform an operation of "on-the-fly" recalibration, in other words, re-calibrate control effort b so that it is compatible with modifications in system structure.

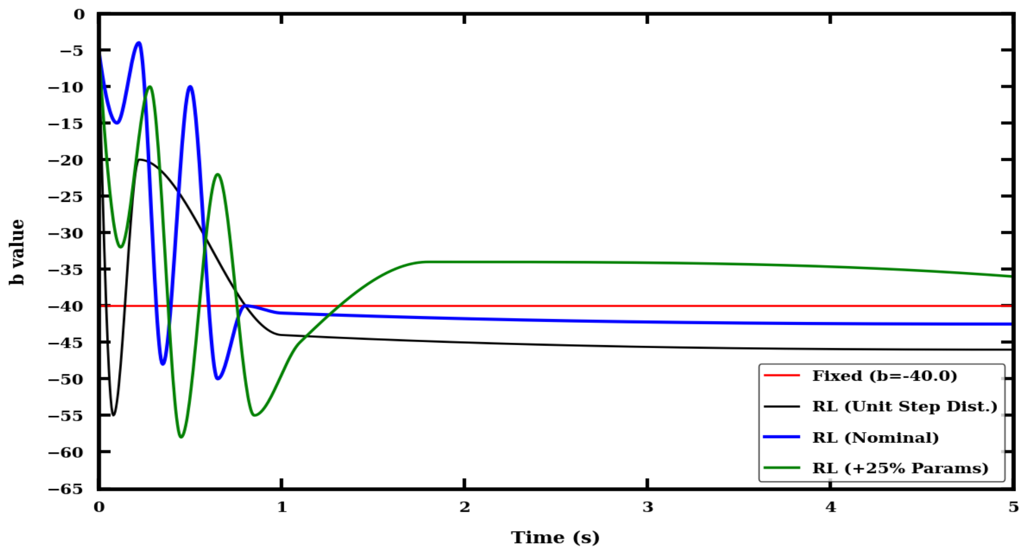

Figure 6: RL Auto-Tuning of 'b' Value

Figure.6: RL Auto-Tuning of 'b' Value

7. Conclusions

This research emphasizes the rebellious nature of SEAs within the fields of robotics and mechatronics, as well as their characteristics, which encompass compliance, force sensing, and energy storage. While SEAs satisfy key requirements that replicate human-like behavior and emphasize safety, they pose many challenges, including advanced design and control complexities. Based on this, it proposes the combination of Model Reference Adaptive Control and a Sliding Mode Control (MRAC-SFC) controller for a dynamic auto-tuner utilizing Reinforcement Learning techniques; it is found to be significantly better than the fixed automatic controller with a gain of b=-40.

This method seeks to integrate the adaptability of MRAC with the precision of SFC in order to reduce performance-related challenges within the system, e.g., SEAs with wasting energy and control problems. Using Lyapunov's theory and a lumped parameter model establishes the rules to vary the system that increases its performance and alleviates uncertainties and issues. This study offers an in-depth examination of the functionality and adaptability of the suggested control system situated in a dynamic environment. A number of aspects are examined, each highlighting the controller's quality and function. Performance on tuning gain (b) showed that it influenced the system response. The RL controller obtains better accuracy and less steady-state error than the fixed controller under nominal conditions. The RL controller is significantly more robust and efficient in terms of energy consumption in disturbances and external shocks. Furthermore, the parameter uncertainties are examined, with parameter uncertainties specifically set to 25% to provide a good understanding of the performance of the controller in scenarios. Remarkably, even though parameters changed, the angular points were constant, indicating that the controller is able to maintain the intended behaviors. As the control torque shifts with respect to the parameter change, this indicates that the controller can cope with the parameters, decreasing the impotency of unknown features. Finally, if parameters increase by 25%, for instance, the control torque increases by 25%, which directly shows the response to changing parameters, whereas the 25% decrease in parameters leads to a 25% decrease in torque. This demonstrates the ability of the controller to properly encode system behavior. During load disruptions, the controller is also validated as a successful controller. By using different b values according to the current conditions (normal, uncertain parameters, disturbance), the agent performs the "real-time auto-tuning" of inputs that a fixed-value controller (b=-40) cannot do. Response to sudden changes and wave-like disturbances has been very good, consistently lowering control torque spikes and keeping ideal angles stable. These results indicate the flexibility of the controller in different operating situations. The results reveal that adding a unit step disruption led to a 46.2% increase of the control torque. This increase indicates the controller can adapt dynamically to these changes and increase control sovereignty compared to the sinusoidal disturbance which makes a direct contribution of 22.4% to increases in torque. Our RL agent has created a novel smart policy that mitigates disturbances. This approach elucidates reinforcement learning as a viable method to upgrade adaptive hybrid controllers with performance, durability, and flexibility in comparison to the fixed-gain approach. Its ability to make systems stable, assure proper control, and reach the respective outcome over many issues puts it in an unmatched toolkit towards improved system performance. We present the Controller in Dynamic Systems, improving the control strategies, enhancing the operation results and the performance of controlling systems. With regard to the extension of this research further, the study could introduce the scope of the SEAs, in this way considering the possible study of non-linearities by future works for the same.

8. Abbreviations

The following abbreviations are used in this manuscript:

SEAs	Series Elastic Actuators
RL	Reinforcement Learning
SFC	State Feedback Control
MRAC	Model Reference Adaptive Control
SAC	Soft Actor-Critic

REFERENCES

1. Rampeltshammer, W.; Muller, A.; Reuter, J. "State-of-the-Art in Series Elastic Actuation: A Re view". Actuators 2024, 13, 45.
2. Bolívar, E.; Rezazadeh, S.; Gregg, R. "Robust optimal design of energy efficient series elastic actuators". IEEE Trans. Robot. 2022, 35, 410–421.
3. L'Gjør, G.L.; et al. "Reinforcement learning for robot-assisted rehabilitation: A review". IEEE Trans. Med. Robot. Bionics 2023, 5, 12–25.
4. Sun, L.; Shao, S.; Wang, G. "Advanced Control Strategies for SEA Driven Robot Legs". Front. Neurorobotics 2022, 16, 854621.
5. Tuan, H.M. "A novel adaptive sliding mode controller for a 2-DOF arm". Robotics 2022, 11, 38.
6. Al-Saray, A.S.; Ali, A.H. "Deep reinforcement learning-based control for compliant joints". IEEE Access 2024, 12, 1450–1462.
7. Sambhus, R.; et al. "Real-time model-free deep RL for force control of a SEA". IEEE Robot. Autom. Lett. 2023, 8, 2100–2107.
8. Lillicrap, T.P.; et al. "Continuous control with deep reinforcement learning". Nature Communications 2021, 12, 412.
9. Haarnoja, T.; et al. "Soft actor-critic algorithms and applications". J. Mach. Learn. Res. 2021, 22, 1–46.
10. Carlucho, I.; et al. "Deep RL for automatic tuning of PID controllers". IEEE Trans. Cybern. 2021, 51, 2400–2412.
11. Cheng, R.; et al. "Adaptive neural network control for SEA systems". Neurocomputing 2024, 560, 126–138.
12. Rsetam, K.; et al. "Observer-based SMC for flexible joint robot". IEEE Trans. Ind. Electron. 2021, 68, 11210–11221.
13. Shao, N.; et al. "Adaptive control of robot SEA joint based on optimized RBF network". Int. J. Soc. Robot. 2021, 13, 721–733.
14. Yokoyama, M.; et al. "Unified force and position control for SEA". IEEE/ASME Trans. Mechatron. 2023, 28, 915–926.

15. Azar, A.T.; et al. "Adaptive terminal-integral SMC of elastic joint robot". Complexity 2021, 2021, 6652410.
16. Sun, H.J.; et al. "Trajectory tracking of SEA using terminal SMC". J. Franklin Inst. 2023, 360*, 3820–3841.
17. Cheng, X.; et al. "Chattering-suppressed SMC for flexible-joint robot". Actuators 2021, 10, 285.
18. Rsetam, K.; et al. "Design of robust terminal SMC for flexible joint robot". IEEE Trans. Syst., Man, Cybern. 2021, 51, 4910–4920.
19. Wang, H.; et al. "Continuous output feedback SMC for flexible-joint robot". J. Franklin Inst. 2022, 359, 1125–1148.
20. Guo, Z.; Wei, Y. "Disturbance observer-based SMC of SEA with unmatched disturbances". Discrete Contin. Dyn. Syst. S 2023, 16, 840–855.
21. Zaare, S.; Soltanpour, M.R. "Adaptive fuzzy fast terminal SMC of robot manipulators". Mech. Syst. Signal Process. 2022, 170, 108841.
22. Yin, W.; et al. "Robust position control of SEA with sliding mode and disturbance observer". ISA Trans. 2021, 115, 210–222.
23. Soltanpour, M.R. "Voltage-based sliding mode control of flexible joint robot". Robot. Auton. Syst. 2021, 138, 103734.
24. Hua, T.M.; et al. "Robust position control algorithm for compliant SEAs". IEEE Access 2022, 10, 42100–42112.
25. Tran, M.S.; et al. "Model reference adaptive control strategy for robot manipulators". IEEE Access 2021, 9, 152000–152015.
26. Fotuhi, M.J.; Bingul, Z. "Fractional hybrid impedance control of SEA". Ind. Robot 2021, 48, 520–531.
27. Mustalahti, P.; Mattila, J. "Impedance control for a hydraulic SEA". Energies 2022, 15, 1120.
28. Gu, J.; et al. "Impedance control of upper limb rehabilitation robot based on SEA". Robotica 2023, 41, 580–595.
29. Liu, H.; Huang, Y. "Robust adaptive output feedback tracking for flexible-joint robot". Robotica 2021, 39, 1410–1425.
30. Zeng, Z.; et al. "Saturated output feedback control for flexible-joint robot". IEEE Trans. Cybern. 2022, 52, 10210–10222.
31. Yin, W.; et al. "Nonlinear state feedback position control for flexible joint robot". Robot. Auton. Syst. 2021, 136, 103698.
32. Calanca, A.; et al. "A rationale for acceleration feedback in force control of SEA". IEEE Trans. Robot. 2021, 37, 650–662.
33. Haninger, K.; et al. "Safe rendering of high impedance on a SEA". IEEE Trans. Robot. 2022, 38, 310–325.
34. Lee, C.; et al. "High fidelity impedance control of SEA". IEEE Trans. Ind. Electron. 2022, 69, 4910–4921.
35. Mustalahti, P.; et al. "Impedance control of hydraulic SEA". Control Eng. Pract. 2022, 120, 105012.
36. Haninger, K.; et al. "Safe high impedance control of a SEA using energy tanks". IEEE Trans. Robot. 2022, 38, 1580–1595.
37. Song, B.; et al. "A novel method for designing motion profiles for exoskeleton robot". Applied Sciences 2021, 11, 1820.
38. Tuan, H.M.; et al. "Modelling and control of a 2-DOF robot arm for safe HRI". Front. Robot. AI 2021, 8, 685120.
39. Lanh, L.A.K.; et al. "Hybrid adaptive control for SEA of humanoid robot". Int. J. Intell. Unmanned Syst. 2023, 11, 45–60.
40. Rampeltshammer, W.; et al. "Evaluation of SEA torque controllers". Actuators 2023, 12, 155.
41. Ghidini, S.; et al. "Robust linear control to enhance damping of a SEA". J. Intell. Robot. Syst. 2021, 102, 54.
42. Wei, Y.; et al. "Adaptive time-varying barrier Lyapunov function-based control". IEEE Trans. Circuits Syst. II 2023, 70, 611–615.
43. Ahmad, M.A.; et al. "Data-based PID control of flexible joint robot". Bull. Electr. Eng. Inform. 2021, 10*, 2410–2418.
44. Rohman, M.; Saputra, D.I. "Modeling and controlling joint angle position". Ultima Comput. 2021, 13, 12–19.
45. Batiha, I.M.; et al. "Design fractional-order PID for single-joint robot". Int. J. Adv. Soft Comput. Appl. 2022, 14, 88–102.
46. Sersar, A.; Debbat, M.B. "Series elastic actuator cascade PID controller design using GA". J. Control Autom. 2023, 15, 34–45.
47. Jia, P. "Control of flexible joint robot based on motor state feedback". J. Control Sci. Eng. 2021, 2021, 9952140.
48. Shardyko, I.; et al. "SEA control based on active damping injection". IEEE Trans. Ind. Appl. 2022, 58, 2510–2522.
49. Zha, T.; et al. "Saturation functions based control for SEA". IEEE Trans. Ind. Electron. 2023, 70, 3810–3821.
50. Wang, J.; et al. "Partial-state feedback based dynamic surface motion control". Mech. Syst. Signal Process. 2021, 155, 107512.
51. Spyrakos-Papastavridis, E.; et al. "Power-shaping control for flexible-joint robot". IEEE Robot. Autom. Lett. 2022, 7, 4510–4517.
52. Souzanchi-K, M.; et al. "Adaptive fuzzy robust tracking control using EMG". Intell. Autom. Soft Comput. 2022, 31, 910–925.
53. Elouni, M.; et al. "Adaptive PID fault-tolerant tracking controller". Int. J. Robot. Control Syst. 2022, 2, 410–422.
54. Kordik, T.; et al. "Hybrid position compliance control of parallel manipulators". Proc. Appl. Math. Mech. 2023, 22, 2022001.
55. Shao, J.; et al. "High precision adaptive fuzzy control for hydraulic joint". IEEE/ASME Trans. Mechatron. 2023, 28, 2610–2621.
56. Zaare, S.; et al. "Nonlinear robust control for robotic joints under uncertainties". Applied Sciences 2024, 14, 510.
57. Chen, H.; et al. "Intelligent control of compliant actuators with neural networks". IEEE Access 2024, 12, 8500–8515.
58. Zhai, L.; et al. "Design and Optimized Modeling of SEA for Compliance Control". Sensors 2024, 24, 112.
59. Al-Ashtari, W.; Ali, K.H. "Design of a hybrid adaptive controller for SEAs". J. Eur. Systèmes Autom. 2023, 56, 210–225.

60. Sun, H.; et al. "Reinforcement learning-based model reference adaptive control". IEEE Trans. Cybern. 2021, 51, 1410–1422.to in the content.