ENERGY CONSUMPTION EVALUATION OF 3 DOF IMPEDANCE-CONTROLLED ROBOTS USING SINUSOIDAL AND SIGMOID INPUT TRAJECTORY FUNCTIONS

Abdullah ERDEMİR

MPG Machinery Production Group Inc. Co.
ORCID ID: 0000-0002-7267-3111

Mete KALYONCU

Prof. Dr., Konya Technical University, Department of Mechanical Engineering
ORCID ID: 0000-0002-2214-7631

ABSTRACT

This study aims to investigate the impact of different input trajectory functions on the energy consumption of a 3 degree-of-freedom (DOF) robot controlled by an impedance controller. Industrial robot manipulators, such as painting robots, require arm position control; however, situations may arise where obstacles exist between the robot and its environment, necessitating the avoidance of harm to both itself and other entities. Impedance control enables the dynamic relationship between the robot and the environment to be managed. The force exerted by the robot on the environment is dependent on the endpoint position of the robot and the corresponding impedance. The input trajectory function employed in impedance control affects the interaction distance of the impedance force, which, in turn, impacts the interaction force and torques in the joints, thereby influencing the robot's energy consumption. To optimize the impedance controller parameters, the Bees Algorithm was used to minimize positioning errors for three input trajectory functions: step, sinusoidal, and sigmoid. The study compared the energy consumption of each input trajectory function and presented the results in numerical and graphical formats. The study concluded that the input trajectory function has a significant impact on the energy consumption of the robot. The step function was found to be the simplest to implement but caused the highest energy consumption. The study contributes to a better understanding of the impact of input trajectory functions on the energy consumption of robots controlled by impedance controllers. The findings could be beneficial in selecting the most energy-efficient trajectory function for a specific robot manipulation task.

Keywords: Trajectory functions, impedance control, energy consumption, 3 DOF robots, The Bees Algorithm.

1. INTRODUCTION

Robotic manipulators have found wide application in various industrial tasks, including painting, assembly, and welding, among others. While some tasks require precise arm position control without environmental interaction, certain applications, such as polishing, cleaning, grinding, and pushing, necessitate robot-environment interaction while avoiding self-damage and environmental harm [1, 2]. Impedance control is an approach employed to regulate the dynamic interplay between the robot manipulator and the environment [3-6]. The force exerted by the

robot on the environment is reliant on the endpoint's position and corresponding impedance in impedance control [7, 8]. The input trajectory function utilized in impedance control directly impacts the impedance force's interaction distance, interaction force, and torques in the joints, all of which impact the robot's energy consumption [9, 10].

Optimization algorithms can be grouped into two primary categories: global optimization algorithms and local search algorithms [11]. Global optimization algorithms, such as Genetic Algorithm [12, 13], Particle Swarm Optimization [14], and The Bees Algorithm [15-17], are designed to explore the entire solution space. On the other hand, local search algorithms like Hooke-Jeeves [18] and Newton Raphson [19] are only concerned with local areas of the solution space. While global optimization algorithms are known for their ability to find the global optimum, local search algorithms can quickly and efficiently identify optimal solutions in limited or constrained regions of the solution space. Selecting an appropriate algorithm depends on the optimization problem's nature and the solution space's characteristics.

This study aims to investigate the energy consumption of a 3 degree-of-freedom (DOF) robot under the influence of an impedance controller, utilizing different input trajectory functions. The goal is to identify the most energy-efficient trajectory function for a specific manipulation task of the robot [20]. To achieve this objective, the study utilizes the Bees Algorithm to optimize the impedance controller parameters, minimizing the positioning errors [21]. The study evaluates three input trajectory functions, namely step, sinusoidal, and sigmoid, by comparing their energy consumption. The study presents the results both numerically and graphically. The findings indicate that the choice of input trajectory function has a significant impact on the energy consumption of the robot.

The practical significance of the present study lies in the identification of the most energy-efficient trajectory function for robotic manipulation tasks that necessitate impedance control. The results can potentially curtail energy consumption and augment the efficiency of robot manipulators in industrial applications. Additionally, the study contributes to the knowledge of how input trajectory functions influence the energy consumption of impedance-controlled robots. These findings offer opportunities for enhancing the design and control of robotic manipulators for various industrial applications.

2. MATHEMATICAL MODEL

The subject of this investigation is a 3 degree-of-freedom (DOF) manipulator consisting of three joints. The manipulator's physical structure comprises three links with masses denoted by m_1, m_2, and m_3, as illustrated in Figure 1.

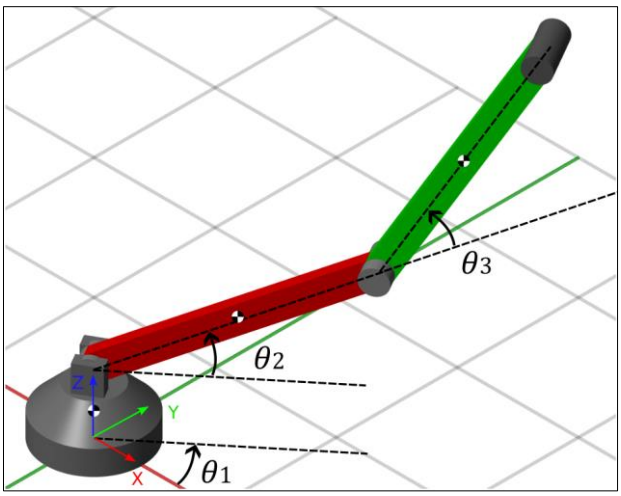


Figure 1. 3 dof robotic system

Forward kinematics is a fundamental method for determining the end-effector position in relation to a reference coordinate system. Denavit and Hartenberg [22] introduced a systematic approach to this technique [20]. The approach involves a sequence of transformations, including d-translation along the z-axis, θ-rotation about the z-axis, a-translation along the x-axis, and α-rotation about the x-axis, to obtain a homogeneous transformation matrix, as shown in Equation (1) and (2).

$$^{i-1}A_i = T_z(d)R_z(\theta)T_x(a)R_x(\alpha) \tag{1}$$

$$^{i-1}A_i = \begin{bmatrix} cos\,\theta_i & -cos\,\alpha_i\,sin\,\theta_i & sin\,\alpha_i\,sin\,\theta_i & a_i\,cos\,\theta_i \\ sin\,\theta_i & cos\,\alpha_i\,cos\,\theta_i & -sin\,\alpha_i\,cos\,\theta_i & a_i\,sin\,\theta_i \\ 0 & sin\,\alpha_i & cos\,\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

Table 1 displays the Denavit-Hartenberg (D-H) parameters of the 3-degree-of-freedom (DOF) robot, which are used to define the kinematic relationship between the robot's joints and end-effectors. These parameters provide a systematic way to obtain the homogeneous transformation matrices that are essential for computing the forward kinematics of the robot.

Table 1. Denavit-Hartenberg Parameters for the 3 Degree-of-Freedom Robot

i	$\boldsymbol{d}$ **[mm]**	$\boldsymbol{\theta}$ [°]	$\boldsymbol{a}$ **[mm]**	$\boldsymbol{\alpha}$[°]
1	d_1	0	0	$\pi/2$
2	0	L_2	0	0
3	0	L_3	0	0

Equation (3) provides a precise specification of the center of mass for each body, which is defined relative to its own coordinate system.

$$\begin{aligned} {}^{1}r_1 &= [0 \quad 0 \quad -d_1/2]^T \\ {}^{2}r_2 &= [-L_2/2 \quad 0 \quad 0]^T \\ {}^{3}r_3 &= [-L_3/2 \quad 0 \quad 0]^T \end{aligned} \tag{3}$$

The calculation of the actuator torque for the i-th joint can be obtained using Equation (4), as described in reference [23]. The acceleration-related symmetric matrix is expressed in Equation (5), which has been previously explained in literature [23, 24]. The impedance torque acting on the i-th joint is denoted as τ_{ei}. The velocity matrix, represented by U_{ij}, provides a mathematical description of the velocity of the i-th body relative to the j-th joint angle (θ_j). The matrix U_{ij} is defined by Equation (6), which establishes a formal relationship between the body's velocity and the joint angle.

$$M(q)\ddot{\theta}_\iota + C\left(\theta_i, \dot{\theta}_\iota\right) + G(\theta_i) = \tau_{act,i} + \tau_{ei} \tag{4}$$

$$M_{ik} = \sum_{j=\max(i,k)}^{n} Tr\left(U_{jk} J_j U_{ji}^T\right) \tag{5}$$

$$U_{ij} = \begin{cases} {}^{0}A_{j-1}\ Q\ {}^{j-1}A_i & , j \le i \\ 0 & , j > i \end{cases} \tag{6}$$

The differentiation of a homogeneous matrix ‘A’ can be obtained through a left multiplication operation with the Q matrix. This matrix encapsulates the rotational aspect of the transformation matrix for revolute joints, and its mathematical formulation is presented in Equation (7). The inclusion of the Q matrix is crucial for obtaining the derivative of the homogeneous matrix A, and it plays a pivotal role in the kinematic analysis of robotic systems. This approach enables a thorough exploration of the movement patterns of robotic manipulators, allowing for a more comprehensive understanding of their behavior.

$$Q = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{7}$$

Equation (5) utilizes the trace operator, which is a widely used mathematical tool in linear algebra for computing the sum of diagonal elements of a matrix. This operator, denoted as Tr, has been previously discussed in literature by Fu et al. [23] and Weisstein [25]. The formal

definition of the trace operator is given by Equation (8), which provides a compact representation of the operator.

$$Tr(a) = \sum_{i=1}^{n} a_{ii} \tag{8}$$

The torques τ_{act1}, τ_{act2}, and τ_{act3} can be obtained by expanding the equation of motion, as represented by Equation (4). The equation of motion is a fundamental tool in physics and engineering for characterizing the dynamics of physical systems. By expanding Equations (9) – (11), the torques necessary for achieving the observed motion of the system can be derived. This approach provides a comprehensive understanding of the system's behavior and enables the calculation of the required actuator torques for precise control of the system.

$$\begin{aligned}\tau_{act1} = &-\frac{1}{12}(12m_3 + m_1)L_2^2\dot{\theta}_1\dot{\theta}_2\sin(2\theta_2) - \frac{1}{12}(\dot{\theta}_2 + \dot{\theta}_3)L_3^2\dot{\theta}_1 m_2\sin(2\theta_2 + 2\theta_3) \\ &- \frac{1}{2}L_2L_3\dot{\theta}_1\dot{\theta}_3 m_3\sin(\theta_3) - \frac{1}{2}d_1 g m_1\sin(\theta_1) + \frac{1}{2}L_2L_3\cos(2\theta_2 + \theta_3)\dot{\theta}_6 m_3 \\ &+ \frac{1}{24}((12m_3 + m_1)L_2^2 + (12m_3 + m_1)L_2^2\cos(2\theta_2) + 12L_2L_3\cos(\theta_3)m_3 \\ &+ 24I_1 + L_3^2 m_2)\dot{\theta}_6 + \frac{1}{24}L_3^2\cos(2\theta_2 + 2\theta_3)\dot{\theta}_6 m_2 + \frac{1}{4}(2m_3 \\ &+ m_2)L_2\cos(-\theta_2 + \theta_1)g + \frac{1}{4}(2m_3 + m_2)L_2\cos(\theta_1 + \theta_2)g \\ &+ \frac{1}{4}L_3\cos(-\theta_2 - \theta_3 + \theta_1)g m_3 + \frac{1}{4}L_3\cos(\theta_1 + \theta_2 + \theta_3)g m_3 - (\frac{1}{2}\dot{\theta}_3 \\ &+ \dot{\theta}_2)\, L_2L_3\dot{\theta}_1 m_3\sin(2\theta_2 + \theta_3)\end{aligned} \tag{9}$$

$$\begin{aligned}\tau_{act2} = &-\frac{1}{2}L_2L_3\dot{\theta}_3^2m_3\sin(\theta_3) - \frac{1}{4}((2m_3+m_2)L_2 + L_3\cos(\theta_3)m_3)\cos(-\theta_2+\theta_1)g \\ &-\frac{1}{4}L_3gm_3\sin(-\theta_2+\theta_1)\sin(\theta_3) - \frac{1}{4}L_3gm_3\sin(\theta_1+\theta_2)\sin(\theta_3) \\ &+\frac{1}{2}((\frac{1}{12}(12m_3+m_1)L_2 + L_3\cos(\theta_3)m_3)L_2 \\ &+\frac{1}{12}L_3^2\cos(2\theta_3)m_2)\dot{\theta}_1^2\sin(2\theta_2) + \frac{1}{2}(\frac{1}{12}L_3m_2\sin(2\theta_3) \\ &+L_2m_3\sin(\theta_3))L_3\cos(2\theta_2)\dot{\theta}_1^2 + \frac{1}{24}((2m_1+24m_3)L_2^2 + 2L_3^2m_2 \\ &+24L_2L_3\cos(\theta_3)m_3)\dot{\theta}_5 + \frac{1}{24}(12L_2L_3\cos(\theta_3)m_3 + 2L_3^2m_2)\dot{\theta}_4 \\ &+\frac{1}{4}((2m_3+m_2)L_2 + L_3\cos(\theta_3)m_3)\cos(\theta_1+\theta_2)g \\ &-L_2L_3\dot{\theta}_2\dot{\theta}_3m_3\sin(\theta_3)\end{aligned} \tag{10}$$

$$\begin{aligned}\tau_{act3} = &-\frac{1}{4}L_3\cos(-\theta_2-\theta_3+\theta_1)gm_3 + \frac{1}{12}L_3^2\dot{\theta}_4m_2 + \frac{1}{2}L_2L_3\dot{\theta}_2^2m_3\sin(\theta_3) \\ &+\frac{1}{24}(12L_2L_3\cos(\theta_3)m_3 + 2L_3^2m_2)\dot{\theta}_5 + \frac{1}{24}L_3^2\dot{\theta}_1^2m_2\sin(2\theta_2+2\theta_3) \\ &+\frac{1}{4}L_2L_3\dot{\theta}_1^2m_3\sin(2\theta_2+\theta_3) + \frac{1}{4}L_2L_3\dot{\theta}_1^2m_3\sin(\theta_3) + \frac{1}{4}L_3\cos(\theta_1+\theta_2 \\ &+\theta_3)gm_3\end{aligned} \tag{11}$$

3. IMPLEMENTATION OF IMPEDANCE CONTROL

Figure 2 illustrates the implementation of impedance control in the robotic system. This control technique necessitates the computation of both the forward kinematics and inverse dynamics equations for the desired point of interaction, based on the robot's joint angles. The deviation between the calculated position obtained from forward kinematics and the desired position is converted into an interaction force by multiplying it with the spring constant and damping coefficients. The resulting force is then multiplied with the transpose of the Jacobian matrix, which produces torques at the joints. The torques calculated using inverse dynamics are then combined with these torques to obtain the final torque signal that is sent to the robot, thereby enabling control.

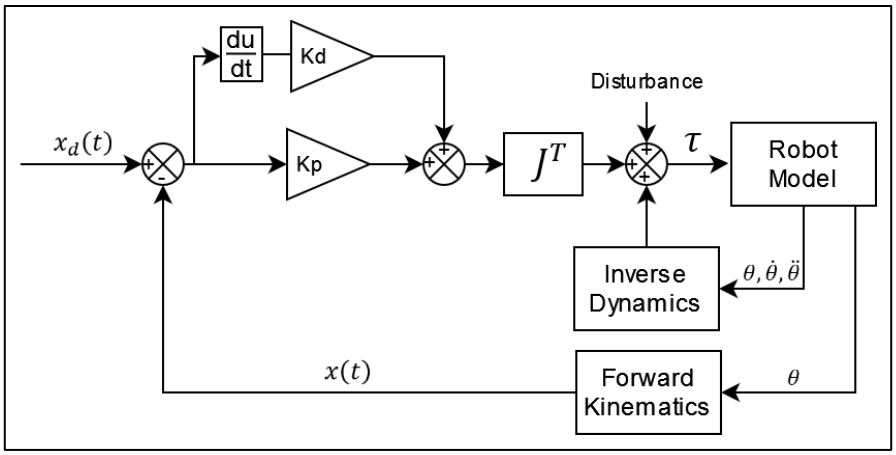


Figure 2. Schematic of Impedance Control Implementation

Equation (12) presents the mathematical expression for the interaction force produced by the Proportional-Integral-Derivative (PID) control. The PID control is a widely used feedback control technique that is employed in various applications to regulate the behavior of dynamic systems.

$$F_{int} = k\begin{bmatrix} x_{3d} - x_3 \\ y_{3d} - y_3 \\ z_{3d} - z_3 \end{bmatrix} + b\begin{bmatrix} \dot{x}_{3d} - \dot{x}_3 \\ \dot{y}_{3d} - \dot{y}_3 \\ \dot{z}_{3d} - \dot{z}_3 \end{bmatrix} + i\int \begin{bmatrix} x_{2d} - x_2 \\ y_{2d} - y_2 \\ z_{3d} - z_3 \end{bmatrix} dt \tag{12}$$

The impedance controller includes three parameters: the spring coefficient (k), the damping coefficient (b), and the integral gain (i). The desired endpoint position of the robotic system, denoted as (x_{3d}, y_{3d} and z_{3d}), is compared to the actual endpoint position (x_3, y_3 and z_3) using Equation (13) which defines the kinematic relationship between the joint angles and the endpoint position.

$$\begin{aligned} \mathrm{x}_3 &= ((L_2 + L_3\cos(\theta_3))\cos(\theta_2) - L_3\sin(\theta_2)\sin(\theta_3))\cos(\theta_1) \\ \mathrm{y}_3 &= ((L_2 + L_3\cos(\theta_3))\cos(\theta_2) - L_3\sin(\theta_2)\sin(\theta_3))\sin(\theta_1) \\ z_3 &= (L_2 + L_3\cos(\theta_3))\sin(\theta_2) + L_3\cos(\theta_2)\sin(\theta_3) + d_1 \end{aligned} \tag{13}$$

To obtain the torques necessary for controlling the robotic system using the interaction force from Equation (12), the Jacobian matrix of the endpoint of the robot must be multiplied with the force. The Jacobian matrix of the system is mathematically represented by Equation (14).

$$J_f = \begin{bmatrix} \frac{dx_3}{d\theta_1} & \frac{dx_3}{d\theta_2} & \frac{dx_3}{d\theta_3} \\ \frac{dy_3}{d\theta_1} & \frac{dy_3}{d\theta_2} & \frac{dy_3}{d\theta_3} \\ \frac{dz_3}{d\theta_1} & \frac{dz_3}{d\theta_2} & \frac{dz_3}{d\theta_3} \end{bmatrix} = \begin{bmatrix} \mathrm{J}_{f11} & \mathrm{J}_{f12} & \mathrm{J}_{f13} \\ \mathrm{J}_{f21} & \mathrm{J}_{f22} & \mathrm{J}_{f23} \\ \mathrm{J}_{f31} & \mathrm{J}_{f32} & \mathrm{J}_{f33} \end{bmatrix} \tag{14}$$

$$J_{f11} = -\big((L_2 + L_3\cos(\theta_3))\cos(\theta_2) - L_3\sin(\theta_2)\sin(\theta_3)\big)\sin(\theta_1)$$
$$J_{f12} = -\big((L_2 + L_3\cos(\theta_3))\sin(\theta_2) + L_3\cos(\theta_2)\sin(\theta_3)\big)\cos(\theta_1)$$
$$J_{f13} = -(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2))L_3\cos(\theta_1)$$
$$J_{f21} = ((L_2 + L_3\cos(\theta_3))\cos(\theta_2) - L_3\sin(\theta_2)\sin(\theta_3))\cos(\theta_1)$$
$$J_{f22} = -((L_2 + L_3\cos(\theta_3))\sin(\theta_2) + L_3\cos(\theta_2)\sin(\theta_3))\sin(\theta_1)$$
$$J_{f23} = -(\cos(\theta_2)\sin(\theta_3) + \cos(\theta_3)\sin(\theta_2))L_3\sin(\theta_1)$$
$$J_{f31} = 0$$
$$J_{f32} = (L_2 + L_3\cos(\theta_3))\cos(\theta_2) - L_3\sin(\theta_2)\sin(\theta_3)$$
$$J_{f33} = (-\sin(\theta_2)\sin(\theta_3) + \cos(\theta_2)\cos(\theta_3))L_3$$

The computation of impedance torques, τ_e, involves multiplying the interaction forces obtained through impedance control with the transpose of the Jacobian matrix, as shown in Equation (15).

$$\begin{aligned}
\tau_e &= J_f^T F_{int} \\
\tau_{e1} &= -((L_2 + L_3\cos(\theta_3))\cos(\theta_2) - L_3\sin(\theta_2)\sin(\theta_3))(-F_y\cos(\theta_1) + F_x\sin(\theta_1)) \\
\tau_{e2} &= (-(L_2 + L_3\cos(\theta_3))F_x\cos(\theta_1) - F_yL_2\sin(\theta_1) - F_yL_3\cos(\theta_3)\sin(\theta_1) \\
&\quad - F_zL_3\sin(\theta_3))\sin(\theta_2) - (-F_zL_2 - F_zL_3\cos(\theta_3) + F_xL_3\cos(\theta_1)\sin(\theta_3) \\
&\quad + F_yL_3\sin(\theta_1)\sin(\theta_3))\cos(\theta_2) \\
\tau_{e3} &= -(((F_x\cos(\theta_1) + F_y\sin(\theta_1))\cos(\theta_3) + F_z\sin(\theta_3))\sin(\theta_2) + ((F_x\cos(\theta_1) \\
&\quad + F_y\sin(\theta_1))\sin(\theta_3) - F_z\cos(\theta_3))\cos(\theta_2))L_3
\end{aligned} \tag{15}$$

In the presented equation (15), the F_{int} force has two components, namely F_x, F_y and F_z, representing its x, y and z directions, respectively. The torques exerted on the system are the sum of actuator torques and the impedance torques, which are obtained by adding the impedance torques from equation (8) and the impedance torques resulting from the multiplication of the interaction forces with the transpose of the Jacobian matrix, as given in equation (15).

4. NUMERICAL APPLICATION

The physical parameters of the system, including the distance d_1 of the endpoint from the base, the lengths of the links L_2 and L_3, and the masses of the links m_1, m_2, and m_3, are specified as follows: $d_1 = 0.2625\,m$, $L_2 = L_3 = 1\,m$ and $m_1 = m_2 = m_3 = 5\,kg$. The robot's endpoint moves in a sequential manner from its initial position at point 0 to point 12, as illustrated in Figure 3.

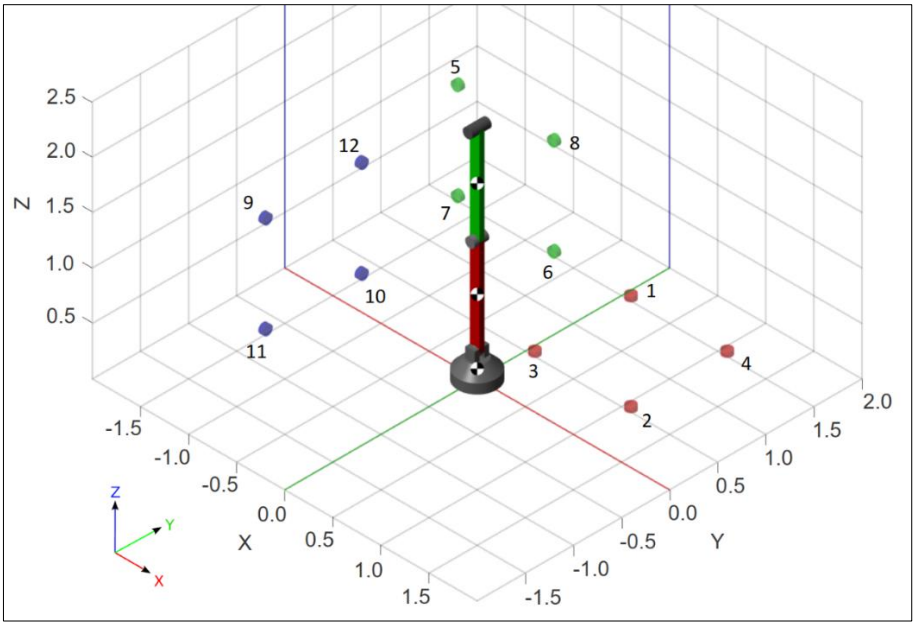


Figure 3. Numeric representation of the application

Table 2 presents the target positions of the robot's endpoint from point 1 to point 12.

Table 2. Desired endpoint positions for sequential movement from point 1 to point 12.

Point No	X [m]	Y [m]	Z [m]	Point No	X [m]	Y [m]	Z [m]
1	0.3	1.3	0.25	7	-1.2	1	0.55
2	1.3	0.3	0.25	8	-0.2	1	1.55
3	0.3	0.3	0.25	9	-1	-1.3	1.55
4	1.3	1.3	0.25	10	-1	-0.3	0.55
5	-1.2	1	1.55	11	-1	-1.3	0.55
6	-0.2	1	0.55	12	-1	-0.3	1.55

The impedance controller parameters in this study are optimized based on several trajectory functions, including a step function described by Equation (16), a sinusoidal function expressed in Equation (17), a sigmoid function given by Equation (18).

$$f_1 = \begin{cases} h_0 & ,t < t_1 \\ h_1 & ,t \geq t_1 \end{cases} \tag{16}$$

$$f_2 = \begin{cases} h_0 & ,t < t_0 \\ h_0 + (h_1 - h_0)\dfrac{\sin\left(\pi(t_m - 0.5)\right) + 1}{2} & ,t_0 \leq t \leq t_1 \\ h_1 & ,t > t_1 \end{cases} \tag{17}$$

$$f_3 = \begin{cases} h_0 & ,t < t_0 \\ h_0 + (h_1 - h_0)\left(\dfrac{1}{1 + e^{-12(t_m - 0.5)}}\right) & ,t_0 \leq t \leq t_1 \\ h_1 & ,t > t_1 \end{cases} \tag{18}$$

The study considers an initial desired position h_0 at time t_0 and a final desired position h_1 at time t_1. The time interval $[t_0, t_1]$ is mapped to the interval $[0,1]$ using the scaled time t_m, as

illustrated in Equation (19).

$$t_m = \frac{t - \mathrm{t}_0}{t_1 - t_0} \tag{19}$$

The trajectory function graphs of the step, sinusoidal and sigmoid functions are shown in Figure 4.

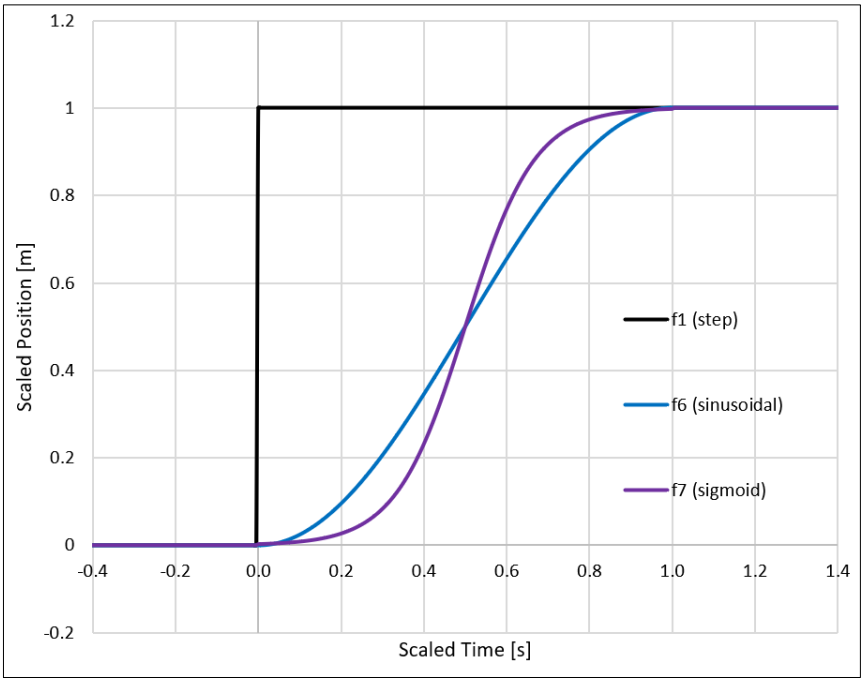


Figure 4. The trajectory function graphs.

The optimization of the coefficients k, b, and i in Equation (12) is achieved by using an objective function defined as Equation (20).

$$f_{obj} = \int_{t_1}^{t_2} (x_{3d} - x_3)^2 + (y_{3d} - y_3)^2 + (z_{3d} - z_3)^2 \tag{20}$$

The optimization of the coefficients k, b, and i in Equation (12) is accomplished through an objective function that involves squaring the difference between the target and actual position, commonly referred to as the distance error. The objective is to minimize this function, thereby reducing the distance error. The parameters used for The Bees Algorithm, which is the optimization technique employed in this study, are presented in Table 3.

Table 3. The parameters of The Bees Algorithm

n	m	e	nep	nsp	ngh	p_{max}	p_{min}
10	7	4	5	3	0.1	[250,50,20]	[0,0,0]

3. RESULTS AND DISCUSSION

The outcomes of optimizing each trajectory function using The Bees Algorithm are tabulated in Table 4. The optimization was carried out for 1000 iterations.

Table 4. Results of The Bees Algorithm optimization

	Initial Conditions			Initial Objective Function	The Bees Algorithm Result			Resulting Objective Function
	k	b	i		k	b	i	
f_1	25	3	0	911.48	248.70	38.41	2.68	17.06
f_2	25	3	0	351.34	245.37	44.91	19.44	1.15
f_3	25	3	0	532.45	241.52	31.06	19.48	1.49

The functions utilized in this study are mathematically described as follows: f_1 represents the step function, f_2 represents the sinusoidal function and f_3 represents the sigmoid function, as defined in Equations (16) through (18). The figure displaying the convergence of the objective function during the optimization process is presented in Figure 5.

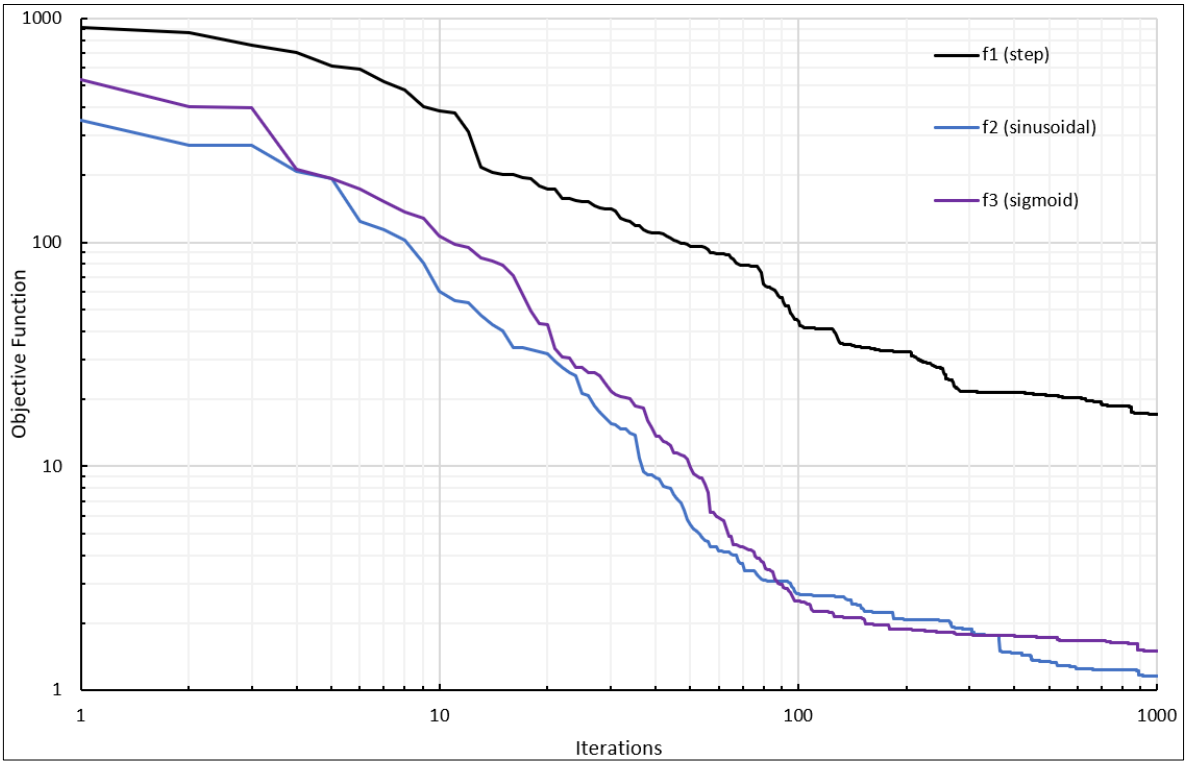


Figure 5. Convergence graph of the objective function.

Based on the scenario, the highest distance error was observed for the step function, and the second-highest error was observed in the sigmoid function, as depicted in Table 4. The final positions of the robot at each trajectory function during the scenario are represented in Figure 6.

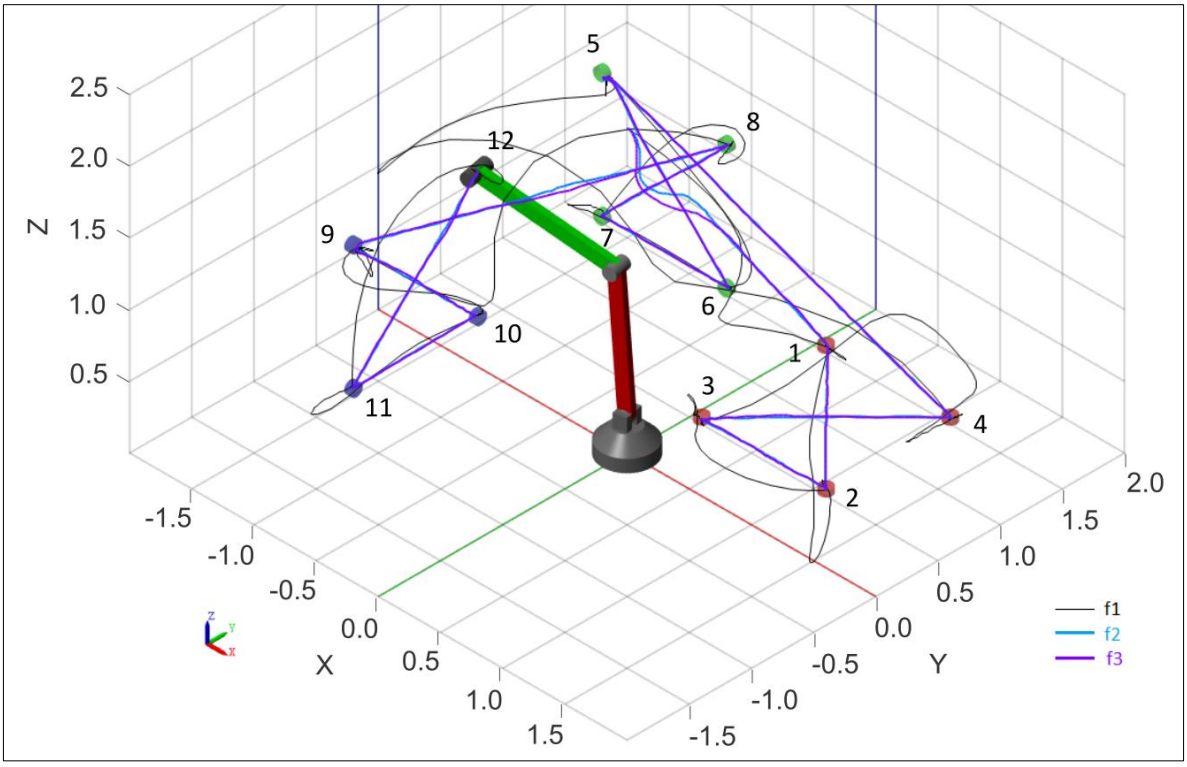
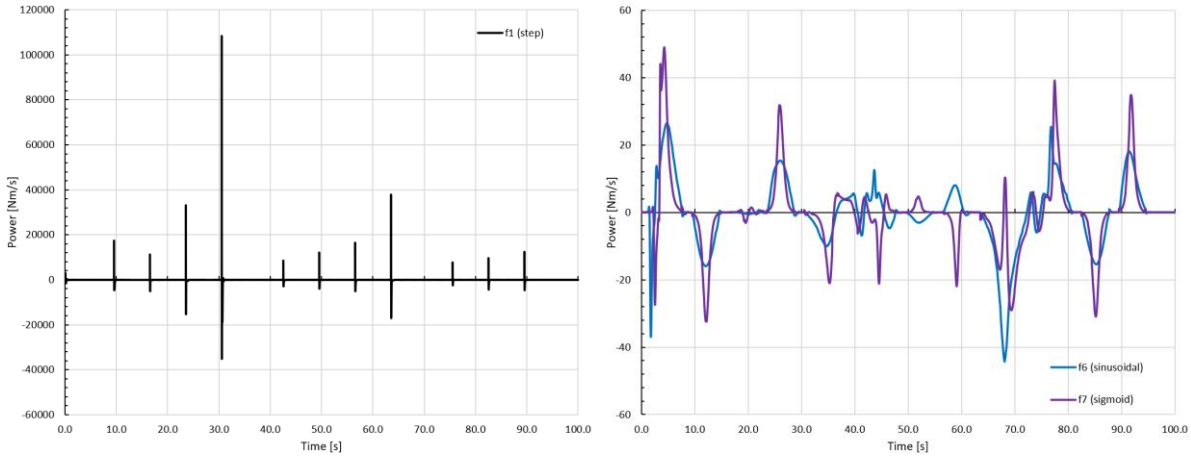


Figure 6. Movement of the robot's endpoint

The graphical representation of the power consumption of each trajectory function is presented in Figure 7.



(a) (b)

Figure 7. The power consumption of f_1 function (a) and f_2 – f_3 functions (b)

Based on the analysis, the step function was found to have significantly higher power consumption compared to the other trajectory functions, and the sigmoid function exhibited the second highest power consumption. The energy consumption values for all the functions have been presented in Table 5 and depicted in Figure 8.

Table 5. Energy consumption analysis of trajectory functions.

Function No	Trajectory Function	Energy Consumption [J]	Energy Saving Ratio (Ref. Step)
f_1	Step	8542.11	0.000%
f_2	Sinusoidal	764.35	91.05%
f_3	Sigmoid	770.96	90.97%

The sinusoidal trajectory function exhibited the minimum energy consumption compared to all other functions.

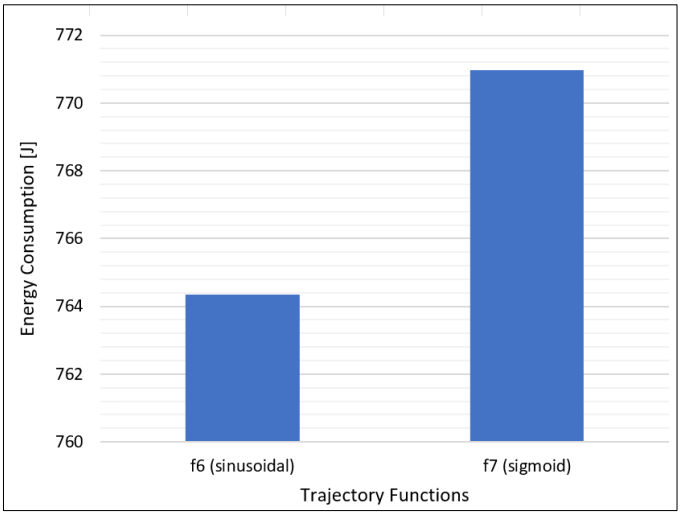


Figure 8. The power consumption of f_2 – f_3 functions

CONCLUSIONS

This research aimed to assess the energy consumption of the three trajectory functions, namely the step, sinusoidal, and sigmoid functions. The results indicated that the sinusoidal function had the lowest energy consumption, while the step function showed the highest. The sigmoid function was identified as the second most efficient. Moreover, even though the step function had the highest energy consumption, it resulted in the highest objective function error and distance error.

In general, the sinusoidal function was found to be the most appropriate trajectory function among the tested ones. It showed a significant reduction of 91.05% in energy consumption compared to the step function, and it also demonstrated the lowest computational cost among all the other functions. In future studies, it would be interesting to explore other trajectory functions based on their performance in terms of distance along the x, y, and z axes, and to compare their energy consumption to achieve more accurate optimization results.

REFERENCES

[1] Bicchi, A., Kumar, V., 2000, *Robotic grasping and contact: A review*, Proc. Proceedings 2000 ICRA. Millennium conference. IEEE international conference on robotics and automation. Symposia proceedings (Cat. No. 00CH37065), IEEE, pp. 348-353.
[2] Pliego-Jiménez, J., Arteaga-Pérez, M. A., 2015, *Adaptive position/force control for robot manipulators in contact with a rigid surface with uncertain parameters,* European Journal of Control, 22(pp. 1-12.
[3] Hogan, N., 1984, *Impedance control: An approach to manipulation*, Proc. 1984 American control conference, IEEE, pp. 304-313.
[4] Hogan, N., 1984, *Impedance control of industrial robots,* Robotics and computer-integrated manufacturing, 1(1), pp. 97-113.
[5] Hogan, N., 1987, *Stable execution of contact tasks using impedance control*, Proc. Proceedings. 1987 IEEE International Conference on Robotics and Automation, IEEE, pp. 1047-1054.
[6] Erdemir, A., Kalyoncu, M., 2023, *Optimal Impedance Control of A 3 DOF Robot*, International Paris Congress on Applied Sciences, Paris, France, pp. 229-240.
[7] Erdemir, A., Kalyoncu, M., 2023, *Optimal Impedance Control of A 2R Planar Robot Manipulator*, Mas 17th International European Conference on Mathematics, Engineering, Natural & Medical Sciences, Cairo, Egypt, pp. 82-92.
[8] Wang, J., Li, Y., 2010, *Hybrid impedance control of a 3-DOF robotic arm used for rehabilitation treatment*, Proc. 2010 IEEE International Conference on Automation Science and Engineering, IEEE, pp. 768-773.
[9] Erdemir, A., Kalyoncu, M., 2023, *Modeling Impedance Control with Limited Interaction Power for A 2R Planar Robot Arm*, 4th Latin American International Congress on Natural and Applied Sciences, Rio de Janeiro, Brazil, pp. 107-119.
[10] Erdemir, A., Kalyoncu, M., 2023, *Polynomial Input Trajectory Functions for Improved Energy Efficiency in 3 DOF Impedance Controlled Robots*, 7th International European Conference on Interdisciplinary Scientific Research, Frankfurt, Germany, pp. 568-581.
[11] Erdemir, A., Kalyoncu, M., 2019, *Optimization of a Multi-Axle Steered Heavy Vehicle Steering Mechanism by using the Bees Algorithm and the Hooke-Jeeves Algorithms Simultaneously*, International Symposium on Automotive Science And Technology (ISASTECH 2019), Ankara/Turkey, pp. 613-622.
[12] Lau, T. L., 1999, *Guided genetic algorithm*, pp. v, 166 leaves.
[13] Ortmann, M., Weber, W., 2001, *Multi-criterion optimization of robot trajectories with evolutionary strategies*, Proc. Proceedings of the 2001 Genetic and Evolutionary Computation Conference. Late-Breaking Papers, Morgan Kaufmann San Francisco, CA, pp. 310-316.
[14] Wang, D., Tan, D., Liu, L., 2018, *Particle swarm optimization algorithm: an overview*, Soft computing, 22(2), pp. 387-408.
[15] Pham, D., Kalyoncu, M., 2009, *Optimisation of a fuzzy logic controller for a flexible single-link robot arm using the Bees Algorithm*, Proc. 2009 7th IEEE International Conference on Industrial Informatics, IEEE, pp. 475-480.
[16] Erdemir, A., Kalyoncu, M., 2015, *Bir Ağır Vasıtanın Çok Akslı Direksiyon Mekanizmasının Arı Algoritması Kullanılarak Optimizasyonu*, Uluslararası Katılımlı 17. Makina Teorisi Sempozyumu UMTS 2015, pp. 421-426.
[17] Pham, D. T., Koç, E., Kalyoncu, M., Tınkır, M., 2008, *Hierarchical PID controller design for a flexible link robot manipulator using The Bees Algorithm*, Proceedings of 6th International Symposium on Intelligent Manufacturing Systems, pp. 757-765.
[18] Hooke, R., Jeeves, T. A., 1961, *``Direct Search''Solution of Numerical and Statistical Problems,* Journal of the ACM (JACM), 8(2), pp. 212-229.
[19] Ypma, T. J., 1995, *Historical development of the Newton–Raphson method,* SIAM review, 37(4), pp. 531-551.
[20] Erdemir, A., Kalyoncu, M., 2023, *Comparison of Energy Consumptions of Input Trajectory Functions in Impedance Controlled 2R Planar Robot Manipulator*, International Euroasia Congress on Scientific Researches and Recent Trends 10, Baku, Azerbaijan, pp. 312-325.
[21] Erdemir, A., Alver, V., Kalyoncu, M., 2022, *Arı Algoritması Kullanılarak Önden Dümenlemeli Bir Aracın Dümenleme Mekanizmasının Optimizasyonu*, International Aegean Conferences on Innovation Technologies & Engineering 6, İzmir, Türkiye, pp. 50-59.

[22] Denavit, J., Hartenberg, R., 1956, *Closure to" Discussions of'A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices'"(1956, ASME J. Appl. Mech., 23, pp. 151–153),* Journal of Applied Mechanics, 23(1), pp. 153.
[23] Fu, K. S., Gonzalez, R., Lee, C. G., 1987, *'Robotics: Control, Sensing, Vision, and Intelligence'*, Tata McGraw-Hill Education, 580 p.
[24] Kurtuhuz, A., Dumitru, C., 2021, *Second Order Dynamic Modelling Of A Trimobil Scara Robot Using A Symbolic Computational Method,* International Journal of Mechatronics and Applied Mechanics, 9), pp. 40-44.
[25] Weisstein, E. W., 2002, *CRC concise encyclopedia of mathematics*, Chapman and Hall/CRC, 1984 p.