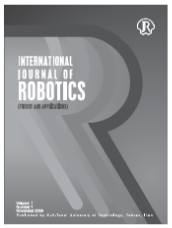

INTERNATIONAL JOURNAL OF ROBOTICS
(THEORY AND APPLICATIONS)
E-mail: IJR@kntu.ac.ir

Optimal Machine Learning based Multiple Impedance Control of a Space Free-Flying Robot

A. Safdari[a], P. Zarafshan[b,*], K. Alipour[a], B. Tarvirdizadeh[a], G. Fang[b]

[a] Department of Mechatronics Engineering, School of Intelligent Systems Engineering, College of Interdisciplinary Science and Technology, University of Tehran, Tehran, Iran
[b] School of Engineering, Design and Built Environment, Western Sydney University, Sydney, Australia

ARTICLE INFO

Article history:
Submit: 2025-08-15
Revise: 2025-09-24
Accept: 2025-10-27

Keywords:
Machine Learning
Multiple Impedance Control
Computational Complexity
Genetic Algorithm

ABSTRACT

Multiple Impedance Control (MIC) in Space Free-Flying Robot (SFFR) is necessary to ensure simultaneous accurate tracking and safe interactions; however, the computation related to grasp in these interactions has become a computational bottleneck, which intensifies with increasing Degrees of Freedom (DoF) and makes on-line control and real-time implementation difficult. Although Machine Learning-based Multiple Impedance Control (ML-MIC) has partly reduced this computational burden, the design of the Machine Learning (ML) network still relies on trial-and-error and does not guarantee optimal reduction of computations. In this paper, an Optimal Machine Learning-based Multiple Impedance Control (OML-MIC) is presented, in which the explicit computation related to the grasp matrix is replaced with a nonlinear approximation based on a Radial Basis Function Neural Network (RBFNN), and the network architecture is optimized using a Genetic Algorithm (GA) to minimize computational cost under accuracy constraints. The proposed method systematically determines the optimal network structure and, while preserving the physical dynamics of the grasp, eliminates the need for heavy linear-algebra operations. Simulation results for planar manipulation of an object by a dual-arm SFFR show that, while meeting the MIC control specifications, OML-MIC reduces the number of multiplication operators by 73.15%, the number of addition operators by 41.27%, and the hidden-layer computations of ML-MIC by 50%. As a result of the structured optimal design, error analysis with statistical quantitative metrics confirms that accuracy remains within the safe-interaction range. These results indicate that integrating architecture optimization with learning-based control provides a reliable path to precise, real-time interaction on robotic platforms with limited resources.

* Corresponding address: School of Engineering, Design and Built Environment, Western Sydney University, Sydney, Australia.
Tel.: +61 432 422 104
E-mail address: payam.zarafshan@gmail.com.

1. Introduction

Over the past two decades, robot control in interaction with the environment has become a prominent research topic, [1]. The earliest studies in this area were motivated by industrial demands to enhance speed and accuracy in manufacturing and assembly processes, which necessitated the integration of robotic systems. The fundamental challenge was that a dexterous robotic manipulator, when subjected to environmental resistance, had to maintain precise following of the desired path, [2]. In modern industrial robots, in addition to position control for following a preplanned reference path, a force-control action is required at the end effector to enable compliant interaction with diverse task environments. For example, when the end effector is equipped with a welding tool, the prescribed path can be followed using a baseline path-following controller without explicitly modeling the welding process. However, improving the end-effector task quality is challenging, because the contact force between the tool and the workpieces may vary with the part geometry, thereby degrading path-following performance. Consequently, the robot must achieve simultaneous path following and force regulation (i.e., hybrid position/force control), [3]. To achieve simultaneous path following and force control, *Impedance Control* (IC) was introduced [4]. The fundamental principles of IC are grounded in a careful analysis of human arm maneuvers and their generalization to the robotic arm. Accordingly, significant results were obtained in applying impedance-based concepts to the control of robots during contact with the environment, [5- 9].

From a physiological viewpoint, the human neuromuscular system configures the fingers according to the object's geometry and stiffness to regulate the object's manipulation; this finger configuration is effectively equivalent to establishing a specific impedance to ensure safe interaction. IC is one of the simplest control strategies that humans perform automatically in various forms; however, achieving the same behavior with robots requires complex algorithms. Therefore, a fundamental challenge in implementing IC is its computational complexity, [10].

From a mechanical viewpoint, impedance pertains to the magnitude of translational mechanical work during contact between two objects; specifically, the magnitude of this work has a direct relationship with force and displacement. Consequently, in problems involving the interaction (or impact) of two objects where neither the contact forces nor the magnitude of displacement can be neglected, the transfer of mechanical energy must be addressed on the basis of interaction dynamics. Therefore, impedance can be regarded as a dynamic operator that determines the output force as a function of the input displacement, [11].

From a control viewpoint, impedance denotes the dynamic relationship between position and force variables, and IC regulates the system's dynamic behavior; in its generalized form, it establishes a dynamic equation that couples the position and force errors. In simple terms, IC provides a tunable specification of how the robot should interact with the environment, enabling safer and more delicate contact, [12]. In essence, position and force controllers pursue similar objectives but each focuses on a single variable: ideal position control corresponds to infinite impedance, where the robot attains the desired position regardless of the applied force; ideal force control corresponds to zero impedance, where the robot exerts the desired force irrespective of its position.

From a mathematical viewpoint, for *Linear Time-Invariant* (LTI) systems, impedance is expressed by an 2^{nd} order linear differential equation describing a mass–spring–damper mechanical system, [13]. In practice, however, the equations governing the system dynamics and the relations imposed by an impedance controller are nonlinear, and designing and deploying nonlinear controllers entails substantial computational and implementation cost.

New robotic applications across diverse domains have motivated research on IC. Representative examples include the IC of lightweight manipulators employed in the aerospace industry, [14]. As these applications have expanded and the challenges of object manipulation control have grown, *Multiple Impedance Control* (MIC) was introduced—denoting the adaptation and regulation of several impedances at distinct parts of the robot, [15]. The substantial computational burden associated with MIC—which scales with the system's *Degrees of Freedom* (DoF)—increases computational cost and often necessitates more expensive controllers, [16]. A contemporary solution to reducing the computational burden of control is to leverage *Machine Learning* (ML) as a branch of *Artificial Intelligence* (AI), [17]. Using ML as a tool to mitigate control-related computational complexity can enhance overall system efficiency, [18]. In the context of IC, ML–based methods have primarily focused on improving robustness and stability under model uncertainty, [18–20]. Nevertheless, enhancing efficiency in MIC remains challenging due to its inherent computational complexity. To reduce the computational complexity of MIC, *Non-model-based Multiple Impedance Control* (NMIC) was introduced on *Space Free-Flying Robot* (SFFR), [16]. The objective of NMIC was to eliminate dependence on the system's dynamic model and to simplify the implementation of MIC on SFFR. By

explicitly removing the dynamic equations and adopting a *Modified Transpose Jacobian* (MTJ) approach, NMIC reduced the computational burden of MIC. Nevertheless, NMIC still required knowledge of physical parameters—such as inertia and stiffness matrices—along with contact-force estimates and error derivatives. In addition, to cancel the required actuation term, NMIC relied on data from the previous time step; this not only tied the controller's accuracy to prior conditions but also made it prone to instability in the presence of noise or impulsive disturbances. Consequently, successful deployment of NMIC depended on careful tuning of sensitivity parameters and the use of smoothing filters to attenuate noise under critical conditions (e.g., impacts or passage through path-change points), indicating a limitation in precise real-time control. These issues not only degraded accuracy under unstable conditions but also hindered generalization to more complex scenarios. To address these challenges, *Machine Learning-based Multiple Impedance Control* (ML-MIC) was introduced on SFFR, [21]. Unlike NMIC, the advantage of ML-MIC is that, to perform the control computations, there is no need to use previous-step data, sensitivity parameters, or smoothing filters, and the computations are generated online with high accuracy. ML-MIC employs ML based on *Radial Basis Function Neural Networks* (RBFNNs) as a nonlinear approximation of the object's non-square grasp matrix; as a result, the counts of multiplication operations decreased by up to 68.52% and of addition operations by up to 39.68%. Furthermore, taking the reduced computation into account, the computational accuracy was maintained at up to 98%.

Despite the considerable improvements achieved by ML-MIC, there is no guarantee of an optimal design of the learning-network architecture nor of an optimal reduction in computation. This work proposes a systematic, Genetic Algorithm (GA)–based method to achieve an optimal reduction of the computational effort of MIC via ML on SFFR. A GA—a search-based optimization technique grounded in the principles of genetics and natural selection—is designed to find optimal solutions for difficult, *Non-deterministic Polynomial-time hard* (NP-hard) problems. Accordingly, the proposed *Optimal ML-MIC* (OML-MIC) not only realizes an optimal reduction in MIC computation, but also synthesizes the ML architecture.

2. Materials and Method

Robotics presents unique challenges for learning algorithms. Grasping—due to factors such as diverse object shapes and states—is one of the most difficult tasks for robot autonomy in terms of perception, planning, and control. During robotic grasp execution, a large share of the computation arises from the grasp matrix, which directly appears in the equations of MIC, [22]. The high computational burden of MIC further adds to the complexity of grasping, whereas real-time execution is critical across a wide range of robotic applications. Unlike humans, who can readily infer how to grasp a particular object, robots still perform well below human standards, although performance is continuously improving due to high demand. Nevertheless, achieving these goals faces numerous challenges, and ML based on *Artificial Neural Networks* (ANNs) has strong potential to address them. Accordingly, in MIC, a particularly complex portion of the computation arises from computing the pseudo-inverse of the grasp matrix:

$$\mathbf{G}^{\#} = \mathbf{W}^{-1}\mathbf{G}^{\mathrm{T}}(\mathbf{G}\mathbf{W}^{-1}\mathbf{G}^{\mathrm{T}}) \qquad (1)$$

where $\mathbf{G}^{\#}$ is the pseudo-inverse of the grasp matrix $\mathbf{G}$, and $\mathbf{W}$ is the weighting matrix. The grasp matrix $\mathbf{G}$ is non-square; consequently, computing the inverse of $\mathbf{G}$, together with the other terms in this relation, makes the implementation of IC on real-time systems difficult and even impractical. The grasp matrix is a physical representation that indicates how the force applied at the contact point relates to the force applied at the object's center of mass, [23]. Therefore, each row of the grasp matrix corresponds to one equation within the set of governing relations between the robot and the object; any dimensionality reduction of the matrix, by any method, removes a physical concept from the equation and can cause the robot to lose the object. Figure 1 shows the object grasped by the SFFR.

One innovative solution to reduce the computational burden of computing the pseudo-inverse of the grasp matrix has been to employ ML based on RBFNNs, [21].

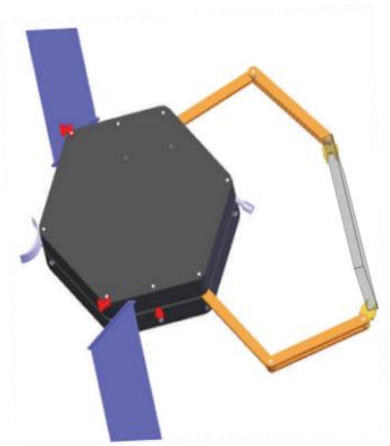

Figure 1. Object grasp by the SFFR with 2-DoF arms (manipulators).

In this approach, an RBFNN—composed of multiple radial basis kernels in its hidden layer(s), analogous to series expansions such as Taylor or Fourier—provides a nonlinear approximation to the entries of the $\mathbf{G}^{\#}$, without discarding the physical dimensionality of the **G**. The proposed ML-MIC method reduced the computational complexity of the MIC process, resulting in shorter execution time and improved MIC system performance. Nevertheless, no explicit rule currently exists for determining the number of hidden layers in an ANNs; this leads to structural suboptimality in the proposed approach and reliance on trial-and-error during ANN design. Moreover, because the designed ANN architecture is not optimal, there is no guarantee of achieving the maximum possible reduction in computational cost.

As illustrated in Figure 2, this study introduces a GA to provide a structured methodology for the optimal design of the ANN architecture aimed at reducing the computational burden of MIC, with emphasis on the pseudo-inverse of the object grasp matrix for the SFFR. In this way, the traditional trial-and-error process for designing the ANN that reduces MIC computations is replaced by a GA-based optimal design procedure, thereby providing a guarantee of convergence to the maximal computational reduction.

Figure 2. Overview of the OML-MIC Method.

3. Control Algorithms

3.1. Multiple Impedance Control Law

Impedance control (IC) is a unified control framework that regulates the dynamic relationship between motion variables and contact forces in a desired manner for manipulation. Accordingly, during unconstrained (free) motion it reduces to position control, whereas during constrained motion it functions as force control, [15]. In robotic manipulation systems, IC is formulated around robot–object interaction and is therefore inadequate for analyzing or regulating the object's manipulation during motion. By contrast, *Multiple Impedance Control* (MIC) explicitly addresses regulation of the object's manipulation by the robot during motion. The MIC law applied to the robot's motion is expressed as follows:

$$\tilde{\mathbf{M}}_{\text{des}}\ddot{\tilde{\mathbf{e}}} + \tilde{\mathbf{K}}_{\text{d}}\dot{\tilde{\mathbf{e}}} + \mathbf{K}_{\text{p}}\tilde{\mathbf{e}} = -\mathbf{F}_{\text{c}} \tag{2}$$

where $\tilde{\mathbf{e}} = \tilde{\mathbf{X}}_{\text{des}} - \tilde{\mathbf{X}}$ denotes the tracking error of the controlled variables, $\tilde{\mathbf{M}}_{\text{des}}$ is the desired mass matrix, $\tilde{\mathbf{K}}_{\text{d}}$ and $\tilde{\mathbf{K}}_{\text{p}}$ are the controller gain matrices, and $\mathbf{F}_{\text{c}}$ represents the contact forces. Furthermore, the MIC law applied to the object's motion is given by:

$$\mathbf{M}_{\text{des}}\ddot{\mathbf{e}} + \mathbf{K}_{\text{d}}\mathbf{e} + \mathbf{K}_{\text{p}}\mathbf{e} = -\mathbf{F}_{\text{c}} \tag{3}$$

where $\mathbf{e} = \mathbf{X}_{\text{des}} - \mathbf{X}$ denotes the object tracking error, $\mathbf{M}_{\text{des}}$, $\mathbf{K}_{\text{d}}$ and $\mathbf{K}_{\text{p}}$ are obtained analogously to $\tilde{\mathbf{M}}_{\text{des}}$, $\tilde{\mathbf{K}}_{\text{d}}$ and $\tilde{\mathbf{K}}_{\text{p}}$.

The concept of grasp in object manipulation is that an object's motion is constrained by applying forces and torques at a set of contact points. Assuming a specific contact model, a grasp can be evaluated by examining the grasp wrench space $\mathcal{W}$, which indicates how the grasp can affect the object through forces and torques. A wrench $\mathcal{W}$ is a vector comprising force $\mathbf{F} \in \mathcal{R}^3$ and torque $\boldsymbol{\tau} \in \mathcal{R}^3$ components applied at the object's center of mass:

$$\mathcal{W} = \begin{bmatrix} \mathbf{F} \\ \boldsymbol{\tau} \end{bmatrix} \in \mathcal{R}^6 \tag{4}$$

any contact force $\mathbf{F}_{\text{i}}$ applied at a point located a distance $\mathbf{r}_{\text{i}}$ from the object's center induces a torque $\boldsymbol{\tau}_{\text{i}} = \mathbf{r}_{\text{i}} \times \mathbf{F}_{\text{i}}$, where $\mathbf{r}_{\text{i}}$ is the vector defining the position of the i^{th} contact point with respect to the object's center of mass ($\| \mathbf{r}_{\text{i}} \| = \text{d}_{\text{i}}$). Therefore, the wrench $\mathcal{W}_{\text{i}}$ can be written as:

$$\mathcal{W}_{\text{i}} = \begin{bmatrix} \mathbf{F}_{\text{i}} \\ \lambda(\mathbf{r}_{\text{i}} \times \mathbf{F}_{\text{i}}) \end{bmatrix} \tag{5}$$

where the constant $\lambda \in \mathcal{R}$ is arbitrary and can be used to scale the torque magnitude. Leveraging the mathematical definition of a wrench $\mathcal{W}$, a grasp specifies how a contact force $\mathbf{F}_{\text{i}}$ maps to the resultant wrench $\mathcal{W}_{\text{i}}$ at the object's center of mass. The grasp matrix (grasp map) **G** provides a linear mapping from each contact force $\mathbf{F}_{\text{i}}$ to the corresponding wrench $\mathcal{W}_{\text{i}}$ at the center of mass.

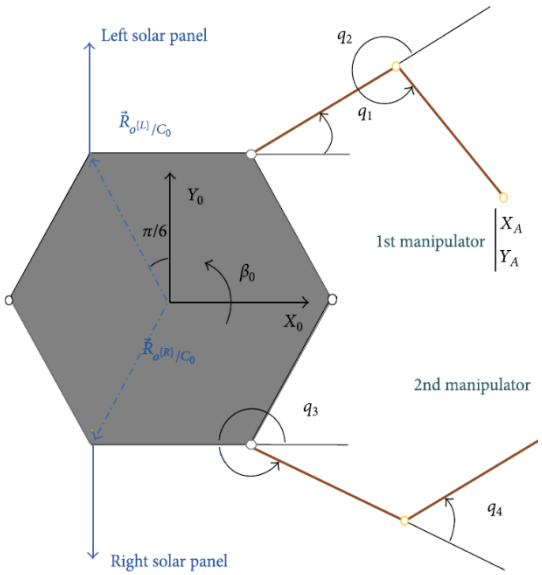

Figure 3. Top view of the SFFR' free-body diagram.

Consequently, the net wrench exerted on the object by all contacts is:

$$\boldsymbol{w} = \sum_{i=1}^{k} \mathbf{G}_i \mathbf{F}_i = \mathbf{G} \begin{bmatrix} \mathbf{F}_1 \\ \vdots \\ \mathbf{F}_k \end{bmatrix} \tag{6}$$

Based on Figure 3, an object of length L is grasped at two revolute-joint contact points and manipulated by the SFFR's 2-DoF manipulators; the object's equations of motion, expressed with respect to the SFFR base frame, are given by:

$$\begin{cases} F_{X_A} + F_{X_B} = m\ddot{X}_o \\ F_{Y_A} + F_{Y_B} = m\ddot{Y}_o \\ \underbrace{\dfrac{(F_{X_B} - F_{X_A})L}{2\,\mathrm{Cos}\,\theta_o}}_{C_{\theta_o}} + \underbrace{\dfrac{(F_{Y_B} - F_{Y_A})L}{2\,\mathrm{Sin}\,\theta_o}}_{S_{\theta_o}} = I\ddot{\theta}_o \end{cases} \tag{7}$$

where (F_{X_A}, F_{Y_A}) are the forces applied to the object by the left manipulator's end-effector, (F_{X_B}, F_{Y_B}) are the coordinates of the right manipulator's end-effector, and θ_o denotes the object orientation. Finally, the grasp matrix **G** for a unit-length object, expressed in the robot's task space, is given by:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ -\dfrac{1}{2C_{\theta_o}} & -\dfrac{1}{2S_{\theta_o}} & \dfrac{1}{2C_{\theta_o}} & \dfrac{1}{2S_{\theta_o}} \end{bmatrix} \tag{8}$$

In MIC, the control forces for object grasping and manipulation —supplied by the manipulators —can be computed using the following relation:

$$\tilde{\mathbf{Q}}_{\text{generalized}} = \tilde{\mathbf{Q}}_{\text{m}} + \tilde{\mathbf{Q}}_{\text{f}} + \tilde{\mathbf{Q}}_{\text{react}} \tag{9}$$

The generalized control-force vector $\tilde{\mathbf{Q}}_{\text{generalized}}$ comprises three fundamental terms. Assuming rigid robot parts and neglecting nonlinear effects, it is computed as follows:

$$\tilde{\mathbf{Q}}_{\text{m}} = \tilde{\mathbf{H}}\tilde{\mathbf{M}}_{\text{des}}^{-1}[(\tilde{\mathbf{M}}_{\text{des}}\ddot{\tilde{\mathbf{X}}}_{\text{des}} + \tilde{\mathbf{K}}_{\text{d}}\dot{\tilde{\mathbf{e}}} + \mathbf{K}_{\text{p}}\tilde{\mathbf{e}}) + \mathbf{F}_{\text{c}}] \tag{10}$$

$$\tilde{\mathbf{Q}}_{\text{f}} = \mathbf{G}^{\#}\left[\mathbf{M}\mathbf{M}_{\text{des}}^{-1}(\mathbf{M}_{\text{des}}\ddot{\mathbf{X}}_{\text{des}} + \mathbf{K}_{\text{d}}\mathbf{e} + \mathbf{K}_{\text{p}}\mathbf{e}) + \mathbf{F}_{\text{c}}\right] \tag{11}$$

$$\tilde{\mathbf{Q}}_{\text{react}} = -\mathbf{G}^{\#}\left[(\mathbf{M}\ddot{\mathbf{X}} - \mathbf{F}_{\text{c}})\right] \tag{12}$$

In Equation (10), $\tilde{\mathbf{H}}$ denotes the positive-definite mass matrix, and $\mathbf{G}^{\#}$ denotes the pseudo-inverse of the grasp matrix. To mitigate the computational burden of evaluating $\mathbf{G}^{\#}$, a RBFNN–based ML surrogate is introduced that approximates $\mathbf{G}^{\#}$ directly, thereby circumventing explicit matrix inversion and achieving substantially lower arithmetic complexity while maintaining high numerical accuracy, [21].

3.2. Machine Learning-based Multiple Impedance Control:

In recent years, *Machine Learning* (ML) has transformed advanced robotics. ML has emerged as an effective strategy for reducing computational burden, increasing computational speed, and improving the performance of robotic control systems, [24]. Among ML methods, *Radial Basis Function Neural Networks* (RBFNNs) are widely used because of their high speed in function approximation, [25]. Formally, an RBFNN maps a p-dimensional input to a k-dimensional output via j kernels:

$$\hat{\mathbf{y}}_{\text{k}} = \sum_{\text{j=1}}^{\text{k}} \boldsymbol{\Phi}_{\text{pj}}\mathbf{w}_{\text{jk}} + \mathbf{b}_{\text{k}} \tag{13}$$

where $\boldsymbol{\Phi}\cdot\mathbf{w}$ is the weighted sum of kernel functions, and **b** is the bias (intercept) of the function approximator. In RBFNNs, the kernels are basis functions (always invertible) and radial (depend on the distance from a central point c). According to Micchelli's theorem, the most widely used radial basis function φ is the Gaussian:

$$\varphi_{\text{j}}(\theta_{\text{o}}) = \exp\left(-\frac{\left\|\theta_{\text{o}} - \text{c}_{\text{j}}\right\|^2}{2\sigma_{\text{j}}^2}\right) \tag{14}$$

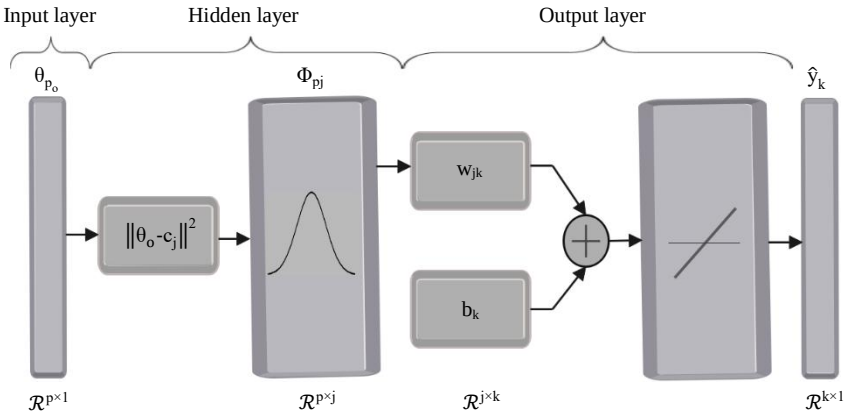

Figure 4. Architecture of the designed RBFNN.

According to Figure 4, by selecting the object orientations $\boldsymbol{\theta}_{p_o}$ as the input pattern vector, the approximation of the $\mathbf{G}^{\#}$ as the output vector $\hat{\mathbf{y}}_k$; accordingly, rewriting Equations (11) and (12) yields:

$$\tilde{\mathbf{Q}}_f = \hat{\mathbf{G}}^{\#}\left[\mathbf{M}\mathbf{M}_{des}^{-1}(\mathbf{M}_{des}\ddot{\mathbf{X}}_{des} + \mathbf{K}_d\dot{\mathbf{e}} + \mathbf{K}_p\mathbf{e}) + \mathbf{F}_c\right]$$
$$\tilde{\mathbf{Q}}_{react} = -\hat{\mathbf{G}}^{\#}\left[(\mathbf{M}\ddot{\mathbf{X}} - \mathbf{F}_c)\right] \quad (15)$$

According to Figure 4, in the designed network, for each input pattern vector representing the object orientation ($p = 1$), a vector formed by two Gaussian basis functions ($j = 2$) is mapped to an output vector containing the 12 elements of the $\mathbf{G}^{\#}$ ($k = 12$).

The most commonly used linear-algebra method for computing a matrix pseudo-inverse in numerical packages such as MATLAB is the *Singular Value Decomposition* (SVD), which factors a matrix into three matrices, [21]:

$$\mathbf{G} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{T} \quad (16)$$

where $\mathbf{G} \in \mathcal{R}^{m\times n}$, $\mathbf{U} \in \mathcal{R}^{m\times m}$ and $\mathbf{V} \in \mathcal{R}^{n\times n}$ are orthogonal matrices, and $\boldsymbol{\Sigma} \in \mathcal{R}^{m\times n}$ is diagonal with non-negative singular values. The $\mathbf{G}^{\#}$ is then defined as:

$$\mathbf{G}^{\#} = \mathbf{V}\boldsymbol{\Sigma}^{\#}\mathbf{U}^{T} \quad (17)$$

where $\boldsymbol{\Sigma}^{\#}$ is the diagonal matrix obtained by inverting the nonzero singular values of $\boldsymbol{\Sigma}$.

For a comparison between RBFNN-based ML-MIC and SVD-based MIC, Table 1 is provided, [20]. Using the RBFNN architecture, the computational burden of evaluating the $\mathbf{G}^{\#}$ for the SFFR was substantially reduced. This reduction was achieved while maintaining high accuracy.

Table 1. Comparison of MIC and ML-MIC methods.

Control types	ML-MIC		MIC
Computational methods	RBFNN		SVD
Layers	Hidden	Output	—
Number of additions (±)	2	36	63
Number of multiplications (×)	10	24	108
Reduction in additions %	39.68 %		—
Reduction in multiplications %	68.52 %		—
CPU Run Time [s]	0.012		0.082
MSE	0.004		0

Nevertheless, ML-MIC does not provide a method to attain the maximum computational reduction nor an optimal network architecture design.

3.3. Optimal Machine Learning-based Multiple Impedance Control:

In current practice, the design of *Artificial Neural Networks* (ANNs) depends heavily on the human designer's expertise. The designer must have sufficient knowledge of network configurations and of the problem's computational domain. As domain complexity increases, manual design becomes increasingly difficult and ultimately unmanageable. For *Radial Basis Function Neural Networks* (RBFNNs), selecting the optimal number of kernels for mapping the input vector to the output vector remains a challenging problem. No explicit rule exists, and the choice is inherently problem-dependent, placing the task in the class of *Non-deterministic Polynomial-time hard* (NP-hard) problems. Consequently, trial-and-error search is time-consuming, lacks guarantees of convergence and generalization, and does not constitute a systematic design methodology. To address such structural optimization problems, nature-inspired metaheuristic methods are commonly employed, [26].

Over billions of years, nature has operated as a vast self-organizing system**,** continually engaging in experimentation, trial-and-error, and optimization. The organisms present in today's

ecosystems are the outcomes of natural selection, which has progressively eliminated weaker structures and strategies while preserving the most suitable ones. This large-scale evolutionary process—without centralized control or complete knowledge of the environment—has produced solutions that are efficient, robust, and adaptable to complex problems. Among nature-inspired metaheuristics**,** the *Genetic Algorithm* (GA) holds a prominent place. GA maintains and gradually evolves a population of candidate solutions**,** employing mechanisms such as selection, crossover, mutation, and the progressive elimination of weak candidates to approach optimal solutions over time. Unlike linear-algebraic methods, GA evaluate a population rather than a single candidate and, through repeated *selection–crossover–mutation* cycles, progressively synthesize improved solutions.

The critical step in optimization is problem formulation and objective-function selection. In this study, the objective is to optimally design the architecture of a RBFNN that maps input vector $\boldsymbol{\theta}_{p_o}$ to the output vector $\mathbf{G}^{\#}$, while minimizing computational cost (cost function). To this end, a GA is employed as the optimizer to determine the optimal number of kernels in the RBFNN; its procedure is as follows:

• *Initialization*:

A population $\boldsymbol{\mathcal{P}}$ of candidate RBFNN architectures is created by sampling the number of radial kernels j from the admissible discrete set:

$$\boldsymbol{\mathcal{P}}^{(0)} = \left\{ j_1^{(0)}, \ldots, j_{N_p}^{(0)} \right\}; \quad j_i^{(0)} \in \Omega_j \tag{18}$$

Each individual encodes a single architecture size j drawn on discrete uniform Ω_j.

• *Evaluation*:

For every candidate j in the current population, the RBFNN is trained on the training subset and assessed on the validation subset against the target $\mathbf{G}^{\#}$. Performance is evaluated by the validation *Root Mean Square Error* (RMSE) $\mathcal{J}$:

$$\mathcal{J}^{(j)} = \sqrt{\frac{1}{N_p} \sum_{i=1}^{N_p} \left\| \hat{\mathbf{G}}^{\#} - \mathbf{G}^{\#} \right\|^2} \tag{19}$$

• *Selection*:

Individuals are ranked by ascending $\mathcal{J}^{(j)}$ and selected with rank-based probabilities, so that architectures with lower $\mathcal{J}^{(j)}$ have a higher chance of reproduction.

• *Crossover*:

Pairs of selected parents $j^A, j^B \in \Omega_j$ are recombined to explore intermediate architectures:

$$j^C = \frac{j^A + j^B}{2} \tag{20}$$

where j^C is the offspring produced by recombining the selected parents.

• *Mutation*:

To preserve diversity and avoid premature convergence, each j^C undergoes a unit integer perturbation with small probability Δ:

$$j^M = j^C + \Delta \tag{21}$$

where j^M results from a genetic mutation.

• *Termination Criteria*:

As is standard in control literature, the termination criterion is satisfied when the network achieves an output RMSE < 0.1 with the minimal possible number of kernels j. As illustrates in Figure 5, the selected architecture is given by:

$$\arg\min \mathcal{J}^{(j)} = j^* \tag{22}$$

Based on Table 2, Using the proposed OML-MIC, the maximum computational reduction is achieved because the GA identifies one hidden-layer kernel (j = 1), whereas ML-MIC used two hidden-layer kernel (j = 2). In OML-MIC, an explicit trade-off between computational reduction and numerical accuracy is enforced; Figures 6.1 to 6.3 report a MSE of 0.005 for the training, validation, and test phases, respectively.

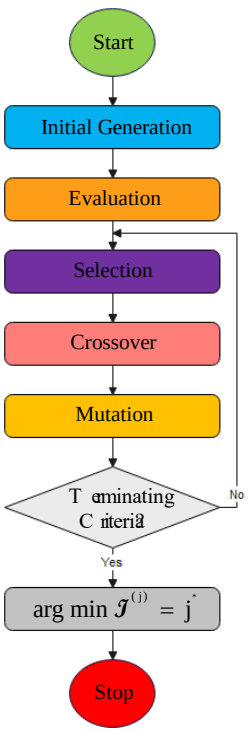

Figure 5. Flowchart presentation of GA.

Table 2. Comparison among OML-MIC, ML-MIC, and MIC.

Control types	OML-MIC		ML-MIC		MIC
Computational methods	RBFNN		RBFNN		SVD
Layers	Hidden	Output	Hidden	Output	—
Number of additions (±)	1	36	2	36	63
Number of multiplications (×)	5	24	10	24	108
Reduction in additions %	41.27 %		39.68 %		—
Reduction in multiplications %	73.15 %		68.52 %		—
CPU Run Time [s]	0.008		0.012		0.082
MSE	0.005		0.004		0

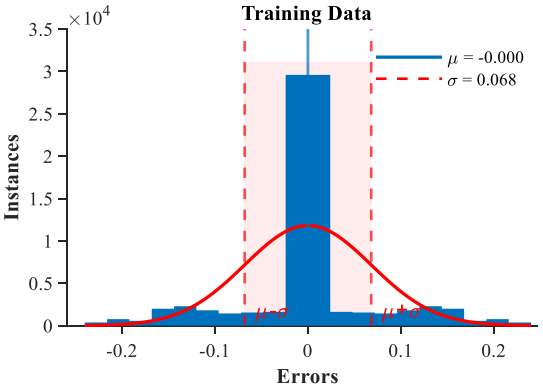

Figure 6.1. Histogram error for training data.

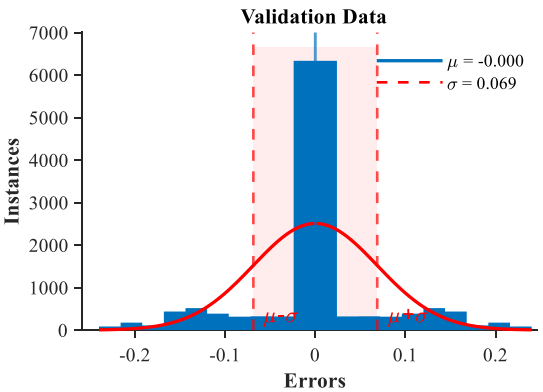

Figure 6.2. Histogram error for validation data.

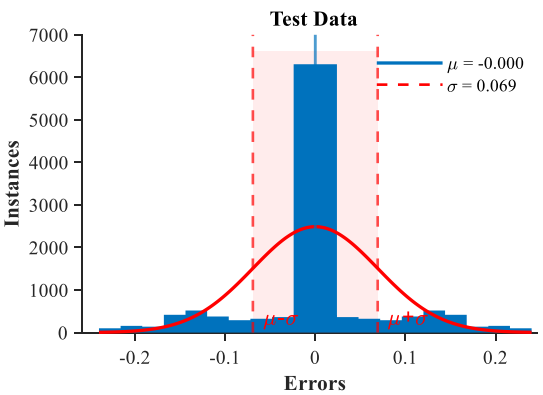

Figure 6.3. Histogram error for test data.

4. Results and Discussions

A comparative evaluation of MIC and OML-MIC is conducted using simulations of planar object manipulation by a 2-DoF SFFR. The SFFR has a total mass of 20 kg with its center of mass located at the base-frame origin, and it manipulates a 2 kg object while tracking a unit-ramp reference in the Cartesian plane. The commanded ramp profiles for both the SFFR motion and the manipulated object trajectory along the X- and Y-axes are shown in Figure 7. Initial conditions—comprising positions and velocities in the relevant coordinates—together with all controller gains are reported in Tables 3 and 4 for the SFFR and the object, respectively. These parameters are held fixed across controllers to attribute any observed performance differences solely to the control laws.

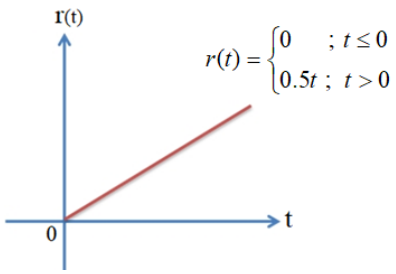

Figure 7. Desired trajectory for both SFFR and object.

Table 3. Initial Conditions; SFFR.

Robot	Initial conditions				Controller gains		
Parameters	Position (m)		Velocity (m/s)		$\tilde{M}_{des}$ (kg)	$\tilde{K}_d$ ($\frac{N}{m}$)	$\tilde{K}_p$ (N/m)
Directions	X	Y	X	Y			
Values	0	0	0	0	20	5	60

Table 4. Initial Conditions; Object.

Object	Initial conditions				Controller gains		
Parameters	Position (m)		Velocity (m/s)		M_{des} (kg)	K_d ($\frac{N}{m}$)	K_p ($\frac{N}{m}$)
Directions	X	Y	X	Y			
Values	$\frac{1}{4}$	$\frac{\sqrt{3}}{4}$	0	0	2	1	30

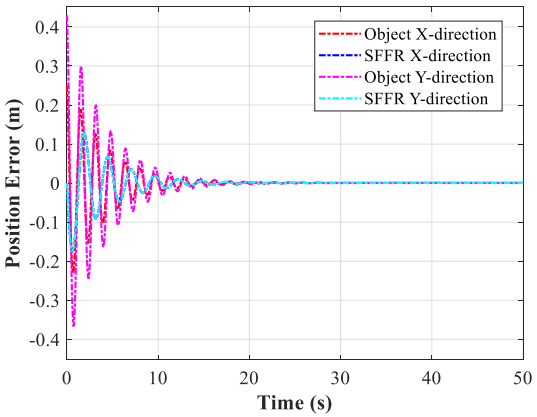

Figure 8.1. Position control; MIC.

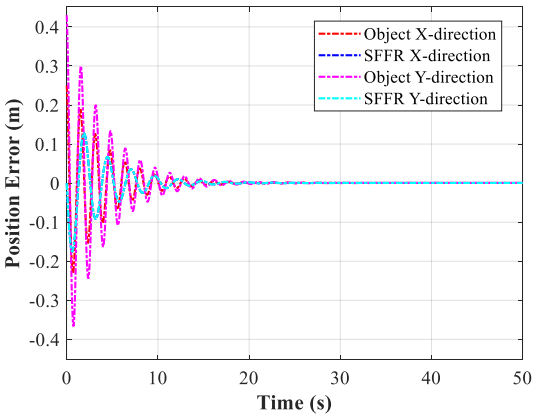

Figure 9.1. Position control; OML-MIC.

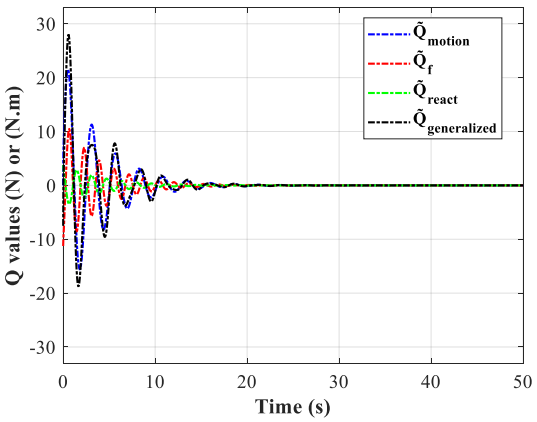

Figure 8.2. Force control (*x*-direction); MIC.

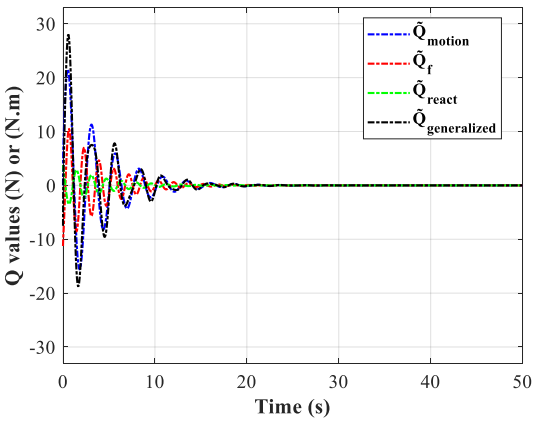

Figure 9.2. Force control (*x*-direction); OML-MIC.

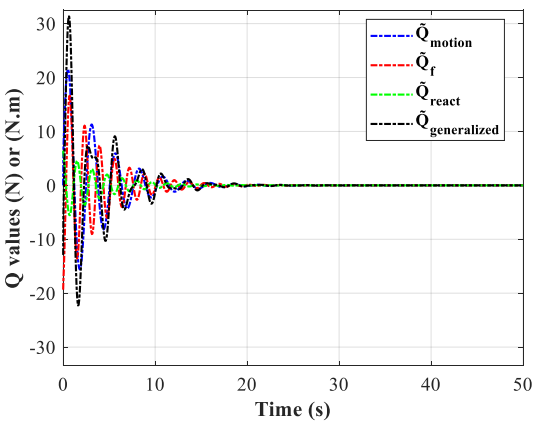

Figure 8.3. Force control (*y*-direction); MIC.

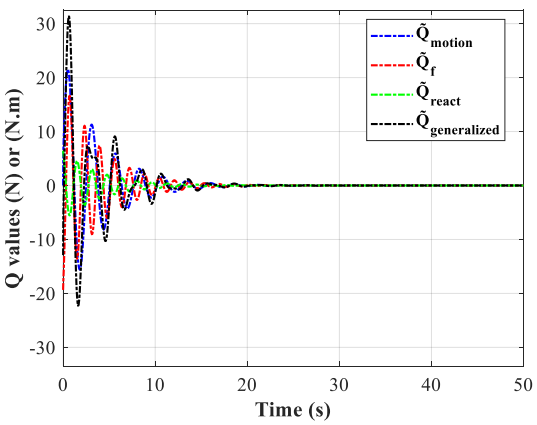

Figure 9.3. Force control (*y*-direction); OML-MIC.

As the baseline, the MIC law is applied to simultaneously regulate task-space position and interaction force in the coupled SFFR–object system, and the responses are reported in Figures 8.1 to 8.3. These results serve as the reference for subsequent analyses and enable a one-to-one comparison with OML-MIC under identical operating conditions, so that any improvement in tracking accuracy or force-damping quality can be attributed directly to the learning architecture and reduced computational complexity.

The principal benefit of the IC framework lies in its unified treatment of task-space motion regulation and interaction-force modulation for the coupled SFFR–object system. Within this scheme, both the SFFR trajectories and the object motion are coordinated while contact forces are shaped to maintain a stable grasp. At closed-loop equilibrium, the effective interaction forces at the SFFR–object interface and the forces applied to the object are balanced such that the desired position and velocity are attained by both subsystems.

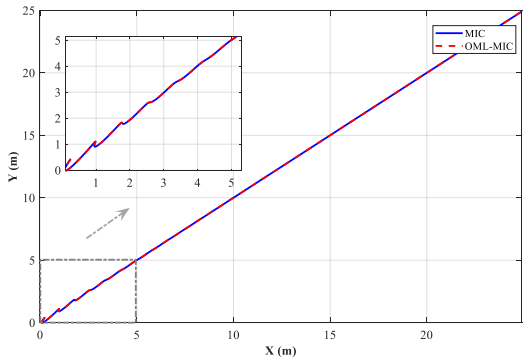

Figure 10.1. SFFR control along the path; MIC and OML-MIC.

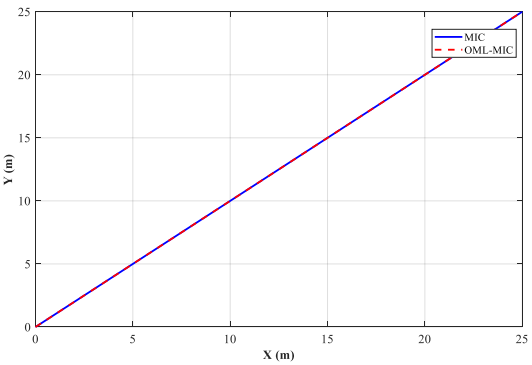

Figure 10.2. Object control along the path; MIC and OML-MIC.

Consequently, convergence of the force control toward zero must not be interpreted as the absence of forces. Rather, it indicates that all effective forces have reached a steady balance and the object resides in a stable state; therefore, no additional control effort is required, and the SFFR's grasp remains secured.

Building on this baseline, the OML-MIC strategy is employed to examine position and force regulation of the system; the corresponding responses are reported in Figures. 9.1 through 9.3. Under identical operating conditions, OML-MIC likewise satisfies the control specifications, exhibiting tracking and interaction behaviors comparable to those observed with MIC. For a side-by-side assessment, Figures 10.1 and 10.2 depict the object and SFFR evolution along their paths, enabling a direct comparison in terms of tracking accuracy, steady-state offsets, and contact-force regulation quality.

5. Conclusions

This study introduces *Optimal Machine Learning-based Multiple Impedance Control* (OML-MIC) as a novel, *Genetic Algorithm* (GA)–optimized *Machine Learning* (ML) approach for computing the pseudo-inverse of the grasp matrix within the framework of *Multiple Impedance Control* (MIC) for a *Space Free-Flying Robot* (SFFR). By targeting a principal computational bottleneck, OML-MIC replaces explicit matrix inversion with a *Radial Basis Function Neural Network* (RBFNN)-based nonlinear surrogate and optimizes the learning architecture via GA so as to minimize arithmetic cost without degrading control quality. With an MIC baseline established under identical operating conditions, OML-MIC satisfied the MIC control specifications while achieving maximal computational reduction and high numerical accuracy. Quantitatively, OML-MIC reduced multiplications by 73.15% and additions by 41.27%, and shortened CPU run time from 0.082 *s* to 0.008 *s*, while maintaining MSE ≈ 0.005 across training, validation, and test phases. From a design standpoint, the GA selected a single hidden-layer kernel (j = 1) for OML-MIC (versus j = 2 for ML-MIC), yielding a clear improvement over ML-MIC. These results demonstrate that structured architecture search at the controller–learner interface can deliver substantial efficiency gains without compromising closed-loop accuracy.

In conclusion, OML-MIC not only achieves the maximum computational reduction for MIC, but also furnishes a systematic methodology for the optimal design of the learning network. This provides a reliable pathway to precise, real-time interaction control on resource-constrained robotic platforms.

References

[1] M. Baniardalani, A. Mansouri, A. Safdari 'External Robotic System for Automating Manual Gear Shifting in Manual Transmission Cars', in *11th International Conference on Control, Instrumentation and Automation (ICCIA)*, IEEE, in press.

[2] A. Rezaee, A. Safdari, and M. J. Fazli, 'Automatic Path Planning for GMAW Welding on Locomotive Bogie in Low Contrast by Welding Robot', *International Journal of Robotics; Theory and Applications*, vol. 10, pp. 22–31, 2024.

[3] C. Jiao, 'Adaptive hybrid impedance control for dual-arm cooperative manipulation with object uncertainties', *Automatica*, vol. 140, 2022.

[4] N. Hogan, 'Impedance control: An approach to manipulation', *American control conference*, 1984.

[5] E. Spyrakos-Papastavridis and J. S. Dai, 'Minimally model-based trajectory tracking and variable impedance control of flexible-joint robots', *IEEE Transactions on Industrial Electronics*, vol. 68, pp. 6031–6041, 2020.

[6] T. Kim *et al*., "Design and force-tracking impedance control of 2-DOF wall-cleaning manipulator via disturbance observer", *IEEE ASME Trans. Mechatron.*, vol. 25, no. 3, pp. 1487–1498, Jun. 2020.

[7] L. Roveda *et al*., "Optimal impedance force-tracking control design with impact formulation for interaction tasks", *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 130–136, Jan. 2016.

[8] L. Roveda and D. Piga, "Robust state dependent Riccati equation variable impedance control for robotic force-tracking tasks", *Int. J. Intell. Robot. Appl.*, vol. 4, no. 4, pp. 507–519, Dec. 2020.

[9] L. Gu and Q. Huang, "Adaptive Impedance Control for Force Tracking in Manipulators Based on Fractional-Order PID", *Applied Sciences*, vol. 13, 2023.

[10] B. Kim and S. Park, 'Impedance Control of Robot Manipulator in Contact Task Using Machine Learning', in *SICE-ICASE International Joint Conference*, IEEE, 2006.

[11] D. Rancourt and N. Hogan, 'Stability in force-production tasks', *Journal of motor behavior*, vol. 33, pp. 193–204, 2001.

[12] X. Zhang *et al*., 'Learning variable impedance control via inverse reinforcement learning for force-related tasks', *IEEE Robotics and Automation Letters*, vol. 6, pp. 2225–2232, 2021.

[13] L. Roveda, 'Adaptive interaction controller for compliant robot base applications', *IEEE Access*, vol. 7, pp. 6553–6561, 2018.

[14] P. Zarafshan, Modeling and Simulation of Dynamical Systems. *Elsevier*, 2024.

[15] S. A. A. Moosavian and E. Papadopoulos, "Cooperative object manipulation with contact impact using multiple impedance control." *International Journal of Control, Automation and Systems* 8 (2010): 314-327.

[16] S. A. A. Moosavian, R. Rastegari, and H. R. Ashtiani, 'Non-model-based multiple impedance control of space free flying robots', in *2007 IEEE International Conference on Control Applications*, IEEE, 2007, pp. 533–538.

[17] G. Beni, 'Swarm intelligence', in *Complex Social and Behavioral Systems: Game Theory and Agent-Based Models*, 2020, pp. 791–818.

[18] R. Martín-Martín *et al*., 'Variable impedance control in end-effector space: An action space for reinforcement learning in contact-rich tasks', in *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, IEEE, 2019, pp. 1010–1017.

[19] H. An *et al*., 'Neural Adaptive impedance control for force tracking in uncertain environment', *Electronics*, vol. 12, no. 3, 2023.

[20] M. H. Hamedani *et al*., 'Intelligent impedance control using wavelet neural network for dynamic contact force tracking in unknown varying environments', *Control Engineering Practice*, vol. 113, 2021.

[21] A. Safdari *et al*., "Machine Learning based Multiple Impedance Control of a Space Free-Flying Robot." *2024 12th RSI International Conference on Robotics and Mechatronics (ICRoM)*. IEEE.

[22] D. Prattichizzo, JC. Trinkle, "Grasping." *Springer handbook of robotics* (2016): 955-988.

[23] S. Sanghavi, J. Ryu, and A. Sabharwal, "Robotic Manipulation and Interaction." Spring 2020.

[24] M. Soori, B. Arezoo, and R. Dastres, 'Artificial intelligence, machine learning and deep learning in advanced robotics, a review', *Cognitive Robotics*, 2023.

[25] G. Montazer *et al*., "Radial basis function neural networks: A review." *Comput. Rev. J* 1.1 (2018): 52-74.

[26] A. Safdari *et al*., 'PID Controller Design for a Mechatronic System using Lion Optimization Algorithm', *International Conference on Applied Research in Electrical Engineering*, 2025.

Biography

Amirhossein Safdari, received his M.Sc. degree as the 1st ranked graduate in Mechatronics Engineering from the University of Tehran, Iran, in 2024. He received his B.Sc. degree with 2nd rank in Mechanical Engineering from Razi University, Iran, in 2022. He is a recognized member of the National Elites Foundation (NEF) and currently serves as an instructor of Artificial Intelligence (AI) at the Scientific Association of the University of Tehran. He has also contributed as a consultant in the domains of smart technologies and urban innovation. His research interests include the application of AI in Control and Robotics.

Payam Zarafshan, is currently an Associate Lecturer at Western Sydney University. He holds the position of an Associate Professor of Mechanical Engineering at the Department of Mechatronics Engineering, Faculty of New Sciences and Technologies, University of Tehran. He received his Ph.D. in Mechanical Engineering from K. N. Toosi University of Technology, Iran. His research has mainly focused on Aerial and Space systems and Agriculture technologies through the application of Automation, Robotics, and Mechatronics. He is currently the founder and supervisor of the AGRicultural INtelligent System Laboratory (AGRINS Lab.) based at the University of Tehran. He has supervised over 28 M.Sc. students and 3 Ph.D. students in different mechatronic fields, including Space Robotic Systems, Irrigation Systems, Pruning Cutters, Water Channels Dredgers, Aerial Pollinators, Cable Robots, and Greenhouse Technologies.

Khalil Alipour, received his B.Sc., M.Sc., and Ph.D. degrees all with honors in Mechanical Engineering from K. N. Toosi University of Technology in 2002, 2004, and 2010, respectively. Currently, he is an associate professor with the Faculty of New Sciences and Technologies at University of Tehran. His research interests are in the areas of Dynamics Modeling, Automatic Control, Motion/Path Planning, and tip-over Stability Analysis of Mobile Robots.

Bahram Tarvirdizadeh, received his Ph.D. degree (in 2012) in Mechanical engineering from the University of Tehran, the symbol of higher education of Iran and the oldest modern university located in Tehran, Iran. Currently, he is an associate professor with the Faculty of New Sciences and Technologies at the University of Tehran. His main research interests include Mechatronics, Robotics, Automation, and Medical and Rehabilitation Mechatronics.

Gu Fang, (Member, IEEE) received the B.E. degree in mechanical automation from the Shanghai University of Technology, China, and the Ph.D. degree from The University of Sydney, Australia. He is currently a Professor of mechatronic engineering with Western Sydney University. He has co-edited two research books. He has published many refereed papers in book chapters, international journals, and conferences. His current research interests include Robotics, Computer Vision, Control Systems, and Artificial Intelligence (AI). He had many research grants from the Australian Research Council (ARC), the National Natural Science Foundation of China, and from industries. He had a Visiting Scholar appointments in UTS, Australia, and Shanghai Jiao Tong University, China. He is a regular referee for various international journals and conferences. He is a member of Engineers Australia