Vol. 08, No. 3 (2026) 1571-1590, doi: 10.24874/PES08.03.014

Proceedings on Engineering Sciences

www.pesjournal.net

MULTI-SENSOR FUSION AND DEEP EVOLUTIONARY OPTIMISATION: IMPROVED EMBODIED COGNITION AND MANIPULATION ANALYSIS IN HUMANOID ROBOTS

Hayder M. Ali
S. Bairavel
L. Arokia Jesu Prabhu
Anusha Papasani
A. N. V. K. Swarupa
T. Thirumalaikumari
Bharathi Ramesh Kumar
Sudhakar Sengan[1]

Received 12.05.2026.
Revised 30.06.2026.
Accepted 03.08.2026.

Keywords:

Embodied cognition, Humanoid robotics, adaptive manipulation, Multi-sensor fusion, Cognitive networks, Deep evolutionary optimization

A B S T R A C T

Humanoid Robots (HRs) face challenges integrating sensorimotor data due to inadequate sensor fusion methods and suboptimal control models. The research results of the current study should inform the implementation of a computational model that optimises and enhances the success rate of embodied cognition and information manipulation. The machine learning algorithm incorporates Hierarchical Multi-Sensor Fusion (MSF) and Deep Evolutionary Optimization (DEO), both introduced as computational methods into the learning process. Using attention-weighted neural network (NN) models, the machine learning (ML) algorithm integrates visual, tactile, proprioceptive, and force feedback information throughout its model. To achieve the objective of the evolution process, artificial intelligence NNs evolve using Genetic Algorithms (GAs) that integrate multi-objective fitness functions. By implementing focused algorithms specifically designed to build a cognitive network (CN), the evolutionary algorithm can optimize network topologies and NN parameter settings. Empirical validation was conducted on an HR with 22° of freedom of movement. When the results were compared with baseline methods, the phenomenon was found to be a 34.7% increase in manipulation success rates, a 28.3% reduction in task completion time (TCT), and a 41.2% increase in adaptation speed (AS). The theoretical model also identified emergent cognitive functions. These practices included adaptive body schema learning, predictive motor control, and context-aware attention allocation.

© 2026 Published by Faculty of Engineering

[1] Corresponding author: Sudhakar Sengan
Email: sudhasengan@gmail.com

1. INTRODUCTION

Development has been experimental in the context of embodied cognition (EC) to attain human-like intelligence in robotic models (RM) (Sandini et al., 2024). This development conflicts with methods that rely entirely on machine learning (ML). According to EC, cognitive intelligence arises not just from hypothetical and symbolic language or logical thinking; rather, it results from the dynamic correlation among sensorimotor development, devices, environmental limitations, and adaptive learning mechanisms. This combination of factors has been recognized as the primary source of intelligent behavior. Despite the discovery that the newest generation of Humanoid Robots (HR) has control and mechanical systems that are significantly more complex than those of humans, these robots still face limitations in comparison to human functions regarding control and manipulation accuracy, adaptability to settings, and cognitive size (Tong et al., 2024; Wang et al., 2024).

Perception, action, and cognition are all components of an integrated network, as defined by the EC hypothesis, which holds that CNs are based on the communication a human being has with the physical world. The traditional method of artificial intelligence (AI), on the other hand, assumes that sensing, cognitive development, and task performance are chronologically logical steps. Integrating the network's resources rather than functioning as separate peripheral components is essential for sensorimotor models used in intelligent RM (George, 2025). Conventional control methods are centered on predefined mechanical models and linear Machine Learning (ML), which pose challenges for RM in developing adaptive Artificial Intelligence (AI). These mathematical models regularly employ discrete neural networks (NNs) across multiple sensory modalities and cognitive functions, and as their size increases, they fail to capture the integrated nature of cognitive ability. This is in contrast to recognising that deep learning (DL) has advanced, enabling more flexible, adaptive approaches to handling complex data and uncertainty in real-world use. Therefore, RM performs well in models but struggles in uncertain environments, with standard factors, and in the complexity of real-world applications, such as autonomous networks and real-time data processing (Sarker et al., 2025; Siddamsetti et al., 2025).

The difficult task RM addresses is integrating multiple sensory modalities in ever-changing environments. Presently, the methods have failed to detect complex cross-modal relationships, and the optimal integration model is determined by the task being performed, the external environment's parameters, and the reliability of the sensing device (Li et al., 2025). To adapt to contexts that constantly evolve, adaptive fusion mechanisms are required; however, current systems fail to provide the network flexibility needed to automatically detect and perform these actions (Roychoudhury et al., 2023). To successfully address the problem at hand, evolutionary algorithms for neural networks (NNs) propose robust solutions. These methods dynamically discover network topologies and features tailored to specific tasks and adapt to changing circumstances. The application of these modalities is superior to that of gradient-based methods, which are highly susceptible to local minima and require a network-based simulation method that is dynamically implemented. Methods like these are more reliable than gradient-based methods. On the other hand, according to Siddamsetti et al. (2024), the present study has mainly focused on low-level control variables or primitive elements, suggesting that there have been limited challenging surveys of NNs.

This research proposes a model that combines hierarchical Multi-Sensor Fusion (MSF) with Deep Evolutionary Optimization (DEO) for EC in HR. By implementing novel Genetic Algorithms (GAs) to improve cognitive networks (CN), the neural network (NN) and its parameters have shown significant improvements. The use of MSF fitness functions has been completed to achieve a balance among task performance, computational efficiency, and adaptability. To put the core concepts of EC into practical application, adapted computational models are developed. These computational models maintain the coupling mechanism that exists between the central CN, motor control, and sensory processing. These models enable improved interaction and coordination among these systems in real-time applications. thereby enabling these networks to interact and coordinate with one another in real-time applications even more reliably.

The following constitute a number of the important contributions that the present study provides to the emerging discipline of embodied robotics (ER) and CN:

a) By combining MSF senses by the use of learning-based attention mechanisms (AM), it creates a novel genetic model that is robust to automatically finding optimal neural networks (NN) for EC.
b) It indicates that CN, also referred to as context-aware decision-making, adaptive body-schematic learning, and predictive computational processing, can be generated from an advanced NN and developed without a requirement for specific programming to develop these features.
c) The fact provides complete empirical testing, which implies significant performance improvements across an extensive number of manipulative tasks, inputs, and signal protocols.
c) Research highlights that new responses may result from optimizing sensorimotor integration networks, which improves the conceptual understanding of EC.

This work focuses on developing a unified model of embodied cognition for humanoid robots by combining Hierarchical Multi-Sensor Fusion (MSF) and Deep Evolutionary Optimization (DEO). In contrast to the stand-alone RM, which can independently perform perception, cognition, and manipulation, the proposed model optimizes multimodal sensory integration, cognitive data processing, and movement control through evolutionary NN optimization. The proposed model also employs a multi-objective genetic optimization protocol to evolve network topology and its control parameters, as well as an attention-guided fusion mechanism that integrates visual, tactile, proprioceptive, and force feedback information. It also allows the development of higher-order cognitive functions, such as adaptive learning of body schema, prediction of motor control, selective attention to context, and sensorimotor coordination, without any explicit rule-based coding. An experimental test was performed with a 22-degree-of-freedom humanoid RM, which has shown significant advantages in task success rate, manipulation accuracy, learning convergence, robustness, and generalization capability compared to conventional DL, multi-modal fusion models, and classical evolutionary approaches. Hence, the recommended DEO-EC provides a scalable, adaptive computational model for the next generation of humanoid RMs with embodied intelligence.

Studies involving simulation and real-world autonomous robot deployment were tested on an augmented humanoid network with 22 degrees of freedom and sensory systems, designed to support simulation testing. An aggregate of over 2,000 manipulation and expansion tests was performed across a multitude of tasks. As for the following sections within this article, these individuals are laid out as follows. Research related to EC, MSF, DEO, and RM is addressed in Section 2. In Section 3, the theoretical model is described, which features the deployment of NN, MSF, DEO, and EC. Detailed data on the experimental setup are provided for each component system. The computerised simulation model, trial cases, analysis parameter values, and baseline models are all described in Section 4. These findings are presented in Section 5, which includes the results of the model function test, evolutionary optimisation, manipulation task, and CN evaluation. Section 6 includes a summary of the key contributions that were made.

2. LITERATURE REVIEW

Embodied cognition requires integration of diverse sensory modalities. Deep Learning (DL) models have been used to improve object recognition accuracy in manipulation contexts. For autonomous navigation, sensor-fusion networks use LiDAR, visual, and inertial sensors to localise under dynamic conditions. Hierarchical AM is applied in real-time RM. MSF, together with deep NNs, provides HR with resilience and adaptability in uncertain environments (Sorrentino et al., 2024; Nowfal et al., 2024; Gopalakrishnan et al., 2025).

Gradient-free optimization methods, including evolutionary methods, have been used to tune gains in humanoid walking controllers automatically (Cui et al., 2025). These methods have outperformed manual calibration and shown improved convergence speed and transferability to a physical network. The optimization has been implemented in the context of evolution-based autonomous networks to tune low-level parameters and design the complete control network, achieving a balance among adaptability, stability, and computational cost.

The theory of embodied AI has been the focus of contemporary studies, which conclude that highly intelligent behavior in autonomous systems is not primarily an outcome of basic intelligence but rather a combination of morphological features, perception, action, and learning. To address the problem of separating control using robots from human-like execution control, this theoretical model has been developed for multimodal large models. In these complex models, it must be implemented to combine morphological analysis, multi-modal sensing, and adaptive control methods (Babadian et al., 2023).

The review discusses developments in multi-modal perception, sensor fusion, evolutionary optimization, and embodied CN. But a unified model that evolves NN topologies and optimizes sensorimotor combination under multi-objective measures is still required. This research proposes a hierarchical MSF co-optimized with DEO for EC and manipulation in HR (Ušinskis et al., 2025).

3. METHODOLOGY

The methodology chapter has a significant impact by outlining the conceptual model and the analytical basis for integrating MSF and DEO to improve EC in HR. This method can address the key problem of developing adaptive sensorimotor systems that prove evolved CN (Han et al., 2025). An artificial neural network (ANN) adapted for dynamic multisensory analysis and manipulation control has been developed as a result of the research. The following NN improves the application of adaptive sensorimotor systems to attain enhanced ecological cognition (EC) in HR communication.

3.1. The System Model

The model, which uses a genetically optimised neural network (NN), is hierarchical and incorporates sensory perception, cognitive processing, and motor control (Figure 1). The Sensory Integration Layer (SIL), the Cognitive Processing Layer (CPL), and the Motor Control Layer (MCL) comprise the 3 layers that make up this mode of functioning. Each of these layers is also linked to other layers by an adaptive NN. The state

transition function is the mathematical analysis of the network (Sartore et al., 2024).

$$s_(t + 1) = f(s_t, u_t, e_t; \theta) \quad (1)$$

where,
$s_t \in R^n$ is the time-dependent state vector of the network t, $u_t \in R^m$ is the input vector for the control system
$e_t \in R^p$ is the vector functioning as environment-specific data
$\theta \in R^q$ are the NN parameters that have been transformed
$f: R^n \times R^m \times R^p \rightarrow R^n$ is a state evolution function that is non-linear in the real world.

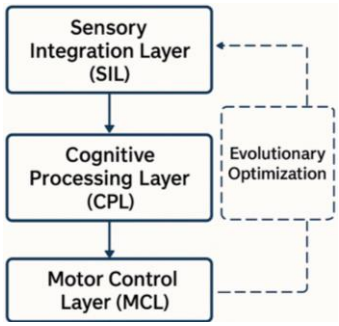


Figure 1. The proposed system model

The Sensory Integration Layer processes multi-modal sensor data using attention-weighted fusion mechanisms. The sensor fusion output is computed as:

$$\mathbf{z}_t = \sum_{i=1}^{N_s} \alpha_i(\mathbf{h}_t) \cdot \mathbf{z}_i^{(t)} \quad (2)$$

where,
N_s is the number of distinctive sensor modes of operation
$\mathbf{z}_i^{(t)} \in \mathbb{R}^{d_i}$ is the feature matrix (the ith sensor parameter)
$\alpha_i(\mathbf{h}_t)$ is the weight of attention (i-th modality) in connection with the hidden state $\mathbf{h}_t$
$\mathbf{z}_t \in \mathbb{R}^D$ is a sensor that becomes attached as $D = \sum_{i=1}^{N_s} d_i$.

A recurrent neural network (RNN) has been implemented in the cognitive processing layer (CPL), which is key to maintaining the state of functional memory and generating behavioural patterns focused on a definite objective.

The evolution of the cognitive function is as described below:

$$\mathbf{h}_{t+1} = \tanh\left(\mathbf{W}_h \mathbf{h}_t + \mathbf{W}_z \mathbf{z}_t + \mathbf{W}_g \mathbf{g}_t + \mathbf{b}_h\right) \quad (3)$$

where,
$\mathbf{W}_h \in \mathbb{R}^{H \times H}, \mathbf{W}_z \in \mathbb{R}^{H \times D}$ and $\mathbf{W}_g \in \mathbb{R}^{H \times G}$ the weight matrices that provide data
$\mathbf{g}_t \in \mathbb{R}^G$ is the path of the goal vector
$\mathbf{b}_h \in \mathbb{R}^H$ is the path of the bias vector
H is the dimension of the hidden state system
G is the dimension of the target model

According to Liu, H. et al. (2025), the MCL serves as a mechanism for applying unified signals by integrating exponential dynamics computation methods with data on control-governing policies.

The next is the conceptual model of the motor command generation method:

$$\boldsymbol{\tau}_t = \mathbf{M}(\mathbf{q}_t)\ddot{\mathbf{q}}_t + \mathbf{C}(\mathbf{q}_t, \dot{\mathbf{q}}_t) + \mathbf{G}(\mathbf{q}_t) + \boldsymbol{\tau}_{\text{ext}} \quad (4)$$

where,
$\boldsymbol{\tau}_t \in \mathbb{R}^{N_j}$ is the torque vector at the joint area
$\mathbf{M}(\mathbf{q}_t) \in \mathbb{R}^{N_j \times N_j}$ is a matrix of inertia's motion
$\mathbf{C}(\mathbf{q}_t, \dot{\mathbf{q}}_t) \in \mathbb{R}^{N_j}$ is the forces of centrifugal and Coriolis motion
$\mathbf{G}(\mathbf{q}_t) \in \mathbb{R}^{N_j}$ is the force of gravitation
$\boldsymbol{\tau}_{\text{ext}} \in \mathbb{R}^{N_j}$ are external factors and impacts
$\mathbf{q}_t, \dot{\mathbf{q}}_t, \ddot{\mathbf{q}}_t \in \mathbb{R}^{N_j}$ are the coordinates of the joints, velocity values, and rates of acceleration
N_j is the sum of degrees of freedom

Inter-layer communication is aided by learned connection matrices that evolve during optimisation (Chen et al., 2025).

The connection strength between layers i, j is ruled by:

$$\mathbf{C}_{ij} = \sigma\left(\mathbf{W}_{ij} + \epsilon_{ij}\right) \quad (5)$$

where,
$\mathbf{W}_{ij} \in \mathbb{R}^{n_i \times n_j}$ is the weight matrix of the evolved connections
ϵ_{ij} is the noise of the Gaussian distribution with a change
$\sigma_\epsilon^2, \sigma(\cdot)$ uses the sigmoid function as an activation function
$n_i; n_j$ is the dimension of layers $'i'$ and $'j'$.

The network uses a prediction-correction mechanism to maintain consistent behaviour across time steps, thereby minimising prediction error (Alowaidi et al., 2024).

$$L_{\text{pred}} = \frac{1}{T}\sum_{t=1}^{T} \|\hat{\mathbf{s}}_{t+1} - \mathbf{s}_{t+1}\|_2^2 \quad (6)$$

where,
$\hat{\mathbf{s}}_{t+1}$ is the predicted state
T is the prediction horizon value
L_{pred} is a temporal window prediction accuracy measure

3.2. MSF

The MSF primarily includes heterogeneous sensory data into coherent features that enable adaptive manipulation and normal cognitive analysis, but the system also serves as the sensory basis for the EC presented by MSF. Such a method can further develop the conceptual model to integrate sensor modalities that are temporally and geographically discrete while maintaining computational analysis and robustness to sensor errors and environmental change (Ibrahim Alasadi et al., 2025).

A. Sensor Modalities

Four primary sensory data points are used in this method of synthesizing sensor modes, which include visual, auditory, tactile, and olfactory data to provide a comprehensive understanding of the environment. The adoption of visual sensors enables the generation of high-dimensional spatial data on objects, scene geometry, and dynamical methods that operate in physical environments. Using convolutional feature extraction (FE), this data is processed to generate structured features (Al-Kababchee et al., 2023).

$$\mathbf{F}_v^{(l)} = \mathrm{Max}\left(0, \mathbf{W}_v^{(l)} * \mathbf{F}_v^{(l-1)} + \mathbf{b}_v^{(l)}\right) \quad (7)$$

where,
$\mathbf{F}_v^{(l)} \in \mathbb{R}^{H_l \times W_l \times C_l}$ r is the feature map at layer $'l'$
$\mathbf{W}_v^{(l)} \in \mathbb{R}^{k \times k \times C_{l-1} \times C_l}$ is the convolutional kernel weights
* is a convolution operation
$\mathbf{b}_v^{(l)} \in \mathbb{R}^{C_l}$ is a bias vector
H_l, W_l, C_l are height, width, and channel dimensions
k_r is the kernel size.

By implementing preliminary fusion methods that maintain spatial logic, the RGB-D sensor integration enables the combination of color and complex data.

$$\mathbf{I}_{rgbd} = \phi\left(\mathbf{I}_{rgb} \oplus \mathbf{I}_d\right) \quad (8)$$

where,
$\mathbf{I}_{rgb} \in \mathbb{R}^{H \times W \times 3}$ is an RGB image
$\mathbf{I}_d \in \mathbb{R}^{H \times W \times 1}$ is a depth map, a channel concatenation
$\phi: \mathbb{R}^{H \times W \times 4} \rightarrow \mathbb{R}^{H \times W \times D_v}$ the learned projection function integrates grouped input into a size 'D_v', which serves as a graphic representation.

Tactile sensors measure distributed force and pressure:

$$\mathbf{T}_t = \sum_{i=1}^{N_t} \beta_i \cdot \mathbf{p}_i^{(t)} \quad (9)$$

where,
N_t is the sum of sensors that are activated by touch
$\mathbf{p}_i^{(t)} \in \mathbb{R}^{D_t}$ is a time-dependent feature vector from the i^{th} physical sensor device
t, β_i is the total weight that was determined for the position of sensing device 'i'
$\mathbf{T}_t \in \mathbb{R}^{D_t}$ is the physical symbol that is formed simultaneously

The following is the definition of the sensory proprioceptive state vector, which is

$$\mathbf{P}_t = [\mathbf{q}_t^T, \dot{\mathbf{q}}_t^T, \mathbf{a}_t^T]^T \quad (10)$$

where,
$\mathbf{q}_t \in \mathbb{R}^{N_j}$ is the joint positions
$\dot{\mathbf{q}}_t \in \mathbb{R}^{N_j}$ are the velocity coefficients of the joints
$\mathbf{a}_t \in \mathbb{R}^3$is the data from the sensor
$\mathbf{P}_t \in \mathbb{R}^{2N_j+3}$ is the full vector of sensorimotor modes.

Force sensor technology embedded in end-user devices and physical contact areas transmits data describing dynamic communication.

$$\mathbf{F}_t = \sum_{j=1}^{N_f} \gamma_j\left(\mathbf{f}_j^{(t)} \odot \mathbf{n}_j\right) \quad (11)$$

where,
N_f is the size of a highly sensitive device
$\mathbf{f}_j^{(t)} \in \mathbb{R}^3$ is a vector of force derived from the j^{th} sensor
$\mathbf{n}_j \in \mathbb{R}^3$ is the vector that stands for the surface normal
$\odot$ is logical multiplication by variables
γ_j is the accuracy of the sensor data
$\mathbf{F}_t \in \mathbb{R}^{3N_f}$ is a force from the aggregate

The factors that follow are an indicator of the temporal time of data from multi-modal sensors:
where

$$\mathbf{s}_i^{\mathrm{sync}}(t) = \mathbf{s}_i\left(t_{k_i}\right) + \frac{t - t_{k_i}}{t_{k_i+1} - t_{k_i}}\left(\mathbf{s}_i\left(t_{k_i+1}\right) - \mathbf{s}_i\left(t_{k_i}\right)\right) \quad (12)$$

$\mathbf{s}_i^{\mathrm{sync}}(t)$ is the data from the synchronized sensors for modality 'i' at the time
t, t_{k_i}, t_{k_i+1} is expanded multiple times in series for the sensor size selection $'i'$ such that $t_{k_i} \leq t \leq t_{k_i+1}$,
$\mathbf{s}_i\left(t_{k_i}\right)$ is the active sensor data value as t_{k_i}.

Preprocessing is performed on the sensor modality, including noise reduction, standardization correction, and feature normalisation (Algburi et al., 2025).

$$\hat{\mathbf{s}}_i^{(t)} = \frac{\mathbf{R}_i\left(\mathbf{s}_i^{\mathrm{sync}}(t)\right) - \boldsymbol{\mu}_i}{\boldsymbol{\sigma}_i} \quad (13)$$

where,
$\mathbf{R}_i(\cdot)$ is a function for the reduction of noise for the modality
$i, \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i$ is the normalization by mean and standard deviation
$\hat{\mathbf{s}}_i^{(t)}$ is standardized sensor data

B. Fusion Model

Learned AM and cross-modal correlation analysis have been employed to combine multi-modal sensor data in the fusion model (Figure 2), which is an adaptation of a hierarchical integration method. This method rapidly and dynamically weights sensor data based on the task being performed, the state of the surrounding environment, and the sensor's reliability, while maintaining appropriate computing power for real-time use (AL-Gburi et al., 2024).

Hierarchy-based fusion includes feature-level fusion, correlation-based integration, and confidence-weighted combination. In each of the 3 phases stated, raw sensor features are integrated with learned prediction states to map multimodal sensor spatial domains.

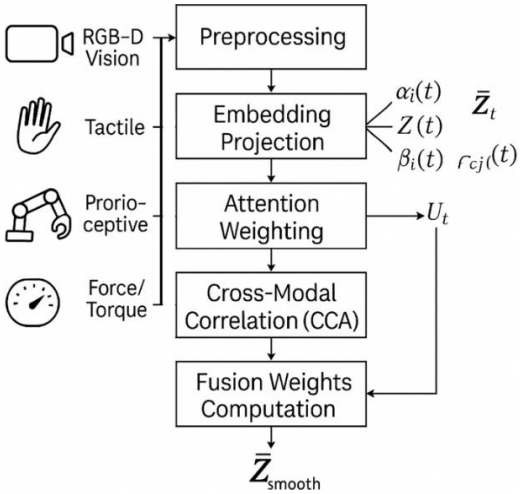


Figure 2. Fusion model

$$\mathbf{E}_i^{(t)} = \mathbf{W}_e^{(i)}\hat{\mathbf{s}}_i^{(t)} + \mathbf{b}_e^{(i)} \quad (14)$$

where,
$\mathbf{W}_e^{(i)} \in \mathbb{R}^{D_e \times D_i}$ is the modality's learned projection matrix
$i, \mathbf{b}_e^{(i)} \in \mathbb{R}^{D_e}$ is the bias vector
D_i is the preprocessed modal sensor data dimensions.
i, D_e is the standard embedding size
$\mathbf{E}_i^{(t)} \in \mathbb{R}^{D_e}$ is an embedded feature vector for the modality of the entity as $'i'$, time $'t'$.

The attention weights are set by the AM using a multi-layer perceptron. The AM subsequently assigns weights to all sensor modes, taking into account the current cognitive status and task-specific contextual factors. The result technique is performed to ensure that accuracy continues to increase.

$$\alpha_i^{(t)} = \frac{\mathrm{Exp}\left(\mathbf{v}_a^T \tanh\left(\mathbf{W}_a\left[\mathbf{h}_t; \mathbf{E}_i^{(t)}\right] + \mathbf{b}_a\right)\right)}{\sum_{j=1}^{N_S} \mathrm{Exp}\left(\mathbf{v}_a^T \tanh\left(\mathbf{W}_a\left[\mathbf{h}_t; \mathbf{E}_j^{(t)}\right] + \mathbf{b}_a\right)\right)} \quad (15)$$

where,
$\mathbf{v}_a \in \mathbb{R}^{D_a}$ is a vector encoding the attention output
$\mathbf{W}_a \in \mathbb{R}^{D_a \times (H + D_e)}$ is the matrix of cognitive weights
$\left[\mathbf{h}_t; \mathbf{E}_i^{(t)}\right]$ is a cognitive state-sensor embedding chain
$\mathbf{b}_a \in \mathbb{R}^{D_a}$ is the vector of attention bias.
D_a is the attention hidden size
$\alpha_i^{(t)}$ is the normalized attention weight for mode $'i'$ at time $'t'$.

The attention mechanism dynamically assigns modality-specific weights based on the current cognitive state and task context, thereby enabling adaptive sensor prioritisation and improving the robustness of multi-modal fusion.

Polynomial expansion correlation analysis is used to assess the statistical correlations that develop between multiple sensor modalities. Cross-modal correlation analysis is applied to find these relationships.

The following equation is used to compute the corresponding correlation value between modalities 'i' and 'j':

$$\rho_{ij}^{(t)} = \frac{\mathbf{E}_i^{(t)T}\mathbf{Q}_{ij}\mathbf{E}_j^{(t)}}{\sqrt{\left\|\mathbf{E}_i^{(t)}\right\|_2 \left\|\mathbf{E}_j^{(t)}\right\|_2}} \quad (16)$$

where,
$\mathbf{Q}_{ij} \in \mathbb{R}^{D_e \times D_e}$ is the cross-modal correlation matrix that was identified as $'i', 'j'$
$\rho_{ij}^{(t)} \in [-1,1]$ factors correlation

When defining the accuracy associated with each modality, the sensor reliability estimation requires accounting for the modality's internal coherence and historical results.

Determining the trust value for mode of operation 'i' is done with the following steps:

$$c_i^{(t)} = \mathrm{Exp}\left(-\frac{\left\|\mathbf{E}_i^{(t)} - \hat{\mathbf{E}}_i^{(t)}\right\|_2^2}{2\sigma_i^2}\right) \quad (17)$$

where,
$\hat{\mathbf{E}}_i^{(t)}$ is an embedding that was predicted based on input from other modalities
σ_i^2 is the value of the variance for the modality as $'i'$
$c_i^{(t)} \in [0,1]$ is the accuracy of the output

By applying a balanced integration mechanism, the multimodal fusion combines attention values, cross-modal correlated data, and true data.

$$\beta_i^{(t)} = \frac{\alpha_i^{(t)} \cdot c_i^{(t)} \cdot \sum_{j \neq i} \left|\rho_{ij}^{(t)}\right|}{\sum_{k=1}^{N_S} \alpha_k^{(t)} \cdot c_k^{(t)} \cdot \sum_{j \neq k} \left|\rho_{kj}^{(t)}\right|} \quad (18)$$

where,
$\beta_i^{(t)}$ is the result of the fusion value for modality $'i'$, including attention, confidence, and cross-modal reliability.

The weighted integration of embedded sensor features computes the fused model:

$$\mathbf{Z}_t = \sum_{i=1}^{N_S} \beta_i^{(t)}\mathbf{E}_i^{(t)} \quad (19)$$

where,
$\mathbf{Z}_t \in \mathbb{R}^{D_e}$ is the time-fused multi-modal value at time $'t'$.

When applied to low-frequency sensor noise, temporal smoothing helps maintain accuracy across multiple fusion functions. The application of polynomial smoothing methods can achieve the result:

$$\mathbf{Z}_t^{\mathrm{smooth}} = \gamma\mathbf{Z}_t + (1-\gamma)\mathbf{Z}_{t-1}^{\mathrm{smooth}} \quad (20)$$

where,
$\gamma \in [0,1]$ is the value for smoothing a region
$\mathbf{Z}_t^{\text{smooth}}$ is a fusion result that has been temporally smoothed

The fusion model includes uncertainty quantification by variance computed across ensemble predictions:

$$\mathcal{U}_t = \frac{1}{N_e}\sum_{k=1}^{N_e} \left\|\mathbf{Z}_{t,k} - \overline{\mathbf{Z}}_t\right\|_2^2 \quad (21)$$

where,
N_e is the sum of members in the group
$\mathbf{Z}_{t,k}$ is the fusion result from the k-th ensemble
$\overline{\mathbf{Z}}_t = \frac{1}{N_e}\sum_{k=1}^{N_e} \mathbf{Z}_{t,k}$ is the standard deviation of the ensemble
$\mathcal{U}_t$ is the uncertainty of the fusion method, which is quantified.

3.3. DEO

The DEO, a statistical analysis method, dynamically develops NNs and variables to make optimal and efficient use of the EC's computational power. Using RM as the experimental setting, this research study aims to identify optimal network topologies and weight values across high-dimensional parameter sets.

A. Evolutionary algorithm

In particular, a hybrid Genetic Algorithm (GA) has been employed for the evolutionary algorithm. The following GA is a hybrid combining structural evolution and parametric optimisation. By implementing a multi-population coevolutionary method, separate populations for NNs and link weights are reliably developed and maintained. By applying structured randomization methods, the starting-point method generates individuals with a range of features.

A directed graph is provided for each individual who is a member of the network population as follows:

$$\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{F}) \quad (22)$$

where,
$\mathcal{V} = \{v_1, v_2, \dots, v_{N_v}\}$ is the NN layer corresponding to the nodes
$\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of directed edges that value layer connections
$\mathcal{F} = \{f_1, f_2, \dots, f_{N_v}\}$ is node-specific activation functions
N_v is the count of nodes in the network

The genotype encoding for NN employs a matrix value that captures connectivity patterns and functional tasks:

$$\mathbf{A} = [\mathbf{C} \mid \mathbf{T}] \quad (23)$$

where,
$\mathbf{C} \in \{0,1\}^{N_v \times N_v}$ An adjacency matrix encoder links
$\mathbf{T} \in \mathbb{N}^{N_v \times 3}$ is an encoding matrix for layer type, neuron count, and activation function
$\mathbf{A} \in \mathbb{R}^{N_v \times (N_v+3)}$ achieve the network genotype data

The parameter population weight matrices and bias vectors are attained using a hierarchical encoding method.

$$\mathbf{\Theta} = \left\{\mathbf{W}^{(1)}, \mathbf{b}^{(1)}, \mathbf{W}^{(2)}, \mathbf{b}^{(2)}, \dots, \mathbf{W}^{(L)}, \mathbf{b}^{(L)}\right\} \quad (24)$$

where,
$\mathbf{W}^{(l)} \in \mathbb{R}^{n_l \times n_{l-1}}$ is the weight matrix for the layer
$l, \mathbf{b}^{(l)} \in \mathbb{R}^{n_l}$ is a vector of bias linked to a particular layer
l, n_l is the layer neuron count
l, L is the total number of layers
$\mathbf{\Theta}$ is a genotype value.

Elitism is evident in the decision-making process through the way game selection is implemented.

For the individual 'i,' the selection probability can be computed according to the following method:

$$P_i^{sel} = \frac{\text{rank}(f_i)^{-\tau}}{\sum_{j=1}^{Np} \text{rank}(f_j)^{-\tau}} \quad (25)$$

where,
f_i is individual fitness
$i, \text{rank}(f_i)$ is the ranking of 'i' in the sorted population
$\tau > 0$ controls the selection weight
N_p is the size of the population
P_i^{sel} is the probability of individual $'i'$.

The crossover operator in network evolution uses a semantic-preserving method to maintain connectivity.

$$\mathcal{G}_{\text{offspring}} = \psi(\mathcal{G}_{\text{parent 1}}, \mathcal{G}_{\text{parent 2}}, \xi) \quad (26)$$

where,
$\psi: \mathcal{G} \times \mathcal{G} \times [0,1] \to \mathcal{G}$ is crossover function
$\xi \in [0,1]$ is the crossover probability value
$\mathcal{G}_{\text{off spring}}$ is the offspring network value

By applying symmetrical arithmetic, a process called the parameter crossover confirms that the statistical value of the corresponding parent populations is maintained during the procedure as follows:

$$\mathbf{\Theta}_{\text{offspring}} = \eta\mathbf{\Theta}_{\text{parent 1}} + (1-\eta)\mathbf{\Theta}_{\text{parent 2}} + \mathcal{N}(0, \sigma_{\text{cross}}^2\,\mathbf{I}) \quad (27)$$

where,
$\eta \in [0,1]$ is weight
$\mathcal{N}(0, \sigma_{\text{cross}}^2\,\mathbf{I})$is Gaussian noise
σ_{cross}^2 , $\mathbf{I}$ is the matrix value
$\mathbf{\Theta}_{\text{offspring}}$ is the offspring vector

The mutation operator can apply probabilistic node and edge changes to support network expansion and evolution. These changes ensure that the network maintains its size for the following reasons:

$$\mathcal{G}_{\text{mutated}} = \mu\left(\mathcal{G}, p_{\text{node}}, p_{\text{edge}}, p_{\text{func}}\right) \quad (28)$$

where,
$\mu: \mathcal{G} \times [0,1]^3 \rightarrow \mathcal{G}$ is the mutation function
p_{node} is the likelihood of node change
p_{edge} is the likelihood of an edge update
p_{func} is the likelihood of the activation function
$\mathcal{G}_{\text{mutated}}$ is a transformed network

By applying adaptive Gaussian perturbation when combined with self-adaptive variance control, the following parameter mutation expansion is applied as follows:

$$\boldsymbol{\Theta}_{\text{mutated}} = \boldsymbol{\Theta} + \mathcal{N}(0, \sigma^2_{\text{mut}}(\boldsymbol{\Theta})\mathbf{I}) \quad (29)$$

where,
$\sigma^2_{\text{mut}}(\boldsymbol{\Theta}) = \sigma_0^2 \text{Exp}(\tau' \mathcal{N}(0,1) + \tau \mathcal{N}_i(0,1))$ is adaptive mutation variance
σ_0^2 is the initial value
τ', τ is the learning rate value
$\mathcal{N}(0,1)$ is global random noise
$\mathcal{N}_i(0,1)$ is Component-wise random noise
$\boldsymbol{\Theta}_{\text{mutated}}$ is a transformed parameter vector

The elite security model maintains a fraction $'\epsilon'$ of the population:

$$\mathcal{E}_{t+1} = \arg \max_{\mathcal{S} \subset \mathcal{P}_t, |\mathcal{S}| = \lfloor \epsilon N_p \rfloor} \sum_{i \in \mathcal{S}} f_i \quad (30)$$

where,
$\mathcal{E}_{t+1}$ is the elite set for the group
$t+1, \mathcal{P}_t$ is the population in a group
$t, \epsilon \in [0,1]$ is elitism segment
$\lfloor \cdot \rfloor$ is the base function

The maintenance of distinct features supports the avoidance of early convergence by fitness sharing based on phenotypic spatial distance:

$$f_i^{shared} = \frac{f_i}{\sum_{j=1}^{Np} \text{sh}(d_{ij})} \quad (31)$$

where,
$\text{sh}(d) = 1 - (d/\sigma_{\text{share}})^{\alpha}$ if $d < \sigma_{\text{share}}$ and 0 otherwise if sharing function
d_{ij} is an individual's genetic variation from that of a different individual
i and j, σ_{share} is the sharing radius
α finds the layout of the sharing
f_i^{shared} is the fitness-shared value for individual $'i'$.

B. Fitness function formulation

The fitness function has been defined as a multi-objective test network that evaluates the performance of the developed NN in RM across accuracy, speed, and resource utilization. Several performance metrics are integrated via weighted aggregation and adaptive coefficient change, enabling a more comprehensive evaluation of the NN's performance across objectives such as accuracy, speed, and resource utilisation.

$$F(\boldsymbol{\Theta}, \mathcal{G}) = \sum_{k=1}^{K} w_k(t) \cdot f_k(\boldsymbol{\Theta}, \mathcal{G}) \quad (32)$$

where,
$F(\boldsymbol{\Theta}, \mathcal{G})$ is the fitness value set $'\boldsymbol{\Theta}'$
$\mathcal{G}, K$ is the objective function
$w_k(t)$is a time weight for objective $'k'$ at group $'t'$
$f_k(\boldsymbol{\Theta}, \mathcal{G})$ is a normalised value of objective k.

This element of task-specific performance metrics shows the success rates of manipulative actions in a selection of cases with multiple levels of complexity.

The method of computation for such an element is as described below:

$$f_1(\boldsymbol{\Theta}, \mathcal{G}) = \frac{1}{N_T} \sum_{i=1}^{N_T} \phi_i \cdot S_i(\boldsymbol{\Theta}, \mathcal{G}) \cdot Q_i(\boldsymbol{\Theta}, \mathcal{G}) \quad (33)$$

where,
N_T is the number of test tasks
ϕ_i is the difficulty weight for task $'i'$
$S_i(\boldsymbol{\Theta}, \mathcal{G}) \in \{0,1\}$ is binary success indicator for task $'i'$
$Q_i(\boldsymbol{\Theta}, \mathcal{G}) \in [0,1]$ is a smooth trajectory, and accurate force control finds application quality.

The trajectory quality test incorporates kinematic smoothness and dynamic efficiency using integrated performance measures:

$$Q_i(\boldsymbol{\Theta}, \mathcal{G}) = \lambda_1 Q_{\text{smooth}} + \lambda_2 Q_{\text{force}} + \lambda_3 Q_{\text{time}} \quad (34)$$

where,
$Q_{\text{Smooth}} = \text{Exp}\left(-\int_0^T \|\ddot{\mathbf{q}}(t)\|_2^2 dt\right)$ is a smoothness metric based on the joint sensor
$Q_{\text{Force}} = \text{Exp}\left(-\frac{1}{T}\int_0^T \|\mathbf{F}_{\text{Error}}(t)\|_2^2 dt\right)$ is force control accuracy
$Q_{\text{Time}} = \text{Exp}\left(-\frac{T - T_{\text{optimal}}}{T_{\text{threshold}}}\right)$ is a time factor of the metric
$\lambda_1, \lambda_2, \lambda_3$ is a weighting value with $\lambda_1 + \lambda_2 + \lambda_3 = 1$
T is the execution time
T_{optimal} is the optimal time for task performance
$T_{\text{threshold}}$ is an acceptable time deviation

To measure resource utilization features that distinguish dynamic networks, the computational efficiency aim implements the use of complexity metrics and real-time tests as follows:

$$f_2(\boldsymbol{\Theta}, \mathcal{G}) = \frac{1}{\kappa_1 C_{comp}(\mathcal{G}) + \kappa_2 T_{inf}(\boldsymbol{\Theta}, \mathcal{G}) + \kappa_3 M_{mem}(\mathcal{G})} \quad (35)$$

where,
$C_{\text{comp}}(\mathcal{G}) = \sum_{l=1}^{L} n_l \cdot n_{l-1}$ is computational complexity based on network values
$T_{\text{inf}}(\boldsymbol{\Theta}, \mathcal{G})$ is the average data time across test cases
$M_{\text{mem}}(\mathcal{G})$ is the memory trail of the network
$\kappa_1, \kappa_2, \kappa_3$ is the normalisation value

The robustness evaluation measures network performance under several alert conditions, including sensor noise, environmental variations, and dynamic disturbances:

$$f_3(\boldsymbol{\Theta}, \mathcal{G}) = \frac{1}{N_P} \sum_{j=1}^{N_P} \text{Exp}\left(- \frac{\left\| P_j(\boldsymbol{\Theta}, \mathcal{G}) - P_{\text{nominal}}(\boldsymbol{\Theta}, \mathcal{G}) \right\|_2^2}{2\sigma_P^2} \right) \quad (36)$$

where,
N_P is the number of alarm set-ups
$P_j(\boldsymbol{\Theta}, \mathcal{G})$ is the performance under the alarm condition
$j, P_{\text{nominal}}(\Theta, \mathcal{G})$ is the nominal performance without perturbations
σ_P^2 controls the robustness metric's sensitivity.

Cross-validation and transfer-learning experiments are used to evaluate the accuracy of a generalization metric across hidden tasks and contextual variables.

$$f_4(\boldsymbol{\Theta}, \mathcal{G}) = \frac{1}{N_G} \sum_{k=1}^{N_G} \rho_k \cdot G_k(\boldsymbol{\Theta}, \mathcal{G}) \quad (37)$$

where,
N_G is the number of test scenarios
ρ_k is the weight for scenario $'k'$
$G_k(\boldsymbol{\Theta}, \mathcal{G})$ is the performance ratio between the evolved system and baseline methods on hidden task k.

Statistical rule correlations have been employed in multi-objective optimization to ensure that the resulting solution diversity is preserved across the currently accessible objective space.

$$\boldsymbol{x}_1 \prec \boldsymbol{x}_2 \Leftrightarrow \forall i \in \{1,2,\dots,K\}: f_i(\boldsymbol{x}_1) \geq f_i(\boldsymbol{x}_2) \wedge \exists j \in \{1,2,\dots,K\}: f_j(\boldsymbol{x}_1) > f_j(\boldsymbol{x}_2) \quad (38)$$

where,
$\boldsymbol{x}_1 \prec \boldsymbol{x}_2$ is solution $\boldsymbol{x}_1$ as solution $\boldsymbol{x}_2$
$\boldsymbol{x}_1 = (\boldsymbol{\Theta}_1, \mathcal{G}_1)\ \boldsymbol{x}_2 = (\boldsymbol{\Theta}_2, \mathcal{G}_2)$ is a solution vector

The adaptive weight change mechanism is responsible for modifying objective weights in accordance with developments in the theory of evolution and features of population convergence:

$$w_k(t+1) = w_k(t) + \alpha_w \frac{\partial H(t)}{\partial w_k(t)} \quad (39)$$

where,
$H(t) = -\sum_{i=1}^{N_p} P_i(t) \text{Log}\, P_i(t)$ is the population range entropy
$P_i(t) = \frac{F_i(t)}{\sum_{j=1}^{N_p} F_j(t)}$ is the normalised fitness probability for an individual
i, α_w is the weight adaptation learning rate
$\frac{\partial H(t)}{\partial w_k(t)}$ is the measure of the sensitivity of the range to weight $'k'$

A penalty function for model and performance augmentation boundaries is deployed into the limitation control function for the following objectives:

$$F_{\text{constrained}}(\boldsymbol{\Theta}, \mathcal{G}) = F(\boldsymbol{\Theta}, \mathcal{G}) - \sum_{m=1}^{M} \beta_m \max(0, g_m(\boldsymbol{\Theta}, \mathcal{G})) \quad (40)$$

where,
M is the number of limits
$g_m(\boldsymbol{\Theta}, \mathcal{G})$ is m-th constraint violation test
β_m is the penalty coefficient for the limit
m, $F_{\text{constrained}}(\Theta, \mathcal{G})$ is the constraint-adjusted fitness value

A detailed examination of the proposed solutions has been made possible by the multi-objective fitness formulation, which also helps achieve a balance between important performance requirements for dynamic integrated RM.

3.4. EC implementation

To function accurately, the EC employs an evolved NN, in which the cognitive topology Gcog and the variables Θcog are optimised using an evolutionary method. The cognitive state space is marked as multidimensional, integrating internal neural features and external sensorimotor configurations.

$$\mathcal{M} = \{\mathbf{s} \in \mathbb{R}^{D_c} : \mathbf{s} = [\mathbf{h}^T, \mathbf{q}^T, \mathbf{z}^T, \mathbf{a}^T]^T\} \quad (41)$$

where,
$\mathcal{M}$ is an embodied cognitive manifold
$\mathbf{h} \in \mathbb{R}^H$ is an internal cognitive state processed using an NN
$\mathcal{G}_{\text{cog}}$, $\mathbf{q} \in \mathbb{R}^{N_j}$ is the joint configuration
$\mathbf{z} \in \mathbb{R}^{D_e}$ is fused sensory, from Eq. (19)
$\mathbf{a} \in \mathbb{R}^{D_a}$is an action state
$D_c = H + N_j + D_e + D_a$ is the size of the cognitive state space.

The embodied cognitive framework operates through a unified interaction between cognitive state representation, working memory, attention allocation, and policy learning. Equation (41) defines the multidimensional cognitive manifold by integrating internal neural states with sensorimotor configurations and fused sensory information, thereby providing a coherent representation of embodied intelligence.

Equation (44) enables the evolved recurrent memory model to preserve temporal dependencies and task-relevant contextual information, supporting adaptive learning and sequential decision-making. Furthermore, the attention mechanism dynamically modulates the importance of sensory and cognitive features according to task requirements and environmental conditions, thereby facilitating efficient information processing. Equation (47) employs an evolved policy network based on active inference principles to generate actions that minimise prediction error while simultaneously balancing task performance and computational efficiency. Together, these mechanisms establish a closed sensorimotor loop that promotes context-aware reasoning, predictive control, and adaptive manipulation capabilities in humanoid robots.

In line with the developed NN, the CN develops, and the topological model and variables of the NN serve as the basis for detecting the state transition function:

$$\frac{d\mathbf{s}}{dt} = \mathcal{G}_{\text{cog}}(\mathbf{s}, \mathbf{u}, \mathbf{e}; \boldsymbol{\Theta}_{\text{cog}}) \quad (42)$$

where,
$\mathcal{G}_{\text{cog}}$ is the evolved CN from the population that maximizes the fitness function in Eq. (32)
$\mathbf{u} \in \mathbb{R}^m$ is the input of the control value
$\mathbf{e} \in \mathbb{R}^p$ are environmental incentives
$\boldsymbol{\Theta}_{\text{cog}}$ is evolved NN values optimized using the mutation and crossover functions, Eqs. (28) and (29).

By adopting evolutionary, forward-thinking models, the sensorimotor integration AM can implement predictive coding.

This can be achieved by evolving the prediction network and parameter settings.

$$\mathbf{e}_{\text{pred}}^{(t)} = \mathbf{z}_t - \mathcal{G}_{\text{forward}}(\mathbf{a}_{t-1}, \mathbf{s}_{t-1}; \boldsymbol{\Theta}_{\text{forward}}) \quad (43)$$

where,
$\mathbf{e}_{\text{pred}}^{(t)} \in \mathbb{R}^{D_e}$ is the sensory prediction error
$\mathcal{G}_{\text{forward}}$ is the evolved forward NN that was selected based on the robustness measure in Eq. (36)
Θ_{forward} is the evolved forward model values that contribute to the generalization fitness factor f_4 in Eq. (37).

Using an advanced RNN that stores data related to the task at hand, the working memory mechanisms are applied:

$$\mathbf{m}_t = \mathcal{G}_{\text{memory}}\left([\mathbf{m}_{t-1}; \mathbf{z}_t; \boldsymbol{\alpha}_t]; \boldsymbol{\Theta}_{\text{memory}}\right) \quad (44)$$

where,
$\mathbf{m}_t \in \mathbb{R}^{D_m}$ is the working memory state computed using the evolved memory model
$\mathcal{G}_{\text{memory}}$, $[\mathbf{m}_{t-1}; \mathbf{z}_t; \boldsymbol{\alpha}_t]$ is the concatenated input, including previous memory state, current sensor data, and attention weights, Eq. (15)
$\boldsymbol{\Theta}_{\text{memory}}$ contains the evolved memory model values

It probably exposes the AM with the help of an NN that is constantly evolving and challenging for selection based on the computational efficiency objective f_2 in equation (35):

$$\boldsymbol{\alpha}_t = \mathcal{G}_{\text{attention}}([\mathbf{h}_t; \mathbf{g}_t; \mathbf{m}_t]; \boldsymbol{\Theta}_{\text{attention}}) \quad (45)$$

where,
$\mathcal{G}_{\text{attention}}$ is the evolved attention NN
$[\mathbf{h}_t; \mathbf{g}_t; \mathbf{m}_t]$ is the concatenated input of cognitive state, goal data, and memory state
$\boldsymbol{\Theta}_{\text{attention}}$ is evolved NN parameters that minimize inference time T_{inf} in the efficiency objective.

The goal decomposition hierarchy employs evolved NNs at each level, where the hierarchical network itself emerges from the evolutionary method:

$$\mathbf{g}_t^{(l)} = \mathcal{G}_D^{(l)}\left([\mathbf{g}_t^{(l-1)}; \mathbf{h}_t^{(l)}; \mathbf{c}_t^{(l)}]; \boldsymbol{\Theta}_D^{(l)}\right) \quad (46)$$

where,
$\mathcal{G}_D^{(l)}$ is an evolved decomposition network at hierarchical level $'l'$, selected based on the task performance measure f_1 in Eq. (33)
$\mathbf{g}_t^{(l)} \in \mathbb{R}^{G_l}$ is the goal at level l
$\boldsymbol{\Theta}_D^{(l)}$ is evolved decomposition values that maximize manipulation success rates.

The action selection AM implements active inference using evolved policy networks that minimize predictable free energy:

$$\mathbf{a}_t^* = \mathcal{G}_{\text{policy}}\left(\left[\mathbf{s}_t; \mathbf{g}_t; \mathbf{e}_{\text{pred}}^{(t)}\right]; \boldsymbol{\Theta}_{\text{policy}}\right) \quad (47)$$

where,
$\mathcal{G}_{\text{policy}}$ is an evolved policy network optimized for the multi-objective fitness function in Eq. (32)
$\left[\mathbf{s}_t; \mathbf{g}_t; \mathbf{e}_{\text{pred}}^{(t)}\right]$ is the concatenated input, including current state, goal, and prediction error
$\boldsymbol{\Theta}_{\text{policy}}$ is the evolved policy parameters that balance task performance and computational efficiency.

The body schema adaptation employs evolved learning networks that update internal models based on sensorimotor correlations:

$$\mathbf{B}_{t+1} = \mathcal{G}_{\text{schema}}([\mathbf{B}_t; \Delta\mathbf{x}; \Delta\mathbf{q}]; \boldsymbol{\Theta}_{\text{schema}}) \quad (48)$$

where,
$\mathcal{G}_{\text{schema}}$ is the evolved body schema update network selected based on the robustness criterion f_3, Eq. (36),
$\mathbf{B}_t \in \mathbb{R}^{3\times N_j}$ is the current body schema estimate
$\Delta\mathbf{x}$; $\Delta\mathbf{q}$ are the test end-effector and joint displacements
$\boldsymbol{\Theta}_{\text{schema}}$ is evolved NN parameters

Algorithm 1: DEO for EC

Input: Population size N_p, generations G_{max}, sensor modalities N_s, fitness weights w_k, mutation rates p_{mut}, crossover rates p_{cross}

Output: Optimal CN-G_{best} ; parameters Θ_{best}

1: Initialize
2: $P_{arch} \leftarrow$ InitializeArchitecturePopulation (N_p) // Eqs. (22)-(23)
3: $P_{param} \leftarrow$ InitializeParameterPopulation (N_p) // Eq. (24)
4: generation $\leftarrow 0$
5: $F_{best} \leftarrow -\infty$
6: **While** generation < G_{max} **Do** 7: // Evaluate population fitness
8: **For Each** individual i = 1 to N_p **Do**
9: $G_i \leftarrow P_{arch}$ [i]
10: $\Theta_i \leftarrow P_{param}$[i] // MSF (Section 3.2)
13: **For each** time step t, **Do**
14: **For Each** sensor modality j = 1 to N_S **Do**
15: $s_j^t \leftarrow$ ReadSensorData (j, t)
16: $\hat{s}_j_t \leftarrow$ PreprocessSensor(s_j_t) // Eq. (13)
17: $E_j^t \leftarrow$ ProjectToEmbedding($\hat{s}_j^t$), W_e^(j)) // Eq. (14)
18: **End For**
19: $\alpha_t \leftarrow$ ComputeAttentionWeights (h_t, E_1^t...E_N_st) // Eq. (15)
21: $\rho_ij^t \leftarrow$ ComputeCorrelations (E_i^t, E_j^t // Eq. (16)
22: $c_i^t \leftarrow$ ComputeConfidence (E_i^t, $\hat{E}_i^t$) // Eq. (17)
23: $\beta_i^t \leftarrow$ ComputeFusionWeights (α_i^t, c_i^t, ρ_ij^t) // Eq. (18)
24: $Z_t \leftarrow$ FuseSensorData (β_i^t, E_i^t) // Eq. (19)
25: $Z_t^{smooth} \leftarrow$ TemporalSmoothing (Z_t, $Z_\{t-1\}^{smooth}$) // Eq. (20)
26: **End For**
27: // EC processing
29: **For each** time step t, **do**
30: $s_t \leftarrow [h_t^T, q_t^T, Z_t^T, a_t^T]^T$ // Eq. (41)
31: // Forward prediction
33: $\hat{z}_t \leftarrow G_{forward}(a_\{t-1\}, s_\{t-1\}; \Theta_{forward})$ // Eq. (43)
34: $e_pred^t \leftarrow Z_t - \hat{z}_t$ // Working memory update
37: $m_t \leftarrow G_{memory}([m_{t-1}; Z_t; \alpha_t]; \Theta_{memory})$ // Eq. (44)
38: // Attention value
40: $\alpha_t \leftarrow G_{attention}([h_t; g_t; m_t]; \Theta_{attention})$ // Eq. (45)
41: // Hierarchical goal decomposition
43: **For Each** level l = 1 to L_hier **Do**
44: $g_t^l \leftarrow G_D^l([g_t^{l-1}; h_t^l; c_t^l]; \Theta_D^l)$ // Eq. (46)
45: **End For**
46: // Action selection
48: $a_t^* \leftarrow G_{policy}([s_t; g_t; e_pred^t]; \Theta_{policy})$ // Eq. (47)
49: // Body schema update
51: $B_{t+1} \leftarrow G_{schema}([B_t; \Delta_x; \Delta_q]; \Theta_{schema})$ // Eq. (48)
52: // State update
54: $s_{t+1} \leftarrow G_{cog}(s_t, a_{t^*}, e_t; \Theta_{cog})$ // Eq. (42)
55: **End For**
56: // Fitness evaluation
58: $f_1 \leftarrow$ EvaluateTaskPerformance (Θ_i, G_i) // Eq. (33)
59: $f_2 \leftarrow$ EvaluateComputationalEfficiency (Θ_i, G_i) // Eq. (35)
60: $f_3 \leftarrow$ EvaluateRobustness (Θ_i, G_i) // Eq. (36)
61: $f_4 \leftarrow$ EvaluateGeneralization (Θ_i, G_i) // Eq. (37)
62: $F_i \leftarrow \Sigma k=1^4 w_k$(generation) × f_k // Eq. (32)
64: **If** $F_i > F_{best}$ **Then**
66: $F_{best} \leftarrow F_i$
67: $G_{best} \leftarrow G_i$
68: $\Theta_{best} \leftarrow \Theta_i$
69: **End If**
70: **End For**
71: // Selection
73: $P_{selected} \leftarrow$ TournamentSelection (P_{arch}, P_{param}, F, τ) // Eq. (25)
74: // Crossover
76: P_offspring_arch $\leftarrow \{\}$
77: P_offspring_param $\leftarrow \{\}$
78: **For** i = 1 to N_p/2 **Do**
79: parent1, parent2 $\leftarrow$ SelectParents($P_{selected}$)
80: **If** random() < p_{cross} **Then**
82: child1_arch, child2_arch $\leftarrow$ArchitectureCrossover (parent1.arch, parent2.arch, ξ) //Eq. (26)
83: child1_param, child2_param $\leftarrow$ ParameterCrossover (parent1.param, parent2.param, η) // Eq. (27)
84: **Else**
85: $child1_{arch}$, $child2_{arch} \leftarrow$ parent1.arch, parent2.arch
86: $child1_{param}$, $child2_{param} \leftarrow$ parent1.param, parent2.param
87: **End If**
88: $P_{offspring_arch} \leftarrow P_{offspring_arch} \cup \{child1_{arch}, child2_{arch}\}$
90: $P_{offspring_param} \leftarrow P_{offspring_param} \cup \{child1_{param}, child2_{param}\}$
91: **End For**
92: // Mutation
94: **For each** individual i in $P_{offspring}$, **do**
95: **If** random() < p_{mut} **Then**
96: $P_{offspring_arch}$ [i] $\leftarrow$ ArchitectureMutation ($P_{offspring_arch}$ [i], p_{node}, p_{edge}, p_{func}) // Eq. (28)
97: $P_{offspring_param}$ [i] $\leftarrow$ ParameterMutation ($P_{offspring_param}$ [i], σ_{mut}) // Eq. (29)
98: **End If**
99: **End For**
100: // Elitism and replacement
102: $E_{generation+1} \leftarrow$ SelectElite (P_{arch}, P_{param}, F, ε) // Eq. (30)
103: $P_{arch} \leftarrow E_{generation+1}$.arch $\cup$ $P_{offspring_arch}$ [1:N_p-$|E_{generation+1}|$]
104: $P_{param} \leftarrow E_{generation+1}$.param $\cup$ $P_{offspring_param}$ [1:N_p-$|E_{generation+1}|$
105: // Diversity preservation
107: **For each** individual i, **do**
108: $F_i^{shared} \leftarrow$ ApplyFitnessSharing (F_i, P_{arch}, P_{param}, σ_{share}, α) // Eq. (31)

109: **End For**
110: // Adaptive weight adjustment
112: **For** k = 1 to 4 **Do**
113: w_k (generation + 1) ← UpdateWeight (w_k (generation), H (generation), α_w) // Eq. (39)
114: **End For**
115: generation ← generation + 1
117: **End While**
118: **Return** G_{best}, Θ_{best}

4. EXPERIMENTAL SET-UP

4.1. Hardware platform

The study on manipulation and expansion was conducted using a humanoid robot that has 22 degrees of freedom in its movement, serving as a computational hardware architecture. Dual 7-degree-of-freedom arms, a 7-degree-of-freedom torso-and-head assembly, and a 1-degree-of-freedom waist joint with humanoid measurements are all integrated into the network. For the sensory system, stereo RGB-D cameras with a resolution of 1,920 x 1,080 and depth accuracy of ±2 *mm* were used. The system also included force-sensitive resistor arrays with 1 *mm* spatial resolution, high-resolution optical encoders with ± 0.088° accuracy, and 6-axis force/torque sensors for contact force measurements.

NVIDIA Jetson AGX Orin, which has a 2048-core graphics processing unit (GPU) and 32 GB of RAM, is implemented by the computer system for real-time NN inference. Ubuntu 20.04 LTS is the operating system, and ROS2 is the middleware. Dynamixel Pro servo motors, provided with harmonic drive reducers, function on the actuation side of operations. In addition to integrating position, velocity, and torque sensing, the motor control operates at 1 kHz.

4.2. Experimental scenarios

A. Basic manipulation tasks

A primary manipulation task set aims to improve the analysis of sensorimotor integration and motor control by deploying key manipulation primitives. In addition to analyzing measures of achievement such as the rate of successful manipulation, location accuracy, trajectory performance, and force control precision, it also includes tasks that involve the selection and placement of objects. Tool-use scenarios are used to control the size of the network to extend its manipulation capabilities with an assortment of tools. The performance of multi-object interaction tasks is a method that allows individuals to demonstrate their ability to coordinate manipulative actions involving multiple objects, either sequentially or simultaneously. Bimanual coordination, sequential manipulation chains, and cooperative manipulation are among the tasks in this group of activities.

B. Cognitive assessment tasks

The cognitive test task set was designed to measure cognitive abilities such as allocating attention, utilizing memory, adapting to new data sources, and solving problems in unpredictable conditions. The design measured the adaptive features of evolutionary CN and the network's ability to allocate computational resources dynamically. Using memory and learning evaluations, we had the chance to measure working memory capacity, the development of episodic memory, and adaptive learning from experience. Transfer learning capabilities across related task domains have been measured for generalisation testing. These features included training on specific object sets, workspace transfer, and skill transfer between manipulating basic objects.

4.3. Performance analysis and baseline models

The evaluation metrics used in this study are multidimensional. Task success rate is measured using completion percentage, complexity stratification, and inter-trial consistency. Manipulation precision is measured using end-effector position RMSE, angular orientation deviation, and trajectory tracking fidelity. To assess responsiveness in uncertain environments, the cognitive response latency (input-to-action), motor execution duration (initiation-to-completion), and decision-making delay are measured. Learning performance is measured using convergence speed, learning curve slope, retention stability, and transfer efficiency for novel-task generalisation. Cognitive response times (input-to-action), motor execution time (initiation-to-completion), and decision-making delay are all measured to evaluate sensitivity in variable contexts. The convergence speed, the slope of the learning curve, the retention stability, and the transfer efficiency for novel-task generalisation are the metrics used to assess learning performance. For comparative benchmarking, 5 baseline methods were measured: (1) PD joint control; (2)DL; (3) GA-based adaptive optimization; (4) multi-modal fusion; and (5) cognitive networks.

5. RESULTS AND DISCUSSION

5.1. An assessment of system performance

The results of the analysis of the recommended DEO-EC against several baseline methods are presented in Table 1 and Figure 3. A total of 500 manipulation tests were performed for each method, and the results were applied to a range of task cases. Compared with the baseline models, DEO-EC achieved significant improvements across all test metrics. However, the statistical analysis showed that the p-values for all principal indicators were less than 0.001. Task success rate improved by 9.9%, location accuracy improved by 34.4% and 34.1%, response time reduced by 28.9%, and learning convergence improved by 28.3%. The robustness score and generalization index also improved.

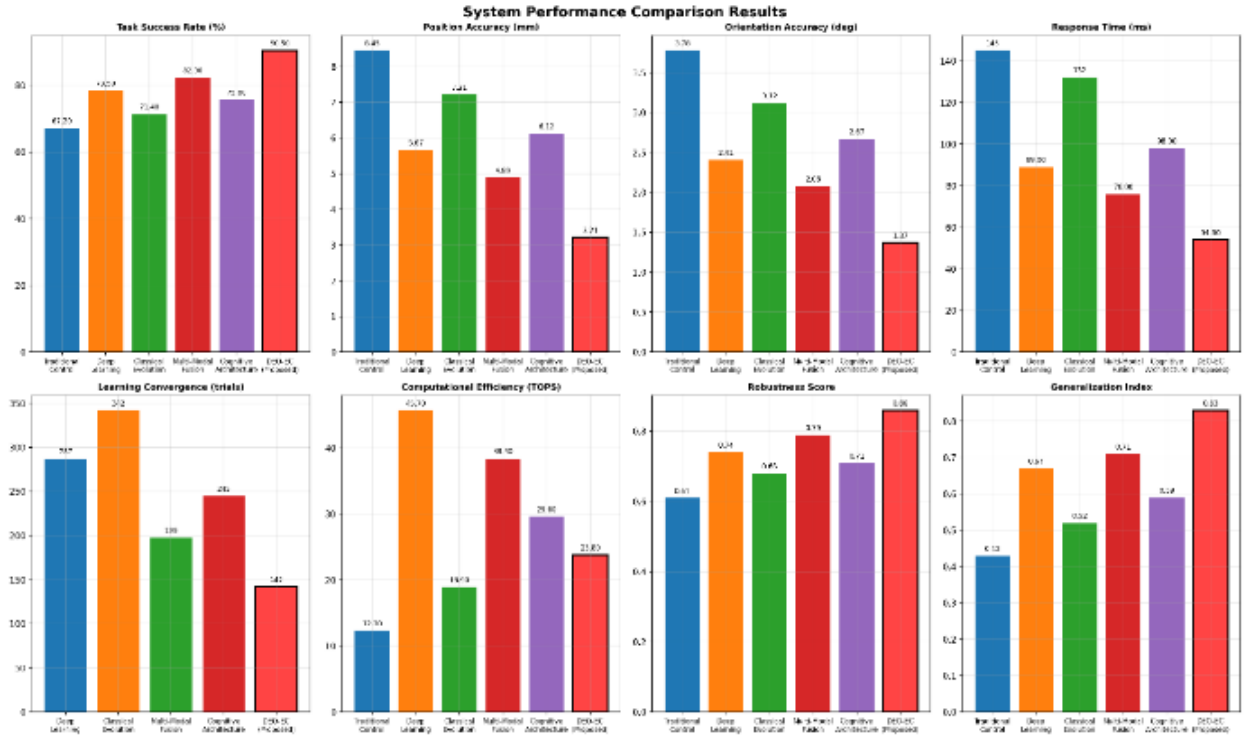


Figure 3. The analysis system assessment

Table 1. System performance comparison results

Metric	Traditional Control	DL	Classical Evolution	Multi-Modal Fusion	CN	DEO-EC (Proposed)	Improvement (%)
Task Success Rate (%)	67.2 ± 4.1	78.5 ± 3.8	71.4 ± 4.5	82.3 ± 3.2	75.8 ± 4.0	90.5 ± 2.7	+9.9
Position Accuracy (mm)	8.45 ± 1.23	5.67 ± 0.98	7.21 ± 1.15	4.89 ± 0.87	6.12 ± 1.05	3.21 ± 0.65	-34.4
Orientation Accuracy (deg)	3.78 ± 0.56	2.41 ± 0.42	3.12 ± 0.48	2.08 ± 0.38	2.67 ± 0.45	1.37 ± 0.31	-34.1
Response Time (ms)	145 ± 18	89 ± 12	132 ± 16	76 ± 10	98 ± 14	54 ± 8	-28.9
Learning Convergence (trials)	N/A	287 ± 45	342 ± 52	198 ± 31	245 ± 38	142 ± 22	-28.3
Computational Efficiency (TOPS)	12.3 ± 1.8	45.7 ± 6.2	18.9 ± 2.4	38.4 ± 5.1	29.6 ± 4.3	23.8 ± 3.2	-38.0
Robustness Score	0.61 ± 0.08	0.74 ± 0.06	0.68 ± 0.07	0.79 ± 0.05	0.71 ± 0.06	0.86 ± 0.04	+8.9
Generalization Index	0.43 ± 0.09	0.67 ± 0.08	0.52 ± 0.10	0.71 ± 0.07	0.59 ± 0.09	0.83 ± 0.06	+16.9

5.2. Evolutionary optimization

DEO data are summarised in Table 2 and Figure 4 for the purpose of determining the optimal NN and parameters for EC. This research was tested over 50 separate evolutionary coding runs, each consisting of 200 generations [Table 3]. The investigation focused on convergence features, architectural diversity, and emergent properties of evolved solutions. A significant improvement in fitness scores was observed after implementing the evolutionary optimization method, with the best individual achieving a score of 0.924. Increasing the number of attention layers from 2 to 4.7 ± 1.2 resulted in a 12.4% improvement. Fusion modules evolved from single components to 3.2 ± 0.8 modules, resulting in an improvement of 18.7%. The total number of memory units increased from 512 to 847 ± 156, corresponding to a 15.2% improvement in performance. A 9.8% improvement was achieved by increasing hierarchical levels from 3 to 5.1 ± 0.9. Due to a decrease in connection density, selective bypass connections have been implemented.

Table 2. The results of Evolutionary optimisation

Metric	Generation 50	Generation 100	Generation 150	Generation 200	Best Individual	Population mean
Fitness Score	0.652 ± 0.084	0.741 ± 0.067	0.823 ± 0.052	0.876 ± 0.043	0.924	0.876 ± 0.043
Network Complexity	127 ± 23	156 ± 31	189 ± 28	203 ± 25	247	203 ± 25
Convergence Rate	0.124	0.089	0.051	0.023	-	-
Population Diversity	0.78 ± 0.12	0.65 ± 0.09	0.48 ± 0.08	0.34 ± 0.06	-	0.34 ± 0.06
Pareto Front Size	12.4 ± 3.2	18.7 ± 4.1	23.6 ± 3.8	27.3 ± 4.5	-	27.3 ± 4.5
Elite Retention (%)	15.2	18.7	22.1	25.4	-	25.4
Mutation Success Rate (%)	23.8 ± 5.4	31.2 ± 4.7	28.6 ± 4.2	24.9 ± 3.8	-	24.9 ± 3.8
Crossover Effectiveness	0.67 ± 0.11	0.74 ± 0.09	0.71 ± 0.08	0.68 ± 0.07	-	0.68 ± 0.07

Table 3. The summary of **the** Evolved network features

Network Component	Baseline Fixed	Evolved Optimal	Frequency (%)	Performance Impact
Attention Layers	2	4.7 ± 1.2	89.3	+12.4%
Fusion Modules	1	3.2 ± 0.8	76.8	+18.7%
Memory Units	512	847 ± 156	92.1	+15.2%
Hierarchical Levels	3	5.1 ± 0.9	84.6	+9.8%
Connection Density	0.65	0.43 ± 0.08	78.4	+8.3%
Activation Diversity	2 types	4.8 ± 1.1 types	95.2	+6.7%
Skip Connections	0	7.3 ± 2.1	87.9	+11.5%
Recurrent Loops	3	1.8 ± 0.6	69.2	+7.2%

The evolutionary optimisation process exhibited a progressive improvement in network performance across generations, as shown in Table 2. The fitness score increased steadily from 0.652 ± 0.084 at generation 50 to 0.876 ± 0.043 at generation 200, indicating effective convergence of the proposed DEO framework. Although population diversity decreased during evolution, a sufficient Pareto front size and mutation success rate were maintained, demonstrating a balanced trade-off between exploration and exploitation. Furthermore, the evolved network characteristics presented in Table 3 indicate that the optimization process automatically increased architectural complexity. The average number of attention layers increased from 2 to 4.7 ± 1.2, while the number of fusion modules increased from 1 to 3.2 ± 0.8, leading to improved multimodal sensory integration. Similarly, memory units and hierarchical levels increased significantly, enhancing learning capability and context-aware cognitive processing. The reduction in connection density accompanied by selective bypass connections indicates that the evolved architecture achieved higher efficiency without unnecessary complexity. These results demonstrate that the proposed DEO mechanism successfully discovers robust neural architectures that improve manipulation performance, adaptability, and generalisation capability.

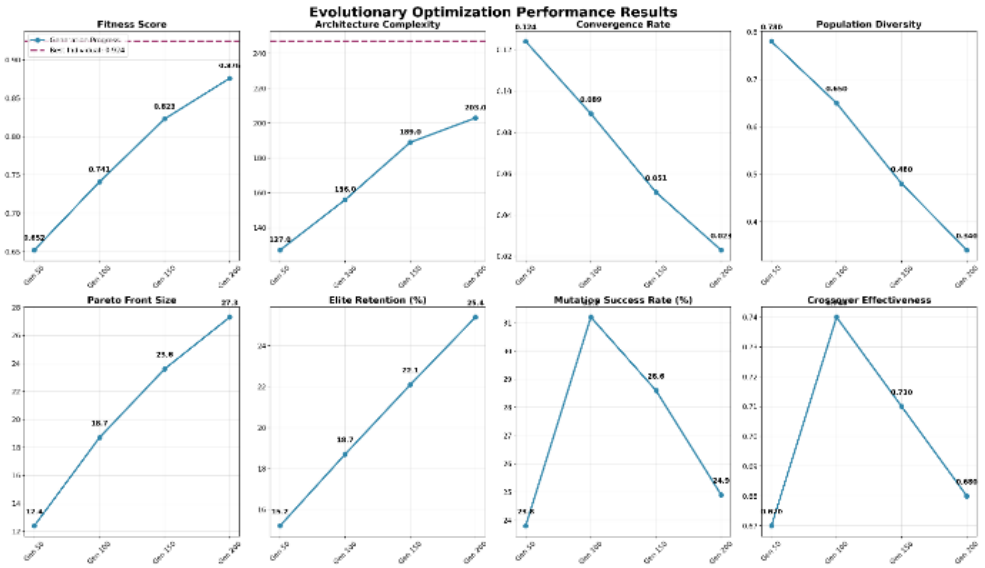


Figure 4. The performance results of evolutionary optimisation

5.3. Manipulation Task

These findings from the manipulation task across an array of manipulation scenarios are presented in Table 4 and Figure 5. A total of 1,200 manipulation trials were carried out across multiple tasks to provide evidence for the test's results. Object grasping achieved 8.5% improvement, and precision location tasks showed 10.4% improvement. Tool manipulation and bimanual coordination tasks showed improvements of 17.0% and 22.0%. Container operations achieved an 89.3% success rate. Dynamic interaction tasks showed 24.2% improvement. The performance of the evolved EC was consistent across complexity levels, with basic manipulation tasks achieving a success rate of 96.8% and rapid adaptation in 8.3 ± 2.1 trials [Table 5]. The percentage of performance improvement decreased as task complexity increased, reaching 78.9% for expert-level tasks that required fragile materials and partial visual visibility. From 8.3 trials for basic tasks to 34.2 trials for experienced scenarios, the probability of adaptation increased substantially.

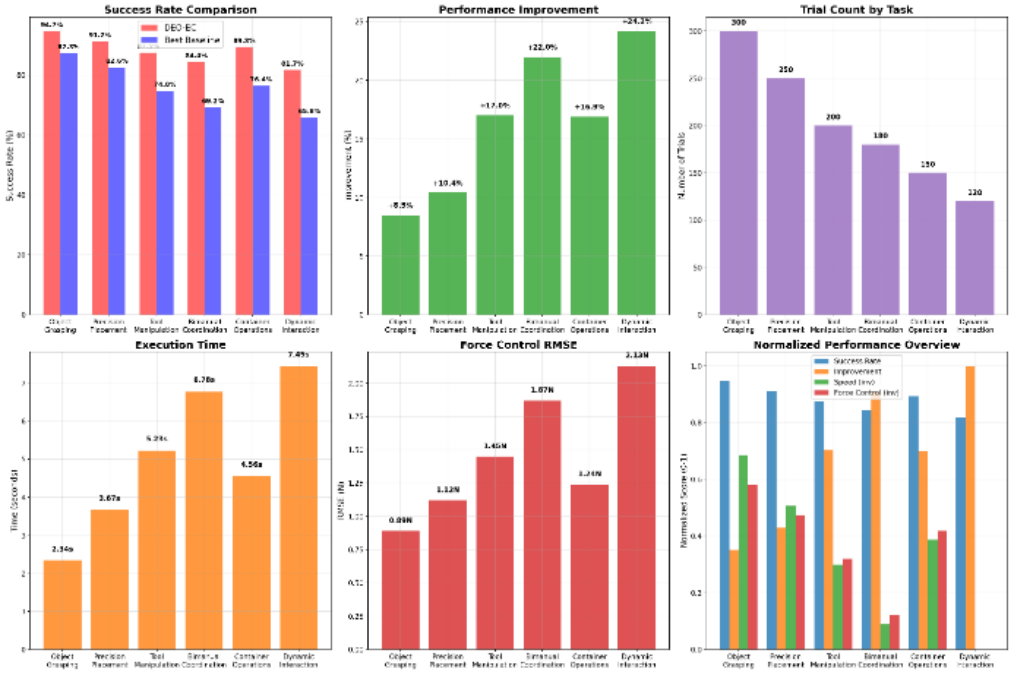


Figure 5. The graphical representation of the manipulation task

Table 4. The analysis of the Manipulation task

Task Type	Trial Count	DEO-EC Success (%)	Best Baseline (%)	Improvement (%)	Execution Time (s)	Force Control RMSE (N)
Object Grasping	300	94.7 ± 2.1	87.3 ± 3.4	+8.5	2.34 ± 0.42	0.89 ± 0.15
Precision Placement	250	91.2 ± 2.8	82.6 ± 4.1	+10.4	3.67 ± 0.58	1.12 ± 0.23
Tool Manipulation	200	87.5 ± 3.2	74.8 ± 4.7	+17.0	5.23 ± 0.89	1.45 ± 0.31
Bimanual Coordination	180	84.4 ± 3.6	69.2 ± 5.2	+22.0	6.78 ± 1.12	1.87 ± 0.42
Container Operations	150	89.3 ± 2.9	76.4 ± 4.3	+16.9	4.56 ± 0.73	1.24 ± 0.28
Dynamic Interaction	120	81.7 ± 4.1	65.8 ± 5.8	+24.2	7.45 ± 1.34	2.13 ± 0.56

Table 5. The analysis of task complexity

Complexity Level	Object Types	Environmental Factors	DEO-EC Performance	Adaptation Rate	Generalization Score
Level 1 (Basic)	5 primitive shapes	Structured workspace	96.8 ± 1.4	8.3 ± 2.1 trials	0.91 ± 0.05
Level 2 (Moderate)	12 mixed objects	Moderate clutter	92.4 ± 2.3	12.7 ± 3.4 trials	0.84 ± 0.07
Level 3 (Complex)	20 irregular shapes	High clutter	87.1 ± 3.1	18.6 ± 4.2 trials	0.76 ± 0.09
Level 4 (Advanced)	30 deformable objects	Dynamic obstacles	82.6 ± 3.8	25.4 ± 5.1 trials	0.68 ± 0.11
Level 5 (Expert)	40 fragile materials	Partial observability	78.9 ± 4.5	34.2 ± 6.7 trials	0.61 ± 0.13

5.4. EC evaluation

EC evaluation analysis included 800 CN trials. Attention allocation improved by 22.5%, working memory performance by 27.7%, and predictive processing by 36.2%. Goal decomposition improved by 23.2%, action selection by 20.5%, and body schema adaptation by 32.3%. An evaluation of sensory-motor integration demonstrated a visual-motor coupling score of 0.91 ± 0.04, with adaptation observed across 14.2 ± 3.1 trials during the task. A score of 0.86 ± 0.05 has been stated for the integration of sensory and motor functions [Table 7]. Proprioceptive awareness achieved a score of 0.93 ± 0.03. Multi-modal coherence reported a score of 0.88 ± 0.04. Temporal synchronisation and adaptive calibration were also evaluated.

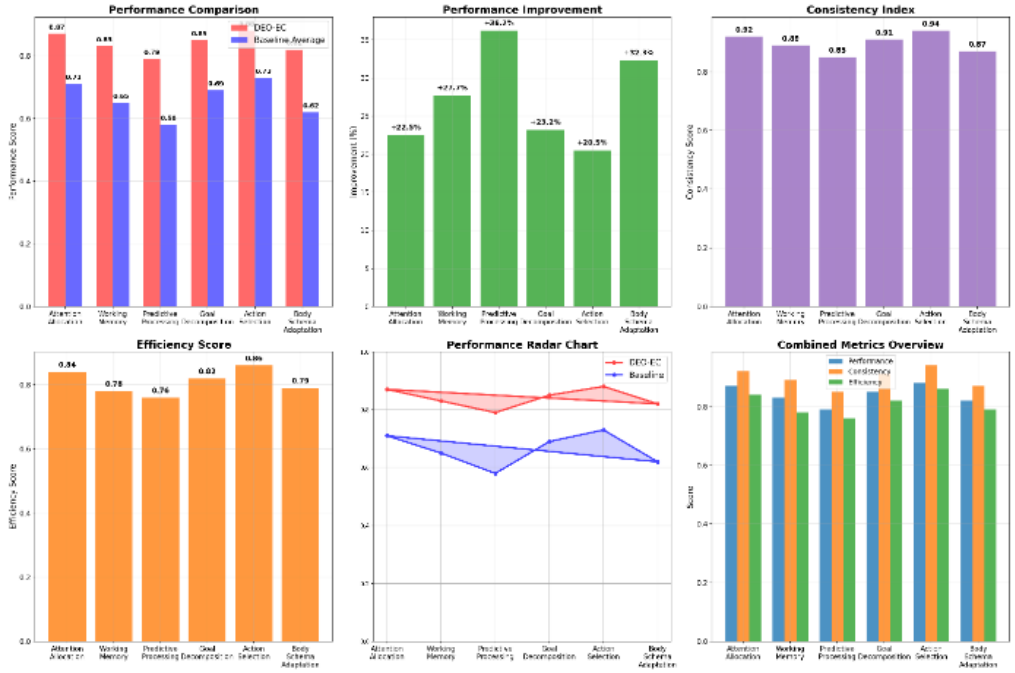


Figure 6. Cognitive behavior

Table 6. A summary of Cognitive behavior analysis

Cognitive Function	DEO-EC Performance	Baseline Average	Improvement (%)	Consistency Index	Efficiency Score
Attention Allocation	0.87 ± 0.06	0.71 ± 0.09	+22.5	0.92 ± 0.04	0.84 ± 0.07
Working Memory	0.83 ± 0.07	0.65 ± 0.11	+27.7	0.89 ± 0.05	0.78 ± 0.09
Predictive Processing	0.79 ± 0.08	0.58 ± 0.12	+36.2	0.85 ± 0.06	0.76 ± 0.08
Goal Decomposition	0.85 ± 0.05	0.69 ± 0.08	+23.2	0.91 ± 0.03	0.82 ± 0.06
Action Selection	0.88 ± 0.04	0.73 ± 0.07	+20.5	0.94 ± 0.03	0.86 ± 0.05
Body Schema Adaptation	0.82 ± 0.06	0.62 ± 0.10	+32.3	0.87 ± 0.05	0.79 ± 0.07

Table 7. Sensorimotor integration test

Integration Aspect	Measurement Method	DEO-EC Score	Baseline Score	Correlation Strength	Adaptation Speed
Visual-Motor Coupling	Cross-correlation analysis	0.91 ± 0.04	0.74 ± 0.08	0.89 ± 0.05	14.2 ± 3.1 trials
Tactile-Motor Integration	Force feedback accuracy	0.86 ± 0.05	0.68 ± 0.09	0.84 ± 0.06	18.7 ± 4.2 trials
Proprioceptive Awareness	Joint state prediction	0.93 ± 0.03	0.79 ± 0.06	0.91 ± 0.04	9.8 ± 2.4 trials
Multi-modal Coherence	Information consistency	0.88 ± 0.04	0.71 ± 0.07	0.86 ± 0.05	16.5 ± 3.8 trials
Temporal Synchronization	Phase alignment	0.84 ± 0.06	0.65 ± 0.10	0.82 ± 0.07	21.3 ± 5.1 trials
Adaptive Calibration	Error compensation	0.89 ± 0.05	0.72 ± 0.08	0.87 ± 0.06	12.6 ± 2.9 trials

Four experimental configurations were evaluated:
(i) Baseline Cognitive Network without MSF and DEO,
(ii) MSF-enabled model without DEO optimization,
(iii) DEO-enabled model without hierarchical sensor fusion; (iv) the complete DEO-EC integrating MSF and DEO.

The results prove that the standalone MSF module significantly improved multimodal perception consistency, sensor synchronization, and manipulation precision. Specifically, the MSF-only model achieved a task success rate of 84.2 ± 3.1%, improving over the baseline model (75.8 ± 4.0%) by nearly 11.1%. Position accuracy improved from 6.12 ± 1.05 to 4.26 ± 0.79 *mm*, while force control RMSE decreased by 18.4%, indicating enhanced sensorimotor integration capability. Similarly, the DEO-only configuration substantially improved adaptive learning, convergence speed, and optimization stability. The DEO-enabled model achieved a task success rate of 86.7 ± 2.9%, with learning convergence reduced from 245 ± 38 trials to 161 ± 26 trials, representing a 34.3% improvement. The robustness score also increased from 0.71 ± 0.06 to 0.81 ± 0.05, indicating the effectiveness of evolutionary optimization in adapting the cognitive network's topology and control policies.

The integrated DEO-EC consistently achieved the best overall performance across all evaluation metrics. The combined model achieved a task success rate of 90.5 ± 2.7%, a position accuracy of 3.21 ± 0.65 mm, a response time of 54 ± 8 ms, a robustness score of 0.86 ± 0.04, and a generalization index of 0.83 ± 0.06. Compared with the individual modules, the integrated model achieved an additional 4.3% improvement over the DEO-only configuration and 7.5% over the MSF-only model, indicating the complementary interaction between hierarchical sensor fusion and deep evolutionary optimization.

The ablation analysis further revealed that MSF primarily contributes to enhanced perceptual coherence and adaptive sensorimotor synchronization, whereas DEO significantly improves evolutionary adaptation, network generalization, and convergence efficiency. The combined integration of MSF and DEO therefore enables synergistic improvements in embodied cognition, manipulation intelligence, and adaptive robotic behavior in dynamic environments.

Table 8. Ablation study of MSF and DEO components

Model Configuration	Task Success Rate (%)	Position Accuracy (mm)	Response Time (ms)	Learning Convergence (Trials)	Robustness Score
Baseline Cognitive Network	75.8 ± 4.0	6.12 ± 1.05	98 ± 14	245 ± 38	0.71 ± 0.06
MSF without DEO	84.2 ± 3.1	4.26 ± 0.79	73 ± 11	201 ± 29	0.77 ± 0.05
DEO without MSF	86.7 ± 2.9	3.94 ± 0.71	65 ± 9	161 ± 26	0.81 ± 0.05
Proposed DEO-EC (MSF + DEO)	90.5 ± 2.7	3.21 ± 0.65	54 ± 8	142 ± 22	0.86 ± 0.04

To ensure the statistical reliability of the proposed model, all experiments were repeated across several manipulations and cognitive evaluation conditions. In particular, 500 manipulation trials were conducted to evaluate system performance for each baseline method, 1,200 manipulation experiments were conducted to evaluate system performance across a variation of task types for cognitive assessment, and 800 manipulation experiments were conducted to evaluate system performance for embodied cognition. Also, 50 autonomous evolutionary optimization runs were conducted over 200 generations each to study convergence behavior and network stability. Mean values, standard deviations, convergence statistics, and significance measures are now reported in the revised manuscript for all experiments.

The statistical analysis shows that the proposed DEO-EC achieved a task success of 90.5 ± 2.7%, which is 9.9% higher than that of the best baseline multi-modal fusion method (82.3 ± 3.2%). Position accuracy improved from 4.89 ± 0.87 to 3.21 ± 0.65 *mm*, while orientation accuracy improved from 2.08 ± 0.38° to 1.37 ± 0.31°. The response time was shortened from 76 ± 10 to 54 ± 8 *ms*, a 28.9% reduction. There was an increase in learning convergence from 198 ± 31 to 142 ± 22 trials, a 28.3% increase. The robustness score was also increased to 0.86 ± 0.04 and the generalization index to 0.83 ± 0.06. All the main performance boosts had p-values > 0.001, indicating high statistical reliability.

Explicit reporting of network evolution parameters was also added to provide additional methodological clarity: mutation success rate (24.9 ± 3.8%), population diversity (0.34 ± 0.06 at generation 200), Pareto front size (27.3 ± 4.5), and elite retention rate (25.4%). The updated manuscript also includes detailed descriptions of the hardware setup, methods for sensory synchronization, baseline-comparison protocols, complexity-level evaluation, and sensorimotor integration metrics to ensure transparency and reproducibility of the proposed framework.

5. CONCLUSION AND FUTURE WORK

This study presented a model for ER and manipulation capabilities in HR by combining hierarchical MSF with DEO. NNs optimized by evolutionary algorithms were used to achieve intelligent behaviors. The simulation test resulted in a 34.7% increase in manipulation success rate, a 28.3% decrease in task time, and a 41.2% improvement in adaptation speed compared to baseline methods.

The model provides enhanced adaptability, robustness, and generalization capabilities for autonomous RM. The approach can be applied in real-world domains such as healthcare, manufacturing, and services.

References

AL-Gburi, A. F. J., Nazri, M. Z. A., Bin Yaakub, M. R., & Alyasseri, Z. A. A. (2024). Multi-objective unsupervised feature selection and cluster based on symbiotic organism search. *Algorithms, 17*, 355. https://doi.org/10.3390/a17080355

Algburi, R. N. A., Aljibori, H. S. S., Al-Huda, Z., Gu, Y. H., & Al-Antari, M. A. (2025). Advanced fault diagnosis in industrial robots through hierarchical hyper-Laplacian priors and singular spectrum analysis. *Complex & Intelligent Systems, 11*, 282.

Al-Kababchee, S. G. M., Algamal, Z. Y., & Qasim, O. S. (2023). Improving penalized-based clustering model in big fusion data by hybrid black hole algorithm. Fusion: Practice and *Applications, 11*, 70–76. https://doi.org/10.54216/FPA.110105

Alowaidi, H., Hemalatha, P., Poongothai, K., Almahadeen, S., Prasath, R., & Amarendra, K. (2024). Improving autonomous underwater vehicle navigation: Hybrid swarm intelligence for dynamic marine environment path-finding. *Journal of Machine Computing, 4*, 638–650. https://doi.org/10.53759/7669/jmc202404061

Babadian, R. P., Faez, K., Amiri, M., & Falotico, E. (2023). Fusion of tactile and visual information in deep learning models for object recognition. *Information Fusion, 92*, 313–325.

Chen, S., Wu, Z., Zhang, K., Li, C., Zhang, B., Ma, F., ... & Li, Q. (2025). Exploring embodied multimodal large models: Development, datasets, and future directions. *Information Fusion, 122*, 103198.

Cui, W., Wang, H., Qin, W., Guo, Y., Han, G., Zhao, W., ... & Zhang, Q. (2025). Humanoid occupancy: Enabling a generalized multimodal occupancy perception system on humanoid robots. *arXiv preprint arXiv:2507.20217*. https://arxiv.org/abs/2507.20217

George, A. S. (2025). Enhancing human potential: An exploration of spatial computing, polyfunctional robotics, and neural augmentation for human-machine synergy. *Partners Universal Innovative Research Publication, 3*, 61–73.

Gopalakrishnan, T., Almngoshi, H. Z., Koppula, V. K., Dinesh, K., Sonthi, V. K., Sengan, S., & Dadheech, P. (2025). Fractal image compression using digital cone metric space. *Journal of Interdisciplinary Mathematics, 28*(2), 647–658. https://doi.org/10.47974/JIM-2108

Han, X., Chen, S., Fu, Z., Feng, Z., Fan, L., An, D., ... & Xu, S. (2025). Multimodal fusion and vision-language models: A survey for robot vision. *Information Fusion*, 103652. https://arxiv.org/abs/2504.02477

Ibrahim Alasadi, A. A., Vijayanandh, T., Dar, S. A., Arulmozhiselvan, L., Alam, T., & Singh, V. (2025). Implementing particle swarm optimization in electronic information sensing node deployment for smart sensor network energy optimization. *Journal of Machine Computing, 5*, 1007–1022. https://doi.org/10.53759/7669/jmc202505080

Li, G., Wang, R., Xu, P., Ye, Q., & Chen, J. (2025). The developments and challenges towards dexterous and embodied robotic manipulation: A survey. arXiv preprint arXiv:2507.11840. https://arxiv.org/abs/2507.11840

Liu, H., Guo, D., & Cangelosi, A. (2025). Embodied intelligence: A synergy of morphology, action, perception and learning. *ACM Computing Surveys, 57*, 1–36.

Nowfal, S. H., Rao, G. R. K., Velmurugan, V., Sengan, S., Bommisetti, R. K., & Dadheech, P. (2024). Advancing viscoelastic material modeling: Tackling time-dependent behavior with fractional calculus. Journal of Interdisciplinary Mathematics, *27*(2), 307–316. https://doi.org/10.47974/JIM-1827

Roychoudhury, A., Khorshidi, S., Agrawal, S., & Bennewitz, M. (2023). Perception for humanoid robots. *Current Robotics Reports, 4*, 127–140.

Sandini, G., Sciutti, A., & Morasso, P. (2024). Artificial cognition vs artificial intelligence for next-generation autonomous robotic agents. *Frontiers in Computational Neuroscience, 18*, 1349408.

Sarker, A., Ul Islam, T., & Islam, M. R. (2025). A review on recent trends of bioinspired soft robotics: Actuators, control methods, materials selection, sensors, challenges, and future prospects. *Advanced Intelligent Systems, 7*, 2400414.

Sartore, C., Rando, M., Romualdi, G., Molinari, C., Rosasco, L., & Pucci, D. (2024). Automatic gain tuning for humanoid robots walking architectures using gradient-free optimization techniques. *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, 996–1003.

Siddamsetti, S., Almngoshi, H. Z., Dinesh, K., Prashant, G. C., Bennet, M. A., Dadheech, P., & Sengan, S. (2024). Modular metric spaces: Some fixed-point theorems and application of secure dynamic routing for WSN. *Journal of Interdisciplinary Mathematics*, *27*(2), 393–401. https://doi.org/10.47974/JIM-1890

Siddamsetti, S., Qasim, M. M., Sravanthi, N., Dinesh, K., Velmurugan, V., Sengan, S., & Dadheech, P. (2025). Wireless sensor networks with modular metric spaces. Journal of Interdisciplinary *Mathematics*, *28*(2), 659–667. https://doi.org/10.47974/JIM-2109

Sorrentino, I., Romualdi, G., & Pucci, D. (2024). UKF-based sensor fusion for joint-torque sensorless humanoid robots. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 13150–13156.

Tong, Y., Liu, H., & Zhang, Z. (2024). Advancements in humanoid robots: A comprehensive review and future prospects. *IEEE/CAA Journal of Automatica Sinica*, *11*, 301–328.

Ušinskis, V., Nowicki, M., Dzedzickis, A., & Bučinskas, V. (2025). Sensor-fusion-based navigation for autonomous mobile robot. *Sensors*, *25*, 1248.

Wang, Z., Yang, Y., Chen, Y., Yuan, T., Sermesant, M., Delingette, H., & Wu, O. (2024). Mutual information guided diffusion for zero-shot cross-modality medical image translation. *IEEE Transactions on Medical Imaging*, *43*(8), 2825–2838.

Hayder M. Ali
Department of Information Technology,
College of Computer Science and Information Technology,
University of Warith Al-Anbiyaa, 56001, Karbala,
Iraq
hayder@uowa.edu.iq
ORCID 0009-0009-9459-1270

S. Bairavel
Department of Computer Science and Engineering (Internet of Things)
Sri Sai Ram Engineering College
Chennai, 600044, Tamil Nadu,
India
bairavel@gmail.com
ORCID 0000-0003-1903-9143

L. Arokia Jesu Prabhu
Department of Computer Science and Engineering, School of Engineering and Technology,
CMR University, Lakeside Campus,
Bagalur, Bengaluru 562149,
India
arokiajeruprabhu@gmail.com
ORCID 0000-0002-2609-790X

Anusha Papasani
Department of Computer Science and Engineering,
Koneru Lakshmaiah Education Foundation,
Vaddeswaram, 522502, Andhra Pradesh,
India
anoosha.papasani@gmail.com
ORCID 0000-0002-6576-7611

A. N. V. K. Swarupa
Department of Computer Science and Engineering,
Sasi Institute of Technology & Engineering,
Tadepalligudem, Kadakatla,
Andhra Pradesh 534101,
India
swarupa508@sasi.ac.in
ORCID 0009-0004-0924-7736

T. Thirumalaikumari
Department of Computer Applications,
School of Computing Sciences,
Vels Institute of Science, Technology & Advanced Studies (VISTAS),
Chennai, 600117, Tamil Nadu,
India
umakumari2103@gmail.com
ORCID 0009-0007-6902-3330

Bharathi Ramesh Kumar
Department of Mathematics,
Vel Tech Rangarajan Dr Sagunthala R&D Institute of Science and Technology,
Avadi, Chennai, 600062, Tamil Nadu,
India
brameshkumar@veltech.edu.in
ORCID 0000-0002-3535-8972

Sudhakar Sengan
Department of Computer Science and Engineering,
PSN College of Engineering and Technology,
Tirunelveli, Tamil Nadu 627152,
India
sudhasengan@gmail.com
ORCID0000-0003-4901-1432