Robotics and Autonomous Systems

Robotics and Autonomous Systems: NeuroEvolution of Dynamic Neural Field Architectures for Human-Robot Joint Action

--Manuscript Draft--

Manuscript Number:	ROBOT-D-26-01378
Article Type:	Research Paper
Keywords:	Dynamic Neural Fields; Joint Action; Human–Robot Interaction; Cognitive Robotics; Neuroevolution; Bio-inspired Control
Corresponding Author:	João G. Cunha, M.D, University of Minho PORTUGAL
First Author:	João G. Cunha
Order of Authors:	João G. Cunha
	Raymond H. Cuijpers
	Wolfram Erlhagen
	Estela Bicho
Abstract:	Achieving fluent and adaptive Human–Robot Interaction (HRI) demands control architectures that generate behaviour through continuous sensorimotor dynamics. Dynamic Neural Fields (DNFs) offer a biologically inspired framework for modelling perception, memory, and action selection, yet designing functional neural-field controllers remains largely manual, limiting their scalability and broader adoption in adaptive robotic systems. Here, we introduce an approach for the automatic synthesis of DNF controllers capable of generating joint-action behaviours. Extending the NeuroEvolution of Augmenting Topologies (NEAT) algorithm to operate on neural-field-based controllers, the approach evolves both intrinsic field dynamics and architectural topology through mutation, crossover, and speciation. We evaluate it in a collaborative packaging task designed to isolate three fundamental joint-action behaviours (complementary action selection, cooperative assistance, and action inhibition) and whose minimal controller can be derived analytically, providing a ground-truth reference for assessing evolved solutions. Across 100 independent runs, the approach discovered viable controllers in 97 cases. The evolved architectures were topologically minimal, with interpretable activation dynamics grounded in known Dynamic Field Theory (DFT) mechanisms. The highest-performing minimal controller was deployed directly to a physical robot without parameter retuning, generalising to object arrangements beyond those used during evolution. Evolution converged on the minimal architecture without any specification of intermediate structure, indicating that behavioural requirements can drive the emergence of the internal processing structure the task requires. These findings demonstrate that functional DNF architectures for joint action can be evolved rather than hand-designed, providing a constructive pathway toward bio-inspired, automatically synthesised control for adaptive HRI.

Highlights

NeuroEvolution of Dynamic Neural Field Architectures for Human-Robot Joint Action

João G. Cunha, Raymond H. Cuijpers, Wolfram Erlhagen, Estela Bicho

- Neuroevolution discovers neural-field controllers for human-robot joint action
- Evolved controllers are topologically minimal with interpretable field dynamics
- Evolved controller transfers to a physical robot without parameter retuning

Highlights

NeuroEvolution of Dynamic Neural Field Architectures for Human-Robot Joint Action

João G. Cunha, Raymond H. Cuijpers, Wolfram Erlhagen, Estela Bicho

- Neuroevolution discovers neural-field controllers for human-robot joint action
- Evolved controllers are topologically minimal with interpretable field dynamics
- Evolved controller transfers to a physical robot without parameter retuning

NeuroEvolution of Dynamic Neural Field Architectures for Human-Robot Joint Action*

João G. Cunha[a,*], Raymond H. Cuijpers[b], Wolfram Erlhagen[c] and Estela Bicho[a]

[a]Centre Algoritmi, University of Minho, Campus de Azurém, Av. da Universidade, Guimarães 4800-058, Portugal
[c]Centre of Mathematics, University of Minho, Campus de Azurém, Av. da Universidade, Guimarães 4800-058, Portugal
[b]Human-Technology Interaction, Eindhoven University of Technology, Het Eeuwsel 53, Eindhoven 5612 AZ, The Netherlands

ARTICLE INFO

Keywords:
Dynamic Neural Fields
Joint Action
Human–Robot Interaction
Cognitive Robotics
Neuroevolution
Bio-inspired Control

ABSTRACT

Achieving fluent and adaptive Human–Robot Interaction (HRI) demands control architectures that generate behaviour through continuous sensorimotor dynamics. Dynamic Neural Fields (DNFs) offer a biologically inspired framework for modelling perception, memory, and action selection, yet designing functional neural-field controllers remains largely manual, limiting their scalability and broader adoption in adaptive robotic systems. Here, we introduce an approach for the automatic synthesis of DNF controllers capable of generating joint-action behaviours. Extending the NeuroEvolution of Augmenting Topologies (NEAT) algorithm to operate on neural-field-based controllers, the approach evolves both intrinsic field dynamics and architectural topology through mutation, crossover, and speciation. We evaluate it in a collaborative packaging task designed to isolate three fundamental joint-action behaviours (complementary action selection, cooperative assistance, and action inhibition) and whose minimal controller can be derived analytically, providing a ground-truth reference for assessing evolved solutions. Across 100 independent runs, the approach discovered viable controllers in 97 cases. The evolved architectures were topologically minimal, with interpretable activation dynamics grounded in known Dynamic Field Theory (DFT) mechanisms. The highest-performing minimal controller was deployed directly to a physical robot without parameter retuning, generalising to object arrangements beyond those used during evolution. Evolution converged on the minimal architecture without any specification of intermediate structure, indicating that behavioural requirements can drive the emergence of the internal processing structure the task requires. These findings demonstrate that functional DNF architectures for joint action can be evolved rather than hand-designed, providing a constructive pathway toward bio-inspired, automatically synthesised control for adaptive HRI.

1. Introduction

Fluent Human–Robot Collaboration (HRC) requires robots that can continuously adapt their behaviour to a human partner's actions, selecting when to help, when to act independently, and when not to interfere. Building controllers that support this kind of real-time coordination remains one of the central challenges in robotics Selvaggio, Cognetti, Nikolaidis, Ivaldi and Siciliano (2021); Billard, Albu-Schaeffer, Beetz, Burgard, Corke, Ciocarlie, Dahiya, Kragic, Goldberg, Nagai and Scaramuzza (2025); Amato, Hutchinson, Garg, Billard, Rus, Tedrake, Park and Goldberg (2025). Meeting this challenge requires architectures that tightly couple perception, memory, and action selection into coherent sensorimotor dynamics, operating under the time constraints and uncertainty of shared interaction Vernon (2025). The field of neurorobotics has emerged as a promising pathway toward intelligent cognitive robots by embedding biologically inspired models of neural computation directly into robotic systems Pfeifer and Bongard (2006); Krichmar (2018); Mompó Alepuz, Papageorgiou and Tolu (2024); Sheng, Nie, Liu, Zhang, Zhou, Liu and Liu (2025).

Dynamic Field Theory (DFT) offers a principled, biologically grounded framework for this kind of bio-inspired embodied coordination Bicho, Mallet and Schöner (2000); Coombes, Beim Graben, Potthast and Wright (2014); Schöner, Spencer and Research Group (2015). Its elementary building blocks, Dynamic Neural Fields (DNFs), describe the temporal evolution of population-level neural activity across continuous feature dimensions such as space or orientation. Stabilised activation bumps in these fields form dynamic representations of task-relevant features, supporting processes such as selective attention, working memory, and decision-making. By coupling fields through structured interactions, complex architectures can be constructed in which the activity of one field shapes the input to another. Such DNF-based architectures have been successfully deployed in robotic systems, enabling joint action with humans, adaptive decision-making, and embodied Human-Robot Interaction (HRI) Schöner, Dose and Engels (1995); Engels and Schöner (1995); Erlhagen and Bicho (2006); Sandamirskaya, Zibner, Schneegans and Schöner (2013); Wojtak, Ferreira, Louro, Bicho and Erlhagen (2023); Chatzistaparaschis, Zhong, Christopoulos and Karydis (2024).

Despite these advances, constructing functional DNF controllers remains a largely manual process. Practitioners must manually specify the architectural organisation of the

*This work was financed by FCT (Portuguese Foundation for Science and Technology) through the R&D Unit Project Scope UID/00319/2025 - Centro ALGORITMI (ALGORITMI/UM) and PhD Fellowship under grant 2022.13948.BD.

*Corresponding author.
id10787@uminho.pt (J.G. Cunha); r.h.cuijpers@tue.nl (R.H. Cuijpers); wolfram.erlhagen@math.uminho.pt (W. Erlhagen); estela.bicho@dei.uminho.pt (E. Bicho)
ORCID(s): 0000-0003-3392-224X (J.G. Cunha); 0000-0001-5980-9208 (R.H. Cuijpers); 0000-0002-8150-3587 (W. Erlhagen); 0000-0003-2715-4037 (E. Bicho)

controller, *i.e.*, the number of fields and their connectivity, the functional role of each field, and the intrinsic parameters governing its dynamics (time constants, resting levels, kernel profiles). Because DNF dynamics are highly non-linear and sensitive to parameter changes, this process is time-consuming, difficult to transfer across tasks, and demands significant domain expertise Igel, Erlhagen and Jancke (2001); Fix (2013). The absence of constructive methods for systematically generating DNF architectures limits their scalability and broader adoption in adaptive robotic systems.

Evolutionary Robotics offers a natural pathway to overcome this limitation. By applying principles of mutation, recombination, and selection, robot controllers can be automatically synthesised and adapted without manual tuning Nolfi and Floreano (2000); Floreano (2023). Among neuroevolution methods, NeuroEvolution of Augmenting Topologies (NEAT) is particularly well suited to this challenge: it begins with minimal architectures and incrementally complexifies them through structural mutations, while protecting innovations via speciation Stanley and Miikkulainen (2002). This combination of gradual elaboration and diversity preservation has made NEAT a foundational method for evolving adaptive controllers Papavasileiou, Cornelis and Jansen (2021).

Building on these two lines of work, we propose an approach for the automatic synthesis of DNF controllers capable of generating joint-action behaviours in embodied HRI. Rather than designing the architecture by hand, we evolve both the intrinsic parameters of individual fields and the topological organisation of the controller, *i.e.* the number of fields, and their connectivity, in service of a concrete collaborative task. Previous work has applied evolutionary strategies to optimise subsets of individual DNF parameters Igel et al. (2001); Schneider, Igel, Klaes, Dinse and Wiemer (2004); Fix (2013); our approach goes further by treating the architecture itself as the object of search. The approach extends NEAT to operate on DNF-based controllers rather than conventional Artificial Neural Networks (ANNs), adapting its genetic encoding, mutation operators, and compatibility metric to the continuous-time, kernel-based nature of neural field dynamics.

We evaluate the approach in a collaborative packaging task designed to isolate three fundamental joint-action behaviours: cooperative assistance, complementary action selection, and action inhibition Sebanz, Bekkering and Knoblich (2006); Bekkering, De Bruijn, Cuijpers, Newman-Norlund, Van Schie and Meulenbroek (2009); Sebanz and Knoblich (2021). The task requires the robot to continuously monitor the human partner and adapt its responses accordingly. Crucially, it cannot be solved by direct input–output inter-field kernel connections alone: the human's hand position must drive opposing responses depending on the context, suppressing the robot's action at the targeted location during complementary and inhibitory behaviour, yet promoting action there during cooperative assistance. Because a single field-to-field connection cannot be simultaneously excitatory and inhibitory, resolving this conflict requires at least one intermediate processing field.

The contributions of this work are fourfold. First, we extend NEAT to operate on DNF-based controllers, adapting its genetic encoding, mutation operators, and compatibility metric to the continuous-time, kernel-based nature of neural fields. Second, we show that this evolutionary synthesis produces functional DNF controllers for adaptive joint action with no manual specification of parameters or internal architecture. Third, because the task admits a derivable minimal solution, we show that evolution reliably discovers this minimal architecture. Fourth, we analyse the evolved field dynamics in terms of known DFT mechanisms and demonstrate direct sim-to-physical transfer without retuning.

The remainder of the paper presents the methodology (Section 2), the experimental evaluation on a collaborative packaging task (Section 3), the results (Section 4), and a discussion of findings, limitations, and future work (Section 5).

2. Methodology

Designing a DNF controller capable of joint action requires both the right internal field dynamics and the right architectural topology. Rather than specifying these by hand, we discover them automatically through an evolutionary process driven by a task-related fitness function: the designer specifies only what counts as correct behaviour, which is generally easier than directly tuning the field parameters and connectivity that produce it Floreano (2023).

The process begins by randomly initialising a population of candidate controllers, each encoded as a genome that specifies a minimal DNF architecture consisting only of input and output fields (the minimum structure needed to couple the controller to the world). Each genome is decoded into a DNF controller and evaluated in simulation: task-specific stimuli are applied to the input fields, the neural-field dynamics unfold over time, and the resulting behaviour is scored by a fitness function that measures how closely the controller satisfies the desired joint-action objectives. Controllers that better approximate the target behaviours are more likely to reproduce, passing their genetic material to a new generation. Genetic operators (crossover and mutation) act on both the parameters of existing fields and the topology of the architecture itself, tuning intrinsic field dynamics and reshaping how activation propagates between fields by introducing new connections and intermediate processing stages. This cycle repeats until a controller emerges that meets the performance criterion (Figure 1).

Three guiding principles adopted from NEAT form the base of our approach: (i) starting from minimal topologies and complexifying through structural mutation; (ii) using historical markings to align genomes during crossover; and (iii) protecting structural innovations through speciation. We extend these principles to operate on DNF-based controllers

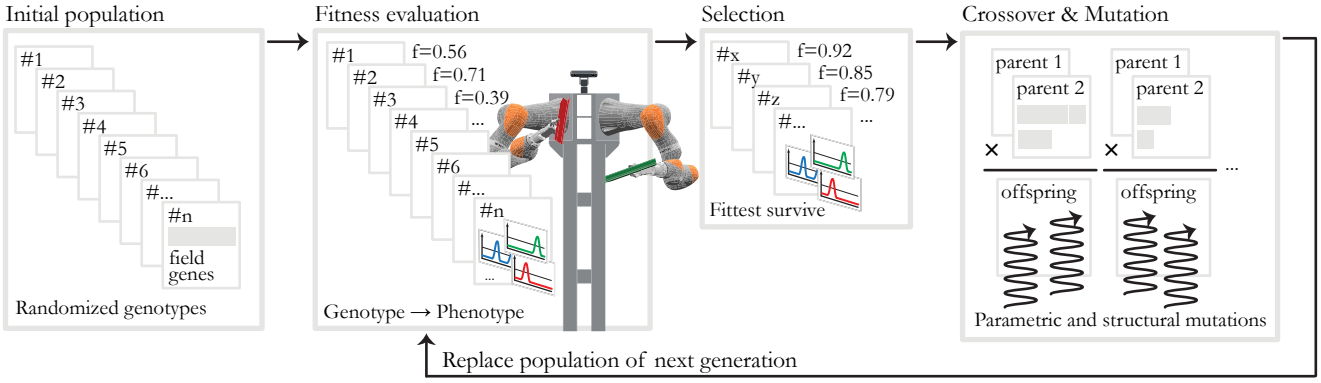


Figure 1: Evolutionary synthesis of DNF-based controllers. A population of randomly initialised genomes, each encoding a minimal DNF architecture, is iteratively evaluated, selected, and varied through crossover and mutation. Over successive generations, this process shapes both the intrinsic dynamics of individual fields and the topological organisation of the architecture, until a controller capable of generating the desired joint-action behaviour emerges.

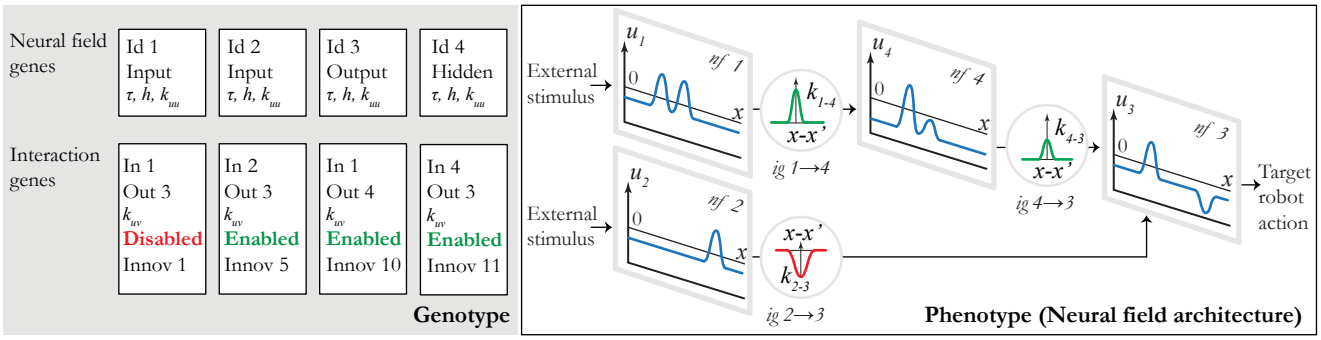


Figure 2: Genotype-to-phenotype mapping. The left panel shows the genomic encoding composed of field and interaction genes. The right panel illustrates the resulting DNF-based architecture, showing how encoded fields interact to form the controller.

rather than conventional ANNs, adapting the genetic encoding, mutation operators, and compatibility metric to the continuous-time, kernel-based nature of neural field dynamics. The following subsections detail each component. For a video explanation of the algorithm, refer to the Supplemental Video at https://youtu.be/Mvc-DXXC8kQ.

2.1. Genetic Encoding

Each candidate controller is represented as a genome composed of two types of genes: field genes and interaction genes (Figure 2). Together, these define the architecture and dynamics of the resulting DNF controller.

2.1.1. Field Genes

Field genes define the fundamental processing units of the architecture: the DNFs themselves. Each field gene encodes: (i) a unique identifier; (ii) a categorical type (input/hidden/output); and (iii) the intrinsic dynamical parameters governing the field's behaviour, including time constant (τ), resting level (h), and the type and parameters of its self-interaction kernel ($k_{uu}(x - x')$).

The temporal evolution of these activation fields is governed by interactions between neural populations, which can be excitatory or inhibitory Amari (1977):

$$\tau \dot{u}(x,t) = -u(x,t) + h + s(x,t) + \int k_{uu}(x - x')g(u(x',t))dx' + q\xi(x,t) \quad (1)$$

where $u(x,t)$ is the average membrane potential at neural position x and time t, see implementation details in Erlhagen and Bicho (2006). The resting level $h(< 0)$ sets the baseline activation of the field. $s(x,t)$ carries time-dependent, spatially localised input arriving either from external stimuli, such as detected object positions or the tracked hand location, or from other fields in the architecture. The time constant τ defines the time scale of the field dynamics. The integral term represents the recurrent interaction within the field, the lateral coupling through which active neurons influence the activation of others, and is determined by two components. The output function $g(u)$ dictates that only neurons surpassing a predetermined threshold of activation contribute to the interaction (e.g., Heaviside or sigmoid function). The self-interaction kernel $k_{uu}(x - x')$ defines the coupling strength between neural populations at positions x and x', and together with τ and h, largely determines the dynamical regime of the field. Finally, ξ is Gaussian white noise of strength q, which models stochastic neural variability and aids decision-making under ambiguous input.

When sufficiently strong input drives the field to a supra-threshold state ($u > 0$), recurrent excitation through the self-interaction kernel can generate a localised peak of activation. Such peaks may be *self-stabilised*, persisting only while external input is maintained, or *self-sustained*, persisting after the input is removed and thus providing a form of working memory. On the other hand, when input is present but insufficient to drive the field above threshold, it produces a sub-threshold peak of activation known as a *pre-shape*, which biases the field toward forming a peak at that location if additional input arrives. Which regime a field operates in depends on the balance between recurrent excitation, inhibition, and resting level (parameters that evolution is free to tune).

Two self-interaction kernel types are supported (Gaussian and Mexican-hat kernels). The Gaussian kernel with global inhibition is defined by:

$$k_{uu}(x - x') = A \exp\left(-\frac{|x - x'|^2}{\sigma^2}\right) - A_{glob} \tag{2}$$

where A and σ define the amplitude and spatial spread of local excitation, respectively, and A_{glob} defines the approximately constant inhibition for distant neurons.

The Mexican-hat kernel is parametrised as a difference of Gaussians with global inhibition:

$$k_{uu}(x - x') = A_{\text{exc}} \exp\left(-\frac{|x - x'|^2}{\sigma_{\text{exc}}^2}\right) - A_{\text{inh}} \exp\left(-\frac{|x - x'|^2}{\sigma_{\text{inh}}^2}\right) - A_{\text{glob}} \tag{3}$$

where A_{exc} and σ_{exc} define the amplitude and spread of the excitatory component, and A_{inh} and σ_{inh} those of the inhibitory component.

Gaussian kernels with global inhibition are well suited to winner-take-all selection, whereas Mexican-hat kernels, through local inhibition, more readily support multi-peak and self-sustained regimes Wojtak, Ferreira, Erlhagen and Bicho (2015); Ferreira, Wojtak, Sousa, Louro, Bicho and Erlhagen (2021); Wojtak et al. (2023). Although the precise regime a field settles into also depends on its excitation–inhibition balance and global inhibition strength, the two kernel types differ in the range of dynamics they readily afford, and no single type covers the full repertoire a multi-field joint-action controller may require. We therefore make both kernel types available and treat the choice between them as an evolvable categorical trait: each field can adopt (and mutate between) the kernel best matched to its functional role, while parameter optimisation tunes the dynamical regime within that choice.

2.1.2. Interaction Genes

While field genes define how each neural field works, interaction genes define how fields are connected. Each interaction gene encodes: (i) source and target field identifiers; (ii) the type and parameters of the inter-field interaction kernel, determining the spatial profile of the projection; (iii) an enabled/disabled flag that controls whether the connection is active in the phenotype; and (iv) a historical innovation number that supports alignment during crossover.

When field u projects to field v, the thresholded output from u provides spatially structured input to v via convolution Zibner, Faubel, Iossifidis and Schoner (2011):

$$s_v(x, t) = \int k_{uv}(x - x') g(u(x', t)) dx' \tag{4}$$

where $s_v(x, t)$ represents the input received by field v at position x and time t, $g(u(x', t))$ is the thresholded output of field u, and $k_{uv}(x - x')$ defines how activity at position x' in u influences position x in field v Ferreira et al. (2021); Wojtak et al. (2015, 2023). As with self-interaction, k_{uv} can adopt Gaussian or Mexican-hat profiles (Equations 2 and 3), and may be excitatory or inhibitory. This means evolution can discover not only which fields are connected, but also whether a projection should excite or inhibit downstream activity and over what spatial extent.

2.2. Mutations

Mutations are the primary source of variation in the evolutionary process, allowing the population to explore new controller designs. Two categories of mutations can occur: *parametric*, and *structural*. Together, they allow evolution to exploit promising dynamics within a given topology and to explore more complex topologies when the task demands it.

2.2.1. Parametric Mutations

Parametric mutations modify the properties of existing genes without changing the architectural topology. For field genes, these include: (i) incremental/decremental adjustments to intrinsic dynamical parameters, including τ and h; (ii) incremental/decremental adjustments to self-interaction kernel parameters, such as A and σ; and (iii) switching between Gaussian and Mexican-hat kernel types. These mutations allow evolution to continuously fine-tune how individual fields respond to input, for instance, sharpening the selectivity of the output field or adjusting the timescale over which a field integrates information about inputs.

2.2.2. Structural Mutations

Structural mutations alter the architectural topology of the phenotype, either by expanding the genome through the introduction of new fields or inter-field connections, or by reconfiguring existing connectivity through toggling. This enables the discovery of more complex controllers when simpler topologies are insufficient for the task.

The *add interaction* mutation establishes a new connection between two previously unlinked fields, introducing a new interaction gene with randomised parameters. This creates an additional pathway through which activation can propagate.

The *add field* mutation inserts a new hidden field along an existing connection. The original interaction gene is disabled and replaced by two new interactions: (i) an input connection from the original source field to the new hidden field, initialised with standard Gaussian parameters; and (ii) an output connection from the new hidden field to the original target, inheriting the characteristics of the disabled interaction. This design aims to minimise disruption to existing functionality while introducing an intermediate processing stage that can be gradually refined through subsequent parametric mutations.

Finally, the *toggle mutation* activates or deactivates existing interaction genes by switching their enabled status. When disabled, a connection is removed from the phenotype without being erased from the genome, effectively altering the architecture's information flow while preserving the genetic material for potential future reactivation. This reversibility allows evolution to explore alternative connectivity configurations and recover previously silenced pathways if selection pressures change.

2.3. Crossover

As genomes expand, they naturally develop varying structures that create alignment issues during recombination, a manifestation of the competing conventions problem in neuroevolution. The original NEAT algorithm addresses this through historical marking mechanisms that track genetic lineage, enabling proper alignment of homologous genes despite topological differences Stanley and Miikkulainen (2002).

This component remains fundamentally unchanged, as the historical marking solution applies equally well to DNF evolution. Each time a structural mutation introduces a new interaction gene, it is assigned a unique global innovation number that reflects when it first appeared in the population. These numbers are permanent and inherited by offspring, creating a chronological record of each genetic element's evolutionary origin.

During crossover, genes are categorised by their innovation numbers into three groups: *matching genes*, present in both parents; *disjoint genes*, present in one parent and falling within the innovation range of the other; and *excess genes*, present in one parent and beyond the innovation range of the other. Matching genes are inherited randomly from either parent, while disjoint and excess genes are taken from the higher-fitness parent, preserving architectural innovations that have already demonstrated their value.

2.4. Speciation

When a structural mutation introduces a new hidden field or connection, the resulting architecture typically performs poorly at first, since its new parameters have not yet been tuned, and the innovation has not had time to demonstrate its value. Without protection, such innovations would be rapidly eliminated by simpler, already-optimised architectures, preventing the population from exploring more complex topologies. To address this, controllers are partitioned into species so that novel structures compete primarily with topologically similar individuals, preserving diversity and giving promising architectural innovations time to mature.

Species assignment is determined by a compatibility distance δ between pairs of genomes, combining structural and functional differences:

$$\delta = \frac{c_e E}{N} + \frac{c_d D}{N} + c_k \overline{K} \tag{5}$$

where E and D are the number of excess and disjoint genes, respectively, and N is the size of the larger genome, normalising for architectural complexity (with a minimum value of 1 for genomes containing fewer than 5 genes). The coefficients c_e, c_d, and c_k control the relative importance of each component. Two genomes are considered compatible and therefore assigned to the same species when δ falls below a threshold δ_t, meaning they share a similar topological structure and inter-field interaction profiles.

The term $\overline{K}$ quantifies the average parametric difference across matching interaction genes. In the original NEAT algorithm, the strength of a connection is represented by a single weight, so comparing two connections is as simple as measuring the difference between those weights. In our framework, however, inter-field interactions are not represented by a single number but by spatial kernels characterised by multiple parameters. We therefore redefine $\overline{K}$:

$$\overline{K} = c_a \sum_{i \in M} |a_i^1 - a_i^2| + c_w \sum_{i \in M} |w_i^1 - w_i^2| \tag{6}$$

where M represents the set of *matching interaction genes*, a_i^1 and a_i^2 are the amplitude parameters of the i-th matching interaction in the first and second genomes, respectively, and w_i^1 and w_i^2 are the corresponding width parameters. The coefficients c_a and c_w weight the relative importance of each.

For Gaussian kernels, amplitude and width correspond directly to A and σ; for Mexican-hat kernels, they refer to the excitatory component's A_{exc} and σ_{exc}. This kernel-aware compatibility metric ensures that two controllers are only grouped when they not only share a similar topology but also implement functionally similar inter-field interactions.

To prevent any single species from dominating the population, fitness sharing is applied within species: each individual's fitness is adjusted relative to the size of its species, and reproduction is allocated proportionally to aggregate adjusted fitness. The evolutionary cycle then proceeds by eliminating under-performing individuals within each species, followed by reproduction among the survivors to generate the next generation.

3. Experimental Evaluation

3.1. Task Selection and Description

We evaluate our approach in a collaborative packaging task designed to isolate three fundamental joint-action behaviors that frequently arise in shared manipulation: complementary action selection, cooperative assistance, and action

inhibition Sebanz et al. (2006); Bekkering et al. (2009); Sebanz and Knoblich (2021).

In the task, a human and a robot jointly pick and place window frames into a container. The workspace contains two types of objects: small frames that either agent can manipulate independently, and large frames that require joint manipulation due to their size. These two object types require different robot responses. When the human approaches a large frame, the robot should assist by grasping the same object (a *cooperative* response). When the human targets a small frame, the robot should avoid interference and select a different small frame if one is available (a *complementary* response). When no such alternative exists, the robot should remain idle (an *action inhibition* response). Together, these three behaviours require the controller to continuously integrate information about object positions and human hand location into context-dependent action selection.

3.2. The Reference Minimal Architecture

Each of these behaviours can be produced in isolation through direct input–output inter-field kernel connections. However, combining all three within a single controller using just these direct connections is not possible, since the hand-position activation must play opposing roles depending on context. For complementary and inhibitory responses, it must inhibit the output field at the human-targeted location, suppressing the robot from selecting that object. For the cooperative response, it must excite the output field at that same location, driving the robot to assist. Since a single inter-field kernel connection cannot be simultaneously excitatory and inhibitory, a direct connection from the hand-position field to the output can support one type of behaviour but not both.

To solve this, at least one intermediate hidden field that separates the two connections is required: a direct inhibitory projection from the hand-position field to the output suppresses action at the human's location, while a second connection routes hand-position activation through a hidden field that integrates it with large-object activation, producing excitatory drive at the output only when the two signals spatially coincide.

One hidden field is thus both necessary and sufficient: this five-field, five-connection architecture is the *minimal controller* for the task, and we adopt it as a reference against which the evolved solutions are evaluated. The evolutionary process must therefore not only tune the intrinsic dynamics of individual fields but also discover new internal structure and the inter-field connectivity that enables this integration.

3.3. Robotic Platform and Implementation

For physical deployment, we used a KUKA LBR iiwa 14 r820 manipulator equipped with an OnRobot RG2 gripper, mounted on an anthropomorphic pedestal Ribeiro, Louro, Vicente, Ribeiro, Rodrigues, Dias, Sousa, Monteiro and Bicho (2024). The workspace consisted of a 60 cm wide tabletop section containing two types of window-frame objects: large green frames (50 cm) requiring joint manipulation, and smaller red frames (40 cm) that could be handled individually[1]. All objects were to be placed into a cardboard box positioned to the side of the robot. A StereoLabs ZED 2i stereo camera provided visual input, detecting object type and position, and tracking the human hand along the relevant continuous spatial dimension. The experimental setup is shown in Figure 3.

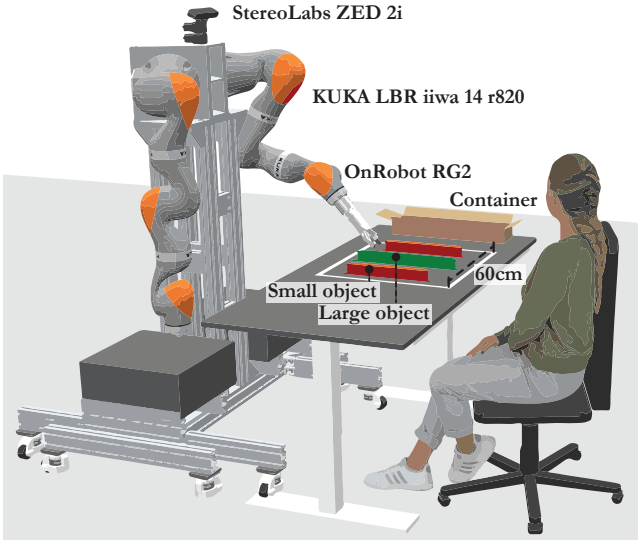


Figure 3: Experimental setup for the collaborative packaging task. A KUKA LBR iiwa 14 r820 manipulator equipped with an OnRobot RG2 gripper is mounted on an anthropomorphic pedestal. The workspace features a 60 cm tabletop area, which contains small (red) and large (green) window-frame objects. A cardboard container is positioned to the left of the robot for object placement. A StereoLabs ZED 2i camera provides visual input, detecting object type and position, and tracking the human hand position.

The robotic system control was implemented within a ROS 2 ecosystem with three main components: (i) a vision-processing node for object detection and human-hand tracking; (ii) a high-level control node running the evolved DNF-based controller (using dynamic-neural-field-composer), which maps perceptual input to task-appropriate robot actions; and (iii) a low-level control node that executes motion plans (using lbr_fri_ros2_stack Huber, Mower, Ourselin, Vercauteren and Bergeles (2024) and MoveIt2) and actuates the gripper (onrobot_ros2_driver). To support reproducibility, all code related to this project is openly available at Jgocunha/neuroevolutionary-joint-action.

3.4. Fitness Evaluation

During evolution, candidate controllers are never tested on the physical robot. Instead, fitness is assessed entirely in simulation by applying controlled input sequences to the DNF architecture and measuring how closely the emergent field dynamics match the desired joint-action behaviours. This allows hundreds of candidate architectures to be evaluated efficiently across generations.

[1] Object colours were chosen to simplify visual detection and carry no behavioural significance for the controller, which operates exclusively on position information.

External stimuli are applied to input fields as spatially localised Gaussian inputs:

$$s_{\text{input}}(x,t) = \sum_i A \exp\left(-\frac{(x-p_i)^2}{2\sigma^2}\right) \tag{7}$$

where p_i denotes stimulus position, A its amplitude, and σ its width. The resulting spatiotemporal activity patterns across input and output fields are then evaluated against a set of behavioural objectives (f_b), combined into a compound fitness score:

$$f = \sum_b w_b f_b \tag{8}$$

where w_b specifies the relative importance of behaviour b.

All evaluations are conducted in a fixed three-object scenario: two small objects at $p = 10$ and $p = 50$, and one large object at $p = 30$. This configuration is sufficient to assess the emergence of all three target behaviours while keeping evaluation feasible.

Fitness is computed from seven behavioural objectives, each weighted equally ($w_b = 1/7$) to avoid privileging any specific joint-action behaviour over others. The objectives are evaluated sequentially, progressing from perceptual representation through increasingly demanding coordination requirements.

First, we verify that the input fields correctly represent the perceptual scene:

- f_1: Correct representation of small objects — two stabilised peaks at $p = 10, 50$ in the first input field.
- f_2: Correct representation of the large object — one stabilised peak at $p = 30$ in the second input field.
- f_3: Correct representation of the human hand — one stabilised peak at $p = 30$ in the third input field.

With the human hand at $p = 30$ (co-located with the large object), we evaluate the cooperative response:

- f_4: Robot assists with the large object — one stabilised peak at $p = 30$ in the output field.

We then evaluate complementary action selection by moving the hand representation sequentially to each small object:

- f_5: When the hand moves to $p = 10$, the robot selects the other small object — one stabilised peak at $p = 50$ in the output field.
- f_6: When the hand moves to $p = 50$, the robot selects the other small object — one stabilised peak at $p = 10$.

Finally, we test action inhibition by removing one small object (at $p = 10$), leaving the hand at $p = 50$, the remaining small object at $p = 50$, and the large object at $p = 30$:

- f_7: The robot remains idle — no supra-threshold activation in the output field, as the human is already targeting the only available small object.

Together, these seven objectives assess whether the evolving architecture can represent all relevant perceptual information, assist when cooperation is required, act complementarily when working in parallel is possible, and withhold action when intervention would cause interference.

3.5. Parameter Settings

The initial population consisted of 1000 controllers, each encoding a minimal architecture of three input fields (representing small-object positions, large-object positions, and human hand positions), and one output field for robot action selection. All field parameters were randomly initialised within predefined bounds, with no inter-field connections or hidden fields present at initialisation. What the designer specifies is thus only the controller's interface to the world: which external stimulus arrives at each input field, and that the output field's activity is read as the robot's action. The intrinsic dynamics of every field, together with the intermediate processing structure and the full inter-field connectivity, are discovered by evolution.

All fields were defined over a one-dimensional spatial dimension of length 60 with $d_x = 1$. For all neural fields, we used a sigmoid output function $g(u)$ with steepness $\beta = 10$, and Gaussian noise of strength $q = 0.015$. Evolution stopped when all behavioural fitness components reached a threshold of $f_{tar} = 0.90$, or after a maximum of 200 generations. Each objective was evaluated over up to 500 simulation steps with a fixed timestep of $\Delta_t = 5$.

For speciation, we used a compatibility threshold of $\delta_t = 4.0$, with coefficients $c_e = 1.0$, $c_d = 0.5$, $c_k = 0.5$, and kernel parameter weights $c_a = c_w = 0.05$. Topological differences were weighted more strongly than parametric ones, in order to preserve architectural diversity without over-fragmenting species due to minor kernel differences.

During reproduction, mutation rates were tuned to favour frequent parametric exploration (50% field-gene mutation, 50% connection-gene mutation) while keeping structural innovations comparatively less common (0.10% new fields, 15% new inter-field connections, 0.5% toggling). This bias reflects a standard principle in neuroevolution Stanley and Miikkulainen (2002): parameters should adapt rapidly to exploit existing topologies, while structural growth must occur gradually to allow sufficient optimisation of new elements before they compete with simpler solutions.

To maintain diversity and mitigate stagnation, several mechanisms modulate reproductive opportunities based on fitness. Within individual species, any species failing to improve over 5 consecutive generations is classified as *stagnant* and excluded from further reproduction. The *champion* (best performing individual) of each species with more than 5 members is copied unchanged into the next generation. At the population level, when global fitness plateaus for 15

generations, reproduction is restricted to the two highest-performing species, redirecting evolutionary pressure toward the most promising lineages.

Performance was robust to moderate variations of these values, indicating that the results do not critically depend on these exact settings. A complete specification of all parameter values is provided in the Supplemental Materials (*hyperparameter_settings.pdf*).

4. Results

4.1. Evolutionary Performance

Across 100 independent evolutionary runs, the approach successfully discovered viable controllers in 97 runs (97%). Successful runs converged in an average of 62.58 generations (median: 53.00, std: 35.61), indicating consistent performance with moderate variation. The fastest successful run required only 18 generations, while the slowest took 199.

The high success rate demonstrates that the evolutionary process reliably discovers topologies capable of generating all three joint-action behaviours without any manual specification of architecture or parameters. The 3 unsuccessful runs discovered topologies of sufficient complexity (3-4 hidden fields, 14-16 inter-field kernel connections) but failed due to insufficient parametric refinement. The needed genes were present in the population, but the internal field and inter-field coupling parameters were never tuned precisely enough to satisfy all behavioural objectives simultaneously within the pre-defined 200 generations. In all three runs, the perceptual objectives and inhibitory response were consistently satisfied, while the cooperative and complementary responses ($f4$, $f5$, $f6$) remained below threshold.

4.2. Architectural Complexity and Topology

The evolved architectures were topologically minimal. Across 97 successful runs, solutions contained on average 1.89 hidden fields (median: 2.00, std: 0.80) and 9.34 enabled inter-field kernel connections (median: 9.00, std: 3.38); the smallest used one hidden field and five connections, the largest four hidden fields and 17 connections. As shown in Figure 4, successful solutions clustered strongly around two topology classes: one hidden field with 5–6 connections, and two hidden fields with 9–11 connections.

We compare these architectures to the described minimal architecture of Section 3.2. Two properties should hold: (i) no evolved successful controller should be less complex (because a simpler solution would imply the derivation is wrong), and (ii) the algorithm should discover the minimal architecture itself. Both are satisfied. No controller across the 97 runs used fewer than one hidden field or five connections, confirming the lower bound. The single most frequently evolved topology was exactly the minimal architecture (one hidden field and five connections), discovered in 15 of 97 runs, more often than any other configuration. Evolution thus did not merely find viable controllers; it repeatedly converged on the minimal one, with no specification of how many fields the controller should contain.

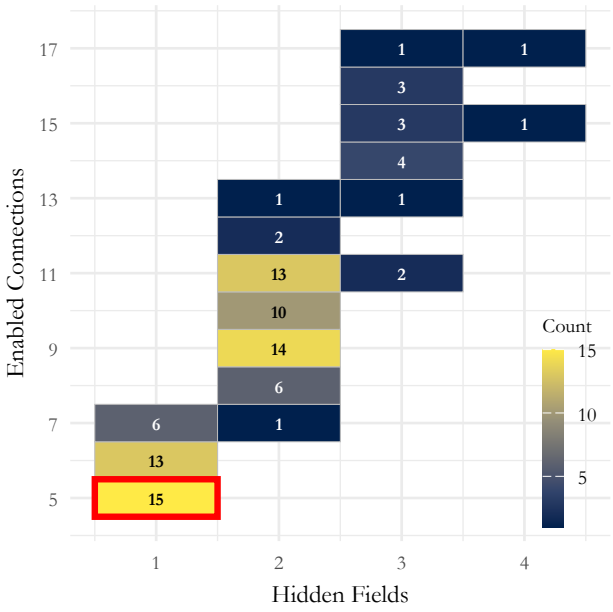


Figure 4: Frequency of successful topologies, as a function of the number of hidden fields and enabled connections. Cell colour and number indicate how many runs converged to each topology. Solutions cluster around minimal configurations, with the most frequent topology being 1 hidden field and 5 connections (15 runs). Topologies with more than 3 hidden fields were rare.

More complex topologies were discovered, but were rare. All successful architectures satisfied the behavioural fitness thresholds regardless of complexity. A Spearman rank-order correlation revealed a strong positive association between topology complexity (total number of field genes and enabled interaction genes in the genome) and generations to threshold ($r_s = 0.73, p < 0.001, n = 97$), indicating that more complex architectures required considerably more generations to optimise (a natural consequence of the larger parameter space that additional fields and connections introduce).

For the remainder of this analysis, we focus on the controller selected for physical deployment. Given that topological minimalism is a stated objective of the approach and that simpler architectures equally satisfied the fitness criteria, we selected the highest-performing solution at minimal topology ($f = 0.97$), whose genome is shown in Figure 5. This architecture consists of three input fields, one output field, one hidden field, and five enabled connections.

4.3. Speciation and Behavioural Emergence

To illustrate how joint-action behaviour emerges through the evolutionary process, we trace the run that produced the deployed architecture, which converged in 31 generations (Figure 6).

The first structural mutation (a connection from the small-object field to the output field - *ig 1-4 innov 1*) appeared as early as generation 2, but was insufficient to disrupt

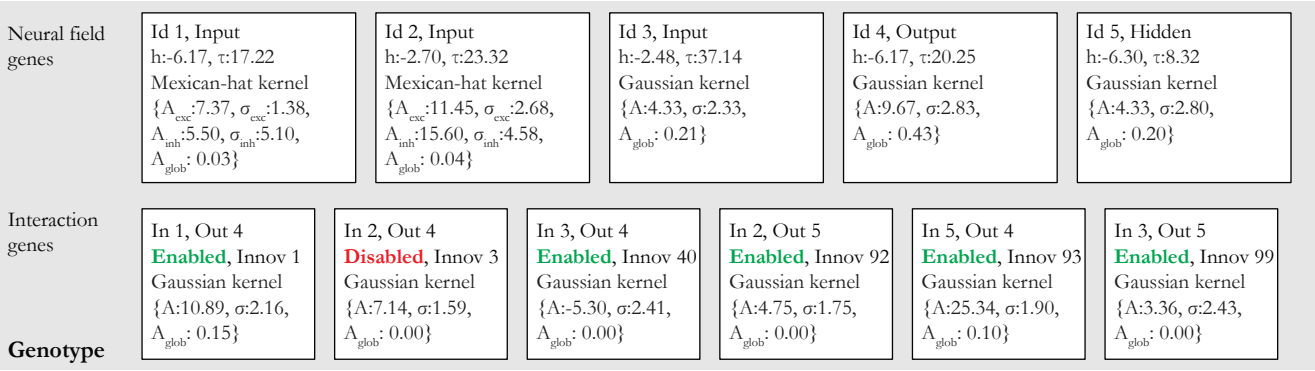


Figure 5: Genome of the highest-performing solution with minimal topology ($f = 0.97$), consisting of five neural fields and five enabled interaction genes. Input fields (*nf1*, *nf2*, *nf3*) receive small-object positions, large-object positions, and human hand position, respectively. The hidden field (*nf5*) receives input from the large-object input field (*nf2*) and the hand-position field (*nf3*), and outputs to the output field (*nf4*). One disabled interaction gene (*nf2* → *nf4*) is present in the genome. All active inter-field projections use Gaussian kernels.

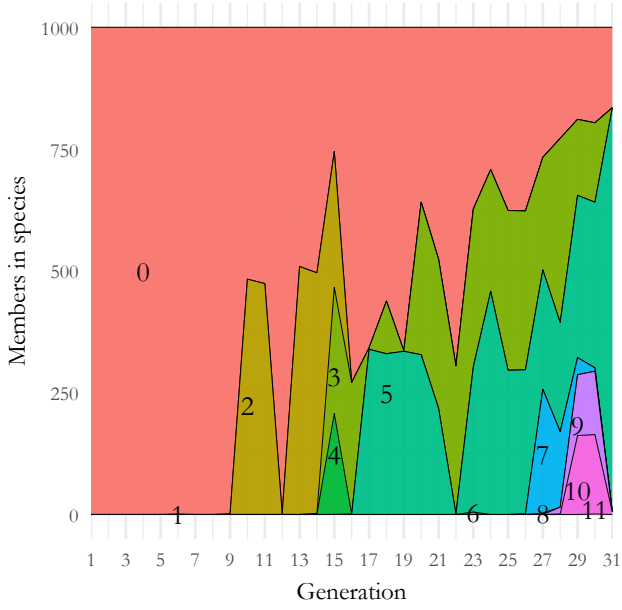


Figure 6: Speciation dynamics across the 31 generations of the run that produced the deployed architecture. Each band represents a distinct species, with height indicating the number of member controllers. A single species dominates the first nine generations. Multiple species emerge and become extinct as competing structural innovations are explored. Species 5, which introduced the hidden field responsible for integrating large-object and hand-position information, survives and ultimately produces the successful controller at generation 31.

species cohesion. For the first nine generations, all controllers remained within a single species (a transient species appeared briefly at generation 6 with a single member but was immediately reabsorbed into the main population). During this period, evolution operated exclusively through parametric mutations and by adding interaction genes.

The first true speciation event occurred at generation 9, triggered by two simultaneous structural innovations. Species 0 acquired a new interaction from the large-object field to the output (*ig 2-4 innov 26*), while species 2 emerged carrying a connection from the hand-position field to the output (*ig 3-4 innov 27*), an inhibitory pathway that would later prove essential for complementary and inhibitory responses. By generation 10, the population had split nearly evenly between these two architectural lineages.

The following generations saw competing innovations and a series of extinctions. Species 2 and 4 both became extinct by generation 16, unable to optimise their parameters within the stagnation window. Species 5, born from species 0 at generation 16 as a single individual, would prove to be the winning lineage.

The most consequential event of the run occurred at generation 22, when species 5 introduced a hidden field (*nf5*) inserted along the large-object-to-output coupling, replacing *ig 2-4 innov 3* with two new interactions (*ig 2-5 innov 92* and *ig 5-4 innov 93*). This created an intermediate processing stage capable of integrating sensory information before propagating to the output field. The immediate consequence was a sharp fitness collapse in species 5 ($f_{champion} = 0.91 \rightarrow 0.69$), as the new hidden field disrupted the existing dynamics before its parameters had been optimised. This is precisely the scenario that speciation is designed to protect against: without a dedicated species, this innovation would have been eliminated immediately by simpler, already-optimised competitors.

Over the following six generations, species 5 recovered steadily through parametric refinement, climbing back to $f_{champion} = 0.91$ by generation 28. Throughout this period, the cooperative response (f_4) remained the last unsatisfied objective. At generation 28, the last critical structural connection was established: *ig 3-5 innov 99*, linking the hand-position field to the hidden field and completing the integration pathway through which large-object and hand-position information converge before reaching the output field (an architectural prerequisite for the cooperative response). In 4 generations, parametric refinements pushed all

seven partial fitness objectives above f_{tar} simultaneously, and the run terminated with the emergence of a solution with $f = 0.97$. At termination, species 5 comprised 829 of the 1000 controllers and was 15 generations old; yet never fully dominated the population, with competing species persisting as minority lineages to the end.

4.4. Evolved Neural Field Dynamics

We analyse the activation dynamics of the deployed architecture in the evaluation scenario presented in Section 3.4 (two small objects at $p = 10$ and $p = 50$, and one large object at $p = 30$). These positions are arbitrary: because the fields encode location continuously, the dynamics depend only on the relative arrangement of the stimuli, provided that objects are separated by more than the excitatory kernel width, so that distinct peaks form rather than merge.

Three structural features of the evolved architecture are key to understanding its dynamics. First, the excitatory connection *'1-4'* propagates the small-object representation in *nf1* to the output field *nf4*, meaning supra-threshold activation in *nf1* drives peak formation in *nf4* at the corresponding locations. When two spatially distinct peaks are present in *nf1*, the evolved selection dynamics of *nf4* ensure that only one peak location is stabilised in the output field. Second, the projection from *nf3* to *nf4* is inhibitory (the only inhibitory coupling in the architecture), continuously inhibiting the output field at the location of the human hand. This inhibition is tuned such that when two small objects are present, and the human hand targets one of them, the output field always selects the other; and when only one small object is present, the inhibition is sufficient to cancel its excitatory contribution entirely. Third, *nf5* receives excitatory projections from both *nf2* and *nf3*: each projection alone pre-shapes *nf5*, but only when the large object and the human hand are spatially co-located do these pre-shapes drive *nf5* to supra-threshold activation. When *nf5* stabilises a peak, its projection to *nf4* is stronger than both the excitatory contribution from *nf1* and the inhibitory contribution from *nf3*. This ensures that the cooperative response always takes precedence over complementary or inhibitory behaviour.

4.4.1. Cooperative Response

When the human hand approaches the large object (*nf3* $p = 30$, *nf2* $p = 30$), *nf5* receives co-localized excitation from *'3-5'* and *'2-5'* and forms a supra-threshold peak at $p = 30$, this peak drives strong excitation onto *nf4* at the same location. Although *nf3* simultaneously inhibits *nf4* at $p = 30$, the excitation from *nf5* overcomes this inhibition, and *nf4* stabilizes a supra-threshold peak at $p = 30$, with pre-shaping at $p = 10$ and $p = 50$. The robot selects the large object and assists the human (Figure 7a).

4.4.2. Complementary Response

When the human hand moves to a small object (*nf3* $p = 10$), *nf5* receives spatially non-overlapping inputs from *nf2* ($p = 30$) and *nf3* ($p = 10$). Because these do not coincide, neither location reaches threshold, leaving only sub-threshold pre-shapes at $p = 10$ and $p = 30$. Without a supra-threshold *nf5* peak, the strong excitatory drive onto *nf4* at $p = 30$ is absent. Meanwhile, *nf3*'s inhibitory projection suppresses *nf4* at $p = 10$, eliminating competition from the human-targeted location. The remaining excitation from *nf1* at $p = 50$ is sufficient to push *nf4* over threshold, and the robot selects the other available small object (Figure 7b).

4.4.3. Inhibitory Response

When the human targets the only remaining small object (*nf3* peak at $p = 10$, *nf1* peak only at $p = 10$), the same dynamics unfold as in the complementary case, *i.e.* *nf5* remains sub-threshold, and *nf3* inhibits *nf4* at $p = 10$. However, unlike the complementary case, no alternative small object is available to drive *nf4* above threshold. The output field remains entirely sub-threshold, and the robot does not act (Figure 7c).

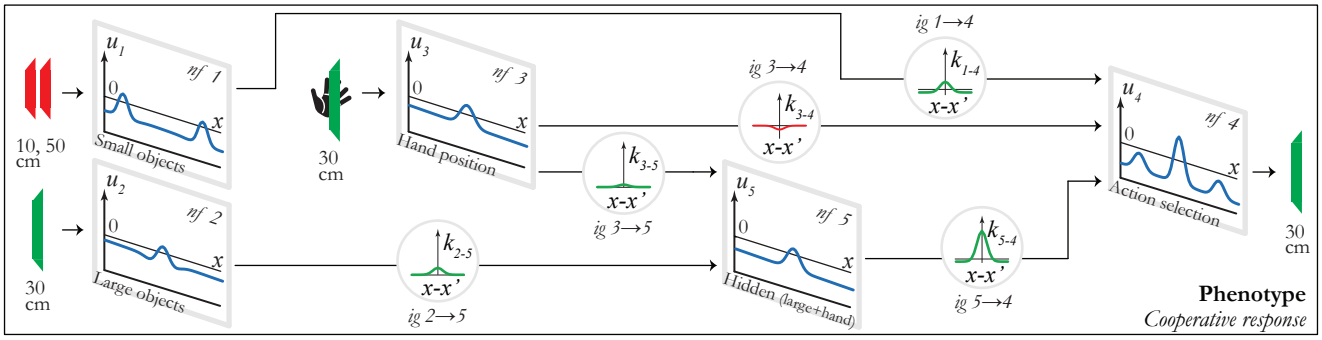
4.5. Physical Deployment

The evolved controller was deployed directly on the physical platform without any modification or retuning of parameters. The same architecture and parameter values that emerged from simulation were used directly, within the ROS2 control stack described in Section 3.3. All three behaviours manifested correctly during physical interaction. The robot continuously monitored the human partner's hand position and adapted its responses in real time, without explicit communication or predefined action sequences. Representative moments of complementary and cooperative behaviour during physical deployment are shown in Figure 8.

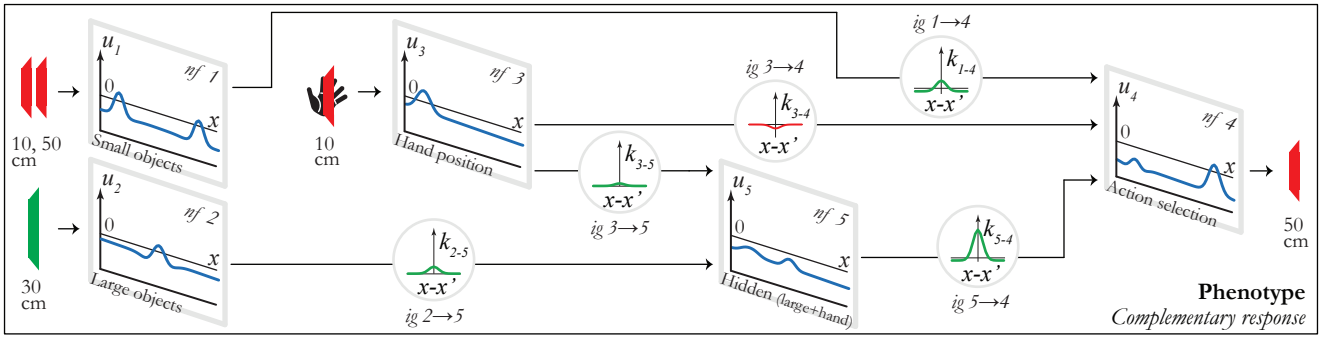
Beyond the fixed three-object configuration used during evolution, the controller was evaluated on a variety of object arrangements presented in different spatial orders and locations: two large and two small, two small and one large, two large and one small, two large only, two small only, one large and one small, and single-object configurations. Video recordings of the generated behaviours during physical deployment across these different object configurations are available as a Supplemental Video at https://youtu.be/_73pyThFsyY.

Across the 23 trials recorded in the ∼18-minute video (Table 1), the robot performed 43 pick-up actions (27 cooperative and 16 complementary), and each one matched the behaviour defined by the task logic; no incorrect or failed responses were observed. The number of times these responses occur is determined by the object configuration. Because every large object requires joint manipulation, the robot's cooperative pick-ups equal the number of large objects[2] presented ($N_{\text{coop}} = \sum \text{trials} \times N_{\text{large}}$). And because each small object is handled by just one agent, the robot's complementary pick-ups plus the human's small-object pick-ups account for all the small objects ($N_{\text{comp}} + N_{\text{small,human}} = \sum \text{trials} \times N_{\text{small}}$); this includes cases where the robot simply takes the one small object remaining if the human isn't reaching for it. The table also reports the inhibitory responses. Unlike the cooperative or complementary responses, these

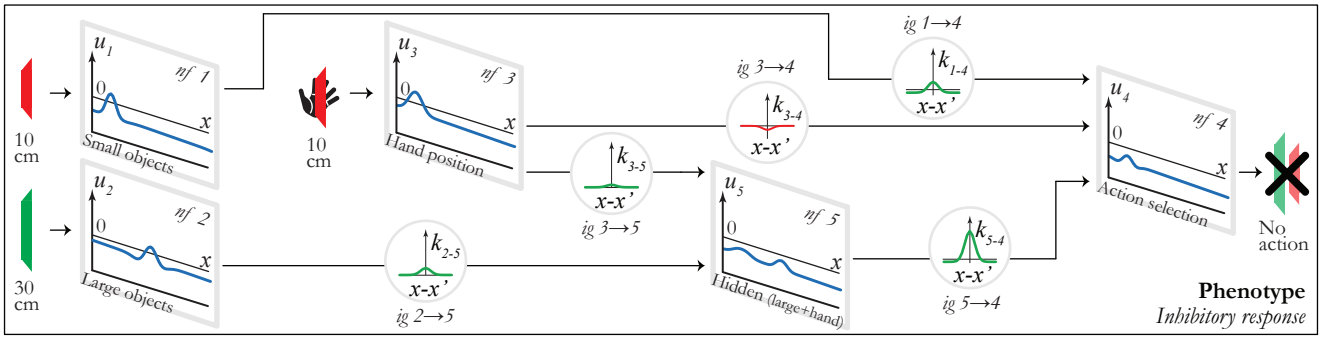
[2] The human's large-object pick-ups aren't shown in the table, since they are always equal to N_{coop}.



(a) *Cooperative response*. The human hand and the large object are co-located at $p = 30$. *nf5* forms a supra-threshold peak at $p = 30$, driving the output field *nf4* to select the large object position. Sub-threshold pre-shapes are visible at $p = 10$ and $p = 50$.



(b) *Complementary response*. The human hand targets the small object at $p = 10$. *nf5* remains sub-threshold. *nf3*'s inhibitory projection suppresses *nf4* at $p = 10$, and the robot selects the alternative small object at $p = 50$.



(c) *Inhibitory response*. The human targets the only available small object at $p = 10$. *nf5* remains sub-threshold and no alternative object is available to drive *nf4* above threshold. The robot withholds action.

Figure 7: Emergence of cooperative, complementary, and inhibitory behaviours from the evolved controller.

can occur any number of times as the human moves between objects, which is why they are most frequent in small-object-rich configurations and absent when only large objects are present.

The generalisation across configurations is a direct consequence of the continuous spatial encoding properties inherent to DNF-based representations. Rather than encoding discrete symbols, stabilised activation peaks represent object positions along a metric dimension, allowing the controller to respond appropriately to any object location, configuration, or quantity without further tuning or reconfiguration Erlhagen and Bicho (2006); Bicho, Erlhagen, Louro and Costa E Silva (2011).

5. Discussion and Conclusions

The results of this work demonstrate that functional Dynamic Neural Field (DNF)-based controllers capable of adaptive joint action can be automatically synthesised through neuroevolution, without any manual specification of architecture or parameters. With 97% success rate (97 out of 100 runs), the evolutionary process reliably discovered minimal architectures that generated all three target behaviours (cooperative assistance, complementary action selection, and action inhibition) and transferred directly to a physical robotic platform without retuning.

A consistent finding across runs was the emergence of topologically minimal architectures. Evolution did not

Figure 8: Physical deployment of the evolved DNF controller in the collaborative packaging task. To the left, the *complementary response*: the workspace contains two small red frames and one large green frame. The human grasps the right small frame, and the robot independently selects the left small frame. To the right, the *cooperative response*: the workspace contains two large green and two small red frames. The human grasps the left large frame, and the robot coordinates by grasping the same object.

Table 1
Physical deployment across 23 trials (∼18 min of footage) spanning multiple object configurations and spatial orders. For each configuration (N_{small}, N_{large}), the robot's pick-ups are defined by the task logic: cooperative pick-ups equal the number of large objects (each handled jointly), and complementary pick-ups plus the human's small-object pick-ups (Human small) equal the number of small objects (each handled by one agent). Inhibitory responses are transient withdrawals (the robot suppressing action and re-planning without grasping) and may recur within a trial. Every response matched the expected behaviour.

Config.			Robot pick-ups		Human	Robot
N_{small}	N_{large}	Trials	Coop.	Comp.	small	Inhib.
1	0	3	0	2	1	6
0	1	2	2	0	0	0
2	0	2	0	3	1	8
0	2	4	8	0	0	0
1	1	4	4	4	0	6
2	1	3	3	3	3	20
1	2	2	4	1	1	4
2	2	3	6	3	3	18
Total		**23**	**27**	**16**	**9**	**62**

produce large, over-parameterised controllers; it converged on the small structures sufficient for the task. These minimal structures reflect the underlying task complexity: as described in Section 3.2, the collaborative packaging task requires at least one hidden field to resolve a conflict in the role of hand-position information, and the most frequent evolved topology contained exactly that: a single hidden field and five connections. That evolution independently and repeatedly converged on the minimum architecture predicted by this analysis, without any specification of how many hidden fields the controller should contain or how they should be connected, suggests that well-defined behavioural requirements can drive the emergence of the necessary internal structure through evolutionary search alone.

The speciation dynamics observed in the traced run show a general pattern in how this structural discovery unfolds. A structural mutation that introduces a hidden field initially reduces fitness, as the new parameters have not yet been tuned, and the innovation cannot yet demonstrate its value. Without speciation, this innovation would be immediately eliminated by simpler, already-optimised competitors. The protection given by a dedicated species provides the new solution time to mature through parametric refinement, eventually enabling behavioural capabilities that the previous topology could not support. This "two-phase dynamic", *i.e.* structural innovation followed by parametric exploitation, reflects the general mechanism through which neuroevolution with speciation navigates the tradeoff between exploration and exploitation in the space of architectures.

The transfer from simulation to physical deployment, and the controller's correct responses to object configurations beyond those "seen" during evaluation, can be attributed to the continuous spatial encoding properties of DNF-based representations. Stabilised activation peaks represent object positions along a metric dimension rather than as discrete symbols, and the controller does not know about specific objects, only about positions and their spatial relationships. Any object placed anywhere in the workspace will generate a stimulus that the field dynamics can respond to, provided it falls within the representational range. This is not a property that evolution had to discover; it is an inherent characteristic of DNFs that the evolutionary process preserved and exploited Schöner et al. (2015); Bicho et al. (2011).

Several limitations of the current work should be acknowledged. The fitness function remains manually designed and task-specific: while our approach automates the discovery of the architecture and parameters, it still requires a human designer to specify what constitutes correct behaviour via weighted behavioural objectives. This is a dependency that multi-objective optimisation methods could partially address in future work Abramovich and Moshaiov (2016); Schrum, Capps, Steckel, Volz and Risi (2023),

by replacing scalar fitness aggregation with Pareto-based selection and reducing the sensitivity of results to weight choices. The task itself, while sufficient to demonstrate the approach, is structured. The workspace is controlled, the objects are limited, and the interaction is constrained to a single spatial dimension. The behavioural evaluation was conducted without quantitative Human-Robot Interaction (HRI) metrics such as fluency, workload, or task efficiency Hoffman (2019); an important direction for future work that would situate the approach within the broader HRI evaluation literature.

These limitations point to several directions for future work. Applying the framework to more complex joint-action scenarios, e.g., shared decision-making, or multi-robot collaboration, will test the scalability of structural innovation under richer behavioural requirements. However, as task complexity grows and more hidden fields and connections are required, the interpretability of the evolved architectures may diminish. In the task studied here, the behaviour of each field can be traced back to identifiable Dynamic Field Theory (DFT) mechanisms; whether this transparency is preserved in larger, more elaborated architectures is an open question and an important direction for future work. Extending the genetic encoding to support multi-dimensional neural fields would broaden the range of perceptual and motor representations available to the evolved architectures. At the theoretical level, the relationship between task complexity and the architectural complexity that evolution discovers remains an open and interesting question.

In conclusion, this work establishes that DNF control architectures for Human-Robot Collaboration (HRC) can be evolved rather than hand-designed. The resulting controllers are interpretable in terms of DFT principles, their structure is minimal, and their dynamics can be traced back to the excitatory and inhibitory field interactions that underlie biological neural computation. By bridging neuroevolution and DFT, this work opens a constructive pathway toward adaptive, interpretable, and automatically synthesised neurorobotic controllers for embodied HRI.

CRediT authorship contribution statement

João G. Cunha: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Raymond H. Cuijpers:** Conceptualization, Supervision, Writing – review & editing. **Wolfram Erlhagen:** Conceptualization, Supervision, Writing – review & editing. **Estela Bicho:** Conceptualization, Project administration, Resources, Supervision, Writing – review & editing.

References

Abramovich, O., Moshaiov, A., 2016. Multi-objective topology and weight evolution of neuro-controllers, in: 2016 IEEE Congress on Evolutionary Computation (CEC), IEEE, Vancouver, BC, Canada. pp. 670–677. URL: http://ieeexplore.ieee.org/document/7743857/, doi:10.1109/CEC.2016.7743857.

Amari, S.i., 1977. Dynamics of pattern formation in lateral-inhibition type neural fields. Biol. Cybern. 27, 77–87. URL: http://link.springer.com/10.1007/BF00337259, doi:10.1007/BF00337259.

Amato, N.M., Hutchinson, S., Garg, A., Billard, A., Rus, D., Tedrake, R., Park, F., Goldberg, K., 2025. "Data will solve robotics and automation: True or false?": A debate. Sci. Robot. 10, eaea7897. URL: https://www.science.org/doi/10.1126/scirobotics.aea7897, doi:10.1126/scirobotics.aea7897.

Bekkering, H., De Bruijn, E.R.A., Cuijpers, R.H., Newman-Norlund, R., Van Schie, H.T., Meulenbroek, R., 2009. Joint Action: Neurocognitive Mechanisms Supporting Human Interaction. Topics in Cognitive Science 1, 340–352. URL: https://onlinelibrary.wiley.com/doi/10.1111/j.1756-8765.2009.01023.x, doi:10.1111/j.1756-8765.2009.01023.x.

Bicho, E., Erlhagen, W., Louro, L., Costa E Silva, E., 2011. Neuro-cognitive mechanisms of decision making in joint action: A human–robot interaction study. Human Movement Science 30, 846–868. URL: https://linkinghub.elsevier.com/retrieve/pii/S0167945710001508, doi:10.1016/j.humov.2010.08.012.

Bicho, E., Mallet, P., Schöner, G., 2000. Target Representation on an Autonomous Vehicle with Low-Level Sensors. The International Journal of Robotics Research 19, 424–447. URL: https://journals.sagepub.com/doi/10.1177/02783640022066950, doi:10.1177/02783640022066950.

Billard, A., Albu-Schaeffer, A., Beetz, M., Burgard, W., Corke, P., Ciocarlie, M., Dahiya, R., Kragic, D., Goldberg, K., Nagai, Y., Scaramuzza, D., 2025. A roadmap for AI in robotics. Nat Mach Intell 7, 818–824. URL: https://www.nature.com/articles/s42256-025-01050-6, doi:10.1038/s42256-025-01050-6.

Chatziparaschis, D., Zhong, S., Christopoulos, V., Karydis, K., 2024. Adaptive Environment-Aware Robotic Arm Reaching Based on a Bio-Inspired Neurodynamical Computational Framework, in: 2024 33rd IEEE International Conference on Robot and Human Interactive Communication (ROMAN), IEEE, Pasadena, CA, USA. pp. 510–515. URL: https://ieeexplore.ieee.org/document/10731226/, doi:10.1109/RO-MAN60168.2024.10731226.

Coombes, S., Beim Graben, P., Potthast, R., Wright, J., 2014. Neural fields: theory and applications. Springer, Heidelberg.

Engels, C., Schöner, G., 1995. Dynamic fields endow behavior-based robots with representations. Robotics and Autonomous Systems 14, 55–77. URL: https://linkinghub.elsevier.com/retrieve/pii/0921889094000203, doi:10.1016/0921-8890(94)00020-3.

Erlhagen, W., Bicho, E., 2006. The dynamic neural field approach to cognitive robotics. J. Neural Eng. 3, R36–R54. URL: https://iopscience.iop.org/article/10.1088/1741-2560/3/3/R02, doi:10.1088/1741-2560/3/3/R02.

Ferreira, F., Wojtak, W., Sousa, E., Louro, L., Bicho, E., Erlhagen, W., 2021. Rapid Learning of Complex Sequences With Time Constraints: A Dynamic Neural Field Model. IEEE Trans. Cogn. Dev. Syst. 13, 853–864. URL: https://ieeexplore.ieee.org/document/9085956/, doi:10.1109/TCDS.2020.2991789.

Fix, J., 2013. Template based black-box optimization of dynamic neural fields. Neural Networks 46, 40–49. URL: https://linkinghub.elsevier.com/retrieve/pii/S089360801300124X, doi:10.1016/j.neunet.2013.04.008.

Floreano, D., 2023. Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies. MIT PRESS, S.l. URL: https://mitpress.mit.edu/9780262547734/bio-inspired-artificial-intelligence/. oCLC: 1371014935.

Hoffman, G., 2019. Evaluating Fluency in Human–Robot Collaboration. IEEE Trans. Human-Mach. Syst. 49, 209–218. URL: https://ieeexplore.ieee.org/document/8678448/, doi:10.1109/THMS.2019.2904558.

Huber, M., Mower, C.E., Ourselin, S., Vercauteren, T., Bergeles, C., 2024. LBR-Stack: ROS 2 and Python Integration of KUKA FRI forMed and IIWA Robots. JOSS 9, 6138. URL: https://joss.theoj.org/papers/10.21105/joss.06138, doi:10.21105/joss.06138.

Igel, C., Erlhagen, W., Jancke, D., 2001. Optimization of dynamic neural fields. Neurocomputing 36, 225–233. URL: https://linkinghub.elsevier.com/retrieve/pii/S0925231200003283, doi:10.1016/

S0925-2312(00)00328-3.

Krichmar, J.L., 2018. Neurorobotics—A Thriving Community and a Promising Pathway Toward Intelligent Cognitive Robots. Front. Neurorobot. 12, 42. URL: https://www.frontiersin.org/article/10.3389/fnbot.2018.00042/full, doi:10.3389/fnbot.2018.00042.

Mompó Alepuz, A., Papageorgiou, D., Tolu, S., 2024. Brain-inspired biomimetic robot control: a review. Front. Neurorobot. 18, 1395617. URL: https://www.frontiersin.org/articles/10.3389/fnbot.2024.1395617/full, doi:10.3389/fnbot.2024.1395617.

Nolfi, S., Floreano, D., 2000. Evolutionary robotics: the biology, intelligence, and technology of self-organizing machines. Intelligent Robotics and Autonomous Agents Series, MIT Press, Cambridge. URL: https://ieeexplore.ieee.org/book/6267313.

Papavasileiou, E., Cornelis, J., Jansen, B., 2021. A Systematic Literature Review of the Successors of "NeuroEvolution of Augmenting Topologies". Evolutionary Computation 29, 1–73. URL: https://direct.mit.edu/evco/article/29/1/1-73/97341, doi:10.1162/evco_a_00282.

Pfeifer, R., Bongard, J., 2006. How the Body Shapes the Way We Think: A New View of Intelligence. The MIT Press. URL: https://direct.mit.edu/books/book/2035/how-the-body-shapes-the-way-we-thinka-new-view-of, doi:10.7551/mitpress/3585.001.0001.

Ribeiro, N., Louro, L., Vicente, P., Ribeiro, P., Rodrigues, D., Dias, P., Sousa, E., Monteiro, S., Bicho, E., 2024. RAMBO — Robotic Anthropomorphic Manipulator for Bimanual Operations, in: 2024 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), IEEE, Paredes de Coura, Portugal. pp. 157–164. URL: https://ieeexplore.ieee.org/document/10535933/, doi:10.1109/ICARSC61747.2024.10535933.

Sandamirskaya, Y., Zibner, S.K., Schneegans, S., Schöner, G., 2013. Using Dynamic Field Theory to extend the embodiment stance toward higher cognition. New Ideas in Psychology 31, 322–339. URL: https://linkinghub.elsevier.com/retrieve/pii/S0732118X13000111, doi:10.1016/j.newideapsych.2013.01.002.

Schneider, S., Igel, C., Klaes, C., Dinse, H.R., Wiemer, J.C., 2004. Evolutionary Adaptation of Nonlinear Dynamical Systems in Computational Neuroscience. Genetic Programming and Evolvable Machines 5, 215–227. URL: http://link.springer.com/10.1023/B:GENP.0000023689.70987.6a, doi:10.1023/B:GENP.0000023689.70987.6a.

Schrum, J., Capps, B., Steckel, K., Volz, V., Risi, S., 2023. Hybrid Encoding for Generating Large Scale Game Level Patterns With Local Variations. IEEE Trans. Games 15, 46–55. URL: https://ieeexplore.ieee.org/document/9764394/, doi:10.1109/TG.2022.3170730.

Schöner, G., Dose, M., Engels, C., 1995. Dynamics of behavior: Theory and applications for autonomous robot architectures. Robotics and Autonomous Systems 16, 213–245. URL: https://linkinghub.elsevier.com/retrieve/pii/0921889095000496, doi:10.1016/0921-8890(95)00049-6.

Schöner, G., Spencer, J., Research Group, D., 2015. Dynamic Thinking: A Primer on Dynamic Field Theory. Oxford University Press, Oxford. URL: https://doi.org/10.1093/acprof:oso/9780199300563.001.0001, doi:10.1093/acprof:oso/9780199300563.001.0001.

Sebanz, N., Bekkering, H., Knoblich, G., 2006. Joint action: bodies and minds moving together. Trends in Cognitive Sciences 10, 70–76. URL: https://linkinghub.elsevier.com/retrieve/pii/S1364661305003566, doi:10.1016/j.tics.2005.12.009.

Sebanz, N., Knoblich, G., 2021. Progress in Joint-Action Research. Curr Dir Psychol Sci 30, 138–143. URL: https://journals.sagepub.com/doi/10.1177/0963721420984425, doi:10.1177/0963721420984425.

Selvaggio, M., Cognetti, M., Nikolaidis, S., Ivaldi, S., Siciliano, B., 2021. Autonomy in Physical Human-Robot Interaction: A Brief Survey. IEEE Robot. Autom. Lett. 6, 7989–7996. URL: https://ieeexplore.ieee.org/document/9501975/, doi:10.1109/LRA.2021.3100603.

Sheng, Y., Nie, W., Liu, Z., Zhang, H., Zhou, H., Liu, Z., Liu, H., 2025. Biomimetic Robotics and Intelligence: A Survey. SmartBot 1, e12010. URL: https://onlinelibrary.wiley.com/doi/10.1002/smb2.12010, doi:10.1002/smb2.12010.

Stanley, K.O., Miikkulainen, R., 2002. Evolving Neural Networks through Augmenting Topologies. Evolutionary Computation 10, 99–127. URL: https://direct.mit.edu/evco/article/10/2/99-127/1123, doi:10.1162/106365602320169811.

Vernon, D., 2025. The Future of Research in Cognitive Robotics: Foundation Models or Developmental Cognitive Models? adv. robot. res. , e202500066URL: https://advanced.onlinelibrary.wiley.com/doi/10.1002/adrr.202500066, doi:10.1002/adrr.202500066.

Wojtak, W., Ferreira, F., Erlhagen, W., Bicho, E., 2015. Learning joint representations for order and timing of perceptual-motor sequences: A dynamic neural field approach, in: 2015 International Joint Conference on Neural Networks (IJCNN), IEEE, Killarney, Ireland. pp. 1–7. URL: http://ieeexplore.ieee.org/document/7280717/, doi:10.1109/IJCNN.2015.7280717.

Wojtak, W., Ferreira, F., Louro, L., Bicho, E., Erlhagen, W., 2023. Adaptive timing in a dynamic field architecture for natural human–robot interactions. Cognitive Systems Research 82, 101148. URL: https://linkinghub.elsevier.com/retrieve/pii/S1389041723000761, doi:10.1016/j.cogsys.2023.101148.

Zibner, S.K.U., Faubel, C., Iossifidis, I., Schoner, G., 2011. Dynamic Neural Fields as Building Blocks of a Cortex-Inspired Architecture for Robotic Scene Representation. IEEE Trans. Auton. Mental Dev. 3, 74–91. URL: http://ieeexplore.ieee.org/document/5709974/, doi:10.1109/TAMD.2011.2109714.

Declaration of interests

☒The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

RAS_Video1_Explanation_Supplemental_Material.mp4

Click here to access/download

Supplementary material

RAS_Video1_Explanation_Supplemental_Material.mp4

RAS_Video2_HumanRobotJointAction_Supplemental_Material.mp4

Research Data

Click here to download Research data
https://github.com/jgocunha/neuroevolutionary-joint-action