Neural Computing and Applications
https://doi.org/10.1007/s00521-020-05224-8

ORIGINAL ARTICLE

A neural integrator model for planning and value-based decision making of a robotics assistant

Weronika Wojtak[1,2] · Flora Ferreira[1] · Paulo Vicente[2] · Luís Louro[2] · Estela Bicho[2] · Wolfram Erlhagen[1]

Received: 25 November 2019 / Accepted: 17 July 2020

Abstract

Modern manufacturing and assembly environments are characterized by a high variability in the built process which challenges human–robot cooperation. To reduce the cognitive workload of the operator, the robot should not only be able to learn from experience but also to plan and decide autonomously. Here, we present an approach based on Dynamic Neural Fields that apply brain-like computations to endow a robot with these cognitive functions. A neural integrator is used to model the gradual accumulation of sensory and other evidence as time-varying persistent activity of neural populations. The decision to act is modeled by a competitive dynamics between neural populations linked to different motor behaviors. They receive the persistent activation pattern of the integrators as input. In the first experiment, a robot learns rapidly by observation the sequential order of object transfers between an assistant and an operator to subsequently substitute the assistant in the joint task. The results show that the robot is able to proactively plan the series of handovers in the correct order. In the second experiment, a mobile robot searches at two different workbenches for a specific object to deliver it to an operator. The object may appear at the two locations in a certain time period with independent probabilities unknown to the robot. The trial-by-trial decision under uncertainty is biased by the accumulated evidence of past successes and choices. The choice behavior over a longer period reveals that the robot achieves a high search efficiency in stationary as well as dynamic environments.

Keywords Dynamic Neural Field · Neural integrator · Assembly robot · Value-based decision making · Sequence learning

1 Introduction

In recent years, the paradigm shift from mass production to mass customization represents new challenges but also new opportunities for human–robot interactions in industrial assembly processes Hu et al. (2011); Krüger et al. (2009). Traditionally, industrial robots perform assembly steps in isolation from people, continuously repeating a carefully predefined sequence of actions. The appearance of of-the-shelf industrial robots like ABB's Frida, Rethink's Sawyer or KUKA's LBR iiwa, certified to safely operate alongside humans, makes it in principle possible to involve robots as coworkers in the final assembly process. However, to efficiently support the operator in customer-oriented production, the robots should be able to quickly adapt to changing tasks and production sequences that come along with product variations. This means that the robots should be equipped with some higher-level cognitive functions like action planning, learning and decision making that allow them to increase their scope of operation autonomously Bannat et al. (2010); Lemaignan et al. (2017); Tsarouchi et al. (2016). Otherwise, the need to closely supervise and control the actions of the robotics helper would further increase the already high cognitive workload for the human operator in complex assembly processes. Even if the robot is not directly involved in the assembly work and only supports the operator by delivering parts and tools, a proactive attitude to timely hold out the object that the operator needs next has a major impact on user

✉ Weronika Wojtak
w.wojtak@dei.uminho.pt

Wolfram Erlhagen
wolfram.erlhagen@math.uminho.pt

[1] Research Centre of Mathematics, University of Minho, 4800-058 Guimarães, Portugal

[2] Research Centre Algoritmi, University of Minho, 4800-058 Guimarães, Portugal

Published online: 07 August 2020

satisfaction Huang and Mutlu (2016); Koene et al. (2014); Pinheiro et al. (2010).

A promising research line to overcome the lack of autonomy of today's industrial robots is to implement in their control architecture models of human cognition. While classical AI-inspired approaches are based on the logical manipulation of abstract symbols Agostini et al. (2017); Lemaignan et al. (2017); Mayer et al. (2011), the emerging area of neurorobotics exploits the fundamentals of neuronal dynamics as a basis of higher-level cognitive functions Bicho et al. (2010); Cox and Krichmar (2009); Erlhagen and Bicho (2006); Kozma (2008). A gradual accumulation of information over time is a neural computation of critical importance to the organization of task-appropriate behavior Cain and Shea-Brown (2012). Persistent, or slowly varying, neural population activity described in many cortical and subcortical brain areas is commonly believed to represent a neural correlate of evidence accumulation for linking perception and action Brody and Hanks (2016); Haller et al. (2018). Persistent activity has been observed in tasks in which information no longer present in the immediate environment has to be maintained for future adaptive behavior (working memory), during an interval prior to or in preparation of a forthcoming goal-directed action (motor planning), or during the process of choosing one option from a set of alternatives on the basis of sensory evidence (perceptual decision making) (for review see Curtis and Lee (2010)). Moreover, sustained population activity that appears to be systematically modulated by the outcome of previous actions across multiple trials is thought to represent a neural substrate for learning in an unfamiliar and stochastic environment Histed et al. (2009).

One proposed neural mechanism for accumulation of inputs and self-sustained activity over behaviorally relevant timescales is based on excitatory feedback in recurrent neural networks Seung et al. (2000). A critical requirement is the fine-tuning of the synaptic interaction strengths. It ensures that the feedback is kept in exactly the right range so that the sustained activity at any time represents the time integral of past inputs. To solve the fine-tuning problem and increase robustness, additional mechanisms like for instance the use of bistable neural units with different activation thresholds have been proposed Koulakov et al. (2002), (for review of different computational approaches see Sakai et al. 2006). Due to the bistability, the network dynamics exhibits multiple stable states even with imprecisely tuned feedback. While the increased robustness of the neural integrator certainly favors the application in real-world robotics experiments, it comes with the price of a loss of sensitivity to weaker evidence. Only a sufficiently strong input may switch between adjacent stable states.

In this paper, we present a novel neural integrator model Wojtak et al. (2016) based on the theoretical framework of Dynamic Neural Fields (DNF) that provides a continuous attractor to perform temporal integration of external and internal inputs of any strength. At the same time, the integrator mechanism is robust to perturbations in the network connectivity. We test the model in two robotics applications with the ultimate goal to advance the autonomy of future robotics assistants. The focus of the experiments is on autonomous decision making guided by past experiences. To give a broader perspective on HRI applications beyond the specific examples, we explore in Discussion how the experimental results may be exploited to address some of the key challenges that have been identified for close human–robot collaborative work in a manufacturing environment Lin (2017); Tsarouchi et al. (2016).

The first robotics experiment is a pipe assembly task in which a robot first watches a human assistant grasping a series of pipes to hand them over to an operator performing the assembly steps. The goal for the robot is to learn the serial order of handovers to subsequently substitute the human assistant in the joint task. Order learning is guided by the information provided by the vision system about the color and the length of each pipe. During observation, the neural integrator of a memory field establishes a gradient of persistent activations over distinct neural subpopulations tuned to a specific color-length combination. The integration time from input onset to the end of the assembly sequence defines the level of persistent activity. Consequently, the neural representation of the first pipe to be grasped has the highest activation, whereas the representation of the last pipe has the lowest. During joint execution, the competitive neurodynamics of a decision field, which receives the stable activation gradient as input, drives the sequential initiation of proactive handovers of all pipes in the correct order.

The second robotics experiment is a simulated material handling system in which a mobile robot transports parts or subassembly parts between workstations. Specifically, the robot has to search at two workbenches where a required part may appear with a certain probability in a fixed period of time. Humans and other animals show in a scenario with constant probabilities a choice behavior known as the matching strategy: the fraction of choices made to any option will match the fraction of total success (or reward) earned from that option Herrnstein (1961); Lau and Glimcher (2005). Using the neural integrator model to incorporate short-term memories of past choices and past successes in the decision process endows the robot with the matching behavior. Importantly, we show that the robot autonomously adapts its internal valuation of the competing alternatives and consequently its choice behavior to unsigned changes in the success probabilities Sugrue et al.

(2004). The robot is thus better able to cope with specific challenges in more flexible and uncertain manufacturing systems.

The present work on planning and value-based decision making of a proactive service robot in an industrial environment complements our earlier research on a DNF architecture for natural human–robot cooperation. Here, the focus was on the implementation of an action understanding and goal inference capacity that allows the robot to coordinate its decisions and behaviors with a human in order to achieve a common goal Bicho et al. (2011b); Erlhagen and Bicho (2014).

The remainder of this article is organized as follows: The next section summarizes basic principles of DNF theory and introduces the mathematical models. Section 3 is divided into two subsections in which we describe the two robotics applications, introduce the DNF architectures for sequence learning and value-based decision making and present results of the robotics experiments. Finally, we discuss the results and related HRI research and give an outlook on future work. Some technical implementation issues and the model parameters are given in Appendix.

2 Dynamic field models

2.1 General principles and field equation

Dynamic Neural Fields (DNFs) represent a theoretical framework for developing cognitive robot control architectures which is inspired by fundamental cortical processing mechanisms Erlhagen and Bicho (2006). Sensory information guiding behavior in cognitive tasks is represented by supra-threshold activity patterns or bumps in recurrently connected neural networks. These patterns are initially triggered by transient inputs from external sources but become self-sustained due to the recurrent interactions within the neural populations. In many robotics applications, the Dynamic Neural Fields are defined over continuous metric dimensions such as the direction in which the robot has to move or the size and color of an object to be manipulated Bicho et al. (2000); Faubel and Schöner (2008); Ferreira et al. (2014). Sensory information about a specific value along the coded dimension defines the field location where the self-stabilized bump develops. Due to the assumed translation invariance of the neural interactions, the field dynamics supports a spatial continuum of persistent activity patterns known as a "continuous attractor" Brody et al. (2003).

The field model first proposed by proposed by Amari [2] is specifically popular in applications since it favors analytical treatment Amari (1977); Coombes (2005); Ferreira et al. (2016):

$$\tau\frac{\partial u(\mathbf{r},t)}{\partial t} = -u(\mathbf{r},t) + \int_{\Omega} w(\mathbf{r},\mathbf{r}')f(u(\mathbf{r}',t)-\theta)\mathrm{d}\mathbf{r}' \\ - h + I(\mathbf{r},t) + \epsilon^{1/2}\mathrm{d}W(\mathbf{r},t), \tag{1}$$

where $u(\mathbf{r},t)$ represents the activity at time t of a neuron at field position $\mathbf{r}$ in a spatial domain Ω which is usually a subset of $\mathbb{R}^d$. For the present applications, fields with dimensions $d=1$ and $d=2$ are used.

The parameter $\tau > 0$ defines the timescale, $I(\mathbf{r},t)$ represents a time-dependent, localized input centered at site $\mathbf{r}$, and $h > 0$ defines the stable resting state of a field without external input. $w(\mathbf{r},\mathbf{r}')$ is the synaptic weight distribution which determines the connection strength between interacting neurons at positions $\mathbf{r}$ and $\mathbf{r}'$. The distribution is assumed to depend on the Euclidian distance, $|\mathbf{r}-\mathbf{r}'|$, with excitation dominating at shorter ranges and surround inhibition. An example is given by a Gaussian function minus a constant:

$$w_{lat}(\mathbf{r}) = A_{lat}e^{\left(-\mathbf{r}^2/2\sigma_{lat}^2\right)} - g_{lat}, \tag{2}$$

with $A_{lat} > g_{lat} > 0$ and $\sigma_{lat} > 0$. Another widely used connectivity function of lateral inhibition type has Mexican hat shape. It is given by the difference of two Gaussians:

$$w_{mex}(\mathbf{r}) = A_{ex}e^{\left(-\mathbf{r}^2/2\sigma_{ex}^2\right)} - A_{in}e^{\left(-\mathbf{r}^2/2\sigma_{in}^2\right)} - g_{mex}, \tag{3}$$

where $A_{ex} > A_{in} > 0$ and $\sigma_{in} > \sigma_{ex} > 0$ and $g_{mex} > 0$.

The term $f(u-\theta)$ denotes the firing rate function with threshold $\theta \geq 0$. A typical example is a smooth sigmoidal function with steepness parameter β

$$f(u) = \frac{1}{1+e^{-\beta(u-\theta)}}. \tag{4}$$

For $\beta \rightarrow \infty$, f converges to the Heaviside function H defined by

$$H(u) = \begin{cases} 1 & \text{if} \quad u > \theta, \\ 0 & \quad \text{otherwise} \end{cases}, \tag{5}$$

which we use throughout the paper. The threshold is chosen as $\theta = 0$ Amari (1977). Finally, the additive noise term $\mathrm{d}W(\mathbf{r},t)$ describes the increment of a spatially dependent Wiener process with amplitude $\epsilon \ll 1$. We use the noise term to break the symmetry in decision processes in which different choices get equal or nearly equal support from input sources.

2.2 Two-field neural integrator

In applications of the Amari model, a sufficiently strong transient input is used to switch between a stable resting state and the bump attractor, thus implementing a memory function. However, since the shape of the bump is exclusively defined by the recurrent interactions within the population, the model cannot be used to memorize in addition the strength or duration of sensory signals nor can it be applied as an integrator of inputs arriving at different points in time. The new field model formalized by two coupled Amari equations overcomes this limitation Wojtak et al. (2016). Bump solutions lie on a two-dimensional attractor, defined by a continuum of positions and amplitudes. Bumps with varying amplitudes may thus be used to robustly represent the accumulation of sensory and other evidence over time.

The activities at time t and spatial position $\mathbf{r}$ of the two coupled fields, $u(\mathbf{r},t)$ and $v(\mathbf{r},t)$, are governed by the equations:

$$\begin{aligned}\tau\frac{\partial u(\mathbf{r},t)}{\partial t} &= -u(\mathbf{r},t)+v(\mathbf{r},t)\\ &\quad+\int_{\Omega} w_{mex}(\mathbf{r},\mathbf{r}')f(u(\mathbf{r}',t)-\theta)\mathrm{d}\mathbf{r}' \\ &\quad+I(\mathbf{r},t)+\epsilon^{1/2}\mathrm{d}W(\mathbf{r},t),\end{aligned} \tag{6a}$$

$$\begin{aligned}\tau\frac{\partial v(\mathbf{r},t)}{\partial t} &= -v(\mathbf{r},t)+u(\mathbf{r},t)\\ &\quad-\int_{\Omega} w_{mex}(\mathbf{r},\mathbf{r}')f(u(\mathbf{r}',t)-\theta)\mathrm{d}\mathbf{r}'.\end{aligned} \tag{6b}$$

2.3 Multi-item memory versus decision making

Figure 1 shows two examples of input-driven bump solutions in a one-dimensional field that are used in robotics applications to implement two fundamental cognitive functions, working memory and decision making. The evolution of a single bump shown in Fig. 1a is governed by equation (1) with strong lateral inhibition (2), whereas the dynamics of the two-field model (6) supports the multi-bump solution shown in Fig. 1b. Both fields received simultaneously three transient inputs of equal duration, but different strengths, A_{I_j}, modeled as Gaussians centered at positions x_{c_j}:

$$I(x)=\sum_{j=1}^{3}A_{I_j}e^{\left(-(x-x_{c_j})^2/2\sigma_I^2\right)}. \tag{7}$$

The inputs may represent for instance sensory information about movement direction to three potential targets. In the Amari case, the strong competition between stimulated field sites guarantees that only one bump may exist at all time. The bump represents in this example a decision to engage in the choice getting the strongest sensory support. The multi-bump pattern, on the other hand, can be seen as a multi-item memory of all possible choices including their relative strengths.

2.4 Choosing parameter values

The design of DNF architectures for robot control is based on localized activation patterns as fundamental units representing task-relevant information. For the designer, it is therefore important to understand the conditions for the model parameters that guarantee their existence and stability. We briefly summarize here the main ideas of an analytical approach developed by Amari [2]. For a more detailed tutorial in the context of robotics applications see Erlhagen and Bicho (2006). A recent overview of analytical and numerical tools for DNFs can be found in Coombes et al. (2014).

Amari's approach is based on the assumption that for the choice of Heaviside step function (5), the pattern formation process can be understood by analyzing the evolution equations of the bump boundaries. Amari defines the finite region of suprathreshold activity of a single stationary bump as $R(u)=\{x|u(x)>\theta\}=(x_1,x_2)$ with $u(x_1)=u(x_2)=\theta$ and derives a simple condition for the existence of a bump of width $\Delta=x_2-x_1$. In the absence of input, the existence of this solution is determined by the roots of

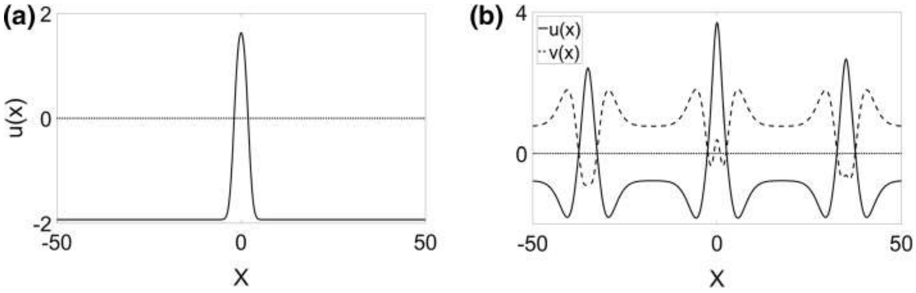

Fig. 1 Input-driven steady-state solutions of models (1) and (6) are shown. Input parameters are $A_{I_j}\in\{1.5,4,2\}$ and $\sigma_I=1.5$, and input duration is $d_{I_j}=1$. **a** Single bump of Amari model (1). The connectivity function is given by (2) with $A_{lat}=1.5$, $\sigma_{lat}=1$ and $g_{lat}=0.5$. **b** Three bump in $u(x)$ (solid line) of the two-field model (6) and corresponding activity pattern of $v(x)$ (dashed line). The connectivity function is given by (3) with $A_{ex}=3$, $A_{in}=1.5$, $\sigma_{ex}=2$, $\sigma_{in}=4$ and $g_{mex}=0.1$

$$F(\Delta) = -\theta + W(\Delta) = 0, \tag{8}$$

where $W(x)$ is the integral of weight function $w(x)$

$$W(x) = \int_0^x w(y)\mathrm{d}y. \tag{9}$$

The analogous condition for the bump existence in the two-field model is determined by the roots of

$$F(\Delta) = -2\theta + K + W(\Delta) = 0, \tag{10}$$

where K equals the sum of the initial states at $t = 0$ of the two fields, that is, $K = u(x, 0) + v(x, 0)$, which we choose to be constant in the applications. Figure 2 illustrates the existence conditions for the two interactions kernels and a Heaviside nonlinearity with threshold $\theta = 0$ used in model simulations of Fig. 1. For both models, the bump solution is stable if

$$\mathrm{d}F(\Delta)/\mathrm{d}\Delta < 0, \tag{11}$$

and unstable otherwise.

Figure 3 shows that for range of threshold values, $0 \le \theta < W_{max}$ for the Amari model and $0 \le \theta < (W_{max} + K)/2$ for the two-field model, two bump solutions co-exist which have different widths, $\Delta' < \Delta$. The wider bump is stable (solid line), whereas the narrower bump is unstable (dashed line). With the other model parameters fixed, the shape of the stable bump can be controlled by varying the parameter θ. This is evident for the bump width shown in Fig. 3. Using tools from numerical bifurcation analysis Rankin et al. (2014), further insights into the dependence of the bump shape on model parameters can be gained by tracking the bump solution when a specific model parameter is changed. An example is shown in Fig. 4 where the maximum bump activity is plotted as function of θ. To summarize, there exists a whole range of suitable parameter values and this range can be determined analytically.

The evolution of bumps is triggered by localized input from connected fields with suprathreshold activity patterns and/or localized input from external sources (e.g., robot camera). We model the external input for convenience as a Gaussian function with strength parameter A_I and standard deviation σ_I. Since for the two-field neural integrator, the width of a transient input affects bump width, $2\sigma_I$ is chosen in order to approximately match the bump width Δ defined by condition (10). Since for the current applications, only the peak position of the bump matters, σ_I is not a sensible parameter. Input strength A_I and input duration, however, play an important role. The accumulated evidence from internal and external input sources is reflected in the bump height which affects decision processes. While the two-field model integrates and memorizes localized input of any strength, we use the Amari model in a dynamic regime in which input strength may make a difference. For the choice $\theta > 0$, the two bump solutions defined by condition (8) co-exist with a stable resting state, $u(x) = 0 \; \forall x$ Amari (1977). This means that only a sufficiently strong input, $A_I > \theta$, may drive the evolution of a self-stabilized bump. Weaker input "pre-activates" neural populations only, and the activity decays to resting level when the input is removed.

For the building and numerically solving of complex control architectures consisting of many coupled DNFs (e.g., the HRI architectures in Bicho et al. (2011b); Erlhagen and Bicho (2014)) open-source software frameworks such as Cedar Lomp et al. (2016) exist. They provide graphical programming interfaces which allow the user to individually parameterize each field along the lines just discussed. Cedar also allows to integrate simulations of a robot, of its sensors and of its working environment.

3 Robotics applications

3.1 Sequence learning and planning in a pipe assembly task

The first experiment is a pipe assembly task in which the robot learns the sequential order of handing over different pipes to an operator who performs the assembly steps. We apply a learning by demonstration paradigm which has been successfully used in the past to teach robots sequential tasks Billard et al. (2008); Erlhagen et al. (2006); Sousa

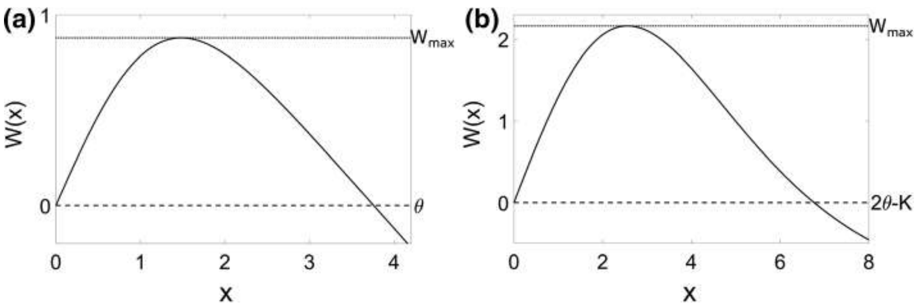

Fig. 2 Integral $W(x)$ of weight function $w(x)$ given by (2) (**a**) and (3) (**b**). Parameters of the weight functions as in Fig. 1

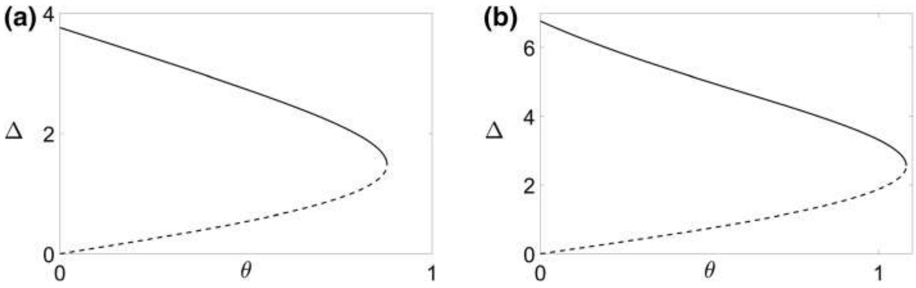

Fig. 3 Stability of one-bump solutions in the Amari model (**a**) and the two-field model (**b**). Parameters of connectivity functions (2) and (3) as in Fig. 1

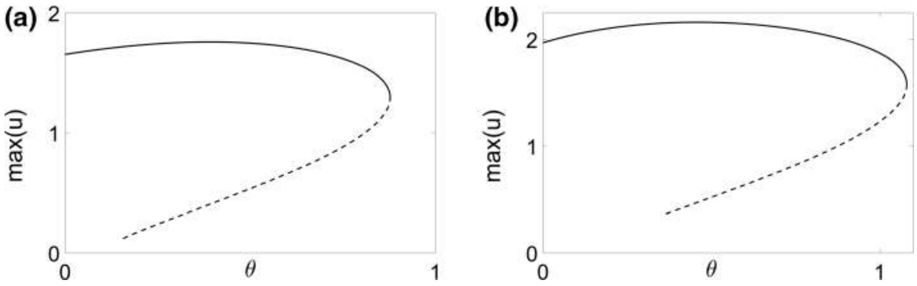

Fig. 4 Bifurcation curves showing one-bump solutions of the Amari model (**a**) and the two-field model (**b**)

et al. (2015). An important practical pre-requisite is that robot learning should be efficient and fast since the human tutors cannot be expected to repeat the demonstration of the action sequence several times Pardowitz et al. (2007). Since the robot is not directly involved in the assembly work, only the sequence of pipe transfers between the assistant and the operator is demonstrated for simplicity. Note however that the neural computations support fast serial-order learning of the entire assembly sequence including the manipulation of pipes initially located in the operator's workspace Sousa et al. [25]. The learning is guided by the information provided by the vision system about the length and the hue value in color space coordinates of each pipe. The combined feature information allows the robot to distinguish the different pipes. For simplicity, the three possible pipe lengths are labeled in the following as short, medium and long, respectively. Following the learning by demonstration phase, the robot demonstrates the acquired task knowledge by assisting the human operator in the assembly work. To support fluency of joint task executing, the robot has to plan and initiate the handover of a particular pipe in anticipation of the worker's need.

For the experiment, we used the collaborative robot Sawyer designed by the company "Rethink Robotics." It is featured by a 7 degrees of freedom robot arm with 1.26 meters reach that operates in work cells defined for humans. The "head" mounted on top consists of a LCD display and a camera system. An additional camera system is mounted on the arm. We used the head camera to provide the information about length and color at the moment when the "giver" has transported the pipe to the exchange position and the "receiver" touches it. During joint task execution, the reach-grasp-transport trajectory of the robot arm to a pre-defined exchange position is generated using the "HUMP" planner developed by our group Silva et al. (2015). It guarantees human-like features of the robot arm movements.

3.1.1 Model architecture

Figure 5 presents a sketch of the DNF model implemented as part of the control architecture of the robot Sawyer. It consists of several interconnected two-dimensional fields spanned over the input dimensions $(x, y) = (\text{length}, \text{color})$.

The model generalizes an earlier one-dimensional sequence learning model which we have applied to teach a robot by demonstration a color-coded musical sequence Ferreira et al. (2014). From a computational perspective, the important innovation is that the neural integrator simplifies the processing and storage of serial information. No additional neural mechanisms (e.g., a threshold accommodation dynamics used in Ferreira et al. (2014)) have to be applied to account for stable bumps with varying amplitudes.

The perceptual field u_{per}, governed by the Amari equation (1) with kernel (2), receives a two-dimensional Gaussian input representing the length–color combination of a specific pipe that the vision system detects at the exchange position. The neural dynamics in the sequence memory layer is governed by the coupled neural integrator equations, u_m and v_m, given by (6) with kernel (3). The u_m population receives two types of excitatory input (solid lines):

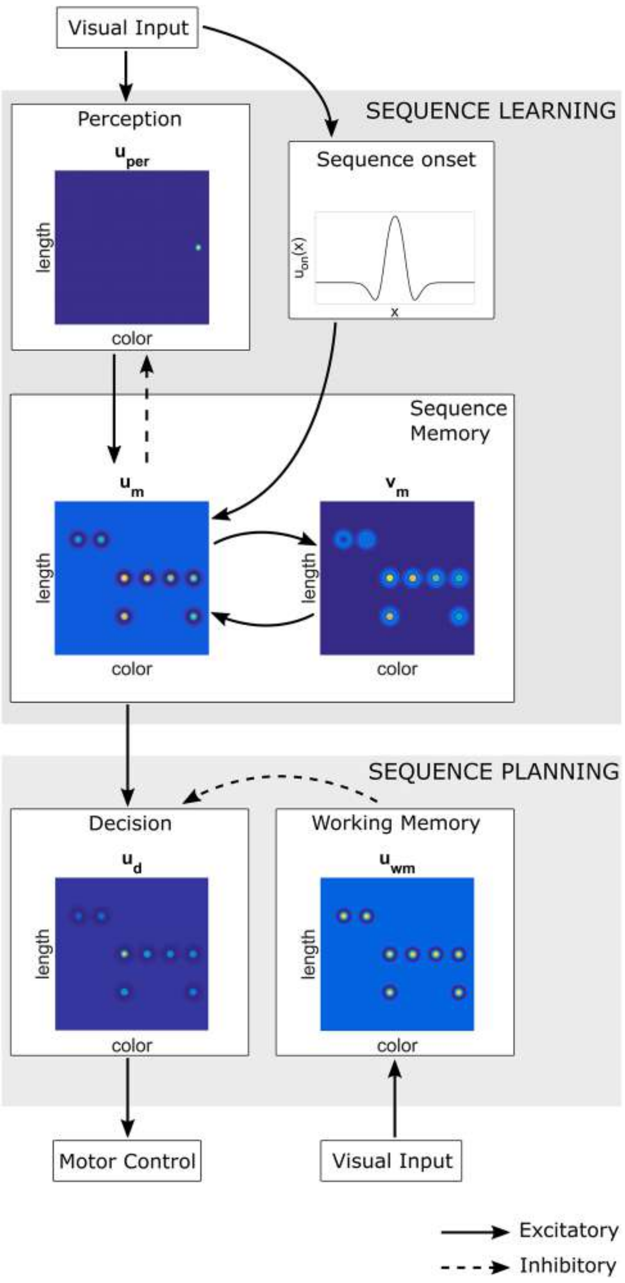

Fig. 5 Schematic view of the DNF model with several interconnected fields implementing sequence learning and sequence planning

$$I(x,y,t) = u_{per}(x,y,t)f(u_{per}(x,y,t)) + I_c f(u_m(x,y,t)). \tag{12}$$

The first term on the right side describes the input from the perceptual field at position (*x*, *y*) where sensory information has triggered the evolution of a suprathreshold activity pattern. Through inhibitory feedback connections (dashed line), the bump in u_{per} becomes destabilized (and the input disappears) once a memory bump in (u_m, v_m) has been established at the corresponding position. The second term describes a constant input I_c to all neurons in u_m, which is defined by the summed suprathreshold activation in the "sequence onset" field u_{on}:

$$I_c = \kappa \int_{\{x: u_{on}(x) \geq 0\}} f(u_{on}(x))\mathrm{d}x, \tag{13}$$

where $\kappa > 0$ defines the input strength.

A bump in this one-dimensional field represents the memory of an additional color cue that signals to the robot the onset of the sequence demonstration. Due to the multiplicative gating by the suprathreshold activity, $f(u_m)$, the integration of the constant input, I_c, manifests only at sites where the transient input from u_{per} has already driven the evolution of a bump. As a consequence, the earlier a certain pipe has been manipulated during demonstration, the higher is its memory bump. An activation gradient established in (u_m, v_m) thus encodes serial order. Gradient-based modeling approaches to serial order are known in the literature as competitive cuing or ordinal models Rhodes et al. (2004). The integration of the constant input stops when a second color cue (not shown), signaling the end of the sequence demonstration, destabilizes the bump in u_{on}.

During joint execution of the assembly task, the choice which pipe to manipulate next is made in the decision field u_d governed by equation (1) with kernel (2). It receives the stationary activation gradient of u_m as subthreshold, excitatory input:

$$I(x,y) = u_m(x,y). \tag{14}$$

The planning and execution of the whole sequence starts with the sequence onset signal. It triggers the continuous increase of the baseline activity h_d. A simple linear dynamics is used with a timescale $\tau_d = 1/\kappa$ controlled by the strength parameter $\kappa > 0$:

$$\tau_d \frac{\mathrm{d}h_d(t)}{\mathrm{d}t} = \int_{\{x: u_{on}(x) \geq 0\}} f(u_{on}(x))\mathrm{d}x, \;\; h_d(t_0) = h_{d_0} < 0. \tag{15}$$

The subpopulation of u_d with highest pre-activation reaches the threshold for creating a bump first. The moment of reaching the threshold is used to trigger the reach-to-grasp movement toward the specific pipe. The bump is destabilized by inhibitory input (dashed line) from a bump in the working memory field u_{wm} which is initially driven by visual input. This input indicates that the hand of the human worker has reached the exchange position to receive the pipe. Subsequently, the neural representation of the next pipe to be manipulated by the robot becomes active in u_d. This autonomous planning process continues until the memory representation of the last pipe with the lowest pre-activation has been processed.

To enable stable multi-bump solutions in the working memory field u_{wm} governed a dimensional Amari equation (1), an oscillatory connectivity function Ferreira et al. (2016); Laing et al. (2002) is used:

$$w_{osc}(\mathbf{r}) = A_{osc}e^{-b|\mathbf{r}|}(b\sin|\mathbf{r}| + \cos\mathbf{r}), \tag{16}$$

where A_{osc} controls the amplitude and $b > 0$ the rate at which the oscillations decay with distance.

3.1.2 Results—sequence learning and planning

The robot Sawyer watches the human assistant handing over 4 pipes to the worker in the following order: pink-medium → orange-medium → orange-short → dark-blue-long. Figure 6 shows two snapshots at the beginning and the end of the demonstration together with the corresponding activity patterns in the sequence memory field. At time $t = 15\mathbf{s}$ a single bump has evolved representing the pink-medium pipe. Immediately after the transfer of the dark-blue-long pipe, at time $t = 57\mathbf{s}$, all 4 pipes are represented in memory with bump amplitudes reflecting serial order. Figure 7 depicts the time course of the sequential bump formation relative to sequence onset at time $t = 0$.

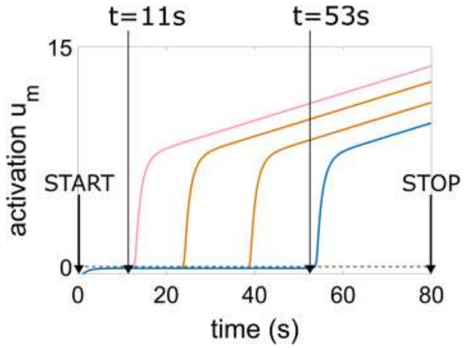

Fig. 7 Temporal evolution of population activities in the memory field u_m during sequence demonstration. The start and stop signals are presented at times $t = 0$ and $t = 80\mathbf{s}$, respectively

During joint task execution together with the human operator, Sawyer takes the role of the giver. Figure 8a depicts the temporal evolution of suprathreshold activity of the subpopulations in the decision field u_d representing motor plans directed toward the different pipes (solid lines). The field dynamics activates the individual handovers in the correct serial order. The suprathreshold activity of each subpopulation decays back to resting state due to the inhibition from bumps evolving at corresponding sites in the working memory field u_{wm} (dashed lines). The

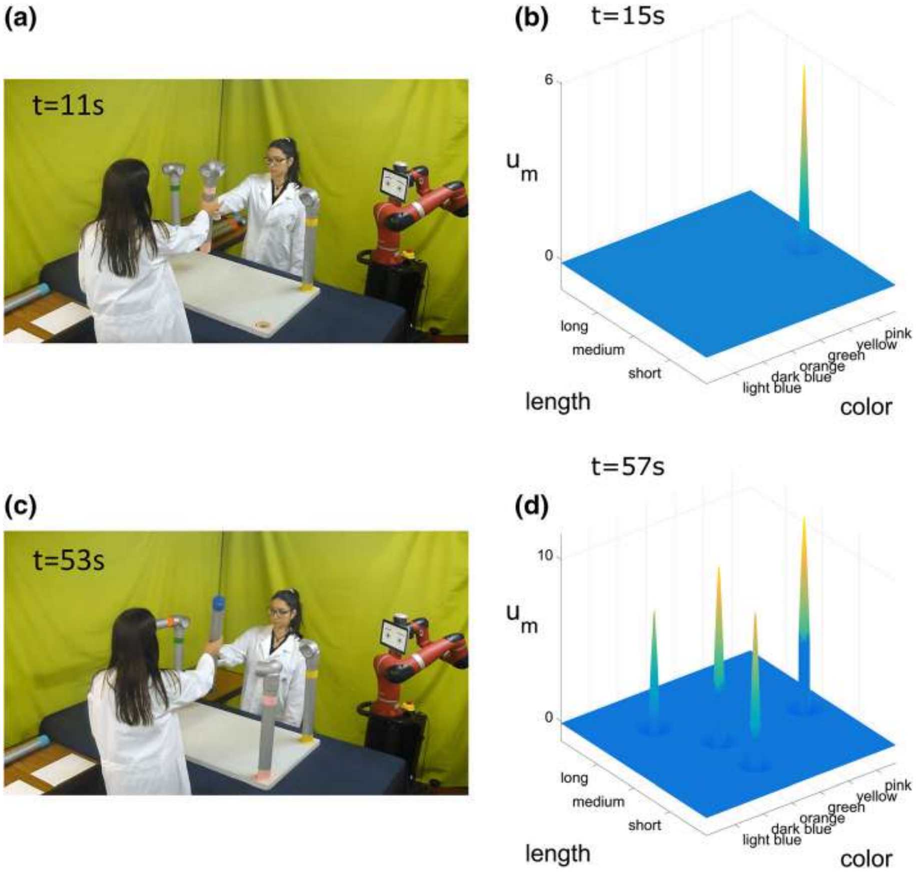

Fig. 6 Snapshots of the sequence demonstration (**a** and **c**) and corresponding bump formation in the memory field u_m spanned over the dimensions color and length (**b** and **d**) (color figure online)

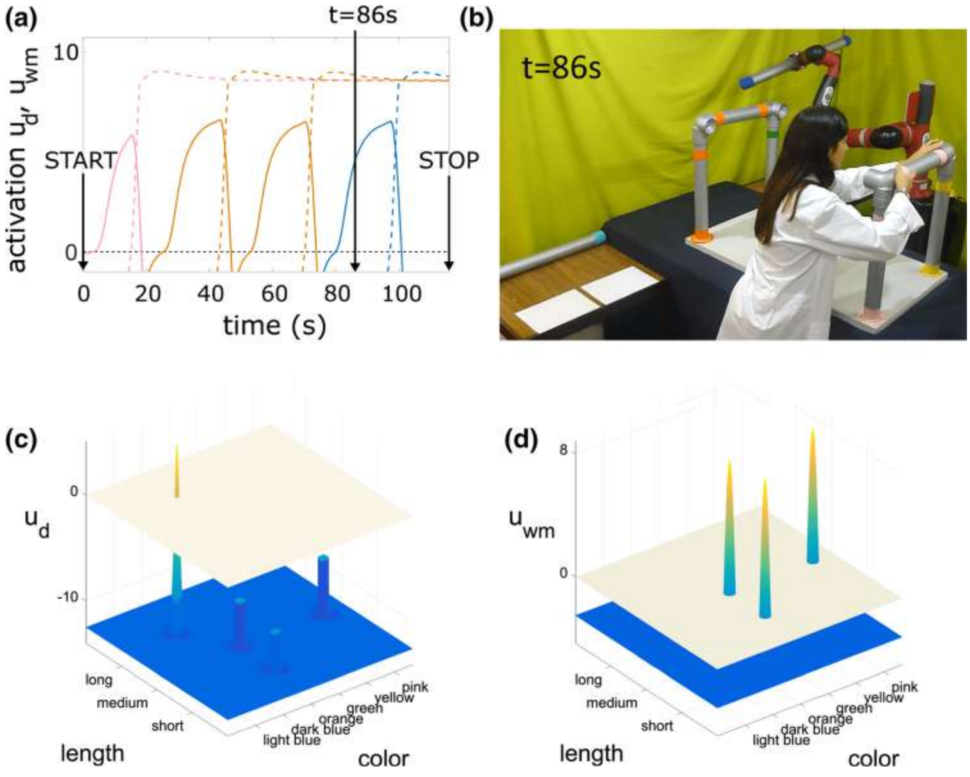

Fig. 8 **a** Comparison of the temporal evolution of population activities in the decision field u_d (solid line) and the working memory field u_{wm} (dashed line). The start and stop signals are presented at $t = 0$ and $t = 110$**s**, respectively. **b** The worker is still mounting a pipe, while Sawyer is already grasping the next one. **c** Bump formation in the decision field u_d, and **d** multi-bump pattern in the working memory field u_{wm} at time $t = 86$**s**

video snapshot (Fig. 8b) taken at time $t = 86$**s** shows the worker mounting a pipe that she had picked from her workspace. By comparing the population activity in the decision field at the same time (Fig. 8c), it becomes clear that Sawyer already starts the handover of the dark-blue-long pipe that the worker has to assemble next. Note that the duration of the whole sequence during joint execution is significantly longer than during demonstration (compare Fig. 7) since the operator assembles additional pipes located within reach on her side.

As shown in Figure 9, the robot is able to accelerate the planning of the entire transfer sequence to adapt to an operator executing the assembly steps with higher speed. This acceleration can be achieved by applying an additional input to the decision field. It may for instance represent a request gesture signaling to the robot that the operator awaits the next pipe Erlhagen and Bicho (2014). The additional input either increases the baseline activity h_{d_0} or the slope of linear ramp-to-threshold dynamics (15). Since in modern assembly manufacturing the ordering of activities is often left to the discretion of the operator, the robot should also be able to adapt to changes in serial order of task execution Wilcox et al. (2013). The activation-based learning implemented in the DNF model ensures that a single-task demonstration is sufficient to establish an activation gradient representing the new order. However, to memorize simultaneously the order preferences of different operators, activation gradients in separate memory fields have to be established (for a DNF representation of different serial orders in a single memory field see Sousa et al. (2015)). During joint task execution, the read-out of the correct memory representation can be ensured, in principle, by a multiplicative gating of the different memory inputs (defined by (14)) to the decision field u_d with a user-specific sensory signal. Only the memory pattern which gets this additional support preshapes u_d and consequently affects sequence planning.

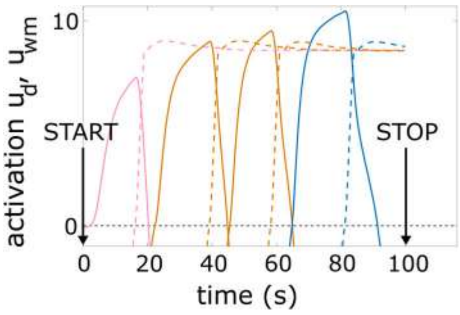

Fig. 9 Comparison of temporal evolution of population activities in the decision field u_d (solid line) and the working memory field u_{wm} (dashed line) for a faster sequence planning and execution. The start and stop signals are presented at times $t = 0$ and $t = 100$**s**, respectively

3.2 Value-based decision making in a material handling task

The second experiment is inspired by challenges for material handling robots in more flexible and therefore less predictable manufacturing processes Choe et al. (2015). Such robots play a central role in industrial assembly lines by transporting parts or subassembly parts between work-stations. Figure 10a shows a top view of a simulated environment in which a mobile robot equipped with a forklift has to pick up a bulky object positioned on a pallet to deliver it to an operator located in an uploading area. For future real-world tests, it is interesting to notice that the DNF approach to cognition is highly compatible with the attractor dynamics approach to autonomous navigation of mobile robots Bicho et al. (2000) (see Discussion). Recently, the approach has been successfully tested in a factory environment cluttered with stationary and moving obstacles Machado et al. (2019); Louro et al. (2019).

For the material handling task, the main challenge for the robot is to deal with an environment characterized by uncertainty. The searched object may arrive in a certain time interval ΔT with independent probabilities at two possible locations, A and B, which are hidden from the robot's camera view. To maximize the success rate of the object search over a longer time period, the robot has to adapt its choice behavior to the statistics of the environment.

We assume that during the duration T of the experiment, the robot is forced to make every ΔT time units a decision to search at A or B and then deliver the object at the upload area, or, in case of no success, return directly to the park position. For the evaluation of the robot performance thus exist $N = T/\Delta T$ trials. For simplicity, we further assume the existence of an external computer clock to trigger the start of each decision process. Note that neural integrator model (6) may be used as well to autonomously measure and reproduce time intervals Wojtak et al. (2019). We leave this interesting issue for future work (see

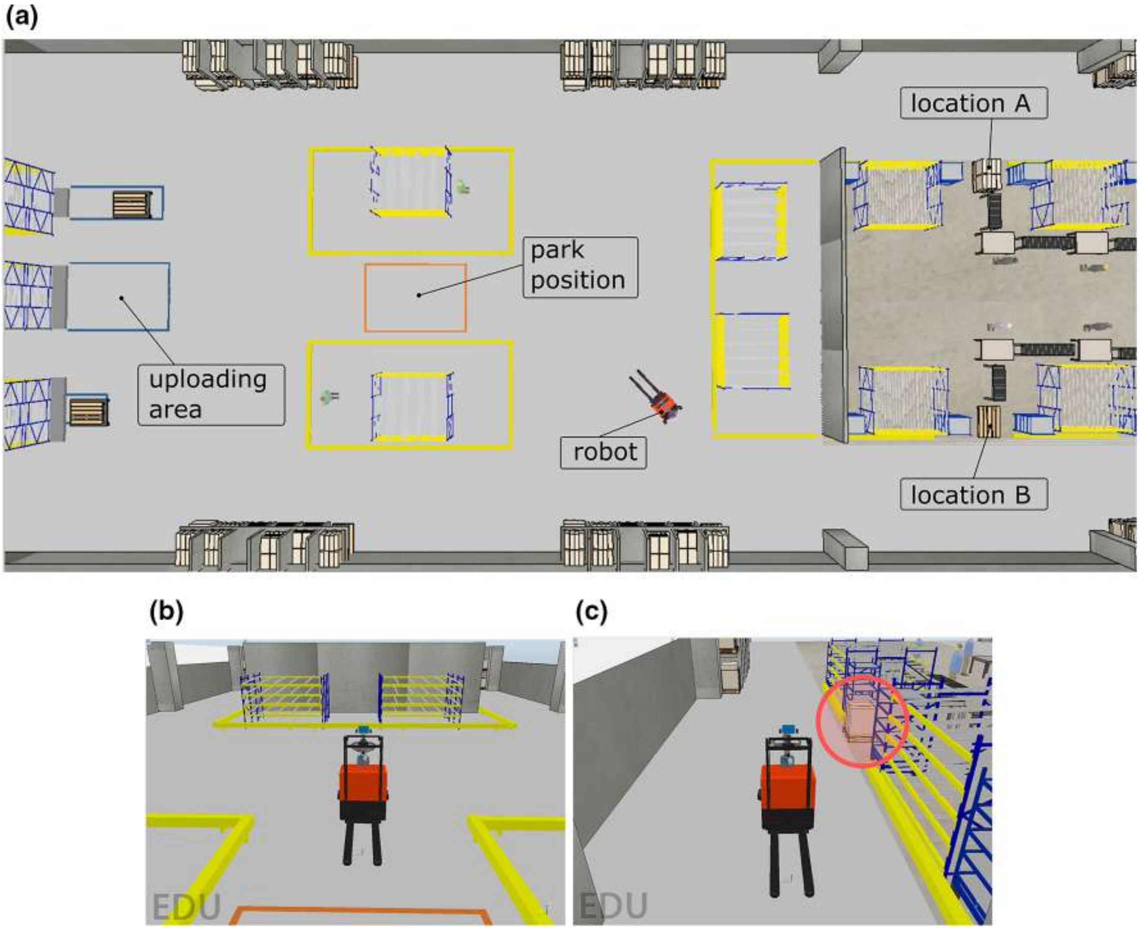

Fig. 10 **a** Top view of a simulated factory environment with a park position for the mobile robot, two target locations A and B, and an uploading area. **b** Robot at the moment of making a decision to search the object at location A (left side) or location B (right side). **c** Robot finds the object at location A (left) to pick it up and deliver it at the uploading area

Discussion). Since the probabilities of finding the specific object at the two locations are independent, in each trial, both locations, only one of the locations or none of the locations, guarantee success. Once the object has arrived at a particular location, it stays there until the robot picks it up.

3.2.1 Model architecture

The DNF model guiding the robot's decisions is inspired by findings in neurophysiological and behavioral studies investigating choice behavior in conceptually similar tasks with humans and other animals Lau and Glimcher (2005); Sugrue et al. (2004). They suggest that it is necessary to integrate in the decision process both past choices and past successes in order to capture an optimal probabilistic strategy in stationary as well as dynamically changing environments.

Figure 11 depicts the model architecture with three coupled fields spanned over the behavioral dimension "movement direction." Bumps in the decision field u_d at locations encoding target A to the left or target B to the right of the start position drive the activation of two neural integrators (u_r, v_r) and (u_c, v_c). They represent, respectively, cumulative success and cumulative choice for each of the options. The input to the u_c population is given by

$$I_c(x,t) = \int_{\Omega} w_{lat}(x-y) f(u_d(y,t)-\theta)\mathrm{d}y, \tag{17}$$

whereas the u_r population receives the input

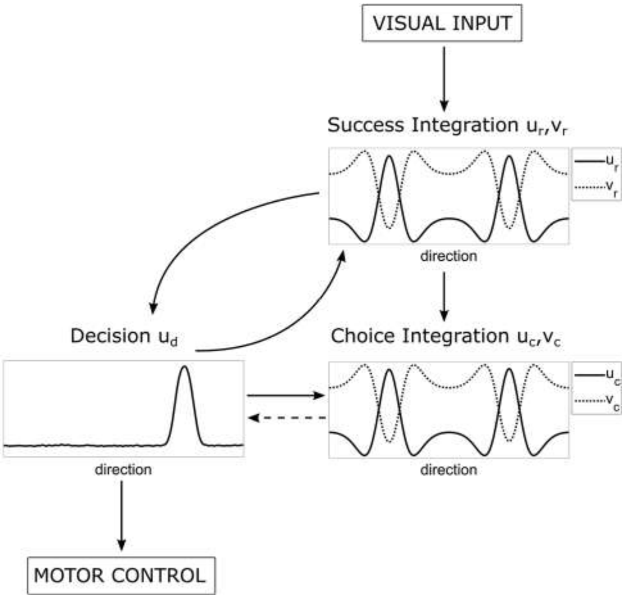

Fig. 11 Model architecture consisting of a decision field u_d, and fields for choice and success integration (u_c, v_c) and (u_r, v_r), respectively. Solid arrows indicate excitatory and dashed arrows inhibitory connections between fields

$$I_r(x,t) = K_r f(u_d(x,t)-\theta)R(x), \tag{18}$$

where $K_r > 0$ is a scaling factor. The integration of suprathreshold activation in u_d is gated by a function $R(x)$ which indicates whether the robot's camera has detected the color-coded object or not, that is, $R(x) = 1$ if the object is found and $R(x) = 0$ if not.

The decision field u_d receives in trial n the summed activation of the two success integration fields at the end of trial $n-1$, $u_{r_{n-1}} + v_{r_{n-1}}$, as excitatory input, whereas the summed activation of the two choice integration fields, $u_{c_{n-1}} + v_{c_{n-1}}$, is taken as inhibitory input:

$$\begin{aligned} I_d(x,t) = {} & (u_{r_{n-1}}(x) + v_{r_{n-1}}(x)) \\ & - c_d(u_{c_{n-1}}(x) + v_{c_{n-1}}(x)), \end{aligned} \tag{19}$$

where $c_d > 0$ is a scaling parameter. The different input signs may be understood by the fact that the simple strategy to always choose the target with higher success probability will not maximize the overall success rate. Due to the persistence of the object at the target location, the likelihood to find the object at the less likely location increases with the time elapsed from the last choices. The robot should thus visit from time to time also this location. The net effect of the inputs to u_d at the beginning of each trial is a preshaping of the neural subpopulations encoding locations A and B. It sets on a trial-by-trial basis the initial condition for the bump formation which is initiated by ramp-to-threshold dynamics (15). An open question is how many past trials should inform the current decision. The issue of the integration timescale is particularly relevant for environments in which the value (success frequency) of each option may change without warning Iigaya et al. (2019). Following the discussion in Sugrue et al. (2004), we address this issue by resetting the activation pattern of the integrators to their initial values after a fixed number of $N_{re} = 8$ trials.

3.2.2 Results—value-based decision making

In the first simulation experiment, we assume that the mobile robot knows the true success probability of each option either through learning in a stationary environment or by instruction. Under this condition, matching behavior is known to represent the optimal strategy for maximizing the overall success Lau and Glimcher (2005); Sugrue et al. (2004). The "matching law" states that an agent allocates choices in a proportion that matches the relative success experienced on these choices. For the present two-choice search task, this translates to write the following formula in an equation environment $N_i/(N_A + N_B) = S_i/(S_A + S_B)$, $i = A, B$, where N_i represents the number of times location i has been chosen and S_i the number of times the robot has found the object at that location. The prior task knowledge is modeled as additional Gaussian inputs to the subpopulations representing directions A and B with strengths proportional to the success probabilities Erlhagen and Schöner (2002):

$$I_{prob}(x) = A_{P_A} e^{\left(-(x-x_{P_A})^2/2\sigma_P^2\right)} + A_{P_B} e^{\left(-(x-x_{P_B})^2/2\sigma_P^2\right)}, \quad (20)$$

where $0 < A_{P_A} < 1$ and $0 < A_{P_B} < 1$.

The decision process in the first trial thus starts from a two-peak, bimodal resting state shown in Figure 12.

Figure 13 illustrates the decision process in four successive trials in which the prior probabilities are 30% and 40% for target A and target B, respectively. The activity patterns in u_d at the start (top) and the end of each trial (middle) together with the pattern in u_c at the end of the trial (bottom) are shown. The decision in the first trial reflects the prior information that the location B has higher success probability. In the second trial, the robot decides to go again to B despite the fact that the prior decision was unsuccessful and the inhibition from the bump in u_c has reduced the preshape at position B. The decision is again unsuccessful. In the third trial, the robot changes its decision and finds the object at A. The switch is due to the further increased inhibition from the pattern in u_c reflecting two B choices. In the fourth trial, the robot moves again to A due to the excitation from u_r, reflecting the success in the last trial (not shown), and the relatively smaller inhibition from u_c at A compared to B.

Table 1 summarizes the results of eight successive trials with the choice pattern (B, B, A, A, B, B, A, B). It shows that in the fourth trial no object was available and in three trials with an object available at A the wrong decision was made. To systematically analyze the search efficiency of the robot and compare it with the efficiency of two alternative search strategies, we run blocks of $N = 1000$ trials with different probability ratios for locations A and B. Search efficiency is defined as the total success achieved divided by the number of available objects. The simple alternative strategies are (1) to go always to the location with higher success probability, which we call the "Most Likely" (ML) strategy, and (2) to stay at the option if successful and change the option if not, known in the literature as "Win-Stay-Lose-Switch" (WSLS) strategy. The results in Table 2 show that for all tested ratios, the DNF model outperforms the two alternative strategies. The difference in search efficiency is in particular evident for cases in which both targets have similar but relatively low success probabilities (e.g., 30/40). Moreover, the distribution of choices among the two alternatives generated by the DNF model closely matches the success ratio.

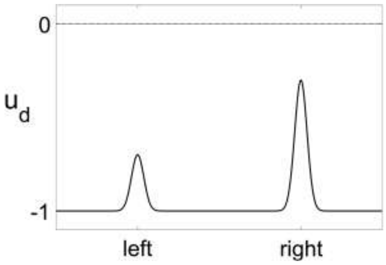

Fig. 12 Initial state of the decision field u_d at the beginning of trial $n = 1$ which is shaped by the prior knowledge about the success probabilities for A and B given by (20) with $A_{P_A} = 0.3$, $A_{P_B} = 0.7$, $\sigma_P = 0.75$

In a second experiment, we tested the situation in which the robot searches the object at A and B without prior knowledge about the success probability at each location. In addition, the probabilities change after $N = 100$ trials without warning and the robot has to adjust the distribution of decisions among the alternatives to this new situation to guarantee a high search efficiency. The first decision process in u_d now starts from equally pre-activated populations representing the possible movement directions A and B (compare Fig. 12). Figure 14a shows the activity distribution in u_d at the beginning of trial 16 in a block in which the probabilities for A and B are 30% and 40%, respectively. The activity pattern is shaped by the summed activations of the choice and success integrators, $u_c + v_c$ (Fig. 14b) and $u_r + v_r$ (Fig. 14c), respectively. The preshape pattern predicts the decision to go to target B. Figure 14 (bottom row) depicts the same activation snapshots after trial 110 which is part of a block in which the success probabilities have changed (at trial 100) to 60% for A and 20% for B. Now the preshape pattern in u_d predicts location A as current choice. Figure 15 compares for both blocks of $N = 100$ trials the cumulative choices of target A and target B (blue curve) with the average ratio of success. Two features of the robot's behavior are notable. First, the robot appears to adjust quite quickly its decisions to the unsigned change in success probabilities. Second, the robot chooses the location with lower probability more often than expected if it knew the true success probabilities, a well-known phenomenon described in the Cognitive Science literature as undermatching Iigaya et al. (2019); Lau and Glimcher (2005). Importantly, however, the robot shows a high search efficiency in both blocks with 85.7% and 73.8%, respectively. Again, the DNF model outperforms the two alternative strategies (compare Table 3).

In the present study, the two integrators were reset to their initial values every 8 trials. Intuitively it is clear that the integration time which maximizes the success rate for a given time period depends on the statistics of the environment. Under relatively stable conditions, it is better to take into account a large number of past choices and successes to estimate the current values of competing options. This is equivalent to introducing a constant prior belief in

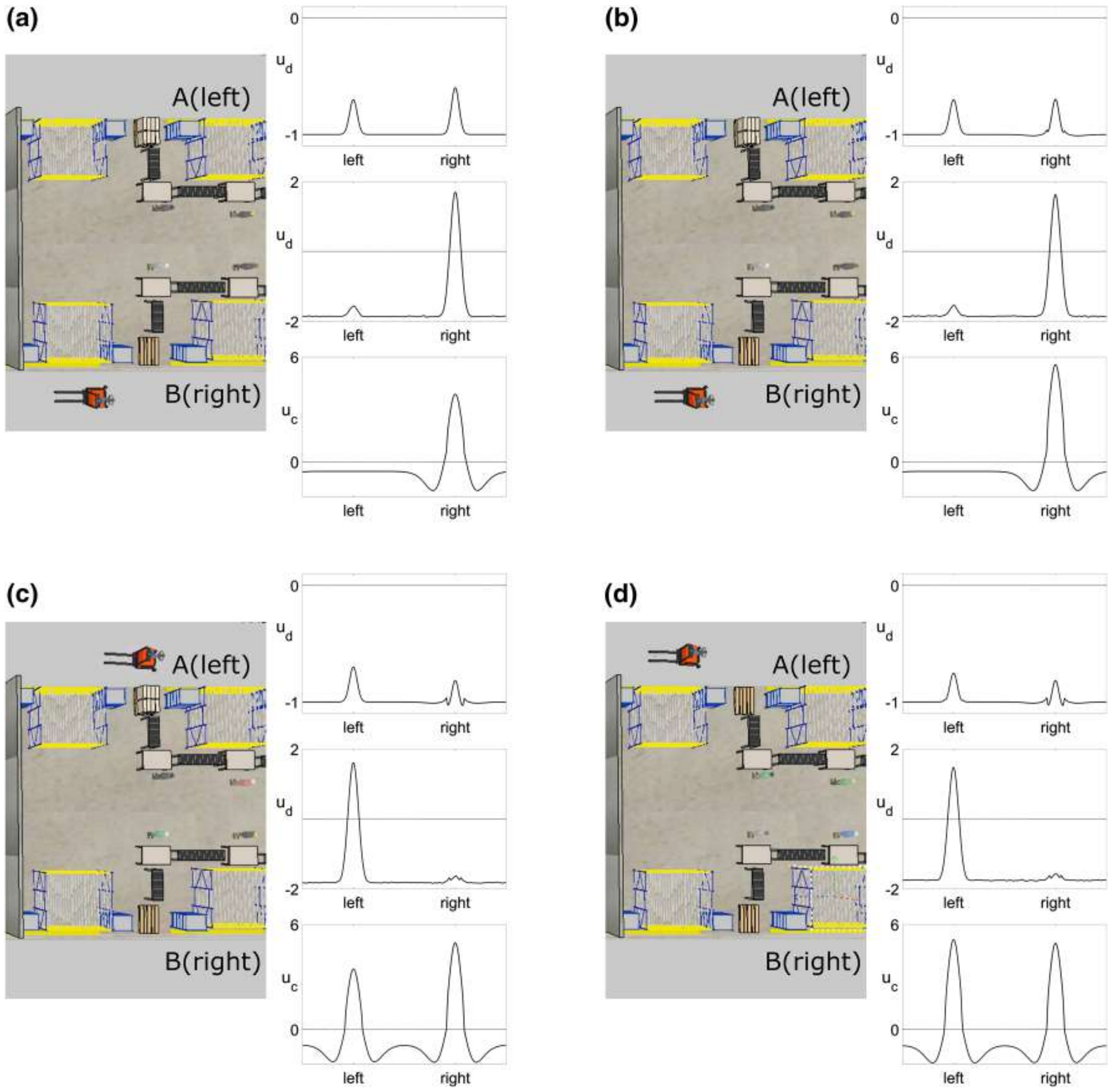

Fig. 13 Snapshots of the simulator illustrating the robot's decisions in the first 4 trials of the experiment. For details see the text and Table 1

Table 1 Availability of objects in locations A and B, robot's decisions and decision outcomes in a series of 8 successive trials. The success probabilities for A and B are 30% and 40%, respectively

Trial no.	Available objects		Decision	Success
	Left	Right		
1	1	0	Right	No
2	1	0	Right	No
3	1	0	Left	Yes
4	0	0	Left	No
5	0	1	Right	Yes
6	1	0	Right	No
7	1	1	Left	Yes
8	0	1	Right	Yes

the model like shown in Figure 12. In nonstationary environments, however, the robot should weight recent events more heavily, as old ones may not be informative anymore for the current choice. The question how the neural integrators might autonomously adapt the integration time window to optimize search efficiency is a current research topic Iigaya et al. (2019) but goes beyond the scope of this paper. It is interesting to notice however that the neural activations representing both options in (u_c, v_c) might be used to "count" the total number of past choices since the last reset. All one has to assume is that the reset mechanism is triggered when the total activation in the choice integration field reaches a pre-defined read-out threshold.

Table 2 The performance of the DNF model is compared with the performances of the ML and WSLS strategies in five blocks with different probability ratios for the two locations

Success probability (left/right)	Choice ratio (left/right)	Success ratio (left/right)	Search efficiency (DNF model)	Search efficiency (ML strategy)	Search efficiency (WSLS strategy)
30/70	24.0/76.0	23.5/76.5	76.5%	70.0%	64.7%
30/40	40.1/59.9	40.1/59.9	79.0%	57.1%	67.1%
60/70	43.2/56.8	42.4/57.6	65.9%	53.9%	55.4%
50/20	73.2/26.8	74.4/25.6	82.3%	71.4%	70.6%
30/30	49.7/50.3	49.8/50.2	81.0%	50.0%	69.7%

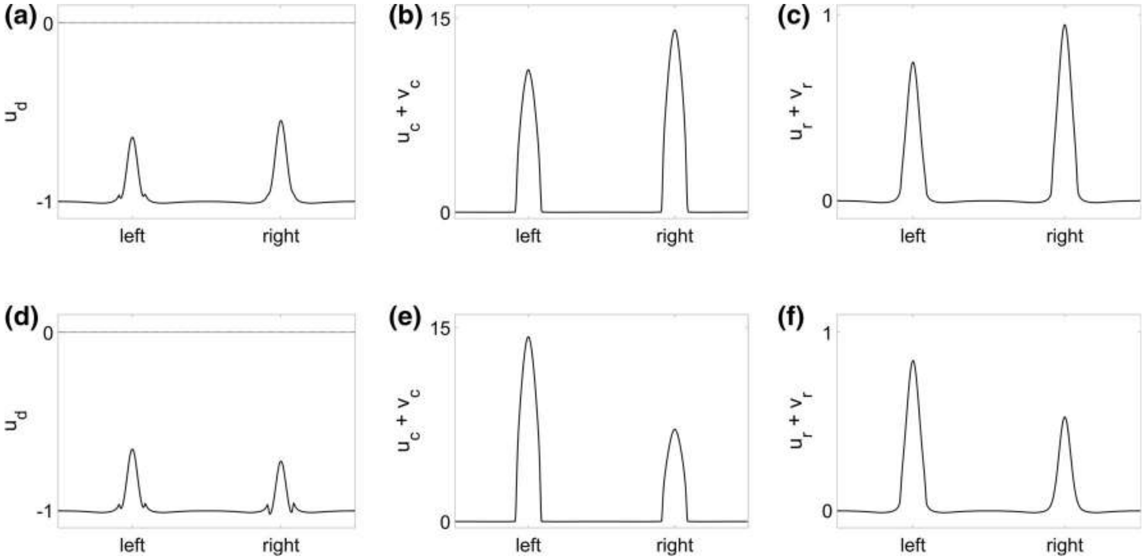

Fig. 14 **a**–**c** First block of trials with 30% and 40% success probability for A and B, respectively. The resting state at the beginning of trial 16 is shown in the decision field u_d (**a**) which is shaped by the inputs from the choice integrator (**b**) and the success integrator (**c**). **d**–**f** Second block of trials with 60% and 20% success probability for A and B, respectively. The resting state at the beginning of trial 110 in shown in the decision field u_d (**d**) which is shaped by the inputs from the choice integrator (**e**) and the success integrator (**f**)

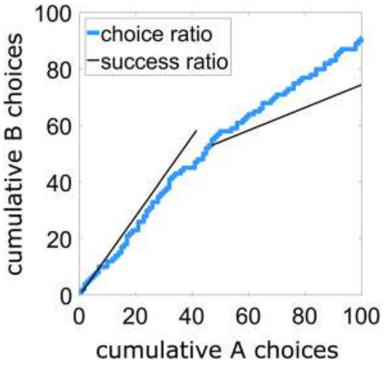

Fig. 15 Dynamic matching behavior. The blue curve indicates the cumulative choices of target A and target B. The black lines represent the average ratio of success (B/A) within each of the two blocks of $N = 100$ trials: B/A = 40/30 in the 1st block and 20/60 in the 2nd block. Search efficiency: 85.7% (1st block) and 73.8% (2nd block)

4 Discussion

In this study, we tested neurocomputational processing principles that support natural and efficient human–robot cooperation by endowing the robot with true autonomy in planning and decision making. The two test scenarios are taken from modern manufacturing and assembly environments that are characterized by a high variability in the built process. To provide a productivity benefit, a robotic assistant should be able to efficiently acquire and adapt knowledge about the order of activities and workflow and to make decisions in partly unknown environments.

The DNF approach is consistent with the hypothesis that computational principles supporting the flexible control of cognitive behaviors may be understood from a dynamical systems perspective Remington et al. (2018); Schöner

Table 3 The performance of the DNF model is compared with the performances of the ML and WSLS strategies in a dynamic environment with two blocks with different probability ratios for the two locations. The change in the probabilities after $N = 100$ trials is not signaled to the robot. For the ML strategy, we assume that the robot still believes in the second block that location B is the more likely one

Block	Success probability (left/right)	Choice ratio (left/right)	Success ratio (left/right)	Search efficiency (DNF model)	Search efficiency (ML strategy)	Search efficiency (WSLS strategy)
1	30/40	47.0/53.0	41.7/58.3	85.7 %	57.1 %	71.4%
2	60/20	58.0/42.0	71.2/28.8	73.8 %	25.0 %	66.2%

(2019). Computations determined by strong recurrent interactions between neurons can be rapidly reconfigured without any change in the network structure by controlling the system's inputs and initial conditions. In the two test scenarios, the robot's decision process is modeled in the field u_d as a transition from a stable resting state to a bump attractor. The resting state appears to be preshaped by inputs from connected populations that represent memorized sensory information about serial order of object transfers or the accumulated evidence about the choice and success history of competing target locations. For the DNF framework, the main novelty is that stable bump solutions of the two-field model support a continuous temporal integration of sensory and other evidence. There is no need to refer to additional processing mechanisms that are specifically designed to model stable bumps with a range of possible amplitudes, (e.g., Erlhagen and Schöner (2002); Ferreira et al. (2014); Koulakov et al. (2002)).

Our neurodynamics approach with brain-like representations differs fundamentally from more traditional AI approaches to human–robot cooperation that use logic-based knowledge representations operating on high-level symbols. Their abstract meaning is easy for humans to understand which favors human–robot interactions based on spoken commands Agostini et al. (2017). A more intuitive interaction with a human operator can be achieved in principle by enriching the robot's knowledge base with heuristics and preferences (formulated as production rules) that humans use in a specific task. A control architecture based on SOAR is a modern example which pursues this paradigm Mayer et al. (2011). The main drawback of the approach based on static symbols is that the representations and task rules have to be hand-crafted by a human expert since the robot is not able to self-generate internal representations. Having the human programmer in the loop to cope with dynamically changing task conditions of a modern manufacturing environment represents an often intractable burden. In the DNF approach, the fields are spanned over dimensions that a human operator may identify as being task relevant. The multi-bump pattern in u_m is not only able to represent all possible combinations of sensed object features in the continuous color-size space but also automatically adapts to different orders and lengths (including repeated items, Ferreira et al. (2014)) of demonstrated handover or other object manipulation sequences. The computational principles guiding the trial-by-trial target selection in the decision field spanned over movement direction can be easily extended to more than two competing targets and additional input factors that may bias the decision process (e.g., travel effort/time).

The presented computational principles and results may be used to address several key challenges that have been identified for close human–robot collaborative work in manufacturing environments Lin (2017); Tsarouchi et al. (2016). First, preferences about task completion are prone to change since the order of activities in many manual processes is left to the discretion of the human operators Wilcox et al. (2013). As already discussed before, the learning by demonstration paradigm can be in principle applied to quickly teach different serial orders represented by activation gradients in separate dynamic fields. In a recent robotics experiment Sousa et al. (2015), we have used the spontaneous recall of different gradient-based sequence memories as input to a recurrent neural network (RNN). It slowly and gradually established through weight-based learning long-term associations between neural representations of different assembly steps. The ultimate goal was to extract generalized task knowledge from user demonstrations which includes the possibility to follow multiple serial orders to achieve a desired end state and the learning of causal relationships between assembly steps (e.g., steps 1 to 4 have to be finished in arbitrary order before step 5 can be executed). The spontaneous recall of serial order memory during off-line periods, which the self-stabilizing properties of the neural field representations support, is of great practical importance for robot learning since human tutors cannot be expected to demonstrate the task execution many times.

As shown in Figs. 6 and 8, the one-shot learning of serial order allows the robot to act in advance of the operator's need, rather than just reacting to a request. For efficient and successful joint action it is highly desirable that an assistant shows such a proactive attitude also when the operator makes an error Lin (2017). To avoid time-consuming disassembly steps, the robot should be able to detect and communicate errors before they manifest in the construction work (a capacity termed proactive resilience in engineered industrial systems Wang et al. (2018a)). In previous research, we have developed a control architecture for natural HRI based on the DNF framework. It implements a highly context-sensitive mapping of an observed action of the co-worker onto an adequate complementary robot behavior. This mapping takes into account different task-related and user-related factors including an error monitoring capacity Bicho et al. (2011a, 2012); Silva et al. (2016). Neural populations encode in their suprathreshold activity a mismatch between which assembly step the operator should execute (shared task knowledge) and the predicted assembly step inferred from the observed motor behavior. For the pipe assembly paradigm, assuming that the robot has learned the serial order of the entire assembly sequence, it may detect a serial order error at the moment when the operator grasps a specific pipe. An adequate complementary robot behavior would be to inform the operator verbally or with a communicative gesture about the error Bicho et al. (2010).

Ideally, the robot should be able to inform the operator even before she has started to grasps the pipe. In many circumstances, humans are able to infer the motor intention of ongoing goal-directed movements solely based on the observed kinematics of arm and hand Zunino et al. (2020). We are currently exploring the application of deep neural networks (DNNs) LeCun et al. (2015) as a data-driven approach to human action recognition and understanding in a manufacturing environment. The central concept behind all deep learning methodology is the automated discovery of a hierarchy of features or representations that reflect the semantic content of raw sensory data on different levels of abstraction. These representations may then be used to implement hierarchical processes that predict sensory events. Recent experiments show that DNN architectures may outperform humans in such complex classification tasks like predicting different final intentions of the same class of grasping movements (e.g., grasping for placing or grasping for passing a certain object, Zunino et al. (2020)). For typical HRI applications, the type of the part or the tool the operator is going to manipulate may help the intention classification (e.g., grasping a screw driver Wang et al. (2018b)).

While there is a growing interest in using deep learning in robotics, there are currently still only few robotics applications that exploit the function approximation capacity of deep networks to mediate the learning, storage and interpolation of input representations to predict the value of states and actions. One reason for this situation is that, in practice, deep architectures require a huge amount of (often labeled) training data which are quite expensive to collect with physical robots (for review see Sünderhauf et al. (2018)). Even if simulated data are used in part to substitute the real-world experiments, the amount of training time is usually huge when trying to optimize the large amount of network parameters involved. We see the dynamic field approach together with a learning by demonstration paradigm as a highly attractive compromise for HRI in an industrial environment. It lies on a spectrum along which the relative and complementary contributions of programming and data vary. On the one end of the spectrum, classical AI focuses on reasoning with rules defined by experts, with little or no adaptation and learning involved. On the other end of the spectrum lies deep learning which replaces programming by training of a network architecture (the training procedure itself is a programmed algorithm) with the promise to find relevant representations automatically. As shown in the two robotics examples, the DNF approach relies on a limited number of computational principles such as a memory function and a continuous neural integrator based on persistent activity that are built into the system. Importantly, these principles are powerful and general enough to support highly efficient learning of other related tasks. For instance, we have recently shown that the build-up of an activation gradient during task demonstration can be used to learn a joint representation of the serial order and the timing of perceptual and motor events. This allows a robot, in principle, to predict temporal information online, a capacity that is missing in most current robotics systems Ferreira et al. (2020). Teaching by demonstration is attractive since it facilitates task specification by individuals without expertise in robot programming. Moreover, like a “good” teacher in a social learning situation, an expert operator may draw the attention of a learning robot to features that guide sequence learning in an assembly paradigm or may explicitly instruct the robot to take the accumulated evidence from past choices and rewards into account in its current search decision. In case that sufficient data are available and adequate features are difficult to identify even for a human expert (like in the example of reading intention from motion), deep learning techniques can be applied to automatically extract low-dimensional features on which field representations can be built.

The focus of the presented experiments was on showing autonomous decision making of a robot guided by past experiences. In dynamically changing manufacturing environments populated by other agents, it is important that the robot also takes online sensory information into account. A relevant example is a search task in which a team of two mobile robots is looking for specific objects as supply for a manufacturing process. To guarantee an efficient team strategy, the decision process of each robot should be biased by predicted movement targets of the other robot to prevent from searching at the same location. This prediction could be based on a trajectory extrapolation of the current heading direction provided by the robot camera. The predicted target location of the co-worker then acts as additional inhibitory input to the target selection process sketched in Fig. 11 (for a DNF implementation see Erlhagen and Bicho (2006)).

In case that sensory information about potential target locations distributed in a relatively large search area is available, a two-step decision process proposed for sensory networks guiding the navigation of multiple mobile agents could be beneficial Wei et al. (2017). First, a decision is made to move to an area with the highest probability of finding a particular object (this probability may change dynamically if other search robots are involved). The neural integrator accumulates sensory and past information leading to clusters of broad activity distributions in direction space. The cluster with the currently highest total activation will be selected as a first target in a competitive decision process. Once the robot has reached the target area, a refined search with more localized representations of sensory and other evidence can be performed. In both steps, the selected target direction is used as input to a path planning module driven by a nonlinear attractor dynamics in which targets are modeled as attractive and obstacles as repulsive forces. The nonlinear attractor approach to autonomous navigation has been successfully tested in highly dynamic industrial environments Louro et al. (2019); Machado et al. (2019).

In future work, we plan to exploit the neural computations of the integrator model in an existing DNF-based architecture for fluent joint action in a shared task. The challenge will be to smoothly integrate a series of actions serving an operator like in the present joint assembly study with assembly work that the robot itself performs. Since a tight synchronization of activities between robot and operator is often required to achieve a shared goal Wilcox et al. (2013), the robot should not only be able to adapt to changing preferences for workflow but also to flexibly time its goal-directed actions. We have recently applied the neural integrator model to measure and reproduce time intervals between sensory-motor events Wojtak et al. (2019). As a key processing mechanism to achieve adaptive motor timing, a bump reflecting measured duration in its amplitude affects either the resting state or the slope of the ramp-to-threshold dynamics in the decision field.

In conclusion, the experimental results support the notion that the brain-like computations offered by the DNF framework may lead the way to endow robots with higher cognition necessary to autonomously operate in dynamic environments. Persistent neural population activity that DNF models explain has been linked to a variety of fundamental cognitive processes like memory, planning, decision making, timing and learning. The capacity of the neural integrator to continuously accumulate sensory and past information over time is of critical importance to the organization of appropriate behavior in human–robot collaborative work.

Acknowledgements The work received financial support from FCT through the PhD fellowships PD/BD/128183/2016 and SFRH/BD/124912/2016, the project “Neurofield” (PTDC/MAT-APL/31393/2017) and the research centre CMAT within the project UID/MAT/00013/2020.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Initial conditions and parameters

Initial conditions

Assembly task

For the model simulations, the initial conditions of the fields governed by the Amari dynamics, u_{per}, u_{on}, u_{wm} and u_d are defined by the inhibition parameter h. For the coupled two-field model the initial conditions are given by:

$$u_m(x, y, 0) = -1, \tag{21a}$$

$$v_m(x, y, 0) = -0.25 - u_m(x, y, 0). \tag{21b}$$

Value-based decision-making task

The initial condition of the decision field at the start of simulation trial n is given by:

$$u_{d_n}(x, 0) = \begin{cases} I_{prob}(x) - h_{d_0} \quad \text{if} & n = 1, \\ (u_{r_{n-1}}(x) + v_{r_{n-1}}(x)) & \\ \quad - c_d(u_{c_{n-1}}(x) + v_{c_{n-1}}(x)) - h_{d_0}, & \text{otherwise.} \end{cases} \tag{22}$$

The initial condition for the choice integration layer (u_c, v_c) in the first trial and after each reset is given by

$$u_c(x, 0) = -0.5, \tag{23a}$$

$$v_c(x, 0) = -u_c(x, 0). \tag{23b}$$

The initial condition for the success integration layer (u_r, v_r) in the first trial and after each reset is given by

$$u_r(x,0) = -0.5, \tag{24a}$$

$$v_r(x,0) = I_{prob}(x) - u_r(x,0). \tag{24b}$$

Model parameters

See Tables 4 and 5.

Table 4 Parameter values of the field equations used for sequence learning and planning

Perception field u_{per}	
τ_{per}	3
w_{lat}	$A_{lat} = 6,\ \sigma_{lat} = 0.65,\ g_{lat} = 2$
I_t	$A_I = 6,\ \sigma_I = 0.75,\ g_I = 0$
h_{per}	1.5

Sequence onset field u_{on}	
τ_{on}	1
w_{mex}	$A_{ex} = 4,\ A_{in} = 2,\ \sigma_{ex} = 1.5,\ \sigma_{in} = 2.5,\ g_{mex} = 0.15$
I_{on}	$A_I = 1.25,\ \sigma_I = 1.5,\ g_I = 0$
h_{on}	0.5

Memory field u_m, v_m	
τ_m	3
w_{mex}	$A_{ex} = 6,\ A_{in} = 3.5,\ \sigma_{ex} = 1.5,\ \sigma_{in} = 2.25,\ g_{mex} = 0$
κ	0.04

Decision field u_d	
τ_d	2
w_{lat}	$A_{lat} = 5,\ \sigma_{lat} = 0.75,\ g_{lat} = 1.5$
h_d	$h_{d_0} = 13.5$, $\kappa = 0.0085$ (fast recall), $\kappa = 0.004$ (slower recall)

Working memory field u_{wm}	
τ_{wm}	2
w_{osc}	$A_{osc} = 2.4,\ b = 0.7$
h_{wm}	2.5

Table 5 Parameter values of the field equations used for value-based decision making

Decision field u_d	
τ_d	1
w_{lat}	$A_{lat} = 2,\ \sigma_{lat} = 0.75,\ g_{lat} = 1$
$\epsilon^{1/2}$	0.025
h_d	$h_{d_0} = 1,\ \tau_{h_d} = 7$
c_d	0.05

Choice integration field u_c, v_c	
τ_c	1
w_{mex}	$A_{ex} = 4,\ A_{in} = 2,\ \sigma_{ex} = 1.5,\ \sigma_{in} = 3,\ g_{mex} = 0.25$

Success integration field u_r, v_r	
τ_r	1
w_{mex}	$A_{ex} = 4,\ A_{in} = 2,\ \sigma_{ex} = 1.5,\ \sigma_{in} = 3,\ g_{mex} = 0.25$
K_r	$K_r = 0.02$ (static case), $K_r = 0.035$ (dynamic case)

Numerical model simulations

Numerical simulations of the model were done in MATLAB using a forward Euler method with parameters given in Table 6. To compute the spatial convolution of w and f we employ a fast Fourier transform (FFT), using MATLAB's in-built functions `fft` and `ifft` to perform the Fourier transform and the inverse Fourier transform, respectively.

Table 6 Spatial and temporal discretization of the neural field models

Assembly task	
space	$L = 50,\ N = 1000,\ \mathrm{dx} = 2L/N = 0.1$
time	$\Delta T = 100,\ M = 10000,\ \mathrm{dt} = \Delta T/M = 0.01$

Value-based decision making	
space	$L = 20,\ N = 1000,\ \mathrm{dx} = 2L/N = 0.04$
time	$\Delta T = 10,\ M = 1000,\ \mathrm{dt} = \Delta T/M = 0.01$

References

Agostini A, Torras C, Woergoetter F (2017) Efficient interactive decision-making framework for robotic applications. Artif Intell 247:187–212. https://doi.org/10.1016/j.artint.2015.04.004

Amari S (1977) Dynamics of pattern formation in lateral-inhibition type neural fields. Biol Cybern 27(2):77–87. https://doi.org/10.1007/BF00337259

Bannat A, Bautze T, Beetz M, Blume J, Diepold K, Ertelt C, Geiger F, Gmeiner T, Gyger T, Knoll A et al (2010) Artificial cognition in production systems. IEEE Trans Autom Sci Eng 8(1):148–174. https://doi.org/10.1109/TASE.2010.2053534

Bicho E, Mallet P, Schöner G (2000) Target representation on an autonomous vehicle with low-level sensors. Int J Robot Res 19(5):424–447. https://doi.org/10.1177/02783640022066950

Bicho E, Louro L, Erlhagen W (2010) Integrating verbal and nonverbal communication in a dynamic neural field architecture for human-robot interaction. Front Neurorobot 4:5. https://doi.org/10.3389/fnbot.2010.00005

Bicho E, Erlhagen W, Louro L, Costa e Silva E, Silva R, Hipolito N (2011a) A dynamic field approach to goal inference, error detection and anticipatory action selection in human-robot collaboration. New Front Human-Robot Interact. https://doi.org/10.1075/ais.2.10bic

Bicho E, Erlhagen W, Louro L, e Silva EC (2011b) Neuro-cognitive mechanisms of decision making in joint action: a human-robot interaction study. Human Mov Sci 30(5):846–868. https://doi.org/10.1016/j.humov.2010.08.012

Bicho E, Erlhagen W, Sousa E, Louro L, Hipólito N, Silva E, Silva R, Ferreira F, Machado T, Hulstijn M, et al. (2012) The power of prediction: robots that read intentions. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, pp 5458–5459, https://doi.org/10.1109/IROS.2012.6386297

Billard A, Calinon S, Dillmann R, Schaal S (2008) Robot programming by demonstration. Springer Handb Robot. https://doi.org/10.1007/978-3-540-30301-5_60

Brody CD, Hanks TD (2016) Neural underpinnings of the evidence accumulator. Curr Opin Neurobiol 37:149–157. https://doi.org/10.1016/j.conb.2016.01.003

Brody CD, Romo R, Kepecs A (2003) Basic mechanisms for graded persistent activity: discrete attractors, continuous attractors, and dynamic representations. Curr Opin Neurobiol 13(2):204–211. https://doi.org/10.1016/S0959-4388(03)00050-3

Cain N, Shea-Brown E (2012) Computational models of decision making: integration, stability, and noise. Curr Opin Neurobiol 22(6):1047–1053. https://doi.org/10.1016/j.conb.2012.04.013

Choe P, Tew JD, Tong S (2015) Effect of cognitive automation in a material handling system on manufacturing flexibility. Int J Product Econ 170:891–899. https://doi.org/10.1016/j.ijpe.2015.01.018

Coombes S (2005) Waves, bumps, and patterns in neural field theories. Biol Cybern 93(2):91–108. https://doi.org/10.1007/s00422-005-0574-y

Coombes S, beim Graben P, Potthast R, Wright J (2014) Neural fields: theory and applications. Springer, Berlin. https://doi.org/10.1007/978-3-642-54593-1

Cox BR, Krichmar JL (2009) Neuromodulation as a robot controller. IEEE Robot Autom Mag 16(3):72–80. https://doi.org/10.1109/MRA.2009.933628

Curtis CE, Lee D (2010) Beyond working memory: the role of persistent activity in decision making. Trends Cognit Sci 14(5):216–222. https://doi.org/10.1016/j.tics.2010.03.006

Erlhagen W, Bicho E (2006) The dynamic neural field approach to cognitive robotics. J Neural Eng 3:36–54. https://doi.org/10.1088/1741-2560/3/3/R02

Erlhagen W, Bicho E (2014) A dynamic field approach to natural and efficient human-robot collaboration. In: Pothast R, Weight J, Coombes S, Beim Graben P (eds) Neural fields: theory and applications. Springer, Berlin, pp 341–365. https://doi.org/10.1007/978-3-642-54593-1_13

Erlhagen W, Schöner G (2002) Dynamic field theory of movement preparation. Psychol Rev 109(3):545. https://doi.org/10.1037/0033-295X.109.3.545

Erlhagen W, Mukovskiy A, Bicho E, Panin G, Kiss C, Knoll A, Van Schie H, Bekkering H (2006) Goal-directed imitation for robots: a bio-inspired approach to action understanding and skill learning. Robot Auton Syst 54(5):353–360. https://doi.org/10.1016/j.robot.2006.01.004

Faubel C, Schöner G (2008) Learning to recognize objects on the fly: a neurally based dynamic field approach. Neural Netw 21(4):562–576. https://doi.org/10.1016/j.neunet.2008.03.007

Ferreira F, Erlhagen W, Sousa E, Louro L, Bicho E (2014) Learning a musical sequence by observation: A robotics implementation of a dynamic neural field model. In: Development and Learning and Epigenetic Robotics (ICDL-Epirob), 2014 Joint IEEE International Conferences on, pp 157–162, https://doi.org/10.1109/DEVLRN.2014.6982973

Ferreira F, Erlhagen W, Bicho E (2016) Multi-bump solutions in a neural field model with external inputs. Phys D Nonlinear Phenomen 326:32–51. https://doi.org/10.1016/j.physd.2016.01.009

Ferreira F, Wojtak W, Sousa E, Louro L, Bicho E, Erlhagen W (2020) Rapid learning of complex sequences with time constraints: a dynamic neural field model. IEEE Trans Cognit Develop Syst. https://doi.org/10.1109/TCDS.2020.2991789

Haller M, Case J, Crone NE, Chang EF, King-Stephens D, Laxer KD, Weber PB, Parvizi J, Knight RT, Shestyuk AY (2018) Persistent neuronal activity in human prefrontal cortex links perception and action. Nat Human Behav 2(1):80. https://doi.org/10.1038/s41562-017-0267-2

Herrnstein RJ (1961) Relative and absolute strength of response as a function of frequency of reinforcement. J Exp Anal Behav 4(3):267–272. https://doi.org/10.1901/jeab.1961.4-267

Histed MH, Pasupathy A, Miller EK (2009) Learning substrates in the primate prefrontal cortex and striatum: sustained activity related to successful actions. Neuron 63(2):244–253. https://doi.org/10.1016/j.neuron.2009.06.019

Hu SJ, Ko J, Weyand L, ElMaraghy HA, Lien TK, Koren Y, Bley H, Chryssolouris G, Nasr N, Shpitalni M (2011) Assembly system design and operations for product variety. CIRP Ann 60(2):715–733. https://doi.org/10.1016/j.cirp.2011.05.004

Huang CM, Mutlu B (2016) Anticipatory robot control for efficient human-robot collaboration. In: The eleventh ACM/IEEE international conference on human robot interaction, IEEE Press, pp 83–90, https://doi.org/10.1109/HRI.2016.7451737

Iigaya K, Ahmadian Y, Sugrue LP, Corrado GS, Loewenstein Y, Newsome WT, Fusi S (2019) Deviation from the matching law reflects an optimal strategy involving learning over multiple timescales. Nat Commun 10(1):1466. https://doi.org/10.1038/s41467-019-09388-3

Koene A, Remazeilles A, Prada M, Garzo A, Puerto M, Endo S, Wing AM (2014) Relative importance of spatial and temporal precision for user satisfaction in human-robot object handover interactions. In: Third International Symposium on New Frontiers in Human-Robot Interaction

Koulakov AA, Raghavachari S, Kepecs A, Lisman JE (2002) Model for a robust neural integrator. Nat Neurosci 5(8):775. https://doi.org/10.1038/nn893

Kozma R (2008) Intentional systems: review of neurodynamics, modeling, and robotics implementation. Phys Life Rev 5(1):1–21. https://doi.org/10.1016/j.plrev.2007.10.002

Krüger J, Lien TK, Verl A (2009) Cooperation of human and machines in assembly lines. CIRP Ann 58(2):628–646. https://doi.org/10.1016/j.cirp.2009.09.009

Laing CR, Troy WC, Gutkin B, Ermentrout GB (2002) Multiple bumps in a neuronal model of working memory. SIAM J Appl Math 63(1):62–97. https://doi.org/10.1137/S0036139901389495

Lau B, Glimcher PW (2005) Dynamic response-by-response models of matching behavior in rhesus monkeys. J Exp Anal Behav 84(3):555–579. https://doi.org/10.1901/jeab.2005.110-04

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444. https://doi.org/10.1038/nature14539

Lemaignan S, Warnier M, Sisbot EA, Clodic A, Alami R (2017) Artificial cognition for social human-robot interaction:an implementation. Artif Intell 247:45–69. https://doi.org/10.1016/j.artint.2016.07.002

Lin Y (2017) Toward intelligent human machine interactions. Mech Eng 139(06):S4–S8. https://doi.org/10.1115/1.2017-Jun-4

Lomp O, Richter M, Zibner SK, Schöner G (2016) Developing dynamic field theory architectures for embodied cognitive systems with cedar. Front Neurorobotics 10:14. https://doi.org/10.3389/fnbot.2016.00014

Louro L, Malheiro T, Guimarães P, Machado T, Monteiro S, Vaz Silva S, Erlhagen W, Bicho E (2019) Motion control for autonomous Tugger vehicles in dynamic factory floors shared with human operators. In: IEEE 45th Annual Conference of Industrial Electronics Society (IECON'2019), Lisbon, Portugal, October 14-17

Machado T, Malheiro T, Monteiro S, Erlhagen W, Bicho E (2019) Attractor dynamics approach to joint transportation by autonomous robots: theory, implementation and validation on the factory floor. Auton Robots 43(3):589–610. https://doi.org/10.1007/s10514-018-9729-2

Mayer MP, Schlick CM, Ewert D, Behnen D, Kuz S, Odenthal B, Kausch B (2011) Automation of robotic assembly processes on the basis of an architecture of human cognition. Product Eng 5(4):423–431. https://doi.org/10.1007/s11740-011-0316-z

Pardowitz M, Knoop S, Dillmann R, Zollner RD (2007) Incremental learning of tasks from user demonstrations, past experiences, and vocal comments. IEEE Trans Syst Man Cybern Part B Cybern 37(2):322–332. https://doi.org/10.1109/TSMCB.2006.886951

Pinheiro M, Bicho E, Erlhagen W (2010) A dynamic neural field architecture for a pro-active assistant robot. In: 2010 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics, IEEE, pp 777–784, https://doi.org/10.1109/BIOROB.2010.5627812

Rankin J, Avitabile D, Baladron J, Faye G, Lloyd DJ (2014) Continuation of localized coherent structures in nonlocal neural field equations. SIAM J Sci Comput 36(1):B70–B93. https://doi.org/10.1137/130918721

Remington ED, Egger SW, Narain D, Wang J, Jazayeri M (2018) A dynamical systems perspective on flexible motor timing. Trends Cognit Sci 22(10):938–952. https://doi.org/10.1016/j.tics.2018.07.010

Rhodes BJ, Bullock D, Verwey WB, Averbeck BB, Page MP (2004) Learning and production of movement sequences: behavioral, neurophysiological, and modeling perspectives. Human Movement Sci 23(5):699–746. https://doi.org/10.1016/j.humov.2004.10.008

Sakai Y, Okamoto H, Fukai T (2006) Computational algorithms and neuronal network models underlying decision processes. Neural Netw 19(8):1091–1105. https://doi.org/10.1016/j.neunet.2006.05.034

Schöner G (2019) The dynamics of neural populations capture the laws of the mind. Topics Cognit Sci. https://doi.org/10.1111/tops.12453

Seung HS, Lee DD, Reis BY, Tank DW (2000) Stability of the memory of eye position in a recurrent network of conductance-based model neurons. Neuron 26(1):259–271. https://doi.org/10.1016/s0896-6273(00)81155-1

Silva EC, Costa M, Araújo J, Machado D, Louro L, Erlhagen W, Bicho E (2015) Towards human-like bimanual movements in anthropomorphic robots: a nonlinear optimization approach. Appl Math Inf Sci Int J. https://doi.org/10.12785/amis/090210

Silva R, Louro L, Malheiro T, Erlhagen W, Bicho E (2016) Combining intention and emotional state inference in a dynamic neural field architecture for human-robot joint action. Adapt Behav 24(5):350–372. https://doi.org/10.1177/1059712316665451

Sousa E, Erlhagen W, Ferreira F, Bicho E (2015) Off-line simulation inspires insight: a neurodynamics approach to efficient robot task learning. Neural Netw 72:123–139. https://doi.org/10.1016/j.neunet.2015.09.002

Sugrue LP, Corrado GS, Newsome WT (2004) Matching behavior and the representation of value in the parietal cortex. Science 304(5678):1782–1787. https://doi.org/10.1126/science.1094765

Sünderhauf N, Brock O, Scheirer W, Hadsell R, Fox D, Leitner J, Upcroft B, Abbeel P, Burgard W, Milford M et al (2018) The limits and potentials of deep learning for robotics. Int J Robot Res 37(4–5):405–420. https://doi.org/10.1177/0278364918770733

Tsarouchi P, Makris S, Chryssolouris G (2016) Human-robot interaction review and challenges on task planning and programming. Int J Comput Integr Manuf 29(8):916–931. https://doi.org/10.1080/0951192X.2015.1130251

Wang J, Dou R, Muddada RR, Zhang W (2018a) Management of a holistic supply chain network for proactive resilience: theory and case study. Comput Indus Eng 125:668–677. https://doi.org/10.1016/j.cie.2017.12.021

Wang P, Liu H, Wang L, Gao RX (2018b) Deep learning-based human motion recognition for predictive context-aware human-robot collaboration. CIRP Ann 67(1):17–20. https://doi.org/10.1016/j.cirp.2018.04.066

Wei W, Song H, Li W, Shen P, Vasilakos A (2017) Gradient-driven parking navigation using a continuous information potential field based on wireless sensor network. Inf Sci 408:100–114. https://doi.org/10.1016/j.ins.2017.04.042

Wilcox R, Nikolaidis S, Shah J (2013) Optimization of temporal dynamics for adaptive human-robot interaction in assembly manufacturing. Robotics. https://doi.org/10.15607/RSS.2012.VIII.056

Wojtak W, Coombes S, Bicho E, Erlhagen W (2016) Combining spatial and parametric working memory in a dynamic neural field model. In: Villa A, Masulli P, Pons Rivero A (eds) Artificial Neural Networks and Machine Learning—ICANN 2016, Lecture Notes in Computer Science, Springer, vol 9886, pp 411–418, https://doi.org/10.1007/978-3-319-44778-0_48

Wojtak W, Ferreira F, Bicho E, Erlhagen W (2019) Neural field model for measuring and reproducing time intervals. In: Tetko IV, Kůrková V, Karpov P, Theis F (eds) Artificial Neural Networks and Machine Learning - ICANN 2019: Theoretical neural computation. Springer, Cham, pp 327–338. https://doi.org/10.1007/978-3-030-30487-4_26

Zunino A, Cavazza J, Volpi R, Morerio P, Cavallo A, Becchio C, Murino V (2020) Predicting intentions from motion: the subject-adversarial adaptation approach. Int J Comput Vis 128(1):220–239. https://doi.org/10.1007/s11263-019-01234-9