A robotic architecture for action selection and behavioral organization inspired by human cognition

Mathis Richter*, Yulia Sandamirskaya†, Gregor Schöner‡
Institut für Neuroinformatik, Ruhr-Universität Bochum
Universitätsstr. 150, 44780 Bochum, Germany
Email: *mathis.richter@ini.rub.de, †yulia.sandamirskaya@ini.rub.de, ‡gregor.schoener@ini.rub.de

***Abstract*— Robotic agents that interact with humans and perform complex, everyday tasks in natural environments will require a system to autonomously organize their behavior. Current systems for robotic behavioral organization typically abstract from the low-level sensory-motor embodiment of the robot, leading to a gap between the level at which a sequence of actions is planned and the levels of perception and motor control. This gap is a major bottleneck for the autonomy of systems in complex, dynamic environments. To address this issue, we present a neural-dynamic framework for behavioral organization, in which the action selection mechanism is tightly coupled to the agent's sensory-motor systems. The elementary behaviors (EBs) of the robot are dynamically organized into sequences based on task-specific behavioral constraints and online perceptual information. We demonstrate the viability of our approach by implementing a neural-dynamic architecture on the humanoid robot NAO. The system is capable of producing sequences of EBs that are directed at objects (e.g., grasping and pointing). The sequences are flexible in that the robot autonomously adapts the individual EBs and their sequential order in response to changes in the sensed environment. The architecture can accommodate different tasks and can be articulated for different robotic platforms. Its neural-dynamic substrate is particularly well-suited for learning and adaptation.**

I. INTRODUCTION

One of the long-standing goals of research in robotics and artificial intelligence is the design of autonomous artificial agents capable of interacting with humans and of helping humans in their natural environments. The complexity, time dependence, and unpredictability of such environments, that include the human user, challenge the robot's perception systems. Humans, on the contrary, accomplish tasks in such natural environments with ease. Reaching the smoothness, speed, and pliability of human action is a key motivation, so it is natural to seek inspiration for solutions in how humans achieve this. A relevant insight from work on human cognition is that planning is closely coupled to perceptual information, observed, for instance, in online-updating of reaching movement at any time during movement preparation and execution [1]. Robotic approaches to behavioral organization, in contrast, have typically linked sensory information to the action planning system through fixed abstract interfaces and at discrete moments in time when an update is made.

For instance, in the classical sense-plan-act paradigm, the sequence of actions is elaborated by a planning algorithm, which operates on objects and states, abstracted from the physical sensory signals by fixed interfaces (e.g., [2], [3], [4]). Such architectures make strong demands on the quality of perceptual information and have been criticized for their lack of flexibility. When augmented by probabilistic techniques [5], such approaches achieve considerable performance but rely on the adequacy of the probabilistic models of the environment, which is often violated in real-world environments.

As an alternative, proponents of the behavior-based approach [6], [7] suggest that a robot's behavior should be generated by combining behavioral modules, each of which has access to appropriate sensory information and guides a motor function. Behavior-based approaches range from purely reactive schemes [8] to methods invoking some extent of planning [9], [10]. Since the most radical, state-less reactive approaches do not scale to more robust and complex behaviors (as argued in [11], [12]), hybrid architectures were developed that combine the strengths of behavior-based and of deliberative approaches (see, e.g., [13]). The limited scalability of behavior-based autonomous robot systems has led to efforts to add structure to the architectures that facilitates the accomplishment of complex tasks [14], [11].

The behavior-based approaches share with the sense-plan-act approaches the attempt to deal with the coordination of behaviors at a level that abstracts from the sensory details. The processing of rich and complex sensory information is delegated to the lower, sensory layers. Analogously, the details of motor control are shifted into a motor execution layer (see, e.g., [15]). Although a useful heuristic, we believe that this shift prevents systems from achieving the smoothness and reliability desired in the behavioral organization of cognitive robots. We argue that the sensory-motor layer has to be integrated more intimately with the systems that generate behavioral sequences. This is because the core elements of sequence generation—the initiation and termination of an action, the selection of the next action, and the estimation of the parameters of an upcoming action—critically depend on such sensory-motor processes.

The advantage of such tight integration has been recognized early on [16] and becomes particularly critical when the robot's actions are directed at objects in the world, such as in manipulation tasks. In such tasks, the low-level perceptual functions, such as segmentation, localization, and estimation are intertwined with action selection, action initiation and termination. For instance, detection of a target object on the sensory array requires that the object's representation is stabilized against noise, occlusions, and distractors before an action directed at this object may be initiated. If the object moves, the location and pose of the object must be tracked, potentially leading to the replanning of the action sequence. To enable integration of the perceptual, motor, and cognitive (i.e., action selection) processes, we need a theoretical and mathematical language that is shared by the sensory-motor and the cognitive level.

We propose such a language, inspired by theoretical approaches to modeling human embodied cognition [17]. We humans solve similar problems as those arising in autonomous robots while acting in only loosely constrained environments. The human nervous system generates behavior based on uncertain and incomplete sensory information, which may vary in time in unpredictable ways. Analogously to the debates about sense-plan-act vs. behavior-based autonomous robotics [18], the theoretical stance of *embodied cognition* is aimed at overcoming the traditional segregation of cognitive functions from the sensory motor processing that is at the basis of human behavior.

In the field of embodied cognition, the attractor dynamics approach [17] is one of the successful theoretical frameworks. The attractor dynamics approach to behavior generation has also been shown capable of generating robotic behavior [19], e.g. to integrate target acquisition and obstacle avoidance on robot vehicles [20] or robot arms [21]. Moreover, the attractor dynamics integrate the low-level sensory-motor layers, at which information is noisy and highly variable, with more abstract cognitive layers. At both layers, meaningful states of the system are represented by attractors of the underlying neural dynamics, providing stability and robustness.

Dynamic Field Theory (DFT) [17] is an extension of the attractor dynamics approach to embodied cognition that endows neural-dynamic systems with graded representations, enabling richer perceptual representations as well as representations of plans and intentions. The dynamics of (neural) activation fields may model metric working memory, the detection of salient inputs, attentional processes, as well as processes of selection among alternatives. In each case, instabilities mark the decisions, separating qualitatively different attractor states [22].

In this paper, we sketch how DFT may be used to organize robotic behaviors. The core functions of the architecture are to select an elementary behavior under the appropriate conditions, activate the selected behavior at an appropriate moment, and deactivate the behavior when successful completion is signaled. Since all decisions are stabilized against fluctuations in sensory or internal state variables, the architecture may be coupled to low-level sensors and simple motor interfaces. To demonstrate how this coupling may be achieved and how the DFT architecture may generate action sequences, we implement the architecture on a humanoid robotic platform. The action sequences are flexible in that a reorganization occurs autonomously in response to changes in the environment. We describe the mathematical formulation of the model and provide means of implementing the architecture for different tasks and on different hardware, as well as applying learning techniques.

II. THEORY AND METHODS

A. Dynamic Field Theory

Dynamic Field Theory (DFT) [17] is a variant of the attractor dynamics approach to embodied cognition, in which states of a behaving agent are described by continuous activation functions defined over behaviorally relevant parameter spaces of different dimensionality. These dynamic activation functions—the dynamic fields (DFs)—evolve in time according to the dynamic equation (1).

$$\tau \dot{u}(x,t) = -u(x,t) + h + S(x,t) + \int f(u(x',t))\,\omega(x-x')\,dx' \quad (1)$$

In Eq. (1), $u(x,t)$ is the activation of the DF; x is the parameter space, e.g., the one-dimensional hue space of the input from the vision sensor, the two-dimensional visual space, or the three-dimensional space of target positions of the end-effector of the agent; t is time; τ is the relaxation constant of the dynamics; $h<0$ is a resting level to which the DF converges without input and working memory of previous activity; $S(x,t)$ is the external input to the DF; $f(\cdot)$ is a sigmoidal non-linearity that shapes the output of the DF. The last term formalizes the homogeneous lateral connectivity (and interaction) in the DF: the output of the DF is convolved with a Gaussian interaction kernel, $\omega(x-x')$. The interaction kernel has two components, local excitation and global or mid-range inhibition, which stabilize the localized peak(s) solution of the DF.

The lateral interaction within the dynamic field also provides for the working memory in the DF: at sufficient levels of local excitation, the DF sustains the activity peak even if the initial input ceases. Within DFT, localized peaks are units of representation. The distributed, ambiguous, fluctuating, and noisy input typical for low-level physical sensors leads to localized, consistent, persistent, stable states in the DFs. An instability separates a peak-attractor of the DF's dynamics from a sub-threshold attractor and thus the field's dynamics performs and stabilizes a "detection" decision about presence of a particular characteristics, as well as a "selection" decision between competing inputs, and supports memory for previously encountered states. These properties form the basis for the power of DFT as interface between low-level sensory-motor and more abstract cognitive computations [23].

Since the behaviorally meaningful states in DFT are attractors, we need a dynamical mechanism to destabilize the current state and to proceed to the next state in a behavioral sequence. To address this need, we have recently implemented the principle of intentionality [24] in DFT. In particular, we argued that each action-related state of the system's dynamics needs a correspondent condition of satisfaction (CoS) state, which signals accomplishment of the current action and triggers a sequential transition [25], [26].

Here, we further develop this mechanism, introducing a concept of elementary behavior (EB), which comprises both high-level representation of the intention and CoS of an action, as well as sensory-motor representations, which receive input directly from sensors of the robot. A number of EBs are organized in an architecture, in which different constraints (e.g., precondition or competition) between the EBs are represented by dynamical nodes.

Next, we describe the DFT architecture for behavior organization in detail. The entire architecture is built from dynamic fields (DFs), each described by Eq. (1), where x are spaces of different dimensionality (here, zero to three). The DFs of the architecture are coupled through weighted connections, a mechanism described in [27]. The detailed mathematical description of the model is presented in Section V.

B. Model

1) Elementary behavior: A task, such as grasping an object, may be subdivided into motor and perceptual processes (e.g., locating the target object, moving the end-effector, or closing the gripper), all of which we refer to as *elementary behaviors* (EBs). Each EB has two dynamical elements: an *intention* and a corresponding *condition of satisfaction* (CoS). Both elements are represented by zero-dimensional DFs, that is, by neural-dynamic nodes. The dynamics of the nodes is bi-stable with an "on" and an "off" attractor. The intention and CoS nodes are coupled to multi-dimensional DFs that represent the graded parameter(s) of the intention and of the CoS, respectively (see Fig. 1).

An active intention node induces a localized peak of activation in the intention field. This peak impacts on the sensory-motor dynamics downstream from the intention field. In particular, it provides input to perceptual DFs over relevant feature spaces that are accessible to the sensors of the robot. It also provides input to DFs that represent movement parameters and are coupled to an effector dynamics (see Fig. 2 for an exemplary architecture). Activation peaks in perceptual and motor

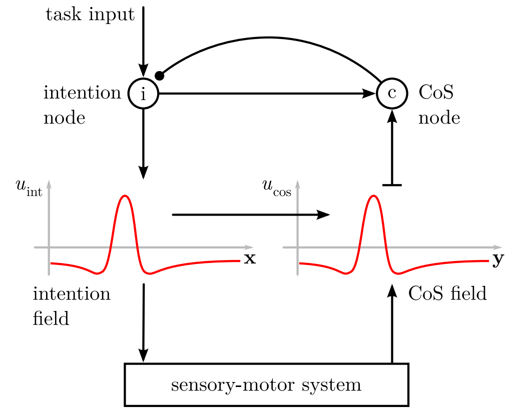


Fig. 1: DFT-based model of an elementary behavior (EB). See text for details.

DFs set attractors of the motor dynamics, which drive effectors of the robot. At the same time, the CoS field receives input from both the intention field and the perceptual fields. If these two inputs match, a peak in the CoS field arises and signals the successful completion of the EB. In this way, the CoS field detects that the predicted sensory consequences of the motor act have been observed. The space over which the CoS field is defined is chosen accordingly. An activity peak in the CoS field activates the CoS node, which in turn inhibits the intention node of the EB. Consequently, the peak in the intention field decays and the impact of the intention of this EB on the sensory-motor systems of the agent ceases. The system transitions to the next EB.

2) Architecture: The neural-dynamic structure of elementary behaviors (EBs) ensures that each action within a behavioral sequence is initiated, executed by the motor system, and brought to an end. A set of EBs comprises the behavioral repertory of the agent. Activation of these EBs has to be organized in time. To accomplish this, we introduce interneurons (i.e., dynamical nodes), which encode constraints on the sequential activation of EBs. In the architecture that we present here, two types of constraints are important: the *precondition* constraint and the *suppression* constraint.

Fig. 2 illustrates, amongst others, a precondition between the EBs "open gripper" (EB1) and "move arm" (EB2). The precondition constraint is encoded by a dynamical node that inhibits the intention node of EB2, which should not be activated unless the gripper of the robot is open. The precondition node is, in turn, inhibited by the CoS node of EB1. Thus, when the CoS node is activated, the intention node of EB2 is disinhibited. As a result, EB1 and EB2 are activated sequentially. A suppression constraint, e.g. between the EBs "move left arm" and "move right arm" (not shown in Fig. 2), suppresses one of these EBs as long as the other one is active, leading to a competition between the EBs. A number of precondition and suppression constraints implicitly encode a sequence. For instance, the constraints shown in Fig. 2 result in the grasping behavior of the robot.

3) Tasks and switching dynamics: A set of EBs with associated behavioral constraints forms a task, which is realized by one or multiple complete behavioral sequences—examples are pointing or grasping. A dynamical task node (not shown in Fig. 2) activates the intention nodes of all EBs involved in a task as well as the respective constraint nodes. The sequence emerges then from the activation of those EBs that do not have preconditions, or whose preconditions have been met. Fig. 2 illustrates portions of an architecture for the task of grasping.

III. RESULTS

In this paper, we present three sets of results. First, we demonstrate the basic functionality of the architecture by implementing different tasks (i.e., sequences of EBs) in simulation and on the robot. Second, we show how the perceptual, bottom-up input impacts on the behavioral sequences along with the top-down constraints between EBs. And finally, we demonstrate the ability of the architecture to detect changes in the perceptual state that require returning to a previous EB in the sequence.

All robotic experiments were performed on a humanoid robot NAO in a table-top scenario. NAO was facing a table of 30 cm height with several colored objects (1x2.5x5 cm) in different arrangements for different runs of the experiments. The target object was placed in the 15x15 cm area in front of the robot, because of the limited operation range of the arms of the version of NAO we used. The color camera of the robot provided input to the perceptual DFs, the dynamics of which effectively performed detection, segmentation, and selection of the colored objects. The raw camera input and input from the other robotic sensors was used to emphasize that our approach stays robust even if linked directly to a low-level sensory-motor system.

A. Basic demonstration: Acting out task-driven behavioral sequences

The architecture that is partially shown in Fig. 2 is comprised of the EBs “find object color”, “move head”, “open left/right gripper”, “find end-effector marker color”, “move left/right arm”, “visual servoing”, and “close left/right gripper”. When the task node “grasping” is activated, it sends excitatory input to the intention nodes of these EBs and to the precondition nodes shown in Fig. 2.

The EBs “find color” and “move head” do not have preconditions and can become active. The intention field of the EB “move head” requires a perceptual input in order to be activated, which specifies the location of the target object in the image. This perceptual input is delivered when the EB “find object” succeeds. The intention field of “move head” sets an attractor for the dynamics that controls the movement of the robot’s head, making it center the target object in the camera image. When the CoS of the EB “move head” detects the presence of the target object in the central part of the camera image, it is activated and inhibits the precondition nodes of the EBs “open left/right gripper”. Since the target object is located in the right portion of the workspace, the intention field of the “open right gripper” field is activated. When this EB is completed, the “move right arm” EB is activated analogously. The intention field of this EB sets an attractor for the arm movement dynamics, estimated from the pitch and yaw of the robot’s head. The EB “move right arm” is finished when the robot’s gripper appears in the camera input. The EB “visual servoing” is then activated until the robot’s thumb is aligned with the target object. This match is detected in the CoS field of the EB “visual servoing” and the final EB “close right gripper” is activated. The whole sequence is completed when the robot’s hand grasps the object.

In simulation, we modeled sensory inputs as localized distributions over dimensions of the perceptual DFs. The exemplary tasks which we looked at in simulation were five different sequences of the EBs depicted in Fig. 2. We conducted 20 runs of the simulation for each task, varying locations and strengths of the perceptual inputs to verify the robustness of the DF representations and the switching mechanism between the EBs. We also tested at which amount of noise in neural dynamics and sensors the behavior of the system becomes unstable.

In the robotic implementation, we let the humanoid robot NAO act out two of the six simulated sequences that result in meaningful behaviors (i.e., “grasp” and “point”). Fig. 3a illustrates the pointing sequence by showing the time courses of the activations of the intention nodes involved in the task. Fig. 3b shows the analogous plot for the grasping task. The two tasks differ in the order of some EBs (i.e., opening/closing the gripper). Comparing Fig. 3a and 3b additionally shows that the architecture can deal with varying durations of execution time for each EB in the sequence (e.g., “visual servoing”), since the successful completion of EBs is determined based on perceptual input only. We have tested performance in these tasks at different arrangements and colors of the target and distractor objects.

B. Influence of the perceptual input on the sequences

In this experiment we demonstrate that not only task-specific behavioral constraints determine the behavioral sequence, but the perceptual input from the environment also influences the executed EBs. In particular, we demonstrate how the architecture stabilizes a selection decision to move the left or right arm depending on the position of the target object on the table. Out of 40 trials, in which the object is situated in different locations to the left and to the right of the robot in 20 cases each, the robot always chooses the correct arm to grasp the object. Even if the target object is close to the midline, there are

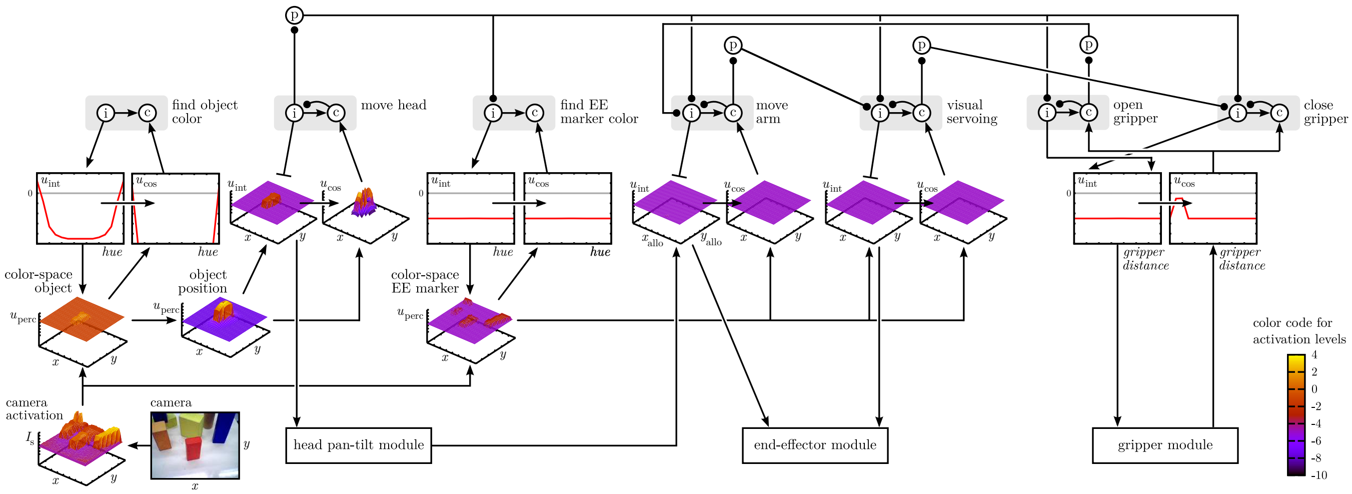


Fig. 2: DFT architecture allowing for tasks that consist of an arm movement toward a target object of an arbitrary color. This particular setup is intended to lead the controlled robot to grasp the target object. Other tasks, such as pointing or pushing of the target object, can be realized by enforcing different behavioral constraints through task input (not shown here). Pointing arrows represent excitatory input, whereas arrows with circular heads denote inhibitory connections. Arrows ending in an orthogonal line represent a global boost of the corresponding DF.

no oscillations between the “left” and “right” decisions. Fig. 4 shows the trajectories of both end-effectors for six exemplary trials with different target locations. In all trials, only one of the end-effectors moves, while the other remains stationary. Some of the trajectories (e.g., trial 2 and 5) also show the switch from the first approximate arm movement of the EB “move arm” to the visual servoing of the end-effector.

C. Reactivating EBs and returning to earlier elements of a sequence

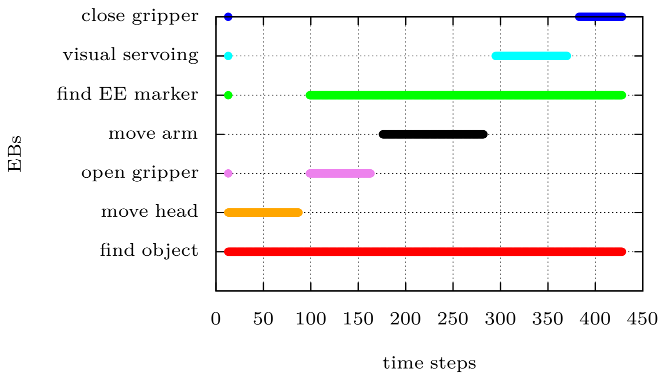
Here, we demonstrate the flexibility of the architecture. The robot is able to detect if the environmental situation changes and returns to prior elements within a sequence. To accomplish this, the CoS node of some EBs (called reactivating EBs) loses its activation if the sensory input that induced it ceases. The suppression and precondition nodes that were inhibited by the CoS of such an EB get reactivated. These nodes inhibit any active intention for which the reactivated EB has to be completed. Since the execution of any EB depends on the perceived state of the environment, the subsequent behavioral sequence may differ from the one before the CoS of the reactivating EB ceased.

During execution of the “grasping” task, described in Section III-A, the target object is moved from the right to the left side of the table when the robots starts to move its right arm within the EB “move right arm”. Since the object is no longer in the center of the robot’s camera, the peak in the CoS field of the EB “move head” decays. The CoS node loses its activation as well and since it no longer inhibits the intention node of the EB “move head”, the intention node is reactivated. The precondition node of the EB “move right arm” is also reactivated and suppresses this EB. Since the target object is now perceived on the left side of the workspace, it provides input to the intention field of the EBs “open/close left gripper” and “move left arm”. The intention fields of these EB are activated in the subsequent behavioral sequence—the robot autonomously uses its other arm to grasp the object.

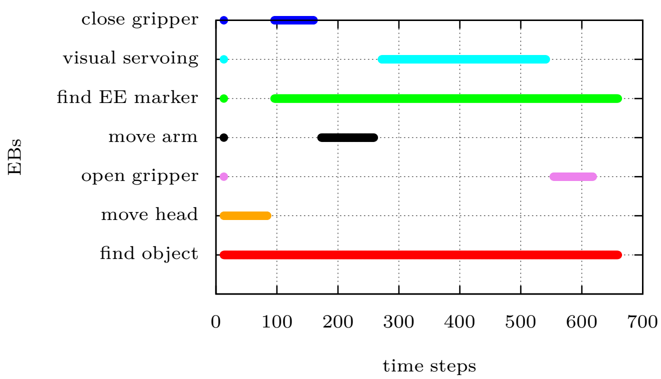
IV. CONCLUSIONS

The framework based on Dynamic Field Theory that we presented here is aimed to organize behavior in autonomous robots while continuously coupling to sensory information that may be low-level, time-varying, and fluctuating. The neural dynamics enable the system to stabilize decisions against variations of sensory input until a critical level is reached, at which the system may flexibly activate, through an instability, an alternate behavioral sequence. The completion of behavior at any given step in a behavioral sequence is likewise controlled by an instability that triggers the transition to the next step, which is then stabilized in turn.

Conceptually, the DFT framework is compatible with the ideas of behavior-based robotics. Earlier efforts to organize behavioral sequences in such systems did not address the stability issues at a theoretical level [28]. We believe this to be a limiting factor for such architectures and that it makes debugging and parametrically tuning such architectures difficult. In a way, our framework is a renewed building of a behavior-based approach to behavioral organization that now provides an explicit conceptual basis for stability and coupling. Unlike the simulated work [28], most implemented behavior-based robots have addressed navigation, locomotion, or expressive gesture generation [29]. In contrast, in our demonstrations, the sequences included motor behaviors that were directed at

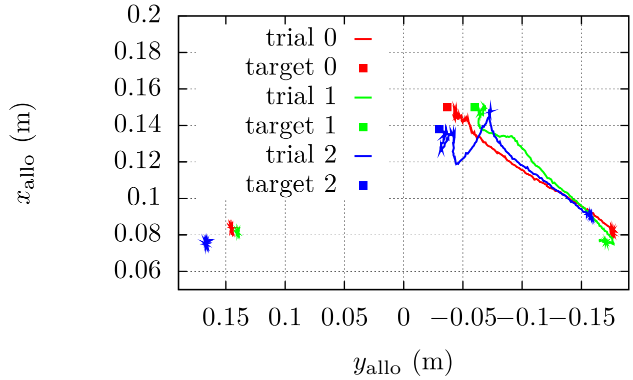


(a) pointing

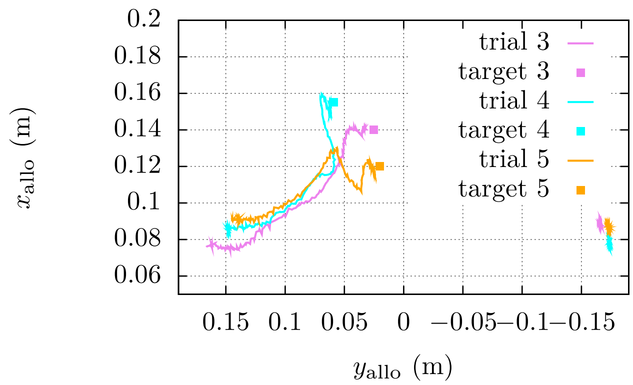


(b) grasping

Fig. 3: Activation patterns of the intention nodes of all EBs relevant to the pointing (a) and grasping (b) tasks. The sequence of EBs emerges autonomously from behavioral precondition constraints, which are activated between EBs by task-specific input. Note the varying amount of time needed for each EB to complete, both among each other and when compared to their counterparts in the grasping task.



(a)



(b)

Fig. 4: Trajectories of both end-effectors for six trials of a grasping task, in which the object is located either to the right (a) or left (b) of the robot. The trajectories are projected onto the table surface and the origin of the coordinate system lies between the robot's feet. The movements result from a sequential activation of the EBs "move left/right arm" and "visual servoing".

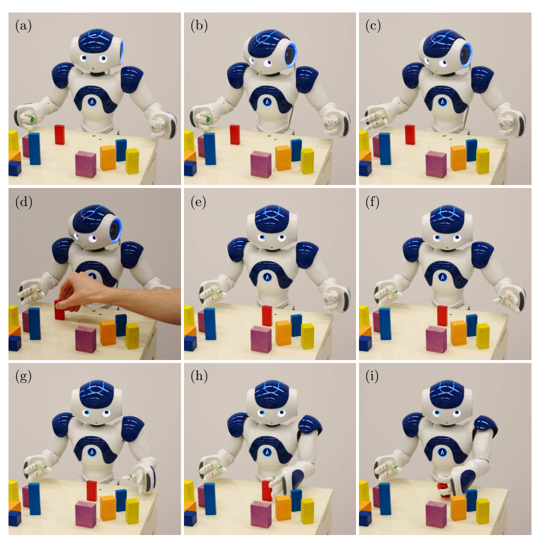


Fig. 5: Photos of a grasping task, in which the target object is moved to the other side of the table during the approach of the robot's right end-effector.

perceived objects. This was possible, because perceptual representations are part of DFT.

Classical, decision theoretical approaches to behavioral organization have typically operated off-line and not investigated online coupling to noisy sensory information. The on-line updating has recently been addressed [30]. However, conceptually these are time-less approaches that make a new decision at every time step. Our approach is, in contrast, based on dynamical coupling. The complete system is a single dynamical system that spends most of its time in an attractor state. We believe this to be an advantage when control-theoretical properties of the coupled effector-environment play a role. In our demonstrations, this becomes visible in Fig. 4, in which both approach and visual servoing were achieved within the unified DFT framework.

The theoretical language of DFT lends itself to include online learning processes in the architecture. Couplings between the different EBs and the associated intention and condition of satisfaction fields may be modified by graded, dynamic learning rules. We are currently integrating methods of value-based reinforcement learning and Hebbian learning into our approach with the goal of enabling the system to learn behavioral constraints from experience.

V. APPENDIX: MATHEMATICAL DESCRIPTION OF THE MODEL

In this section, we present the dynamical equations of the DFT architecture for behavioral organization. First, we introduce a notation for the generic part of the dynamical equation for a dynamic node (Eq. (2)) and a dynamic field (DF) (Eq. (3)):

$$\mathscr{F}_v(v(t)) = -v(t) + h_v + c_v^{\text{exc}} f(v(t)), \tag{2}$$

$$\mathscr{F}_u(u(\mathbf{x},t)) = -u(\mathbf{x},t) + h_u + \int f(u(\mathbf{x}',t))\,\omega(\mathbf{x}-\mathbf{x}')\,d\mathbf{x}' \tag{3}$$

We denote activation of all dynamic nodes with v and all dynamic fields with u; the subscripts are "int" for intention, "cos" for condition of satisfaction, "perc" for perceptual, and "task" for task. h is a negative resting level, t is time. $f(\cdot)$ is a sigmoidal non-linearity that shapes the output of nodes and DFs

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

and ω is a sum-of-Gaussians interaction kernel

$$\omega(\mathbf{x}-\mathbf{x}') = c_u^{\text{exc}} e^{-\frac{(\mathbf{x}-\mathbf{x}')^2}{2\sigma_{\text{exc}}}} - c_u^{\text{inh}} e^{-\frac{(\mathbf{x}-\mathbf{x}')^2}{2\sigma_{\text{inh}}}}. \tag{5}$$

We also introduce the following notation for projections between dynamic nodes and dynamic fields of different dimensionality

$$f_{\text{pr}}(u) = c_{\text{pr}}^u \int f(u(\mathbf{x},t))\,d\bar{\mathbf{x}}, \tag{6}$$

where $\bar{\mathbf{x}}$ is a subspace of the space $\mathbf{x}$ of the dynamic field from which the projection is performed (u here). This subspace is comprised of dimensions that are not present in the space, over which the receiving dynamic field is defined (the field on the left-hand side of the following equations). $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ are different spaces, over which intention, condition of satisfaction, and perceptual fields are defined, respectively.

Taking these notations into account, an elementary behavior within our architecture may be described by the following set of non-linear differential equations.

The *intention node*

$$\tau\dot{v}_{\text{int}}(t) = \mathscr{F}(v_{\text{int}}(t)) + c_{\text{i,t}} f(v_{\text{task}}(t)) - c_{\text{i,c}} f(v_{\text{cos}}(t)) - \sum_{p/s} c_{\text{i},p/s} f(v_{p/s}(t)) \tag{7}$$

receives a positive input from the task node, scaled by a parameter $c_{\text{i,t}}$, a negative input from the respective CoS node, scaled by a parameter $c_{\text{i,c}}$, and a negative input from the active constraints nodes, each scaled by a factor $c_{\text{i},p/s}$.

The *condition of satisfaction (CoS) node*

$$\tau\dot{v}_{\text{cos}}(t) = \mathscr{F}(v_{\text{cos}}(t)) + c_{\text{c,i}}^v f(v_{\text{int}}(t)) + c_{\text{c},c}^{v,u} f_{\text{pr}}(u_{\text{cos}}) \tag{8}$$

receives positive input from the respective intention node, scaled by a factor $c_{\text{c,i}}^v$, and positive input from the CoS field, scaled by $c_{\text{c,c}}^{v,u}$. The projection f_{pr} is typically simply an integral over the DF u_{cos}.

The *intention field*

$$\tau\dot{u}_{\text{int}}(\mathbf{x},t) = \mathscr{F}(u_{\text{int}}(\mathbf{x},t)) + c_{\text{i,i}}^{u,v}(\mathbf{x},t) f(v_{\text{int}}(t)) + c_{\text{i,perc}} f_{\text{pr}}(u_{\text{perc}}) \tag{9}$$

receives a positive input from the intention node, scaled by a weights function $c_{\text{i,i}}^{u,v}(\mathbf{x},t)$, which may vary in a learning process, and a positive input from (typically a single) perceptual DF, scaled by a factor $c_{\text{i,perc}}$ and, if needed, projected down or up.[1]

The *CoS field*

$$\tau\dot{u}_{\text{cos}}(\mathbf{y},t) = \mathscr{F}(u_{\text{cos}}(\mathbf{y},t)) + c_{\text{c,i}}^u(\mathbf{x},\mathbf{y},t) f(u_{\text{int}}(\mathbf{x},t)) + c_{\text{perc,c}} f_{\text{pr}}(u_{\text{perc}}) \tag{10}$$

receives positive input from the intention DF through a weight function $c_{\text{c,i}}^u$, which accomplishes a transformation from the space $\mathbf{x}$ of the intention field to the space $\mathbf{y}$ of the CoS field ($\mathbf{x}$ and $\mathbf{y}$ are often, but not always, identical). The CoS field also receives positive input from (typically a single) perceptual DF through a projection, scaled by a factor $c_{\text{perc,c}}$.

The *precondition constraint node*

$$\tau\dot{v}_{\text{p}}(t) = \mathscr{F}(v_{\text{p}}(t)) + c_{\text{p,task}} f(v_{\text{task}}(t)) - c_{\text{p,c}} f(v_{\text{cos}}(t)) \tag{11}$$

receives positive task input of strength $c_{\text{p,task}}$ and negative input from the connected CoS node, scaled by a factor $c_{\text{p,c}}$. The node needs the CoS input to be deactivated.

The *suppression constraint node*

$$\tau\dot{v}_{\text{s}}(t) = \mathscr{F}(v_{\text{s}}(t)) + c_{\text{s,task}} f(v_{\text{task}}(t)) + c_{\text{s,i}} f(v_{\text{int}}(t)) \tag{12}$$

receives positive task input of strength $c_{\text{p,task}}$ and positive input from the connected intention node, scaled by a factor $c_{\text{s,i}}$. The node is inactive when the intention input ceases.

The *task node*

$$\tau\dot{v}_{\text{task}}(t) = \mathscr{F}(v_{\text{task}}(t)) + I_{\text{context}}(t) \tag{13}$$

is activated by the contextual input $I_{\text{context}}(t)$, which, in our experiments, is simply a numerical input emitted from a GUI, but could also be the output of a perceptual DF module that detects particular contexts in which tasks should be activated.

[1] Up-projection converts input from a lower-dimensional DF to a higher-dimensional DF by repeating input over additional dimensions.

A *perceptual field*

$$\tau \dot{u}_{\mathrm{perc}}(\mathbf{z},t) = \mathscr{F}(u_{\mathrm{perc}}(\mathbf{z},t)) + c_{\mathrm{perc,i}} f_{\mathrm{pr}}(u_{\mathrm{int}}) + I_{\mathrm{s}}(t) \quad (14)$$

receives positive input from the intention DF, scaled by $c_{\mathrm{perc,i}}$ and, typically, projected up (when the perceptual field has a higher dimensionality than the intention field) or down (when the perceptual DF has a lower dimensionality). Additionally, it receives a direct input $I_{\mathrm{s}}(t)$ from the associated sensor, which could, e.g., be a hue-value distribution over the image.

The *motor dynamics*

$$\tau \dot{\phi} = -\phi + \mathit{forcelet}(f_{\mathrm{pr}}(u_{\mathrm{perc}}), f_{\mathrm{pr}}(u_{\mathrm{int}}), \phi) \quad (15)$$

is defined for the motor variable ϕ, the dynamics of which has an attractor set by a *forcelet* that is constructed depending on the peculiarities of the motor controlled by ϕ. The motors of the robot are driven according to the value of ϕ. Examples of use of such dynamics may be found in [20], [21].

A particular architecture is assembled from a number of such elements (see, e.g., Fig. 2). The parameters of DFs and couplings c are subject to neurally inspired (Hebbian) learning and may potentially be tuned based on the experience of the agent using reinforcement learning techniques, imitation, and scaffolding or, alternatively, through evolutionary optimization.

VI. Acknowledgments

The authors gratefully acknowledge the financial support of the European Union Seventh Framework Programme FP7-ICT-2009-6 under Grant Agreement no. 270247—NeuralDynamics.

References

[1] C. Prablanc and O. Martin, “Autonomous control during hand reaching at undetected two-dimensional target displacements,” *Journal of Neurophysiology*, vol. 67, pp. 455–469, 1992.

[2] R. Alami, R. Chatila, S. Fleury, M. Ghallab, and F. Ingrand, “An architecture for autonomy,” *International Journal of Robotics Research*, vol. 17, no. 4, pp. 315–337, 1998.

[3] K. Konolige, K. Myers, E. Ruspini, and A. Saffiotti, “The saphira architecture: A design for autonomy,” *Journal of Experimental and Theoretical Arificial Intelligence, Special Issue on Architectures for Physical Agents*, vol. 9, pp. 215–235, 1997.

[4] A. Orebäck and H. I. Christensen, “Evaluation of architectures for mobile robotics,” *Autonomous Robots*, vol. 14, pp. 33–49, 2003.

[5] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Boston, MA, USA: The MIT Press, 2005.

[6] R. Brooks, “A robust layered control system for a mobile robot,” *Robotics and Automation, IEEE Journal of*, vol. 2, no. 1, pp. 14–23, 1986.

[7] R. C. Arkin, *Behavior-Based Robotics*. MIT Press, Cambridge, M, 1998.

[8] P. Maes, *Designing Autonomous Agents : Theory and Practice from Biology to Engineering and back*, ch. Situated agents can have goals, pp. 49–70. MIT Press, Cambridge, MA, 1990.

[9] M. Nicolescu and M. Mataric, “Linking perception and action in a control architecture for human-robot domains,” *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*, p. 10, 2003.

[10] P. Pirjanian, “Behavior coordination mechanisms – state-of-the-art,” *Institute for Robotics and Intelligent Systems*, 1999.

[11] J. Bryson, “The behavior-oriented design of modular agent intelligence,” *Agent technologies*, 2003.

[12] T. Tyrrell, *Computational mechanisms for action selection.* PhD thesis, University of Edinburgh. Center for Cognitive Science., 1993.

[13] E. Gat, *Artificial Intelligence and Mobile Robots: Case Studies of Successful Robot Systems*, ch. Three-layer architectures, pp. 195–200. MIT Press, Cambridge, MA, 1998.

[14] M. Proetzsch, T. Luksch, and K. Berns, “Development of complex robotic systems using the behavior-based control architecture iB2C,” *Robotics and Autonomous Systems*, vol. 58, pp. 46–67, Jan. 2010.

[15] S. Lemaignan, R. Ros, L. Mösenlechner, R. Alami, and M. Beetz, “ORO, a knowledge management platform for cognitive architectures in robotics,” in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pp. 3548–3553, 2010.

[16] D. Payton, J. Rosenblatt, and D. Keirsey, “Plan guided reaction,” *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 20, no. 6, pp. 1370–1382, 1990.

[17] G. Schöner, “Dynamical systems approaches to cognition,” in *Cambridge Handbook of Computational Cognitive Modeling* (R. Sun, ed.), (Cambridge, UK), pp. 101–126, Cambridge University Press, 2008.

[18] M. L. Anderson, “Embodied cognition: A field guide,” *Artificial Intelligence*, vol. 149, pp. 91–130, 2003.

[19] G. Schöner, M. Dose, and C. Engels, “Dynamics of behavior: Theory and applications for autonomous robot architectures,” *Robotics and Autonomous Systems*, vol. 16, pp. 213–245, 1995.

[20] E. Bicho, P. Mallet, and G. Schöner, “Target representation on an autonomous vehicle with low-level sensors,” *The International Journal of Robotics Research*, vol. 19, pp. 424–447, 2000.

[21] H. Reimann, I. Iossifidis, and G. Schöner, “Autonomous movement generation for manipulators with multiple simultaneous constraints using the attractor dynamics approach,” in *Proceedings of the IEEE International Conference on Robotics and Automation ICRA*, (Shanghai, China), 2011.

[22] J. S. Johnson, J. P. Spencer, and G. Schöner, “Moving to higher ground: The dynamic field theory and the dynamics of visual cognition,” *New Ideas in Psychology*, vol. 26, pp. 227–251, 2008.

[23] Y. Sandamirskaya, S. Zibner, S. Schneegans, and G. Schöner, “Using dynamic field theory to extend the embodiment stance toward higher cognition,” *New Ideas in Psychology*, no. ABiALS special issue, in press.

[24] J. R. Searle, *Intentionality — An essay in the philosophy of mind.* Cambridge University Press, 1983.

[25] Y. Sandamirskaya and G. Schöner, “An embodied account of serial order: How instabilities drive sequence generation,” *Neural Networks*, vol. 23, pp. 1164–1179, December 2010.

[26] Y. Sandamirskaya, M. Richter, and G. Schöner, “A neural-dynamic architecture for behavioral organization of an embodied agent,” in *IEEE International Conference on Development and Learning and on Epigenetic Robotics (ICDL EPIROB 2011)*, 2011.

[27] S. K. U. Zibner, C. Faubel, I. Iossifidis, and G. Schöner, “Dynamic neural fields as building blocks for a cortex-inspired architecture of robotic scene representation,” *IEEE Transactions on Autonomous Mental Development*, vol. 3, no. 1, pp. 74–91, 2011.

[28] P. Maes, “How to do the right thing,” *Connection Science Journal*, vol. 1, no. 3, pp. 291–323, 1989.

[29] M. Proetzsch, T. Luksch, and K. Berns, “Development of complex robotic systems using the behavior-based control architecture ib2c,” *Robot. Auton. Syst.*, vol. 58, pp. 46–67, Jan. 2010.

[30] A. Ferrein, C. Fritz, and G. Lakemeyer, “On-line Decision-Theoretic Golog for Unpredictable Domains,” *Lecture Notes in Computer Science: Knowledge-Based Systems*, vol. 3238, pp. 322–336, 2004.