

2009 Special Issue

Explorations on artificial time perception

Michail Maniadakis^{a,b,*}, Panos Trahanias^a, Jun Tani^b^a Computational Vision and Robotics Laboratory, Foundation for Research and Technology – Hellas, Institute of Computer Science (FORTH-ICS), Crete, Greece^b Laboratory for Behavior and Dynamic Cognition, RIKEN Brain Science Institute (RIKEN-BSI), Wako-shi, Saitama, Japan

ARTICLE INFO

Article history:

Received 5 May 2009

Received in revised form 2 June 2009

Accepted 25 June 2009

Keywords:

Time perception

Time experience

Rule switching

Evolutionary robotics

Wisconsin Card Sorting

ABSTRACT

In the field of biologically inspired cognitive systems, time perception, a fundamental aspect of natural cognition is not sufficiently explored. The majority of existing works ignore the importance of experiencing the flow of time, and the implemented agents are rarely furnished with time processing capacities. The current work aims at shedding light on this largely unexplored issue, focusing on the perception of temporal duration. Specifically, we investigate a rule switching task that consists of repeating trials with dynamic temporal lengths. An evolutionary process is employed to search for neuronal mechanisms that accomplish the underlying task and self-organize time-processing dynamics. Our repeated simulation experiments showed that the capacity of perceiving duration biases the functionality of neural mechanisms with other cognitive responsibilities and additionally that time perception and ordinary cognitive processes may share the same neural resources in the cognitive system. The obtained results are related with previous brain imaging studies on time perception, and they are used to formulate suggestions for the cortical representation of time in biological agents.

© 2009 Elsevier Ltd. All rights reserved.

1. Introduction

Many of the human daily activities rely on our ability to perceive the flow of time (Husserl, 1964). For example, we estimate how much time is left for our visitor to come so that we schedule food preparation, or how our inability to drive fast will cause a delay in the time of arrival to a planned destination. Besides humans, animal behaviors are also influenced by temporal computations (Gallistel & Gibbon, 2000), rendering time perception a fundamental aspect of cognition.

The interpretation of the principles guiding the natural perceptuo-motor loop into a computational context, is now a well established approach for developing Artificial Intelligence (AI) systems. Recent bio-inspired works in the field of robotics emphasize the reciprocal coupling of the control mechanism (i.e. of the central nervous system), the body of the agent, and the operating environment in the development of intelligence (Pfeifer, Lungarella, & Iida, 2007). These approaches are usually referred to with the term *new AI* (Pfeifer & Scheier, 1999; Steels & Brooks, 1984), in order to contrast with the *good-old-fashion AI* (Brooks, 1996; Haugeland, 1985) that aims at implementing universal mechanisms of logical derivation, omitting the role

that situatedness, embodiment, and emergence have in intelligence development (Srensen & Ziemke, 2007). Besides these shortcomings, old-fashion AI has already from its early days understood that time is fundamental aspect of cognition and there is a clear need for reasoning about it. More than 50 years ago, Arthur Prior introduced the Tense Logic (Prior, 1957) that gave rise to the recent time-dedicated calculus systems expressed in the form of Temporal Logics (Fisher, Gabbay, & Vila, 2005; Galton, 1995).

However, in new AI approaches, the importance of perceiving time is not adequately appreciated yet. The majority of works that focus on embodied intelligence concentrate only on the spatial characteristics of environmental interaction, nearly ignoring the temporal aspects of cognition that are necessary to accomplish meaningful perception of real world phenomena (Karmarkar & Buonomano, 2007). Due to the behavioral evaluation of robotic agents that is interpreted in terms of spatial measures, cognitive systems are now equipped with the ability to understand spatial relationships, to reach goals, to accurately mimic behavioral patterns, and others (e.g. Dautenhahn & Nehaniv, 2002; Demiris & Hayes, 2002; Doya, Samejima, Katagiri, & Kawato, 2002; Iizuka & Paolo, 2007; Oztog & Arbib, 2002). However, these robotic systems are lacking the capacity to experience time. In most existing systems, time is only implicitly accounted for, in terms of cognitive state transitions occurring in linearly ordered clock ticks. In other words, time is nothing more than a variable specifying the ordering of events.

In contrast, in biological agents the feeling of time significantly affects the dynamics of cognition, from the way that the low level

* Corresponding author at: Computational Vision and Robotics Laboratory, Foundation for Research and Technology – Hellas, Institute of Computer Science (FORTH-ICS), Crete, Greece.

E-mail address: mmaniada@ics.forth.gr (M. Maniadakis).

sensory-motor loop is perceived, to the performance and control of higher level cognitive processes such as memory organization or reasoning. Despite the importance of temporal information, humans and animals lack a sensory system devoted to the sense of time. This suggests that time perception is carried out by cognitive activities working on top of other sensory modalities. However, even if all our senses were prevented from functioning for a while, we could still notice the passing of time (LePoidevin, 2000). The experience of time is now considered a central element of natural cognition (Varela, 1999) that has recently attracted a large amount of research interest (Droit-Volet & Meck, 2007; Leon & Shadlen, 2003; Wittmann & Paulus, 2008). The current study focuses on one particular aspect of time perception, namely duration (other aspects of experiencing time are discussed in Section 2).

An important issue for cognitive neuroscience is how cortical processes represent time. There are now two major models for the neural representation of time (Ivry & Schlerf, 2008). One emphasizes that the judgment of the duration of a stimulus depends on the operation of dedicated neural mechanisms specialized for representing the temporal relationships between events. Alternatively, according to the second model, the representation of duration may arise from the intrinsic dynamics of neural mechanisms non-dedicated to time perception. The current work aims to explore the plausibility of the two alternative choices by following a cognitive robotics approach. Previous works have stressed that evolutionary robotics (Maniadakis & Trahanias, 2008; Nolfi & Floreano, 2000) can be a useful tool in brain science that can help to uncover important principles of neural processing (Ruppin, 2002). Such examples can be found in several recent studies (Borrett, Jin, & Kwan, 2005; Maniadakis, Hourdakis, & Trahanias, 2007; Suzuki, Floreano, & DiPaolo, 2005). In the current work we follow a similar approach to investigate the time representation issue, avoiding arbitrary favoring any of the two models mentioned above.

In particular, we evolve simple Continuous Time Recurrent Neural Network (CTRNN) controllers (Beer, 1995), being free to self-organize in any direction, revealing the most appropriate mechanism to perceive time duration. Similar to Maniadakis and Tani (2008), we investigate a mobile robot rule switching task. In short, according to our experimental scenario, a simulated robotic agent has to consider unpredictably changing reward signals, in order to switch between behavioral rules choosing the one that is considered correct at a given time period. In order to focus on the time-feeling properties developed in the cognitive system, the rule switching task consists of a series of trials with varying temporal duration. Then, we study the internal mechanisms developed in CTRNNs, exploring the self-organization neurodynamics and how they are constrained by the task's temporal properties. It is noted that, to the best of our knowledge this is the first work that uses the new AI's embodied intelligence principles to explore time perception cognitive dynamics.

The rest of the paper is structured as follows. In Section 2 we discuss important aspects of experiencing time and how they are related to brain science research. Then, we describe the CTRNN models used in the current study. In the following section we described the investigated task providing the details of our experimental setup. In Section 5 we present the evolutionary procedure used to explore the space of CTRNN solutions. In Section 6 we present the results obtained by the independent evolutionary procedures. Additionally, we discuss the common neurocognitive characteristics appearing in all successful solutions. Subsequently, we compare the results of our study with those obtained in our previous work (Maniadakis & Tani, 2008) investigating rule switching using trials of predefined temporal duration. In Section 8 we discuss the effect of the experimental temporal constraints on neural dynamics, and we formulate predictions about possible mechanism of real cortical dynamics. Finally, the last section highlights conclusions and directions for future work.

2. Aspects of time perception

Investigating the issue of time perception is a very complicated task that requires research efforts to be directed along different, yet interconnected topics. In particular, research on time perception needs to address the broad range of temporal experiences we have in our daily life. Among these, we may list five basic experiences (LePoidevin, 2000), that regard:

- Duration, that assigns the moments of a time interval to real-world behaviors and processes. The ways that different intervals might be related (inclusion, succession, overlap, etc.) places independent experiences into a common context that is useful for understanding the causal dynamics of the world.
- Simultaneity, that addresses how events that start and end at close but different moments, are experienced as occurring concurrently (i.e. we feel present as an interval rather than a durationless instant).
- Ordering, that is how we perceive precedence amongst events, i.e. not only perceiving events one after another, but rather perceiving an event as occurring after another considering also the relationships that may link them (Husserl, 1964; Tani, 2004).
- Past, present and future, that respectively regard (i) feelings we once sensed and we can not experience in the same way again, but we can recall an abstract (incomplete) representation of them (ii) feelings that we have here and now, or during short periods perceived as present (the specious present), (iii) feeling that we have never sensed but is possible to do so after some time, being capable of abstractly representing this possibility (in a way different than recalling the past) (Husserl, 1964; Tani, 2004; Varela, 1999).
- Time flow, that is the fundamental feeling of the present that is constantly changing in a unidirectional way, that makes future become present and then become past.

Additionally, from a brain science and artificial intelligence perspective, we are interested in the cognitive mechanisms of biological agents being involved in time perception and how they can be interpreted in a computational context. In particular we are interested in:

- the neural basis of time processing, addressing issues about the existence of separate subsystems for processing long and short time scales, the dedicated or implicit representation of time, and the role of development in acquiring time perception capacity.
- the interaction of time perception with other cognitive processes, investigating how we plan actions, filter information, direct attention to events, form decisions, or how can we understand the common and different properties of two similar behaviors executed in different time scales (i.e. temporal compression).

The topics mentioned above have attracted significant research interest in the fields of philosophy, cognitive science and neuroscience (Droit-Volet & Meck, 2007; Leon & Shadlen, 2003; Wittmann & Paulus, 2008). However, in the field of robotics and embodied intelligent systems, these issues remain largely unexplored. In a first attempt to explore temporal processing in artificial cognitive systems we have recently investigated how the feeling of temporal duration may self-organize in robotic cognitive agents (Maniadakis, Tani, & Trahanias, 2009). In the current work we extend this study comparing also the dynamics of the cognitive system with the dynamics of network structures evolved in a similar experimental setup with different temporal characteristics.

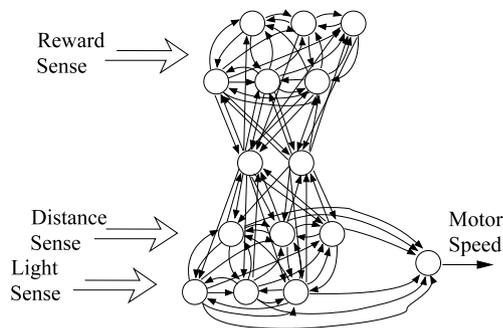


Fig. 1. Schematic representation of the bottleneck CTRNN used in the current study.

3. CTRNN model and input–output connectivity

In order to investigate how time perception affects the self-organization of rule switching capacity in neural dynamics we implement an artificial cognitive system based on a Continuous Time Recurrent Neural Network (CTRNN) (Beer, 1995). Interestingly, in CTRNNs contextual memory is implicitly represented by internal neurodynamics. Therefore, in our experimental setup, the neuronal state is initialized only once in the beginning of the first trial, and then neuronal dynamics continues across trials and phases without resetting (see experimental setup below). In this manner, we speculate that dynamical states will emerge for representing the rule stored in working memory, and additionally, these dynamical states might switch to one another according to the currently adopted rule.

Following our previous study (Maniadakis & Tani, 2008) showing that bottleneck configurations (Paine & Tani, 2005) are more effective in rule switching tasks compared to fully connected CTRNNs, the current work focuses only on the bottleneck architecture. As shown in Fig. 1, we use two bottleneck neurons to separate CTRNN in two levels. The bottleneck neurons loosely segregate information processing in each level, allowing the development of different roles by the groups of higher and lower level neurons. At the same time, the continuous interaction between levels supports their cooperation and the accomplishment of different behavioral patterns by the overall system.

Similar to previous studies (Paine & Tani, 2005; Yamauchi & Beer, 1996) all CTRNN neurons are governed by the standard leaky integrator equation:

$$\frac{d\gamma_i}{dt} = \frac{1}{\tau} \left(-\gamma_i + \sum_{k=1}^R w_{ik}^s I_k + \sum_{m=1}^N w_{im}^p A_m \right) \quad (1)$$

where γ_i is the state (cell potential) of the i -th neuron. All neurons in a network share the same time constant τ in order to avoid explicit differentiation of CTRNN parts. The state of each neuron is updated according to external sensory input I weighted by w^s , and the activity of presynaptic neurons A weighted by w^p . After estimating neural state by Eq. (1), then the activation of the i -th neuron is calculated by the non-linear sigmoid function according to:

$$A_i = \frac{1}{1 + e^{-(\gamma_i - \theta_i)}} \quad (2)$$

where θ_i is the activation bias applied on the i -th neuron.

In order to investigate embodied rule switching, we employ a two wheeled simulated robotic agent equipped with 8 uniformly distributed distance, light and reward sensors. The experiments

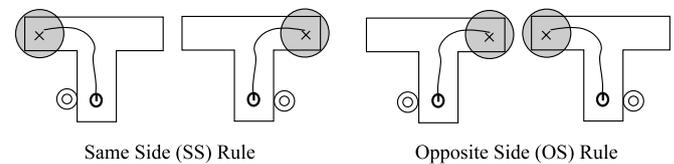


Fig. 2. A schematic representation of the delayed response rules. Light samples are represented by double circles. Goal locations are represented by \times , while reward corresponds to the gray area. The behavioral task asks for controllers capable of switching between the two rules.

discussed here have been carried out using YAKS¹ a simulated version of the real Khepera miniature mobile robot. The simulator has been slightly modified for the needs of the present study (e.g. by integrating a new type of sensors that supports feeling the special environmental signals simulating negative rewards).

4. Experimental setup

The current study is an extension of our previous work (Maniadakis & Tani, 2008), addressing meta-cognitive rule switching dynamics in a mobile-robot version of the classical Wisconsin Card Sorting (WCS) task (Greve, Stickle, Love, Bianchini, & Stanford, 2005; Milner, 1963). The motivation for our experiments is to provide self-organization pressure on simple neural network models which are evolved to accomplish a mobile-robot WCS task with time varying characteristics.

4.1. Mobile robot rule switching task

The task used in the current study is inspired by the rat version of WCS used to investigate rule switching capacity of rodents (Joel, Weiner, & Feldon, 1997). In particular, we assume that a mobile robotic agent is located at the bottom of a T-maze environment (see Fig. 2). At the beginning of a trial, a light sample appears at either the left or the right side of the robot. Depending on the light side, the robot has to move to the end of the corridor, making a 90° turning choice towards the left or right. The side of the light is linked to the choice of the robot according to two different sample-response rules (see Fig. 2). The first is called Same-Side (SS) rule implying that the robotic agent should turn left if the light source appeared at its left side, and it should turn right if the light source appeared at its right side. The second rule is named (OS), implying that robot should turn to the side opposite to the light.

The capacity of the agent to adopt and follow each rule can be evaluated by testing sequences of the above described trials. For example, consider that a human experimenter selects one of the rules (either SS or OS) and asks the agent to follow it for several trials. Based on the side of the light sample, the experimenter provides reward to the side of the T-maze that the robot should turn (see Fig. 2). Thus, every time that the robot gives a correct response, it drives to a reward area, knowing that it follows the right rule.

Turning now to rule switching, the experimenter at a random time (unknown to the robotic agent) changes the rule considered correct, positioning rewards according to a new sample-response rule. The task for the agent now is to discover this rule change, switching its response strategy in accordance to the new rule. The details of the experimental procedure are described below.

In order to explore the capacity of the robotic agent to switch among rules we have divided the sample-response sequence into

¹ The simulator has been developed in the University of Skovde, Sweden, and can be downloaded at <http://www.his.se/iki/yaks>.

$P \in \{1 \dots 10\}$ phases, each one consisting of T_p (randomly determined) trials. The number of trials $T_p \in \{8, 10, 12, 14\}$ is randomly specified, so that the agent can not predict the end of a phase. Let us assume that during the first phase $p = 1$, the experimenter selects SS as the correct rule. Then, for all T_1 trials the agent has to respond to the appearance of light samples at its left or right side (their order is randomly chosen) according to the SS rule. Every time the agent gives a correct response it receives a positive reward indicating it is following the correct rule. In case that the robot turning is not correct, it will drive to an area that no reward exists, indicating that the currently adopted rule is not correct and it should be switched. During phase p , the robot is given six free-of-cost exploratory trials to discover the currently correct rule specified by the experimenter. In the remaining $T_p - 6$ trials the performance of the robotic agent is evaluated in terms of following the desired response rule. If any of these trials is incorrect, the task is immediately terminated without completing the current phase and without investigating the next phases (we note that for the successful CTRNN controllers presented in the Results section, immediate terminations are very rare, occurring on average once every 84 trials).

If the agent completes T_p trials successfully, it moves to the next phase. In the beginning of phase $p + 1$ the experimenter changes the correct rule—to OS for our example. Therefore, reward signals are now positioned by the experimenter according to OS. The agent that is not aware for this change will continue responding according to the previous rule (i.e. SS). In that case, the agent will be unable to get any reward, indicating it is not following the correct rule. In order to get more reward, the robot must reconsider its rule choice, switching to OS. In phase $p + 1$, the robot is given again six free exploratory trials to discover rule switching. In the remaining $T_{p+1} - 6$ trials agent's responses are evaluated according to the correct response rule chosen by the experimenter. If any of these trials is incorrect, the evaluation is interrupted.

If the agent completes T_{p+1} trials successfully, it moves to the next phase. In phase $p + 2$ the experimenter changes again the correct rule – back to SS for our example – and a similar experimental procedure is repeated (i.e. due to the re-location of the reward cues the robotic agent needs to switch the adopted rule to SS). Overall, the task evaluates agent's switching behavior for a maximum of P phases (if all of them are completed successfully).

Trial duration. Due to the iterative nature of the Rule Switching task described above, we investigate robot responses for several trials. At the beginning of each trial the robot is located at a predefined starting position, with its direction randomly specified in the range $[85^\circ - 95^\circ]$ degrees (90° correspond to the direction of the corridor). The robot is kept in the same initial position for five simulation steps, and then it is allowed to navigate freely in the environment, responding to the presentation of the light sample at its left or right side.

The temporal length of each trial is not predefined, but it is determined on-line in a dynamic way. Specifically, each trial ends as soon as the agent reaches the current goal position at a distance of 10 environmental units. Therefore, trials with very fast robot responses will last shorter than those that the agent spends time exploring the environment. Additionally, we have defined an upper bound for the duration of a trial, being 200 simulation steps. At the end of the trial, we automatically reset robot to the starting position, and we are ready to test its behavior for the next trial (that will have again a dynamically determined temporal duration).

It is worth noting here that every time we artificially reset robot to the start position we do not make any artificial change to the robotic cognitive dynamics which are kept continuous, without any interruption (i.e. we do not reset neural state of the CTRNN controller). Following this approach, CTRNN functionality resembles the continuous nature of real brain dynamics.

The described experimental setup is particularly appropriate to uncover temporal differences between SS and OS rules, indicating the development of time perception capacity in CTRNN controllers, as will be described later in Section 6.

Table 1

The incrementally more complex tasks solved in different parts of the evolutionary procedure.

Generation	Task type	Description
1–60	Single phase	CTRNN reset - Task 1: SS CTRNN reset - Task 2: OS
61–140	Two phase	CTRNN reset - Task 1: SS → OS CTRNN reset - Task 2: OS → SS
141–300	Multiple phase	CTRNN reset - Task 1: SS → OS → SS → OS ... SS → OS CTRNN reset - Task 2: OS → SS → OS → SS ... OS → SS

5. Evolutionary procedure

We use a Genetic Algorithm² (GA) to explore how rule switching capacity self-organizes in CTRNN dynamics. In short, we use a population of artificial chromosomes encoding CTRNN controllers (their synaptic weights and neural biases). Each candidate solution encoding a complete CTRNN is tested on tasks examining the ability of the network to switch between rules. We evaluate the performance of all candidate CTRNN controllers assigning them an appropriate fitness value. The scores accomplished by the controllers are used to sort and evolve the population of chromosomes, therefore producing a new generation of CTRNN controllers that is ready for evaluation. This iterative procedure is repeated for a predefined number of generations. The details of the evolutionary procedure are described below.

Incremental evolution. In order to facilitate successful convergence of the evolutionary process we have used an incremental approach investigating gradually more complex versions of the rule switching problem. In the first 60 generations (see Table 1) the evolutionary process asks for robot controllers capable of adopting both SS and OS response rules. Two different tasks are used to evaluate CTRNN controllers. The robotic agent needs to explore the environment in order to discover which rule should be adopted for gaining rewards. Each task consists of only one phase. The accomplishment of Task1 implies that the robot can adopt rule SS, while the accomplishment of Task2 implies that the robot can adopt rule OS. At the beginning of each task the states of all CTRNN neurons are reset to zero, which means that the robot is in a neutral state, without following any rule.

In evolutionary generations 61–140, the tasks are getting more complex asking for controllers capable of one switching step between rules. Therefore tasks consist of two phases. Reward signals that have been properly positioned by the experimenter, indicate the correct response strategy for each phase. The Task1 examines agent's ability to adopt SS and then switch to OS. In a similar way, the Task2 examines robot's ability to first adopt OS and then switch to SS. At the beginning of each task the CTRNN state is reset to zero, but then it is kept continuous implying that special memory pathways have to develop facilitating rule switching from SS to OS and visa versa.

Finally, in generations 141–300 we ask for controllers capable of repeatedly switching between rules. Both Task1 and Task2 are now described by ten phases (see Table 1). Similarly to previous generations CTRNN is reset to zero at the beginning of each task, and then keeps continuous memory state when passing from one phase to the other (i.e. continuously switching between SS and OS rules).

Task evaluation. The accomplishment of tasks is evaluated based on the goal positions of each trial. The goal positions are specified

² The current evolutionary procedure does not mean to represent an artificial counterpart of biological evolution. It only serves our study as a consistent mechanism to explore the domain of solutions for our problem.

Table 2

The average distance of the paths followed by the robot in distinct trials. The second column shows average distance and variance between two SS paths, the third column shows average distance and variance between two OS paths and the fourth column shows average distance and variance between an SS and an OS path.

	SS vs SS intra-distance	OS vs OS intra-distance	SS vs OS inter-distance
Left turn	av:5.24 (var:3.94)	av:2.64 (var:1.52)	av:20.33 (var:9.81)
Right turn	av:4.79 (var:2.22)	av:3.41 (var:1.67)	av:22.42 (var:8.20)

according to (i) the current rule, and (ii) the side of the light sample (see Fig. 2). For each response of the robot, the minimum distance $d_{\min} \in [0, D]$ between the goal and the robot route is used to measure the success of the given robot turning choice (D is the distance between the starting position and the goal). For a task i evaluating the behavior of the robot for p phases, the success on rule switching is given by:

$$E_i = \sum_{q=1}^p \left(\sum_{t=7}^{T_q} \left(1 - \frac{d_{\min}}{D} \right) \right). \quad (3)$$

The evaluation starts from trial $t = 7$ because the first six trials of each phase are exploratory and they are not considered in evaluation. The higher the value of E_i the more rule switches the agent has accomplished.

Fitness measure. The individuals encoding CTRNN controllers are tested on *Task1* and *Task2* described above. The accomplishment of each task is evaluated separately according to Eq. (3). The total fitness of the individual is then estimated by:

$$fit = E_{Task1} \cdot E_{Task2}. \quad (4)$$

We note that the multiplication operator favors individuals that can accomplish (at least partly) both tasks, distinguishing them from the individuals that fail in any one of them.

Evolutionary process. A standard GA with mutation, but without crossover, is employed to evolve randomly initialized populations of 500 encoded CTRNNs. The evolutionary process is driven by the fitness function described in Eq. (4). At the end of each epoch, the $S = 30$ best individuals of the population are used as a basis for producing the individuals of the next generation. The new individuals are generated by randomly selecting and mutating one of the S individuals. Mutation corresponds to the addition of up to 30% noise, in the parameters encoded to the chromosome, while each parameter has a probability of 4% to be mutated.

6. Results

We have run ten GA processes, evolving CTRNN controllers to accomplish the incrementally more complex tasks described above. Six of the evolutionary procedures converged successfully configuring CTRNNs capable of rule switching. Interestingly, even if the corresponding evolutionary procedures have been statistically independent, all obtained results show (qualitatively) similar internal dynamics. Below we discuss the common characteristics among successful neuro-controllers, using as a working example one representative solution.

The performance of the agent during rule switching is demonstrated in Fig. 3. During trials 1–4 the agent follows SS rule, successfully acquiring rewards. Next, in trial 5 the experimenter changes rule to OS. The agent that is not aware of this change fails to accomplish reward for two consecutive trials, but then, in trial 7 it adopts OS. The rule is changed again in trial 15, where the agent is missing the reward. However, this time the agent switches very fast back to SS, accomplishing reward in trial 16, and continues responding according to SS for the rest trials.

We note that the agent follows different trajectories to gain rewards, depending on the rule adopted in each trial. For example, the left turning paths when SS is adopted (see trials 2, 3, 16, 18) are all similar, but different than the right turning paths when OS is adopted (see trials 7, 10, 12, 13). More specifically, we have calculated the average distance between the left or right paths followed by the agent for the case of the two rules. These are demonstrated in Table 2. Clearly, the distance between individual paths of the same rule is rather low (see columns 2, 3), while the distance between paths of different rules is significantly higher (see column 4). Therefore, embodiment and sensory-motor dynamics seem to have considerable correlations with rule encoding, or in other words, they have an important role in discriminating the two rules.

Additionally, we have investigated neural activity in the higher and lower levels of the CTRNN network (see Fig. 4). We observed that in all trials, lower level neurons fluctuate much faster than higher level neurons. This difference implies that higher level neurons are mostly involved in rule encoding and response planing, while the neurons below bottleneck are mostly involved in the execution of higher level plans taking also into account environmental interaction issues (e.g. wall avoidance). This property is an emergent result of evolutionary self-organization, that appears consistently in all CTRNNs capable of rule switching. It is worth emphasizing that our evolutionary design procedure does not artificially force CTRNN to develop different roles in the higher and lower levels. Non-successful networks follow different activation patterns. In the cases that we observe distinct neural activity in the two levels, higher activity either drives the agent always to a single side of the T-maze, or encodes only one of the available rules being unable to switch to the other. There is also the case that the network operates as a whole, being unable to distinguish between light and reward sensory information or being incapable to navigate efficiently in the environment.

Observing further Fig. 4, we see that trials 1 and 4 corresponding to the right turnings of the OS rule last longer compared to the other trials. In order to shed more light on this issue, we have calculated statistics regarding the temporal length of agent responses from a series of randomly initialized runs. In particular, we have estimated the number of simulation steps elapsed between successive positionings of the agent in the starting location. These are shown in Table 3. We see that for the case of left turning, SS and OS have nearly the same (average) duration, which is however not true for the case of right turning. As it is expected, the duration of the right OS turning is significantly longer to the rest. This makes the dynamics adopted for the two rules differentiate in terms of their temporal characteristics which in turn facilitates the agent to track the current rule, when passing from one trial to the next.

Furthermore, statistical information shows that for all four possible responses, duration may vary across independent responses. This is because the duration of trials depends on the time that the agent will reach the goal position (this is usually less than the maximum of 200 simulation steps per trial). A series of turnings for both rules is shown in Fig. 5. We see that the variations in the paths followed by the agent (due to the noise of sensors and actuators) can produce significant delays in the duration of trials.

Turning back to Fig. 4, we can further observe that two patterns of neural activation repeat across trials that correspond to either SS or OS rule (for example note the similarity in trials 1, 4 or 2, 3 and how they compare to trials 7, 9 and 8, 10 respectively). In an attempt to reveal the differences between the two rules, we have investigated neural activity of the original CTRNN configurations evolved for rule switching in a modified single-rule-following task, i.e. always follow only one of the SS or OS without switching

Fig. 3. The response of the agent in 22 consecutive trials (covering three phases). The robot initially follows rule SS, then it switches to OS, and back to SS.

Fig. 4. The activity of neurons in CTRNN layers, while the agent performs ten consecutive sample-response trials. The first line shows activity of a higher level neuron (H-N), the second line shows activity of a bottleneck neuron (B-N), and the third line shows activity of a lower level neuron (L-N). The exact paths followed by the robot at each trial are demonstrated in the last line.

Table 3

The average, minimum and maximum duration of sample-response trials, when the agent turns left and right following either the SS, or the OS rule.

	Rule SS duration Number of sim. steps	Rule OS duration Number of sim. steps
Left turn	av: 159 (min: 155 max: 163)	av: 158 (min: 150 max: 162)
Right turn	av: 154 (min: 148 max: 157)	av: 178 (min: 166 max: 186)

(these results are only briefly discussed here). Specifically, for both rules, we request the agent to perform 100 random turning trials (either left or right) after random perturbation of the neurons in the higher level. For both rules, CTRNN dynamics quickly converged to SS or OS after the perturbations, developing the same pattern of neural activity with the one shown in Fig. 4. Additionally, we have conducted PCA analysis on neural activity that revealed different patterns of principal components for SS and OS. In particular, Fig. 6 shows the phase plot of the first two principal components related to the activity of the higher level neurons (these components encompass 68% of the total information encoded in neural activity). For both rules we observe the same shape to appear in the phase plot, regardless of the perturbations in the initial state. Therefore, each plot represents a distinct invariant set for the corresponding rule. It is noted that neural activity moves on the same invariant sets when the agent is tested on the accomplishment of the original Task1 and Task2. The switching of the adopted rule from SS to OS and *visa versa*, corresponds to neural activity transitions from one invariant set to the other.

In order to explore the mechanisms facilitating rule switching, we have tested the obtained solutions in rule following under no-reward conditions (i.e. when the agent responds correctly reaching the goal position, it is not provided any reward). These experiments showed that SS performance crucially depends on reward signals (i.e. when reward is missing, the agent can not stabilize behavior to SS). However, the agent is capable of following the OS rule for

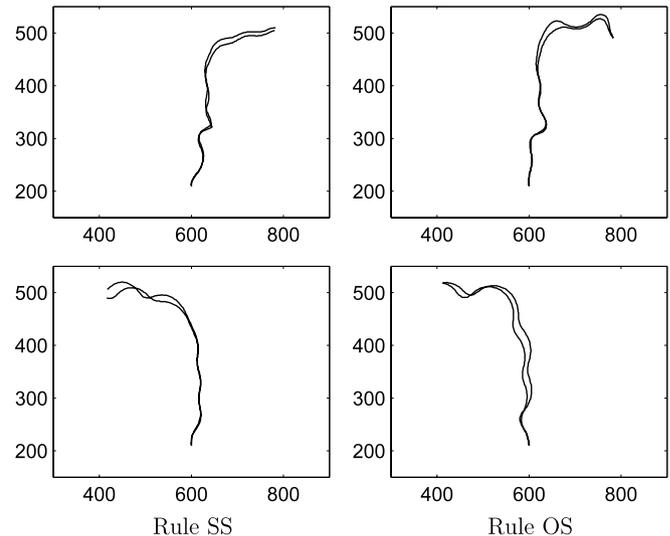


Fig. 5. Indicative left and right turnings of the robotic agent when it follows either the SS or the OS rule (the starting point corresponds to the bottom of the plots). Sensory-motor differences produce variation in the duration of sample response trials.

an adequately long number of trials, even if no reward is provided. But, if missing the expected reward does not affect OS, then what makes the agent shift to SS in the ordinary rule switching task? A careful investigation of trial 5 in Fig. 4 (i.e. an erroneous response after the experimenter has changed the rule), shows that the duration of the underlying trial is considerably larger compared to the previous ones. This is because the agent is unable to meet the goal position, spending the maximum of available simulation steps in the current trial. This makes the cognitive system dynamics and the expected duration of the trial de-synchronize, which results

