

# Time Perception in Shaping Cognitive Neurodynamics of Artificial Agents

Michail Maniadakis, Jun Tani and Panos Trahanias

**Abstract**—Recent research in cognitive systems aims to uncover important aspects of biological cognitive processes and additionally formulate design principles for implementing artificially intelligent systems. Despite the increasing amount of research efforts addressing cognitive phenomena, the issue of time perception and how it is linked to other cognitive processes remains largely unexplored.

In the current paper, we make a first attempt for studying artificial time perception by means of simulated robotic experiments. Specifically, we investigate a behavioral rule switching task consisting of repeating trials with dynamic temporal duration. An evolutionary process is used to search for neuronal mechanisms accomplishing the rule switching task taking also into account its particular temporal characteristics. Our repeated simulation experiments showed that (i) time perception and ordinary cognitive processes may co-exist in the system sharing the same neural resources, and (ii) time perception dynamics bias the functionality of neural mechanisms with other cognitive responsibilities. Finally, in the current paper we make contact of the obtained results with previous brain imaging studies on time perception, and we make predictions for possible time-related dynamics in the real brain.

## I. INTRODUCTION

**R**EALITY is dynamic. Sensing and knowing our world arises through spatiotemporal experiences and interpretations. In fact, dynamics is so essential to reality that a static world is difficult to imagine. Evolution has equipped human brain with the capacity to perceive and understand in a meaningful way the spatial and temporal aspects of everyday phenomena.

The computational interpretation of how human perceptuo-motor loop performs during real world interaction, is a very common approach for developing artificially intelligent systems. Almost two decades ago, the necessity for embodiment and environmental interaction has been comprehended as a key feature for developing intelligent systems [1]. Since then, simulated and real robotic systems have been extensively used as testbeds of the embodied intelligence. Unfortunately, the majority of the existing cognitive and robotic systems, concentrate only on the spatial characteristics of environmental interaction, nearly ignoring the other aspect that is

Michail Maniadakis is a researcher in the Computational Vision and Robotics Laboratory, Institute of Computer Science, Foundation for Research and Technology – Hellas (FORTH), Crete, Greece. Additionally, he is a visiting researcher in the Laboratory for Behavior and Dynamic Cognition, RIKEN Brain Science Institute, Wako-shi, Saitama Japan. (email: mmaniada@ics.forth.gr).

Jun Tani is head of the Laboratory for Behavior and Dynamic Cognition, RIKEN Brain Science Institute, Wako-shi, Saitama Japan. (email: tani@brain.riken.jp).

Panos Trahanias is head of the Computational Vision and Robotics Laboratory, Institute of Computer Science, Foundation for Research and Technology – Hellas (FORTH), Crete, Greece. (email: trahania@ics.forth.gr).

necessary for describing our dynamic world, namely the perception of time [2]. 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. [3], [4], [5], [6], [7]).

However, existing robotic systems are lacking the ability to perceive time passage. In the majority of existing systems, time is only partially observed 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. However, in biological systems, time has a more central role (in memory organization, reasoning, knowledge acquisition, etc.) shaping the dynamics of other pure cognitive processes. There are now two major models for the neural representation of time in the cortex [8]. 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, 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, avoiding to arbitrary favor any of the time perception models described above. As it is suggested in [9], an evolutionary robotics approach [10], [11] can be used to resolve this issue. In particular, we apply evolutionary pressure on simple Continuous Time Recurrent Neural Network (CTRNN) controllers [12], being free to self-organize in any direction, revealing the most appropriate time perception mechanism. Similar to [13], 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.

The rest of the paper is structured as follows. In section II we describe the CTRNN models used in the current study. In section III we described the investigated task providing the details of our experimental setup. Then we present the evolutionary procedure used to explore the space of CTRNN

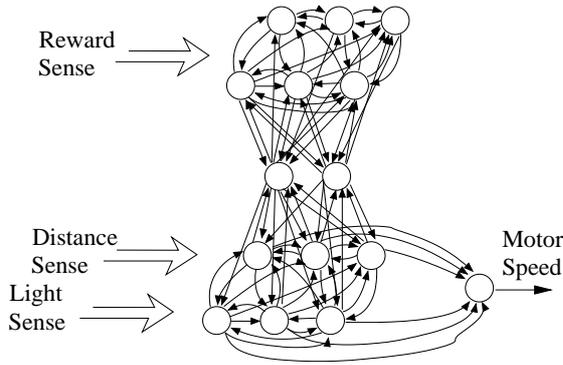


Fig. 1. Schematic representation of the bottleneck CTRNN used in the current study.

solutions. In section V 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 [13] investigating rule switching using trials of predefined temporal duration. In section VII 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.

## II. CTRNN MODEL AND INPUT-OUTPUT CONNECTIVITY

We use Continuous Time Recurrent Neural Network (CTRNN) models [12] to investigate how time perception affects the self-organization of rule switching capacity in neural dynamics. 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. 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 [13] showing that bottleneck configurations [14] 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, maintaining minimum interactions between them. All neurons are governed by the standard leaky integrator equations described in previous studies [15], [14].

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 discussed here have been carried out using

YAKS<sup>1</sup> 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).

## III. EXPERIMENTAL SETUP

The current study is an extension of our previous work [13], addressing meta-cognitive rule switching dynamics in a mobile-robot version of the classical Wisconsin Card Sorting (WCS) task [16], [17]. 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.

### A. 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 [18]. 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, let's assume 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 according 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  $P \in \{1 \dots 10\}$  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

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



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.

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).

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 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 ( $90^\circ$  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,

Single Phase		Two Phases		Multiple Phases	
Generations: 1-60		Generations: 61-140		Generations: 141-300	
Task 1	Task 2	Task 1	Task 2	Task 1	Task 2
Reset	Reset	Reset	Reset	Reset	Reset
CTRNN	CTRNN	CTRNN	CTRNN	CTRNN	CTRNN
SS	OS	SS ↓	OS ↓	SS ↓	OS ↓
		OS	SS	OS ↓	SS ↓
				SS ↓	OS ↓
				...	...
				OS ↓	SS ↓
				SS	OS

TABLE I  
THE INCREMENTALLY MORE COMPLEX TASKS SOLVED IN DIFFERENT PARTS OF THE EVOLUTIONARY PROCEDURE.

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 V.

#### IV. EVOLUTIONARY PROCEDURE

We use a Genetic Algorithm<sup>2</sup> (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

<sup>2</sup>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.

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 I) 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 SS rule, while the accomplishment of *Task2* implies that the robot can adopt OS rule. 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 I). 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 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 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:

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

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 (1). The total fitness of the individual is then estimated by:

$$fit = E_{Task1} \cdot E_{Task2} \quad (2)$$

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 (2). 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.

## V. 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). 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

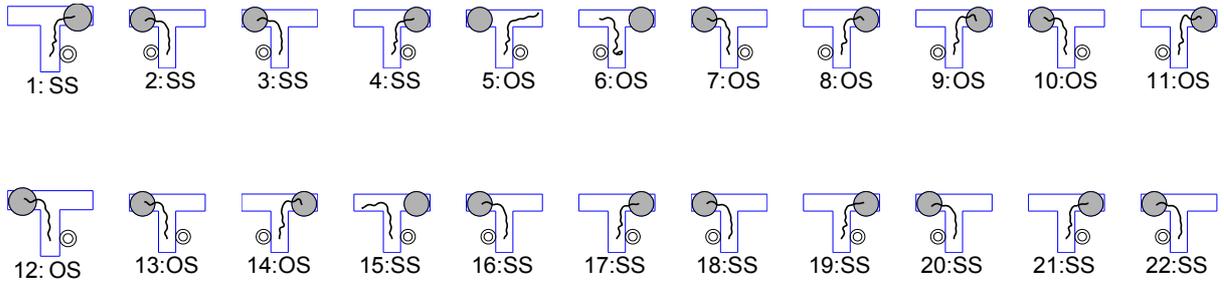


Fig. 3. The response of the agent in 22 consecutive trials (covering three phases). The robot initially follows SS rule, then it switches to OS, and back to SS.

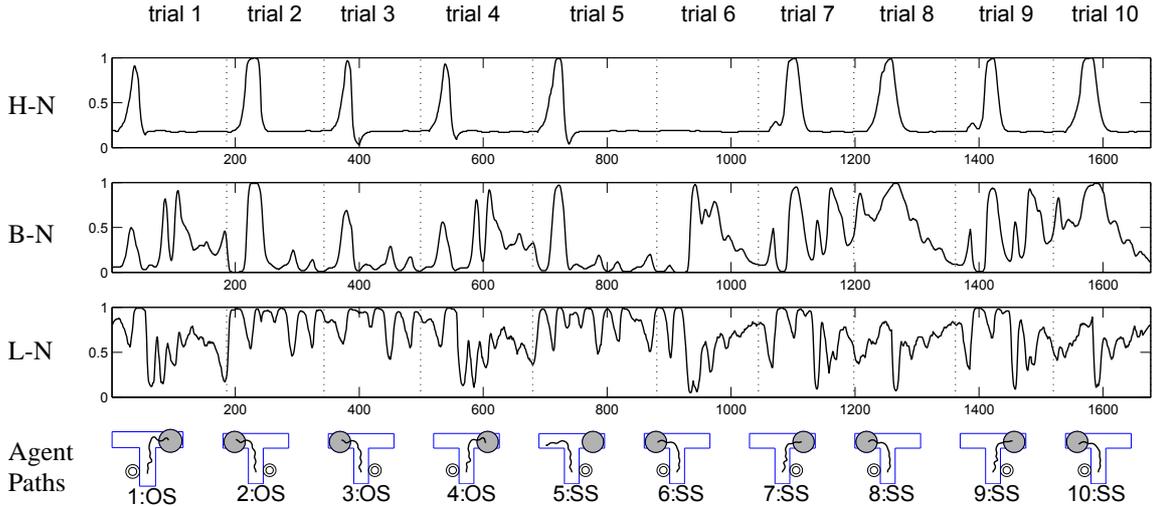


Fig. 4. The activity of neurons in CTRNN layers, while the agent performs ten consecutive sample-response trials. The first line shows activity of a higher level neuron (H-N), the second line shows activity of a bottleneck neuron (B-N), and the third line shows activity of a lower level neuron (L-N).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

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.

After careful examination of Fig 4, we observed two patterns of neural activation (corresponding to either SS or OS rules) that are repeated across trials. Trying to reveal the differences between SS and OS rules, we have conducted attractor analysis taking the phase plots of activities for the neurons H-N and L-N shown in Fig 4, in a modified version of the previous tasks, testing agent’s responses in a no-switching task (i.e. follow only one of the SS or OS). Specifically, for both rules, we ask the agent to perform 100 random turning trials (either left or right) after random perturbation of the neurons in the higher level. For each rule we observed the same shape to appear in the phase plot, regardless of the randomness in the initial state (Fig 5).

	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)

TABLE II  
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.

Therefore, each plot represents a distinct invariant set for the corresponding rule. In our no-switching experiment, CTRNN neuronal state always converge to one of the two invariant sets depending on the reward stimuli specifying the currently correct 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.

Furthermore, during the no-switching task we observed that even if the pattern of neural activation remains macro-

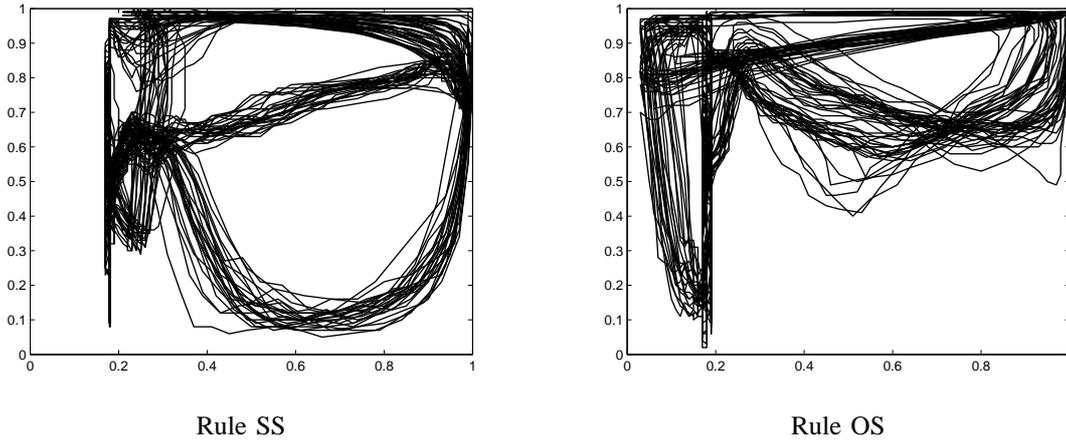


Fig. 5. Phase plots of CTRNN internal activity for 100 sample-response trials, while the agent follows either the SS or the OS rules. The x-axis corresponds to neuron H-N from Fig 4, while the y-axis corresponds to neuron L-N from the same figure.

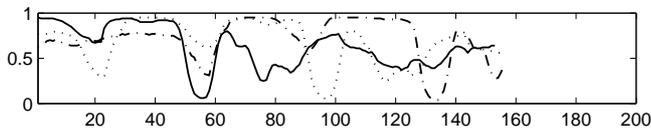


Fig. 6. The shifting of neural activity of lower level neuron, in repeating SS trials. The solid line shows activation in trial 25, the dotted line shows activation in trial 55, and the dashed-dotted line shows activation in trial 85.

scopically the same, the activation corresponding to the end of trials is gradually shifted in a circular mode (Fig 6). This is because the temporal length of trials is dynamic, and depends on how goal position is reached. Statistical information about the temporal length of trials is shown in Table II. Specifically, the duration of each sample-response trial 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 7. 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.

The same shifting phenomenon is also observed in the case of original rule-switching task. Furthermore, now the shifting effect is supported by one more factor. When the experimenter unexpectedly changes the rule, the robot that is not aware of this change is searching for reward in the wrong side of the T-maze, spending the total of the 200 available simulation steps. The longer duration of the erroneous-response trials (in comparison to the durations presented in Table II) will cause a shift of the next trial starting point, relative to the repeating pattern of neural activity.

In terms of phase plots, this shifting corresponds to moving the transition points related to the switching between OS and SS invariant sets. In other words there is not a single transition point from one attractor to the other, but this is possible to occur in several locations of the invariant set.

Finally we have also tested the performance of CTRNN

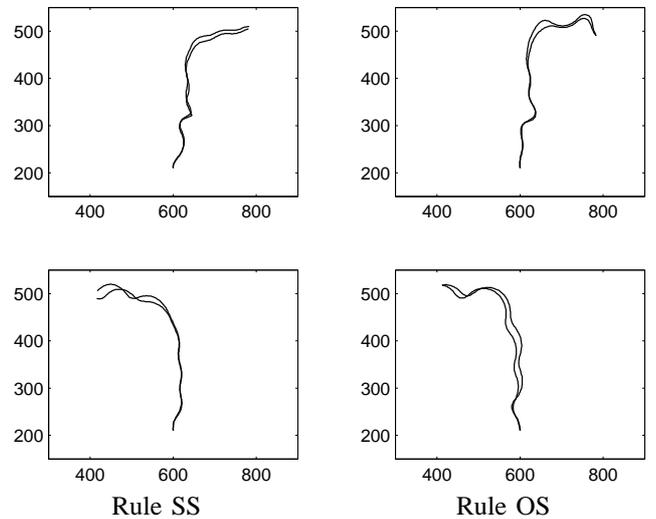


Fig. 7. Indicative left and right turnings of the robotic agent when it follows either the SS or the OS rule. Sensory-motor differences produce variation in the duration of sample response trials.

controller in ‘no-reward’ conditions. In that case, the agent has been possible to adopt the OS rule and follow it for many trials, but it was unable to switch to SS. This means that CTRNN is self-organized assuming OS as the default rule, that can occasionally switch to SS. The internal bias that drives neurodynamics to a preferred invariant set is similar to [6].

It is worth noting that what seems important for the preference of the cognitive system to OS rule, is the different duration of trials when the agent turns left or right (see Table II). This argument is reinforced by the fact that the named result -different duration of trials- appears consistently in all successful CTRNN controllers that were obtained by statistically independent evolutionary processes (i.e. when there is significant difference in the duration of left and right turnings the response rule is preferred against the other). This, in turn,

clearly suggests that the CTRNN controllers have developed an internal time-perception mechanism that can be used to discriminate the two response rules. Subsequent investigation of the CTRNN computational units could not reveal any dedicated subset of neurons with time measure responsibilities. Therefore, in relation to the time perception models proposed in [8], our CTRNN controllers have developed an intrinsic rather than a dedicated mechanism of time perception. In other words, time is not just a variable existing in the neural equations of the CTRNN model, but something much more rich and powerful that can affect the self-organization of neurodynamics. These results are in agreement with recent brain imaging studies showing that time perception shares neural resources with other cognitive processes (in particular working memory) [19], favoring the intrinsic approach.

## VI. STATIC VS DYNAMIC DURATION OF TRIALS

We believe it is important to compare the current results with those obtained in our previous study [13] investigating rule switching assuming a static trial duration. In particular we had investigated switching between the same SS and OS rules with all robotic agent trials lasting exactly 170 simulation steps. At the end of a trial we automatically reset the robot to the start position (without resetting neurocognitive dynamics), and we are ready to test its behavior for the next trial lasting again 170 simulation steps. In order to discriminate the two versions of rule switching we will refer to the problem investigated in the current paper as Dynamic Duration (DD), and the problem investigated in [13] as Static Duration (SD).

The investigation of successful CTRNN controllers in SD rule switching showed that, similar to the current study, rules are encoded in distinct invariant sets, and that embodiment together with environmental interaction significantly facilitate the discrimination of the sample-response rules. Furthermore, each rule is encoded in a separate neural activation pattern in the higher part of the CTRNN. Specifically, Fig 8 shows activation of a higher level neuron for the SS and OS rules. It is worth noting that there was no neural activation shifting across trials and therefore these patterns appeared always in the same form in a single trial. Additionally, note that for both the left and the right turnings of SS, neural activity starts and ends at very high values. In contrast, for the case of OS rule, neural activity starts and ends at very low. This high-low difference facilitates (i) the discrimination between the two rules, and (ii) the binding of left and right turnings linked to the same single rule. It is important to note that the nature of the SD rule switching problem allows the emergence of the above mentioned neurodynamic characteristics due to the perfectly measured and exact temporal duration of trials (all of them lasting 170 simulation steps) that is perfectly synchronized with the activation of neurons. However, the above described high-low mechanism, could not work for our current DD study, because of the variation appearing in the duration of trials (see Table II), preventing their synchronization with neural activity.

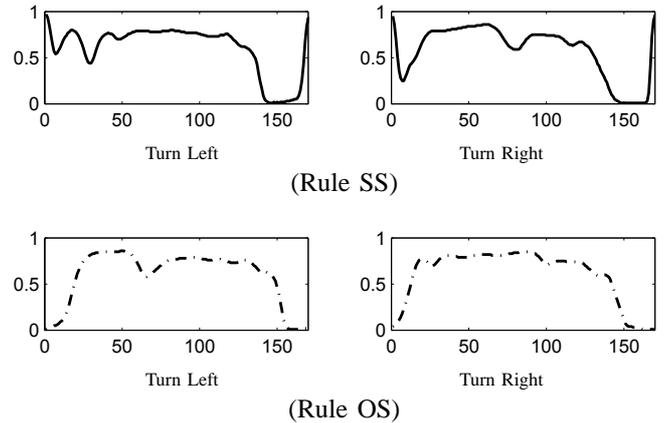


Fig. 8. The activation of one higher level neuron in different trials of the SD rule switching task. In the case of the SS rule (solid line) neural activation starts and ends at high values, while in the case of OS (dashed-dotted line) neural activation starts and ends at low values. The figure is copied from [13].

The short comparison between our current DD and the previous SD rule switching study shows that temporal constraints of the experimental setup significantly bias the self-organization of internal cognitive dynamics. However, the parameter of time is rarely taken into account in robotic cognitive studies and we believe it worths more attention from researchers in order to get a complete picture of cognitive phenomena in biological and artificial agents.

## VII. DISCUSSION-TIME PERCEPTION AND NEURO-COGNITIVE DYNAMICS

In the current study, we have evolved Continuous Time Recurrent Neural Networks (CTRNNs) on a rule-switching task consisting of trials with dynamic temporal duration. The continuous nature of CTRNN controllers is very important for the study of cognitive process related to rule switching, because real brain operates also in a continuous mode [20]. Therefore the present study can reveal new unexplored aspects of brain processes involved in switching from one behavioral strategy to another in conditions of varying trial duration.

The relevance of the obtained CTRNN solutions to real brain is supported by the emergent properties of neurocognitive dynamics. First, we found that time perception may co-exist with other cognitive processes (for our task, those involved in rule discrimination). This is in agreement with [19] arguing that time perception shares common neural resources with other cognitive processes (see also [8] discussing possible models of time perception in neurocognitive systems).

Furthermore, our results are in agreement with [2], that investigates possible mechanisms for encoding temporal duration. According to this study, it is not necessary to have a linear time counter to accomplish (primitive) skills of time perception. Similar to [2], in the current work, CTRNN solutions have been capable of discriminating between the

default OS and the alternative SS rule, with the first having significant differences in the duration of left and right turnings. Additionally, we observed the shaping of invariant sets in cognitive neurodynamics that supports the time-perception capacity of agents, discriminating SS and OS rules having different duration. This is also in agreement with [2].

However, our work addresses an issue that is not studied by previous works. Specifically, due to the varying duration of sample-response trials, and the shifting phenomena that we discussed in the Results section, the DD problem investigated here asks for a time-invariant representation of SS and OS rules. This is accomplished by the development of invariant sets (corresponding to either SS or OS), which can switch not only by means of a single transition point, but by means of a transition surface.

This is accomplished by the development of distinct activation patterns (each one corresponding to either SS or OS) interpreted as distinct invariant sets in phase plots. However, the varying temporal duration of trials require the development of multiple transition points from one invariant set to the other, rather than the common single point transition (this was also the case for our SD results). Therefore, we predict that the shaping of time-invariant representations in continuous cognitive systems (e.g. represented by CTRNN) requires a transition surface to support invariant set switching.

### VIII. CONCLUSIONS

The current study aims to shed light on a largely unexplored aspect of cognition, namely time perception, and its relation to other cognitive processes. By adopting a dynamical systems approach to explore mechanisms shaping neurodynamics we found that the continuous nature of cognition provides to ordinary cognitive processes primitive time perception capacity. Additionally, we found that the temporal constraints of tasks can significantly bias the shaping of internal dynamics of the system.

Overall, the investigation of time is an important parameter/aspect for the complete and in-depth understanding of cognitive processes. The current study is a first attempt towards a rigorous and systematic exploration of the time perception capacity of cognitive agents. In the future, we aim at systematically exploring more aspects of time perception, investigating problems that combine perception of both static and dynamic duration tasks.

### REFERENCES

[1] R. Brooks, "Intelligence without reason." in *Proc. of 12th International Joint Conference on Artificial Intelligence*, J. Mylopoulos and R. Reiter, Eds. Morgan Kaufmann, 1991.

[2] U. R. Karmarkar and D. V. Buonomano, "Timing in the absence of clocks: Encoding time in neural network states," *Neuron*, vol. 53, no. 3, pp. 427 – 438, 2007.

[3] J. Demiris and G. Hayes, "Imitation as a dual-route process featuring predictive and learning components: a biologically plausible computational model," pp. 327–361, 2002.

[4] K. Doya, K. Samejima, K. Katagiri, and M. Kawato, "Multiple model-based reinforcement learning." *Neural Computation*, vol. 14, pp. 1347–1369, 2002.

[5] E. Oztop and M. A. Arbib, "Schema Design and Implementation of the Grasp-Related Mirror Neuron System." *Biological Cybernetics*, vol. 87, pp. 116–140, 2002.

[6] H. Iizuka and E. A. D. Paolo, "Toward spinozist robotics: exploring the minimal dynamics of behavioral preference," *Adaptive Behavior*, vol. 15, no. 4, pp. 359–376, 2007.

[7] K. Dautenhahn and C. L. Nehaniv, Eds., *Imitation in Animals and Artifacts*. Cambridge, MA: MIT Press, 2002.

[8] R. Ivry and J. Schlerf, "Dedicated and intrinsic models of time perception," *Trends in Cognitive Sciences*, vol. 12, no. 7, pp. 273–280, 2008.

[9] E. Ruppin, "Evolutionary autonomous agents: A neuroscience perspective," *Nature Reviews Neuroscience*, vol. 3, no. 2, pp. 132–141, 2002.

[10] S. Nolfi and D. Floreano, *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*. MA: MIT Press/Bradford Books, 2000.

[11] M. Maniadakis and P. Trahanias, "Hierarchical co-evolution of cooperating agents acting in the brain-arena," *Adaptive Behavior*, vol. 16, no. 4, pp. 221–245, 2008.

[12] R. Beer, "On the dynamics of small continuous-time recurrent neural networks," *Adaptive Behavior*, vol. 3, no. 4, pp. 471–511, 1995.

[13] M. Maniadakis and J. Tani, "Dynamical systems account for meta-level cognition," in *10th Int. Conf. on the Simulation of Adaptive Behavior (SAB-08)*, 2008, pp. 311–320.

[14] R. Paine and J. Tani, "How hierarchical control self-organizes in artificial adaptive systems," *Adaptive Behavior*, vol. 13, no. 3, pp. 211–225, 2005.

[15] B. M. Yamauchi and R. D. Beer, "Spatial learning for navigation in dynamic environment," *IEEE Trans. Syst. Man Cybern.*, vol. 26, no. 3, 1996.

[16] B. Milner, "Effect of Different Brain Lesions on Card Sorting," *Archives of Neurology*, vol. 9, pp. 90–100, 1963.

[17] K. Greve, T. Stickley, J. Love, K. Bianchini, and M. Stanford, "Latent structure of the Wisconsin Card Sorting Test: a confirmatory factor analytic study," *Archives of Clinical Neuropsychology*, vol. 20, pp. 355–364, 2005.

[18] D. Joel, I. Weiner, and J. Feldon, "Electrolytic lesions of the medial prefrontal cortex in rats disrupt performance on an analog of the Wisconsin Card Sorting Test, but do not disrupt latent inhibition: implications for animal models of schizophrenia," *Behavioural Brain Research*, vol. 85, pp. 187–201, 1997.

[19] P. Lewis and C. Miall, "Remember the time: a continuous clock," *Trends in Cognitive Sciences*, vol. 10, no. 9, pp. 401–406, 2006.

[20] S. Scherbaum, M. Dshemuchadse, and A. Kalis, "Making decisions with a continuous mind," *Cognitive, Affective, & Behavioral Neuroscience*, vol. 8, no. 4, pp. 454–474, 2008.