Single agents can be constructivist too

Details of authors:
Olivier L. Georgeon,
Université de Lyon, CNRS,
LIRIS, UMR5205,
olivier.georgeon@liris.cnrs.fr

Salima Hassas,
Université de Lyon, CNRS,
LIRIS, UMR5205,
salima.hassas@univ-lyon1.fr

Upshot:
We support Roesch and his co-authors' theoretical stance on constructivist artificial agents, and wish to enrich their "exploration of the functional properties of interaction" with complementary results. By revisiting their experiments with an agent that we developed previously, we explore two issues that they deliberately left aside: autonomous intentionality and dynamic reutilization of knowledge by the agent. Our results reveal an alternative pathway to constructivism that addresses the central question of intentionality in a single agent from the very beginning of its design, suggesting that the property of *distributed processing* proposed by Roesch et al. is not essential to constructivism.

Main text:
1. In their paper "Exploration of the functional properties of interaction: Computer models and pointers for theory", Roesch and his coauthors formulate a constructivist approach to artificial learning in which "knowledge of the world, for an individual, is created from the interaction with the environment, rather than existing in an ontic reality, supposedly pre-existing or available to registration from the physical world" (§1). They propose three models to illustrate this idea, in which a swarm of agents performs different tasks in an environment made of a string of digits. We fully agree with this theoretical stance but we feel that these models do not illustrate it as well as possible. In particular, one might argue that the swarm's knowledge does, in fact, "exist in an ontic reality" since the agents directly "perceive" the digits and apply predefined rules to process the digits for the purpose intended by the designer.

2. Here, we present an alternative model that does not make the knowledge of the environment directly available to registration by the agent. The environment is the string of digits presented in §23, and the agent was designed to produce similar results as proposed in §25: sorting the string. Yet, the agent's observations are reduced to a single bit whose significance depends on the dynamics of the agent's interactions rather than directly reflecting the state of the environment. The agent remains "unaware" that it "exists" at a particular position in a string of digits, and its own goal is not to sort this string. For the agent, the construction of knowledge consists of learning to organize its behavior to fulfill a form of intentionality defined independently of the environment.

Implementation

3. *Initialization*: The environment is a string of 10 digits E_0 =[6, 3, 5, 4, 7, 3, 5, 3, 9, 5] plus an integer p in the interval [0, 9] that represents the agent's position. $E_t[p_t]$ denotes the digit at the agent's position at time t. At time 0, p_0 = 0, thus the current digit $E_0[p_0]$ = 6.

4. *Behaviors*: At time t, the agent chooses an action from amongst the set of three possible actions A = {*step*, *feel*, *swap*}, and then receives a binary observation from amongst the set of two possible observations O = {*true*, *false*}. The set $A{\times}O$ thus contains 6 possible *interactions*. The agent initially ignores the meaning of actions and observations, i.e., it implements no rule to process them specifically. However, each interaction has a predefined *valence* that plays a role in defining the agent's intentionality, as explained below. Unbeknownst to the agent, *step* consists of stepping to the next digit. If this action takes the agent to a greater or equal digit then it produces observation *true* and has a positive valence, otherwise, it produces observation *false* and has a strongly negative valence. *Feel* consists of testing whether the next digit is greater than or equal to the current one, if yes it produces *true*, otherwise *false*. *Feel* interactions have a mildly negative valence. *Swap* consists of trying to swap the current digit with the next; it succeeds only if the current digit is greater than the next, producing observation *true* and a positive valence, and otherwise it does nothing and produces observation *false* and a strongly negative valence. When the agent is at position 9, *step* returns the agent to position 0, *swap* does nothing, and the three actions produce observation *false*. Table 1 summarizes the implementation of these possibilities of interaction.

Table 1: possibilities of interaction available to the agent.

Action	Condition	Effect	Observation	Interaction	Valence
step	$p_t < 9$ and $E_t[p_t] \leq E_t[p_t+1]$	$p_{t+1} = p_t + 1$	true	step_up	4
	$p_t < 9$ and $E_t[p_t] > E_t[p_t+1]$	$p_{t+1} = p_t + 1$	false	step_down	-10
	$p_t = 9$	$p_{t+1} = 0$	false	step_down	-10
feel	$p_t < 9$ and $E_t[p_t] \leq E_t[p_t+1]$	-	true	feel_up	-4
	$p_t < 9$ and $E_t[p_t] > E_t[p_t+1]$	-	false	feel_down	-4
	$p_t = 9$	-	false	feel_down	-4
swap	$p_t < 9$ and $E_t[p_t] \leq E_t[p_t+1]$	-	false	not_swap	-10
	$p_t < 9$ and $E_t[p_t] > E_t[p_t+1]$	$E_{t+1}[p_t+1] = E_t[p_t]$ $E_{t+1}[p_t] = E_t[p_t+1]$	true	swap	4
	$p_t = 9$	-	false	not_swap	-10

5. *Agent*: we used an agent presented previously (Georgeon & Ritter 2012), which was programmed to exhibit two forms of intentionality: the tendency to select sequences of actions that produce well-predicted observations, and the tendency to enact positive interactions while avoiding strongly negative interactions. The former type of intentionality relates to Steel's (2004) autotelic principle (the enjoyment of being *in control* of one's activity), and was implemented as a tendency to record, hierarchically organize, and appropriately re-enact sequences of interactions that capture regularities in the coupling between the agent and the environment. The latter is called interactional motivation (Georgeon, Marshall, & Gay 2012), and was implemented through preferentially engaging in sequences of interactions that have the highest total valence.

Results:

6. Table 2 reports selected strips of behaviors, with the current digit marked in a box. The agent started by randomly picking the *step* action at time 1 and 2. Over time, the agent organized its behavior as if it had discovered that the *feel* action could be used to test the next digit. If this action resulted in the *feel_up* interaction, then the *step_up* interaction could subsequently be enacted, otherwise, the *swap – step_up* sequence could subsequently be enacted. This dynamics resulted in the behavior of "carrying digits to the right". This behavior is illustrated in Table 2 from time 106 to 113: the agent "carried" the "5" digit from position 2 to 4, by repeating the *feel_down – swap – step_up* sequence until the "5" digit got "blocked" by a greater or equal digit (another "5" digit at position 5). This behavior resulted in the string being entirely sorted at time 130.

Table 2: Behavior strips.

Time	Interaction	Environment
0	-	$\boxed{6}$ 3 5 4 7 3 5 3 9 5
1	step_down	6 $\boxed{3}$ 5 4 7 3 5 3 9 5
2	step_up	6 3 $\boxed{5}$ 4 7 3 5 3 9 5
...		
106	feel_down	3 4 $\boxed{5}$ 3 3 5 5 6 7 9
107	swap	3 4 $\boxed{3}$ 5 3 5 5 6 7 9
108	step_up	3 4 3 $\boxed{5}$ 3 5 5 6 7 9
109	feel_down	3 4 3 $\boxed{5}$ 3 5 5 6 7 9
110	swap	3 4 3 $\boxed{3}$ 5 5 5 6 7 9
111	step_up	3 4 3 3 $\boxed{5}$ 5 5 6 7 9
112	feel_up	3 4 3 3 $\boxed{5}$ 5 5 6 7 9
113	step_up	3 4 3 3 5 $\boxed{5}$ 5 6 7 9
...		
130	swap	3 3 $\boxed{3}$ 4 5 5 5 6 7 9

7. Figure 1 reports the agent's behavior until time 200, in terms of what matters to the agent: the enacted interactions, their valence, and the level of control that the agent has over its activity manifested by the length of the sequences intentionally enacted.

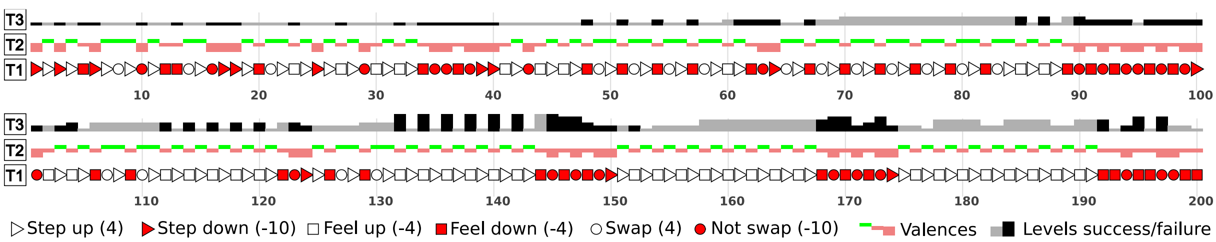

Figure 1: Analysis of the first 200 interactions enacted by the agent. Tape T1: the enacted interactions (the shape represents the action and the color the resulting observation). Tape T2: the valence of the enacted interactions displayed as a bar graph (green when positive, red when negative). Tape T3: The length of the sequences intentionally enacted, displayed as a bar graph. Higher levels of gray indicate better control over the activity; black segments indicate that an intended sequence was interrupted due to the failure to correctly predict the resulting observation. This trace shows that the behavior was unorganized approximately until time 40 (no regularities in the symbols in T1 and the presence of *step_down* and *not_swap* interactions that have strong negative valence represented by high red bars in T2). The agent intentionally enacted the second order sequence *swap – step_up* for the first time during time 68-69 (second level in T3), then the third-order sequence *feel_down – swap – step_up* during time 70-72 (third level in T3), repeating this sequence until time 85. After time 130, the digits were entirely sorted, and the agent engaged in repeating the sequence *feel_up – step_up*, except when reaching the end of the string, in which case it continued experimenting other behaviors (episodes 144 – 150, 168 – 174, and 192 – 200). After time 310 (not shown), the agent resigned itself to merely enacting the *step_down* interaction when reaching the end of the string, acknowledging that it had no better possibilities.

8. In summary, this experiment helps clarify the distinction between the designer's goal (sorting the string, illustrated in Table 2) and the agent's intentionality (being in control and enacting interactions that have positive valence, illustrated in Figure 1). While the agent remained unaware of the underlying structure of the environment, it learned to master sensorimotor contingencies as if it enjoyed being able to predict its activity and to "step up", and disliked "stepping down" and failing to swap digits. The agent learned to use the *feel* action—in spite of its negative valence and the ignorance of its meaning—as an active perception of the environment to inform subsequent behaviors. This activity illustrates the property pointed out by Roesch et al. that "perception is an integral part of the process from which knowledge of the world arises" (§7) and that "Exploration of the environment provides the organism with the ability to sense and become attuned to the laws governing change" (§8). We believe that these properties, associated with the capacity of the agent to engage in incremental learning, qualify the agent as a candidate to illustrate key aspects of constructivism.

9. Roesch et al. conclude by listing the properties of interactions that they judge to be paramount to the constructivist approach: "partial information, exploration, distributed processing, aggregation of information, emergence of knowledge and directedness towards relevant information" (§40). Our results support all of these except the *distributed processing* property (insofar as it applies to a swarm of agents), and suggest the additional property of *intrinsic intentionality*.

References

Georgeon O. & Ritter F. (2012) An intrinsically-motivated schema mechanism to model and simulate emergent cognition. Cognitive Systems Research 15-16: 73-92.

Georgeon O., Marshall J., & Gay S. (2012) Interactional motivation in artificial systems: between extrinsic and intrinsic motivation. In proceedings of the 2nd International Conference on Development and Learning, and on Epigenetic Robotics (EPIROB2012), San Diego: 1-2.

Steels L. (2004) The Autotelic Principle. In: Fumiya I., Pfeifer R., Steels L., & Kunyoshi K. (eds.) Embodied Artificial Intelligence. Springer Verlag: 231-242.