Sense-making and knowledge construction via constructivist learning paradigm

Jianyong Xue, Raphaël Lallement, Matteo Morelli

Sense-making and knowledge construction via constructivist learning paradigm

Jianyong Xue
Université Paris-Saclay
CEA, List
F-91120 Palaiseau, France
jianyong.xue@cea.fr

Raphaël Lallement
Université Paris-Saclay
CEA, List
F-91120 Palaiseau, France
raphael.lallement@cea.fr

Matteo Morelli
Université Paris-Saclay
CEA, List
F-91120 Palaiseau, France
matteo.morelli@cea.fr

***Abstract*—As a knowledge acquisition theory, constructivism describes information processing mechanisms behind infants' cognitive development. When infants play with the world around them, they exhibit amazing abilities to generate novel behaviors in unseen situations and explore actively to learn the best while lacking extrinsic rewards from the environment. These abilities are critical to achieving autonomous intelligent agents (such as robots). In this article, we seek to understand and replicate some of the abilities in infants' play and propose a computational framework based on the constructivist learning paradigm, which enables sense-making and knowledge construction for self-motivated agents. Furthermore, we evaluate the proposed framework for solving the Small Loop Problem (SLP) and compare its performance with reinforcement-based models. A toolkit of Generating and Analyzing Interaction Traces (GAIT) was introduced to report and explain the fine-grained learning process and the formation of structured behaviors after each decision-making. The result shows that the agent has successfully learned to interact with its environment and avoid unfavorable interactions by using regularities discovered through interaction. Moreover, the proposed framework outperforms reinforcement-based models in learning the goals and adapting behaviors in interacting with dynamic environments.**

***Index Terms*—sense-making, knowledge construction, cognitive development, intrinsic motivation, constructivist learning paradigm, sequential learning, self-adaptation**

I. Introduction

Infants are excellent at playing [1]. During the initial phase of cognitive development, they exhibit amazing abilities to explore the environment actively and absorb new knowledge flexibly to generate novel behaviors in being fun. [2]–[4]. These abilities set them apart from most advanced robots. For most artificial agents (such as robots), learning approaches heavily rely on the availability of prior knowledge and specific goals proposed by the system designer, which limits the flexibility of agents in varies of tasks (such as in dynamic scenarios). How can we use observations on infant play to enable a self-motivated agent that can behave in an intelligent and a flexible manner has become a hot topic in research [5], [6].

Over the last decades, a multitude of theories and methods have been devoted to the study of learning mechanisms in infants' early-stage cognitive development, proposed various algorithms targeted at designing and implementing a self-motivated agent as well. For example, developmental learning [7] explores mechanisms that allow continually discover and learn new skills in unseen environments. Intrinsic motivation [8], [9] (such as curiosity, novelty, etc) drives the development of the world-model, as a way to replicate some abilities of infants' interaction. The constructivism as a knowledge acquisition theory proposes that learning happens as result of an internal mental representations and external perceptions from interaction [10], rather than existing in an ontic reality, or available for registration from the physical world [11]. In the view of constructivist learning paradigm, the agent is not designed as as passive observer of reality, but rather constructs as perception of reality through active interaction experience [12].

Inspired from the theory of constructivism, we introduce a computational framework of Constructivist Cognitive Architecture (CCA), as a way towards simulating the early learning mechanism of infants' cognitive development based on theories of enactive cognition, intrinsic motivation, and constructivist epistemology. Meanwhile, the CCA allows a self-motivated agent to autonomously construct the perception of the environment and acquire capabilities of self-adaption and flexibility to generate proper behaviors to tackle with diverse situations in interacting with the environment.

Different to traditional cognitive architectures, the introduced model neither initially endows the agent with prior knowledge of its environment, nor supplies it with knowledge during its learning process. Accordingly, we are not proposing an algorithm that optimizes exploration of a predefined problem-space to reach predefined goal states. Instead, we propose a way for the agent to autonomously encode the interaction experiences and reuse behavioral patterns based on the agent's self-motivation implemented as inborn preference that drive the agent in a proactive way. In addition, we introduce two forms of self-motivation: successfully enacting sequences of interactions (or called autotelic motivation), and preferably enacting interactions that have predefined positive values (or called interactional motivation). Following these drives, the agent autonomously learns regularities afforded by the environment, and constructs hierarchical sequences to perform higher-level behaviors.

The paper is structured as follows. In section II, we introduce related works in traditional artificial intelligence and

current developments in solving the problems we are facing with. In section III, we present the schematic diagram of the CCA and explain the learning process of each parts in it. In section IV, we introduce the methodology and the experimental settings and also introduce the implementation of toolkit GAIT for analyzing all interaction results. Finally, we conclude and provide open issues for possible improvements of our work in the future.

II. Related work

Reinforcement Learning [13] and with its advances [14]–[16] as a most popular way to learning the optimal policies by achieving the maximum cumulative reward of actions. Behavior construction in such systems is not predefined, but learned from interactions with the environment, enabling the partial self-adaptation to situations that were unexpected or unknown at designing time [6], [17]. However, it will become unsuitable in conditions of nonlinearity and nonstationarity of the environments . Besides, it needs further self-adaptation during the interaction in situations where the environment has been changed. Furthermore, the previously learned and stored knowledge could cause interference in the learning of new behavioral patterns [6].

Playful behaviors in the absence of reward are usually described as intrinsic motivations (such as curiosity [1], [7]) that drive the development of the sense-making, as a way to replicate some abilities of infants' playing [1], [18]. Playing capacity in this period likely interacts with infants' powerful abilities to understand and model their environment, which amazingly generates flexible actions with familiar environments and novel behaviors with unfamiliar environments. Driven by intrinsic motivations, [19] adopts an evolutionary perspective and design primary reward functions to measure both emergent intrinsic and extrinsic motivation. [20] stores agent's interactive experience of the environment in an episodic memory, while also spur the robot reaching experiences not yet presented in memory. With intrinsic motivation of curiosity, [9] formalizes curiosity with neural network and a self-model to let the agent challenge the developing world-model , by combing with an error map.

A related but alternative idea comes from the constructivist learning paradigm [2], [6], [21]. In the constructivist paradigm, agent is not a passive observer of reality, but rather constructs a perception of reality through active interaction [4]. The constructivist theory proposes that humans build internal frameworks of knowledge, and acquire new knowledge either through *assimilation* (incorporating new knowledge into their existing framework) or *accommodation* (re-framing internal representations to the newly acquired external knowledge) [4].

Previously, we proposed a causality reconstruction model with constructivist [22] which could let an autonomous agent organize its behavior to fulfill a form of intentionality defined independently of a specific task. With the PetriNet that the agent has learned to predict the consequences of the agent's actions, which explains regularities of interaction through the presence of objects in the agent's surrounding space. In another work, [23] introduces an algorithm for self-motivated hierarchical sequence learning with inspirations from Piaget's theories of early-stage developmental learning. The behavior organization is driven by pre-defined values associated with primitive behavioral patterns. The agent learns increasingly elaborated behaviors through its interactions with its environment. These learned behaviors are gradually organized in a hierarchy that reflects how the agent exploits the hierarchical regularities afforded by the environment.

III. The Constructivist Cognitive Architecture (CCA)

Figure 1 shows the schematic diagram of the CCA, which is mainly composed of the following parts: (a) the stream of enacted interactions timeline, (b) the module of episodic memory, (c) the container of sensorimotor interactions, (d) a hierarchical sequential construction module, (e) the anticipation proposition process, and (f) the behavior selection mechanism. We will introduce each of them in the following sections.

At the bottom of Figure 1, the interaction timeline presents the stream of enacted interactions that occurred overtime. Specifically, enacted interactions are represented by colored symbols in order to indicate different interactions. As the enacted interaction is prepared, the episodic memory records it in the form of hierarchical structure. In the CCA, the elements in the episodic memory are formed by two different kinds of interactions: the primitive interaction and the composite interaction, which are stored in the container of sensorimotor interactions. Specificially, the primitive interaction is a way to represent primitive sensorimotor scheme, the composite interaction is used to represent hierarchical sequential scheme.

With agent's continuously interacting with the environment, new patterns of interaction and higher-level behaviors are gradually constructed, which enhances the hierarchical sequential system on the left top of Figure 1. Hierarchical Sequential System represents the mechanism of learning hierarchical regularities of interactions. It includes the rudimentary learning of regularities of interaction and the recursive learning of composite interactions. The rudimentary learning of regularities of interactions constructs the basic patterns of interaction with previous enacted interaction and current enacted interaction. While the recursive learning of composite interaction gives a bottom-up way to learn higher-level patterns of interaction which are made of lower-level patterns of interactions (or rudimentary composite interactions), which is bottom-up hierarchical sequential learning. Behavior Selection mechanism balances the propositions made by the sequential system and by episodic memory, and then selects the next sequence of interactions to try to enact. For example, an interaction of "moving forward with success" is evoked in episodic memory, then the behavior selection mechanism may select this interaction as the next intended interaction to try to enact.

A. The sensorimotor interaction

In the initialization, the agent is endowed innate experiments and primitive interactions. In our work, the primitive interaction is defined as a tuple of an experiment e_t with its corresponding feedback f_t at time *t*, $i_t = \langle e_t, f_t \rangle$. Additionally, we associate each primitive interaction with a scalar *valence* v_t to qualify the agent's "feeling" from each environmental feedback f_t. Enacting an interaction i_t means that the agent intends an experiment e_t and receives feedback f_t that composes a given interaction at step *t*. The experiment e_t could be a primitive interaction or a series of interactions that the agent is going to enact recursively. Meanwhile, the agent intends an interaction i_t which presents that it performs an experiment e_t while expecting its corresponding feedback f_t at step *t*. With different interactive situations, this "intention" could be that the agent actually enacts interaction $\langle e_t, f_t^{'} \rangle$ if it receives feedback $f_t^{'}$ instead of f_t. If the enacted interaction equals the intended interaction, then the attempted enaction of intended interaction is considered a *success*, otherwise *failure*.

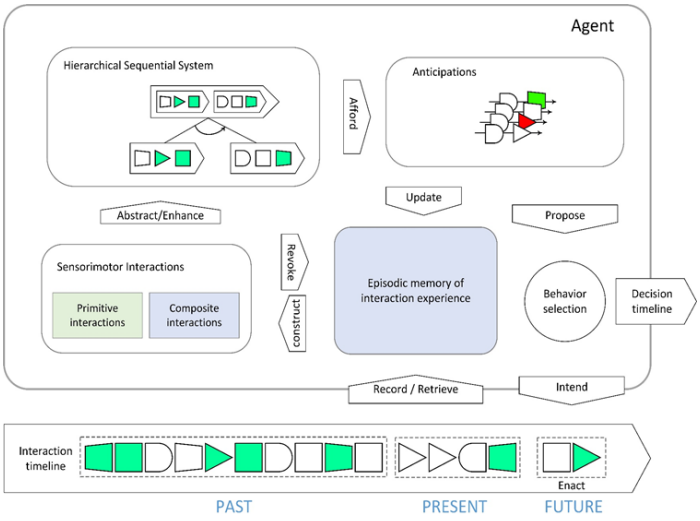


Fig. 1. The schematic diagram of the Constructivist Cognitive Architecture.

At the interaction cycle *t*, agent selects an intended interaction $i_t^i = \langle e_t, f_t \rangle$ and tries to enact with reference to the reactive part of the environment. As a result, the agent receives the enacted interaction i_t^e and memorizes the two-step enacted interactions formed as a sequence $c_t = \langle i_{t-1}^e, i_t^e \rangle$, which is made by the previous and current enacted interactions. The sequence of interaction $\langle i_{t-1}^e, i_t^e \rangle$ is called a *composite interaction* c_t, as the pattern of structured behaviors corresponds to the *assimilation* process in constructivism. The interaction i_{t-1}^e is called c_t's pre-interaction, noted as $pre(\langle i_{t-1}^e, i_t^e \rangle)$, and i_t^e is called c_t's post-interaction and is noted as $post(\langle i_{t-1}^e, i_t^e \rangle)$. The tuple of composite interaction expresses that in the context of i_{t-1}^e, the post interaction i_t^e will be activated to enact in the future. As interaction continues, more complex composite interactions will be emerged (or reinforced) according to the combinations of different kinds of primitive interactions. To better reflect the proximity of pre-interaction and post-interaction in composite interactions, we associate each composite interaction with a *weight* (initialized as "1") and it will be incremented when the same composite interaction has learned again (coincide with the *accommodation* process in constructivism). Moreover, composite interaction's *valence* is the sum of its pre-interaction's and post-interaction's *valence*.

B. Activation process

The set of composite interactions known by the agent at time *t* is defined as C_t and the set $J_t = I \cup C_t$ is the all interactions known to the agent at time *t*. For the next step interaction, the current enacted interaction i_t^e as the context B_t activates previously learned composite interactions as it matches their pre-interaction, then the agent gets the *activated composited interaction* set $A_t = \{a_i \in C_t | pre(a_i) \subset B_t\}$.

Activated interactions A_t propose their post-interaction as anticipations and the agent's decision from these anticipations for the next round. For each post-interactions, *anticipation* is created with a scalar value *proclivity* $p_i \in \Re$ which is computed from the weight w_{a_i} of the activated interaction a_i multiplied by the valence of the proposed post-interaction $v(post(a_i))$, then forms the set AN_t of proposed anticipations: $AN_t = \{anti_i \in AN_t | a_i \in A_t, proclivity_{anti_i} = w_{a_i} \times v(post(a_i))\}$. The proclivity value reflects the regularity of the interaction based on its probability of occurrence and the agent's motivations.

With partial anticipations starting with the same experiments, the *partial similar anticipations (PSAs)*, we regroup all anticipations according to their first experiment of the intended interactions and map them all as different lists with corresponding experiments respectively. Each experiment's proclivity value is calculated as follows:

$$proclivity_{anti^i_{default}} = \sum_{i=1}^{n} w_{a_i} \times v(post(a_i)) \qquad (1)$$

$proclivity_{anti^i_{default}}$ is the proclivity of experiment *i*, *n* refers to the number of anticipations that share the same first-experiment of primitive interaction in their intended interaction, w_{a_i} is the weight of activated composite interactions a_i and the $v(post(a_i))$ is the valence of a_i's post-interaction.

C. Behavior selection and interaction enaction

The intended interaction i_t^i is selected from anticipations whose experiment has the biggest proclivity and enacts the intended interaction of its anticipation with the biggest proclivity. If the intended interaction is composite, the enaction of this intended composite interaction refers to its anticipation's weight w_{a_i} and the threshold $d \in \Re$. The parameter *d* is the threshold which presents the belief of enacting the intended interaction as a whole. Suppose the weight of the anticipation is greater than d and its proclivity value is positive. In that case, the agent will effectively enact all primitive interactions within this intended interaction according to the hierarchical sequential structure recursively. Otherwise, the agent enacts the first primitive interaction of this intended interaction. In

essence, this mechanism ensures that higher-level schemas are sufficiently rehearsed before being enacted as a whole. If the sequence of the intended interactions corresponds to the regularity of interaction, then it is possible that the sequence of this intended interaction can be enacted again. Therefore, the agent can anticipate that performing post interaction's experience will likely produce its result. The agent can thus base its choice of the next interaction on this anticipation.

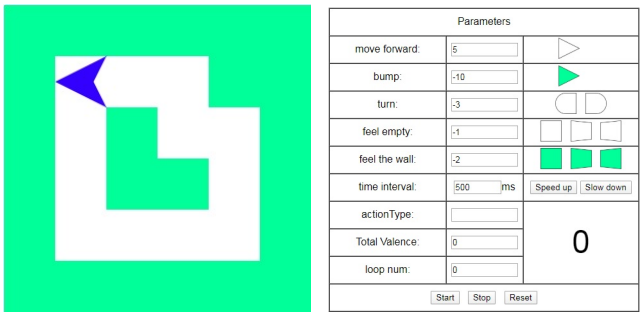
While enacting intended interaction, the agent checks each enacted interaction with intended interaction and compares the result between them. For enacting composite interactions, the flat sequence of enacted interactions constructs a hierarchical structure according to the enaction sequence and the intended composite interaction's structure. With enacted interactions, new composite interactions are constructed or reinforced with their pre-interaction belonging to the context and their post-interaction i_t^e, forming the set of learned or reinforced interaction c_t to be included in C_{t+1}, for supporting affordance of more complicated interaction situations in the future. The set c_t is defined as $c_t = \{\langle i_{t-1}^e, i_t^e\rangle, \langle i_{t-2}^e, \langle i_{t-1}^e, i_t^e\rangle\rangle, \langle\langle i_{t-2}^e, i_{t-1}^e\rangle, i_t^e\rangle\}$, $C_{t+1} = C_t \cup c_t$. A new context B_{t+1} is constructed to include the stabilized interactions in i_t^e and $post(i_t^e)$.

IV. Methodology and Experimental Scenario

To evaluate the proposed model, we set up an experimental scenario in which the agent could move around and touch the environment in three directions: front, left and right. The environment is designed as a Small Loop Problem (SLP) [24], [25] environment, which composed of white squares present paths surrounded by green "walls" (as shown in the left figure of Figure 2). The agent is presented as a blue arrow and initialized with a random direction. Meanwhile, the following shapes of triangle, left and right half-circle square, left and right trapezoid and square respectively represent experiments of moving forward, turn left and turn right, touch left, touch right and touch front. Colors of interactions in white and green indicate possible feedbacks of moving forward successfully or bumping with wall from interacting with the environment.

The SLP as a benchmark to evaluate agents that implement four principles of emergent cognition: environment agnosticism, self-motivation, sequential regularity learning, and spatial regularity learning [26]. Different from most existing benchmarks, the small loop environment does not involve a final goal for the agent to reach, instead, the agent's self-motivation comes from the fact that primitive interactions have different valence. Additionally, our environment can be dynamically modified to verify the adaptability of agent in dynamic environments.

In the interface panel, the experimenter can preset the parameters to control the interaction process (as shown in the right figure of Figure 2), such as valences allocation for primitive interactions, interaction "interval" for speeding up or slowing down the interaction process, "actionType" indicates the current experiment the agent enacts, "Total valence" represents the accumulated valence, and "loopNum" presents the



(a) The environment for interaction. (b) Parameter settings for interacting.

Fig. 2. The Small Loop Problem environment and experimental settings.

decision-making steps. To better investigate the fine-grained learning process and the formation of structured behaviors after each decision-making, we developed a toolkit of "Generating and Analyzing Interaction Traces toolkit" (GAIT) as introduced in the next section.

A. Generating and Analyzing Interaction Traces (GAIT)

The toolkit of GAIT records all the details of each decision-making and its execution results, including all anticipations activated by current enacted interaction, all candidate interaction in each anticipation sorted by their proclivity value, and the enacted interaction of the selected proposed interaction (the intended interaction). In addition, the interactive interface enables to pop out tip windows to display these details selectively. For example, in the area of interaction traces, the mouse move over each enacted interaction will pop up the tip window presenting all composite interactions. The highlighted green rectangles indicate that the composite interactions are newly learned, while the blue ones are already learned before but reinforced in this interaction. Left-clicking on any enacted interactions, the sorted experiment list will pop out with its proclivities. For the case that experiments have proposed anticipations, we highlight them with pink rectangles. Selecting the experiment will display more details on its anticipations and sorted by their own proclivities. As for enacting composite interactions, the light green rectangle fields on the "loop number" can pop out a tip window including the intended composite interactions with its enacted composite interaction to show the detailed interaction process. In order to identify the range of interactions in this composite interaction, we circle all primitive interactions the agent has enacted with a yellow rectangle.

The scroll button at the bottom of the interaction traces can be used to show all previous interactions. In our experiment, the proposed toolkit can support tens of thousands of interactions which makes it easy to retrospect all previous interactions.

B. Interaction traces analysis

As shown in Figure 3, agent starts to enact a default intended interaction (the green right trapezoid) and it receives the same enacted interaction. In step 2, a composite interaction was

formed according to the previous enacted interaction and the current enacted interaction. Particularly, in step 3, the agent not only memorizes the previously enacted interaction and also combines the previously learned composite interaction to construct more higher-level composite interaction.

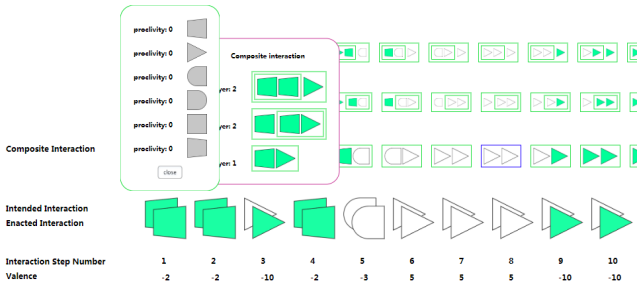


Fig. 3. The first several interactions and composite interactions construct process.

The proposed intended interaction appears at step 9 (as shown in Figure 4), the experiment "move forward" has the highest proclivity and its intended interaction (the white triangle) is activated with the highest proclivity (the proclivity value is 15). Then the agent intends this white triangle (move forward) and gets the same white triangle (the agent successfully moves forward a step), then this enacting intended interaction is a *success*. When the opposite situation happens in step 14, the agent intends the same white triangle but bumps with the wall (a green triangle), thus this interaction is a *failure*.

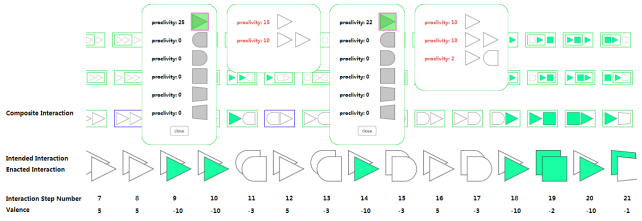


Fig. 4. Enact the same intended interaction with different feedback.

At step 23, the agent is going to enact a composite interaction, with the reason that this anticipation's weight is less than the threshold, then the agent intends the first primitive interaction (left half-circle) and receives the same enacted interaction (as shown in Figure 5). In our implementation, we use different colors of proclivities to identify their anticipations' weight beyond the threshold or not, the red color means the weight is less than the threshold while the green color presents the weight is higher than the threshold.

At step 119, the agent gets an intended composite interaction with the weight beyond the threshold, then it sequentially intends this composite interaction successfully receives the same enacted composite interaction (as shown in Figure 6). The agent combines this enacted composited interaction with previously enacted interaction and learned composite interaction to construct higher-level and more complex composite interaction.

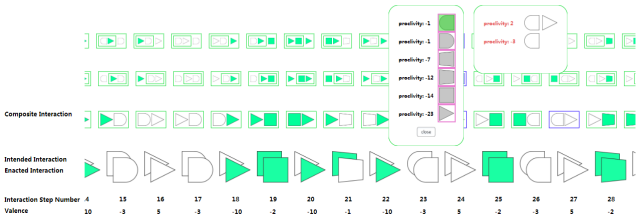


Fig. 5. The enacted composite interaction's weight less than the threshold.

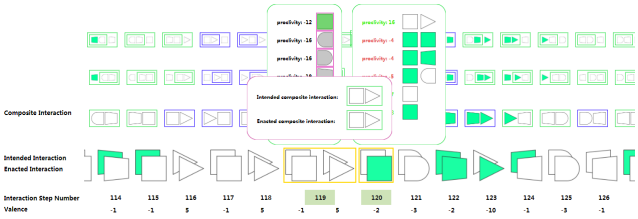


Fig. 6. The agent enacts composite interaction and constructs higher-level composite interaction.

AS interactions continue, the agent successfully interacts with the environment and learns to avoid unfavorable interactions (bumping with the wall) by using sequences that it has learned. More complicated behavioral sequences have been constructed and they can handle appropriately with different situations (as shown in Figure 7).

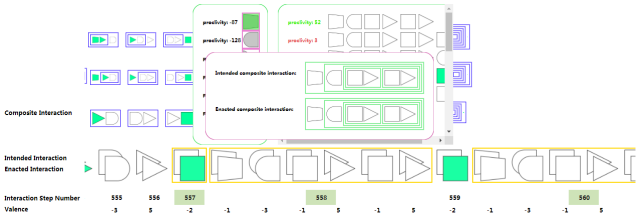


Fig. 7. Enacting complex composite interaction.

C. Performance comparison

We compare the performance of CCA with reinforcement-based method of Q-learning. Figure 8 indicates that both algorithms converge as the interaction proceeds, except that the CCA converges faster. For example, in the top sub-figure, the bumping phenomenon in CCA stops from 357th step, and in Q-Learning from the 1246th step, respectively. Meanwhile, from the perspective of accumulated valences, we find the valence begins to rise as the agent could successfully interact with the environment, and the increase goes gradually faster until stable, which is evidence proving the emergence of sense-making and cognitive development in the agent's interactions with the environment. Accordingly, the CCA gains more

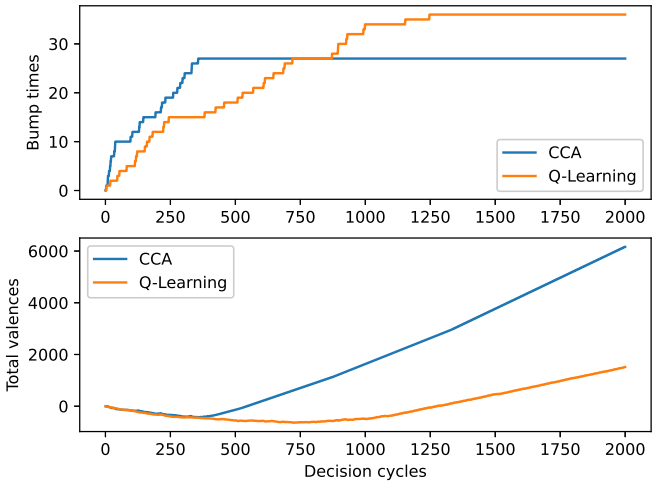


Fig. 8. Performance comparison between CCA and Q-Learning.

valences than the Q-Learning (as shown in the bottom Figure IV-C).

From step 2000, we change the environment to investigate the scalability of two algorithms in adapting the new environment and the flexibility to generate appropriate behaviors in it. As shown in Figure 9, both algorithms adapt to the changes, as well as converge as interaction proceeds. Particularly, the CCA outperforms Q-Learning by a faster discovery of the changes and converges faster than Q-Learning. This advantage can be seen more clearly in the accumulated valence sub-figure, where CCA obtains more valence.

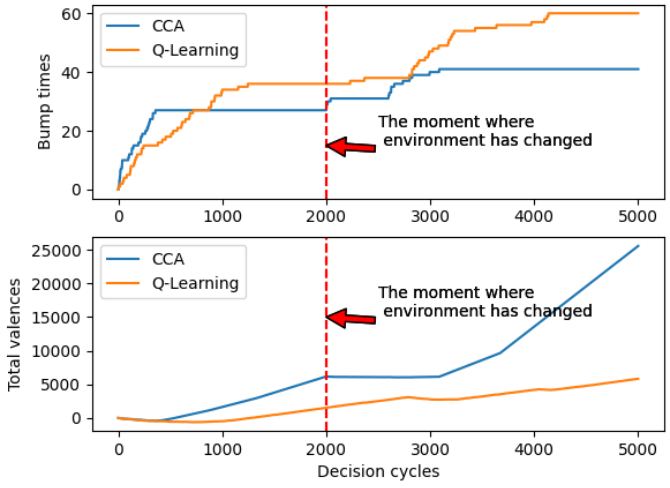


Fig. 9. Performance comparison between CCA and Q-Learning in dynamic environment.

V. Conclusion and Open Issues

This article introduces a computational framework of constructivist cognitive architecture (CCA), as a way of simulating the early learning mechanism of infants' cognitive development and enabling the sense-making and knowledge construction for self-motivated agents. The agent autonomously organizes schemas it learned from interaction into hierarchically structured behaviors, which lets the agent gradually understand the meaning of divers experiments and infer the structure of the environment simultaneously based on the patterns in the stream of interactions feedback traces. Meanwhile, an implementation of GAIT for autonomously generating and analyzing interaction at run-time, which could let us observe the detailed learning process for agent interacting with the environment and each structured behaviors it has learned within each decision-making.

We evaluated the agent's cognition emergence with an improved Small Loop Problem (SLP) environment, in which the changeable environment is designed for simulating agent's performance in different levels of complex scenarios. With interaction traces from GAIT and hierarchically structured behaviors the agent has learned, the agent gradually exploits the hierarchical regularities afforded by the environment and learn to avoid unfavorable interactions using regularities that it has learned.

Nevertheless, the agent has to retrospect all composite interactions to retrieve the one whose pre-interaction matches the current enacted interaction. Being performed interactions, their traces grow progressively longer, hence the agent spends a long time to activate all eligible composite interactions for anticipations. In addition, the utility rate of composite interactions needs to be improved. As an example, the agent activated almost all composite interactions but few are proposed for intending. Although the agent can be easily qualified for the work in the environment designed in this paper, when the environment becomes more complex, this shortcoming becomes more obvious.

Valence initialization is another issue that we need to face. According to the common sense of human beings, assuming the agent gets positive hints if it can successfully take a step forward and the negative feedback for collisions with the wall. As for the agent, it starts interaction without any prior knowledge, which means it should comprehend the feedback of different behaviors from its own interaction with the environment. For the initialization of Valence, it is inevitable to have a certain influence on the cognitive process of the agent to some extent. The setting of the threshold limits the agent performing the proposed interactions, in this paper, this parameter was set manually, and it should have a prediction mechanism that dynamically gates the enaction of the proposed interaction. In the following research, we need to study how to reduce human intervention as much as possible, let the agent explore for itself, and discover how to find the optimal valence allocation strategy from the interaction.

Further work will be mainly focused on optimizing our

model and upgrading our toolkit. For example, we could use a predictive model for better-proposing anticipations for the agent of interacting with the environment in the next round. With memorizing patterns that could improve the learning efficiency and eliminate composite interactions that probably will not use to simplify the activation and proposition processes in the CCA in the future. Also, for selecting proposed interactions to enact, we could flatten composite interactions into sequences of primitive interactions without taking care of its structure. With this hierarchical sequential learning model, we'd like to evaluate the performance of the agent in a multi-agent scenario, which provides more challenges and opportunities to improve its learning ability.

REFERENCES

[1] N. Haber, D. Mrowca, L. Fei-Fei, and D. L. Yamins, "Emergence of structured behaviors from curiosity-based intrinsic motivation," *arXiv preprint arXiv:1802.07461*, 2018.

[2] M. Guériau, F. Armetta, S. Hassas, R. Billot, and N.-E. El Faouzi, "A constructivist approach for a self-adaptive decision-making system: application to road traffic control," in *2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, 2016, pp. 670–677.

[3] Y. Bu, J. Lu, and V. V. Veeravalli, "Active and adaptive sequential learning," *arXiv preprint arXiv:1805.11710*, 2018.

[4] O. L. Georgeon and F. E. Ritter, "An intrinsically-motivated schema mechanism to model and simulate emergent cognition," *Cognitive Systems Research*, vol. 15, pp. 73–92, 2012.

[5] G. Anthes, "Lifelong learning in artificial neural networks," *Communications of the ACM*, vol. 62, no. 6, pp. 13–15, 2019.

[6] M. Guériau, N. Cardozo, and I. Dusparic, "Constructivist approach to state space adaptation in reinforcement learning," *Learning*, vol. 4, no. S3, p. S2, 2019.

[7] P.-Y. Oudeyer, F. Kaplan, and V. V. Hafner, "Intrinsic motivation systems for autonomous mental development," *IEEE transactions on evolutionary computation*, vol. 11, no. 2, pp. 265–286, 2007.

[8] A. Baranes and P.-Y. Oudeyer, "Active learning of inverse models with intrinsically motivated goal exploration in robots," *Robotics and Autonomous Systems*, vol. 61, no. 1, pp. 49–73, 2013.

[9] N. Haber, D. Mrowca, S. Wang, L. F. Fei-Fei, and D. L. Yamins, "Learning to play with intrinsically-motivated, self-aware agents," *Advances in neural information processing systems*, vol. 31, 2018.

[10] F. Guerin, "Constructivism in ai: Prospects, progress and challenges." in *AISB Convention*, 2008, pp. 20–27.

[11] E. B. Roesch, M. Spencer, S. J. Nasuto, T. Tanay, and J. M. Bishop, "Exploration of the functional properties of interaction: computer models and pointers for theory," *Constructivist Foundations*, vol. 9, no. 1, pp. 26–33, 2013.

[12] O. L. Georgeonab and F. E. Ritterb, "An intrinsically-motivated schema mechanism to," *To appear in Cognitive Systems Research (Accepted July 2011)*, 2011.

[13] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[14] T. D. Kulkarni, K. Narasimhan, A. Saeedi, and J. Tenenbaum, "Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation," in *Advances in neural information processing systems*, 2016, pp. 3675–3683.

[15] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[16] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, p. 1.

[17] M. Riedmiller, R. Hafner, T. Lampe, M. Neunert, J. Degrave, T. Van de Wiele, V. Mnih, N. Heess, and J. T. Springenberg, "Learning by playing-solving sparse reward tasks from scratch," *arXiv preprint arXiv:1802.10567*, 2018.

[18] A. E. Stahl and L. Feigenson, "Observing the unexpected enhances infants' learning and exploration," *Science*, vol. 348, no. 6230, pp. 91–94, 2015.

[19] S. Singh, R. L. Lewis, A. G. Barto, and J. Sorg, "Intrinsically motivated reinforcement learning: An evolutionary perspective," *IEEE Transactions on Autonomous Mental Development*, vol. 2, no. 2, pp. 70–82, 2010.

[20] N. Savinov, A. Raichuk, R. Marinier, D. Vincent, M. Pollefeys, T. Lillicrap, and S. Gelly, "Episodic curiosity through reachability," *arXiv preprint arXiv:1810.02274*, 2018.

[21] J. Piaget, *The construction of reality in the child*. Routledge, 2013, vol. 82.

[22] J. Xue, O. L. Georgeon, and M. Gillermin, "Causality reconstruction by an autonomous agent," in *Biologically Inspired Cognitive Architectures Meeting*. Springer, 2018, pp. 347–354.

[23] O. L. Georgeon, J. H. Morgan, and F. E. Ritter, "An algorithm for self-motivated hierarchical sequence learning," in *Proceedings of the International Conference on Cognitive Modeling. Philadelphia, PA. ICCM-164*. Citeseer, 2010, pp. 73–78.

[24] O. L. Georgeon, C. Wolf, and S. Gay, "An enactive approach to autonomous agent and robot learning," in *2013 IEEE Third Joint International Conference on Development and Learning and Epigenetic Robotics (ICDL)*. IEEE, 2013, pp. 1–6.

[25] O. L. Georgeon and J. B. Marshall, "Demonstrating sensemaking emergence in artificial agents: A method and an example," *International Journal of Machine Consciousness*, vol. 5, no. 02, pp. 131–144, 2013.

[26] ——, "The small loop problem: A challenge for artificial emergent cognition," in *Biologically Inspired Cognitive Architectures 2012: Proceedings of the Third Annual Meeting of the BICA Society*. Springer, 2013, pp. 137–144.