

An Event-Schematic, Cooperative, Cognitive Architecture Plays Super Mario

Fabian Schrodtt

Department of Computer Science
Eberhard Karls University of Tübingen
tobias-fabian.schrodtt@uni-tuebingen.de

Yves Röhm

Department of Computer Science
Eberhard Karls University of Tübingen
yves.roehm@student.uni-tuebingen.de

Martin V. Butz

Department of Computer Science
Eberhard Karls University of Tübingen
martin.butz@uni-tuebingen.de

Abstract—We apply the cognitive architecture SEMLINCS to model multi-agent cooperations in a Super Mario game environment. SEMLINCS is a predictive, self-motivated control architecture that learns conceptual, event-oriented schema rules. We show how the developing, general schema rules yield cooperative behavior, taking into account individual beliefs and environmental context. The implemented agents are able to recognize other agents as individual actors, learning about their respective abilities from observation, and considering them in their plans. As a consequence, they are able to simulate changes in their context-dependent scope of action with respect to their own interactions with the environment, interactions of other agents with the environment, as well as interactions between agents, yielding coordinated multi-agent plans. The plans are communicated between the agents and establish a common ground to initiate cooperation. In sum, our results show how cooperative behavior can be planned and coordinated, developing from sensorimotor experience and predictive, event-based structures.

I. INTRODUCTION

Most of the approaches on intelligent, autonomous game agents are robust, but behavior is typically scripted, predictable, and hardly flexible. Current game agents are still rather limited in their speech and learning capabilities as well as in the way they act believably in a self-motivated manner. While novel artificial intelligent agents have been developed over the past decades, the level of intelligence, the interaction capabilities, and the behavioral versatility of these agents are still far from optimal [1], [2].

Besides the lack of truly intelligent game agents, however, the main motivation for this work comes from cognitive science and artificial intelligence. Over the past two decades, two major trends have established themselves in cognitive science. First, *cognition is embodied*, or grounded, in the sensory-, motor-, and body-mediated experiences that humans and other adaptive animals gather in their environment [3]. Second, *brains are predictive encoding systems*, which have evolved to be able to anticipate incoming sensory information, thus learning predominantly from the differences between predicted and actual sensory information [4]–[7]. Combined with the principle of free-energy-based inference, neural learning, as well as active epistemic and motivation-driven inference, a unified brain principle has been proposed [8], [9]. Concurrently, it has been emphasized that event signals may be processed in a unique manner by our brains. The event segmentation theory [10], [11] suggests that humans learn to segment the

continuous sensorimotor stream into *event codes*, which are also closely related to the common coding framework and the theory of event coding [12], [13]. Already in [10] it was proposed that such event codes are very well-suited to be integrated into event schema-based rules, which are closely related to production rules [14] and rules generated by anticipatory behavior control mechanisms [15]. As acknowledged from a cognitive robotics perspective, event-based knowledge structures are as well eligible to be embedded into a *linguistic, grammatical system* [16]–[18].

We apply the principles of predictive coding and active inference and integrate them into a highly modularized, cognitive system architecture. We call the architecture SEMLINCS, which is a loose acronym for SEMantic, SENSory-Motor, SELF-Motivated, Learning, INTelligent Cognitive System [19]. The architecture is motivated by a recent proposition towards a unified subsymbolic computational theory of cognition [20], which puts forward how production rule-like systems (such as SOAR or ACT-R) may be grounded in sensorimotor experiences by means of predictive encodings and free energy-based inference. The theory also emphasizes how active-inference-based, goal-directed behavior may yield a fully autonomous, self-motivated, goal-oriented behavioral system and how conceptual predictive structures may be learned by focusing generalization and segmentation mechanisms on the detection of events and event transitions.

SEMLINCS is essentially a predictive control architecture that learns event schema rules and interacts with its world in a self-motivated, goal- and information-driven manner. It specifies a continuously unfolding cognitive control process that incorporates (i) a self-motivated behavioral system, (ii) event-oriented learning of probabilistic event schema rules, (iii) hierarchical, goal-oriented, probabilistic reasoning, planning, and decision making, (iv) speech comprehension and generation mechanisms, and (v) interactions thereof.

Here, our focus lies on studying artificial, cognitive game agents. Consequently, we offer an implementation of SEMLINCS to control game agents in a Super Mario game environment¹²³. Seeing that the game is in fact rather complex,

¹<https://www.youtube.com/watch?v=AplG6KnOr2Q>

²<https://www.youtube.com/watch?v=ltPj3RIN4Nw>

³https://www.youtube.com/watch?v=GzDt1t_iMU8

the implementation of SEMLINCS faces a diverse collection of tasks. The implemented cognitive game agents are capable of completing Super Mario levels autonomously or cooperatively, solving a variety of deductive problems and interaction tasks. Our implementation focuses on learning and applying schematic rules that enable artificial agents to cause behaviorally relevant intrinsic and extrinsic effects, such as collecting, creating, or destroying objects in the simulated world, carrying other agents, or changing an agent’s internal state, such as the health level. Signals of persistent surprise in these domains can be registered [21], which results in the issuance of event schema learning [20], and which is closely related to the refference principle [22]. As a result, production-rule-like, sensorimotor-grounded event schemas develop from signals of surprise and form predictive models that can be applied for planning. SEMLINCS thus offers a next step towards complete cognitive systems, which include learning techniques and which build a hierarchical, conceptualized model of their environment in order to interact with it in a self-motivated, self-maintenance-oriented manner.

A significant aspect when considering multi-agent architectures inspired by human cognition is cooperation and communication: Unique aspects of human cognition are characterized by social skills like empathy, understanding the perspective of others, building common ground by communication, and engaging in joint activities [23]. As a step towards these abilities, we show that the developing event-oriented, schematic knowledge structures enable the implemented SEMLINCS agents to cooperatively achieve joint goals. Thus, our implementation shows how sensorimotor grounded event codes can enable and thus bootstrap cooperative interactions between artificial agents. SEMLINCS is designed such that the developing knowledge structures and the motivational system can be coupled with a natural language processing component. In our implementation, agents are able to learn from voice inputs of an instructor, follow instructed goals and motivations, and communicate their gathered plans and beliefs to the instructor. Moreover, they can propose to and discuss with other game agents potential joint action plans.

In the following, we provide a general overview of the modular structure of SEMLINCS in application to the Super Mario game environment. Moreover, we outline key aspects for coordinated cooperation in our implementation. We evaluate the system in selected multi-agent deduction tasks, focusing on learning, semantic grounding, and conceptual reasoning with respect to agent-individual abilities, beliefs, and environmental context. The final discussion puts forward the insights gained from our modeling effort, highlights important design choices, as well as current limitations and possible system enhancements.

II. SEMLINCS IN APPLICATION TO SUPER MARIO

Here we give a brief overview of the main characteristics of SEMLINCS in application to the Super Mario game environment. A detailed description is available in [19]. The implementation consists of five interacting modules as seen

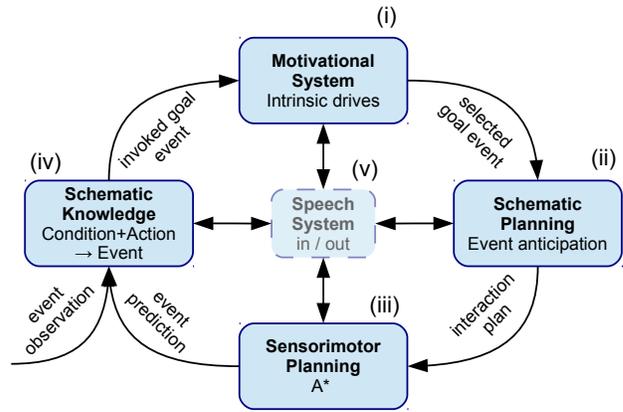


Fig. 1. Overview of the main modules and the cognitive control loop in the implementation of SEMLINCS.

in Figure 1. The motivational system (i) specifies drives that activate goal-effects that are believed to bring the system towards homeostasis. The drives comprise an urge to collect coins, make progress in the level, interact with novel objects, and maintain a specific health level. Goal-effects selected by the motivational system are then processed by an event-anticipatory schematic planning module (ii) that infers a sequence of abstract, environmental interactions that are believed to cause the effects in the current context. The interaction sequence is then planned in terms of actual motor commands by the sensorimotor planning module (iii), which infers a sequence of keystrokes that will result in the desired interactions. Both, the schematic and sensorimotor forward models used for planning are also used to generate forward simulations of the *currently* expected behavioral consequences. These forwards simulations are continuously compared with the actual observations by the event-schematic knowledge and learning module (iv), where significant differences are registered as event transitions that cause the formation of procedural, context-dependent, event-schematic rules. The principle is closely related to Jeffrey Zacks and Barbara Tversky’s event segmentation theory [10], [11] and the refference principle [22]. After a desired goal effect was achieved, the respective drive that caused the goal is lowered, and a new goal is selected, completing an action cycle. The speech system (v) provides a natural user interface to all of these processes, and additionally enables verbal communication between agents. In the following, we focus on the steps most relevant for our implementation of coordinated joint actions: Event-schematic knowledge and planning.

A. Event-Schematic Knowledge and Planning

An *event* can be defined as a certain type of interaction that ends with the completion of that interaction. An *event boundary* marks the end of such an event by co-encoding the encountered extrinsic and intrinsic changes or *effects*. Since the possible interactions with the environment are context-dependent in nature, we describe an *event-schematic rule* as a conditional,

probabilistic mapping from interactions to encountered event boundaries. Production-rule like schemas can be learned by means of Bayesian statistics under assumptions that apply in the Mario environment: Object interactions immediately result in specific effects, such that temporal dependencies can be neglected. Furthermore, the effects always occur locally, such that spatial relations can be neglected. Thus, in the Mario world, interactions can be restricted to *directional collisions*, which may result in particular, immediate *effects*, given a specific, local *context*.

In the SEMLINCS implementation, event boundary detection is implemented by detecting significant sensory changes that the agent does not predict by means of its sensorimotor forward model. Amongst others, these include changes in an agents' health level or the number of collected coins, the destruction or creation of an object, or the action of lifting or dropping an object or another agent.

The *context* for the applicability of a schematic rule, however, is determined by different factors: It includes a *procedural* precondition for an interaction, which specifies in our current implementation the identity of actor and target as well as the intrinsic state of the actor (i.e. its health level). On the other hand, an *environmental context* precondition limits the applicable rules to the current *scope of an action*. That is, the target of a schema rule must be available and the interaction with the target must be expected to lead to the desired effect given the current situation. While the compliance with procedural constraints can be determined easily, the reachability of objects has to be ascertained by an intelligent heuristic, which we describe in the following.

B. Simulating the Scope of Action

The scope of action in a simulated scene is determined by a recursive search based on sensorimotor forward simulations. The search starts at the observed scene or environmental context and then simulates a number of simplified movement primitives in parallel. Each of the simulations results in a number of collisions (or interactions), as well as a new, simulated scene. Sufficiently different scenes are then expanded in the same manner, until the scope of action is sufficiently explored. As a result, it encompasses the reachable positions as well as attainable interactions in a local context as provided by the sensorimotor forward simulation, neglecting, however, the effects that may result from the interactions.

The simulation of changes in the scope of action is accomplished using the abstract, schematic forward simulation of the local environment. In the current implementation, the schematic forward model is applied by a stochastic, effect probability based Dijkstra search. In contrast to the sensorimotor forward model, it neglects the actual motor commands but integrates the estimated, attainable interactions in the local context as provided by the recursive, sensorimotor search. When specific interactions relevant to the scope of action are simulated (for example the destruction of a block) the scope of action is updated.



Fig. 2. Expansion of the scope of action by simulating environmental interactions. Red fields mark the reachable positions, while blue arrows denote the registered object interaction options, while simulating the scope of action. Top row: The scope of action is updated by simulating the destruction of an object. Bottom row: The scope of action is updated by simulating the interaction with another agent.

In the first example shown in Figure 2, an agent aims at collecting a specific item (the coin on the top right). However, this item is blocked by destructible objects (the golden boxes to the right of the agent). Assume that the agent has already learned that it can destroy and collect the respective objects. In the initial situation (top left picture), however, the learned rule about how to collect the coin is not applicable. The schematic planning module thus first simulates the destruction of one of the blocking objects, and then updates the simulated scope of action. When there is more than one destructible object in the current scene, it furthermore has to identify the correct object for destruction, that is, degeneralize the schematic rule with respect to the context (in the example, both objects are suitable). Next, the agent realizes that the desired item can be collected, given that one of the blocks was destroyed, resulting in a schematic action plan.

C. From Schematic Planning to Coordinated Cooperation

Schema structures gathered from sensorimotor experiences can be embedded into hierarchical, context-based planning. Human cognition, however, is highly interactive and social. To enable our architecture to act in multi-agent scenarios, it has to (i) recognize other agents as individual actors (ii) observe and learn about their actions and abilities, (iii) consider them as actors in own plans (iv) consider them as possible interaction targets, and (v) communicate emerging plans. Since agents may have different knowledge and scopes of action, this can already result in simple cooperative behavior, for example, if the destruction of a specific block is needed but in the scope of action of another agent only.

To yield a greater variety of cooperative scenarios, we additionally equip the agents with individual abilities. Specifically, agents are equipped with different jumping heights or the

unique ability to destroy specific blocks. As shown in Figure 2, the agents may then expand their scope of action when considering interactions with other agents during schematic planning. As a consequence, depending on the situation, agents may be committed to include other agents into their plans, as will be shown in the experiments.

While these principles are sufficient to model cooperative planning, additional mechanisms are needed to account for the coordination and communication of plans. In our implementation, all schematic plans are strictly sequential, meaning that only one interaction by one agent is targeted at a time, eliminating the need for a time-dependent execution of plans. The communication of plans is done via the speech system by communicating (grammatical tags corresponding to) the planned, abstract, schematic interaction sequences from the planning agent to possibly involved agents. Neither the concrete, contextualized interaction sequence, nor corresponding sensorimotor plans are communicated. As a consequence, the addressed agent has to infer the concrete instances of targeted objects that the planning agent is talking about. To do so, the agent performs *contextual replanning* to comprehend the proposed plan using his own knowledge – essentially mentally reenacting it. Given that the involved agent has learned a different set of knowledge than the planning agent, it is likely to end up with a different plan and a different overall probability of success. In our current implementation, an involved agent accepts a proposed plan when it does not have another solution for the targeted goal that is more likely successful than the proposed plan given its knowledge. Given the involved agent gets to a different plan, it makes a counterproposal that is always accepted by the initial planning agent. The process of negotiation is shown in Figure 3.

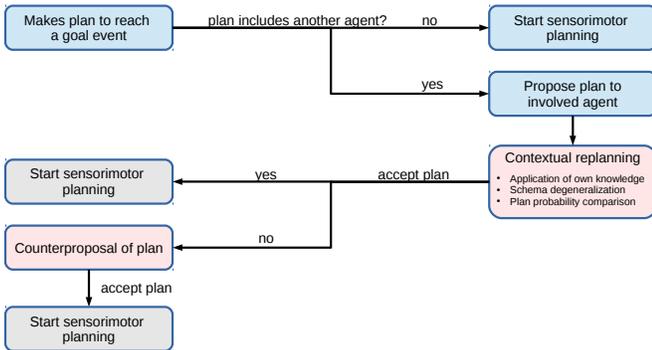


Fig. 3. Negotiation diagram for two agents. Blue boxes: Tasks of the planning agent. Red boxes: Tasks of an agent involved in the initial plan. Grey boxes: Both agents are planning.

III. EVALUATION

We evaluated the resulting cooperative capabilities of SEM-LINCS by creating exemplar scenarios in the Super Mario world, which illustrate the cooperative abilities of the agents. We show two particular, illustrative evaluations. However, we have evaluated SEM-LINCS in various, similar scenarios and have observed the unfolding of similarly well-coordinated

behavior. Videos showcasing these scenarios are available online⁴⁵. An additional scenario showing the negotiation process is also available, but it is not included in this paper because it is not the main focus here⁶.

A. Toad Transports Mario

The first scenario is shown in Figure 5. In the initial scene (top left picture), the agent ‘Mario’ stands on the left, below an object named ‘simple block’ while the agent ‘Toad’ stands close to Mario to the right side. Neither Mario nor Toad have gathered schematic knowledge about their environment so far. Mario is instructed to jump and learns that if he is in his ‘large’ health state and collides with a simple block from the bottom, the block will be destroyed. Next, he is ordered to jump to the right – essentially onto the top of Toad – resulting in Toad carrying Mario and the learning of the option to ‘mount’ Toad and thus be carried around. As Mario is instructed to jump to the right again, he also learns how to dismount Toad. Figure 4 shows a graph of Mario’s schematic knowledge at this point.

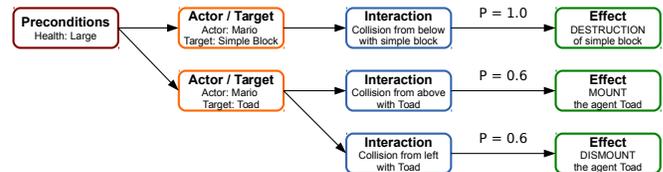


Fig. 4. Mario’s schematic knowledge in scenario 1. The respective entries are put into context by the schematic planning module.

Equipped with this knowledge, Mario is ordered by voice input to ‘destroy a simple block’. This sets as goal effect the destruction of a simple block object which activates planning in the schematic knowledge space. As can be seen in Figure 5, the only simple block is located at the top right in the current context. In this implemented scenario, Toad is able to jump higher than Mario, such that he can jump to the elevation, while Mario is not able to do so. Thus, a direct interaction with the simple block is not possible for Mario as it is not in Mario’s current scope of action.

The schematic planning is thus forced to consider other previously experienced interactions in the context of the current situation. We assume that all agents have full knowledge about the sensorimotor abilities of the others. Thus, inferring that it will expand his scope of action, Mario simulates to jump on the back of Toad, followed by Toad transporting Mario to the elevated location on the right. Because the combined height of Mario and Toad is too tall to pass through the narrow passage where the simple block is located, a dismount interaction is simulated subsequently. Finally, Mario is able to destroy the simple block since it is now in his scope of action.

This interaction plan is then negotiated between the two agents before they start sensorimotor planning. As Toad observed Mario and thus learned the same knowledge entries, he

⁴Scenario 1: https://youtu.be/0zle8L6H-_4

⁵Scenario 2: https://youtu.be/WzOg_WcNDik

⁶Additional Scenario: <https://youtu.be/7RV4QCwDK8U>

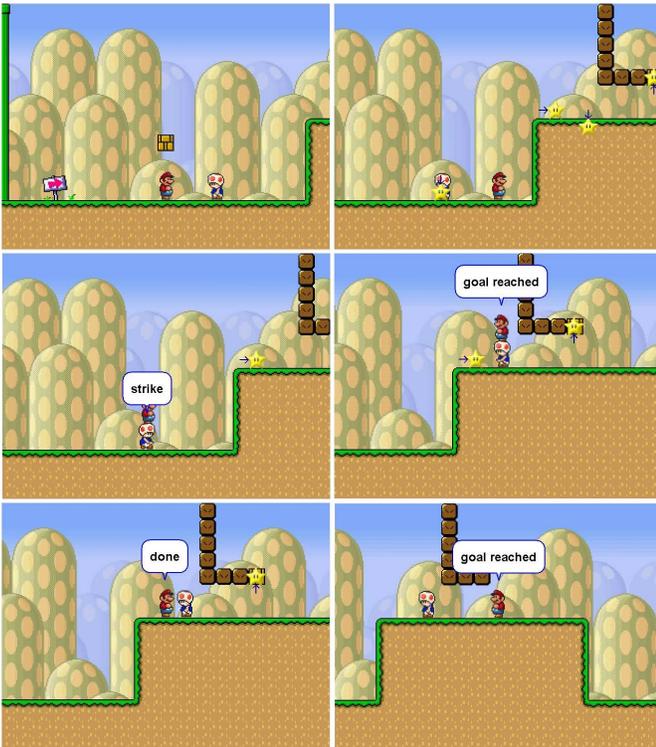


Fig. 5. Scenario 1: Toad helps Mario to destroy a block.

infers the same schematic plan and thus considers the proposal useful and accepts. After the agreement, both agents plan their part of the interaction sequence in terms of keystrokes (top right picture) and wait for the other agent to execute its part when necessary. The resulting execution of the plan is shown in the following pictures: Mario mounting Toad; Toad transporting Mario to the elevated ground; Mario dismounting Toad and finally Mario moving to the simple block and destroying it.

B. Mario Clears a Path for Toad

In the second scenario, shown in Figure 6, Toad is at first instructed to collect the coin object, while Mario is ordered to destroy the simple block (see top left picture). We assume that Toad is not able to destroy a simple block by himself, and does not generalize that he can do so as well. Toad is instructed to increase his number of coins (top right picture). Although he knows that a collision with a coin will yield the desired effect, there is no coin inside his scope of action, since the only coin in the scene is blocked by a simple block. Thus, the schematic planning module anticipates a destruction of the simple block by Mario (bottom left picture), expanding Toad's scope of action. After that, Toad is able to collect the coin (bottom right picture).

Both shown scenarios demonstrate how SEMLINCS agents are able to learn about each other, include each other in their action plans by recognizing individual scopes of action in an environmental context, and coordinate the joint execution of the plans. Communicating cooperative goals to the participat-

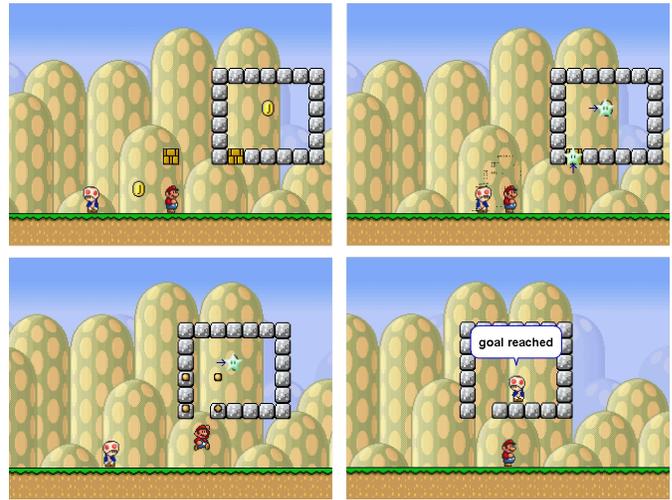


Fig. 6. Scenario 2: Mario helps Toad to collect a coin.

ing agents establishes a common ground, consisting of the final goal an agent wants to achieve as well as the interactions it plans to execute while pursuing the final goal.

IV. CONCLUSION

Humans are able to understand other agents as individual, intentional agents, who have their own knowledge, beliefs, perspectives, abilities, motivations, intentions, and so their own mind. [24]–[26]. Furthermore, we are able to cooperate with others highly flexibly and context-dependently, which requires coordination. This coordination can be supported by communication, helping to establish a common ground about a joint interaction goal.

In the presented work, we showed how social cooperative skills can be realized in artificial agents. To do so, we equipped the agents with different behavioral skills, such that particular goals could only be reached with the help of another agent. To coordinate a required joint action, SEMLINCS had to enable agents to learn about the capabilities of other agents by observing other agent-environment interactions and to assign the learned event schema rules to particular agents. Moreover, our implementation shows how procedural rules can be applied to a local, environmental context, and how sensorimotor and more abstract schematic forward simulations can be distinguished in this process, and applied to build an effective, hierarchical planning structure. Besides the computational insights into the necessary system enhancements, our implementation opens new opportunities for future developments towards even more social, cooperative, artificial cognitive systems.

First of all, currently the agents always cooperate. A conditional cooperation could be based on the creation of an incentive for an agent to share its reward with the participating partner agent. Indeed, it has been shown that a sense of fairness in terms of sharing rewards when team play was necessary is a uniquely human ability [27]. While a sense of fairness is a motivation to share when help was provided – or also possibly when future help is expected, that is, expecting that the partner

will return the favor – a more long term motivation can create social bonds by monitoring social interactions with partners over time and preferring interactions and cooperations with those partners that have shared rewards in the past in a fair manner. Clearly many factors determine if one is willing to cooperate, including social factors, game theory factors, and related aspects – all of which take the expected own effort into account, the expected effort of the cooperating other(s), as well as the expected personal gain and the gain for the others.

It also needs to be noted that currently action plans are executed in a strict, sequential manner. In the real world, however, joint actions are typically executed concurrently, such as when preparing dinner together [25]. Thus, in the near future we will face the challenge to allow the parallel execution of cooperative interactions, which will make the timing partially much more critical.

Although our agents already communicate plans on an abstract, schematic level, all sequential steps of the plans need to be fully verbalized in order to coordinate a joint action at the moment. An alternative would be to simply utter the goal and ask for help, thus expecting the other agent to help under consideration of the known behavioral abilities of the individual agent. Therefore, more elaborate theories of mind would need to be taken into consideration [28]. For example, in the first scenario mentioned above, Toad may realize that he needs to transport Mario to the higher ground on the right to enable Mario to destroy the box up there, because Mario cannot reach this area. Humans are clearly able to utter or even only manually signal a current goal and still come up with a joint plan, without verbally communicating the plan in detail. While verbal communication certainly helps in the coordination process, obvious interactions can also unfold successfully without communication (e.g. letting another pedestrian pass; passing an object out of reach of another person, who apparently needs it). Although the Mario world is rather simple, cooperative interactions of this kind could actually be enabled when enhancing the current SEMLINCS architecture with the option to simulate potential goals of the other agent and plans on how to reach them, thus offering a helping hand wherever it seems necessary.

REFERENCES

- [1] S. M. Lucas, M. Mateas, M. Preuss, P. Spronck, and J. Togelius, "Artificial and Computational Intelligence in Games (Dagstuhl Seminar 12191)," *Dagstuhl Reports*, vol. 2, no. 5, pp. 43–70, 2012. [Online]. Available: <http://drops.dagstuhl.de/opus/volltexte/2012/3651>
- [2] G. N. Yannakakis and J. Togelius, "A panorama of artificial and computational intelligence in games," *Computational Intelligence and AI in Games, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2014.
- [3] L. W. Barsalou, "Grounded cognition," *Annual Review of Psychology*, vol. 59, pp. 617–645, 2008.
- [4] J. Hoffmann, *Vorhersage und Erkenntnis: Die Funktion von Antizipationen in der menschlichen Verhaltenssteuerung und Wahrnehmung. [Anticipation and cognition: The function of anticipations in human behavioral control and perception.]*. Göttingen, Germany: Hogrefe, 1993.
- [5] M. V. Butz, O. Sigaud, and P. Gérard, "Internal models and anticipations in adaptive learning systems," in *Anticipatory Behavior in Adaptive Learning Systems: Foundations, Theories, and Systems*, M. V. Butz, O. Sigaud, and P. Gérard, Eds. Berlin Heidelberg: Springer-Verlag, 2003, pp. 86–109.
- [6] M. V. Butz, "How and why the brain lays the foundations for a conscious self," *Constructivist Foundations*, vol. 4, no. 1, pp. 1–42, 2008.
- [7] K. Friston, "Learning and inference in the brain," *Neural Netw.*, vol. 16, no. 9, pp. 1325–1352, 2003.
- [8] —, "The free-energy principle: a rough guide to the brain?" *Trends in Cognitive Sciences*, vol. 13, no. 7, pp. 293 – 301, 2009.
- [9] A. Clark, "Whatever next? predictive brains, situated agents, and the future of cognitive science," *Behavioral and Brain Science*, vol. 36, pp. 181–253, 2013.
- [10] J. M. Zacks and B. Tversky, "Event structure in perception and conception," *Psychological Bulletin*, vol. 127, no. 1, pp. 3–21, 2001.
- [11] J. M. Zacks, N. K. Speer, K. M. Swallow, T. S. Braver, and J. R. Reynolds, "Event perception: A mind-brain perspective," *Psychological Bulletin*, vol. 133, no. 2, pp. 273–293, 2007.
- [12] B. Hommel, J. Müssele, G. Aschersleben, and W. Prinz, "The theory of event coding (TEC): A framework for perception and action planning," *Behavioral and Brain Sciences*, vol. 24, pp. 849–878, 2001.
- [13] W. Prinz, "A common coding approach to perception and action," in *Relationships between perception and action*, O. Neumann and W. Prinz, Eds. Berlin Heidelberg: Springer-Verlag, 1990, pp. 167–201.
- [14] A. Newell and H. A. Simon, *Human problem solving*. Englewood Cliffs, NJ: Prentice-Hall, 1972.
- [15] M. V. Butz and J. Hoffmann, "Anticipations control behavior: Animal behavior in an anticipatory learning classifier system," *Adaptive Behavior*, vol. 10, pp. 75–96, 2002.
- [16] P. F. Dominey, "Recurrent temporal networks and language acquisition: from corticoatrial neurophysiology to reservoir computing," *Frontiers in Psychology*, vol. 4, pp. 500–, 2013.
- [17] K. Pastra and Y. Aloimonos, "The minimalist grammar of action," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 367, pp. 103–117, 2012.
- [18] F. Wörgötter, A. Agostini, N. Krüger, N. Shylo, and B. Porr, "Cognitive agents—a procedural perspective relying on the predictability of object-action-complexes (OACs)," *Robotics and Autonomous Systems*, vol. 57, no. 4, pp. 420–432, 2009.
- [19] F. Schrodtt, J. Kneissler, S. Ehrenfeld, and M. V. Butz, "Mario becomes cognitive," *TOPICS in Cognitive Science*, in press.
- [20] M. V. Butz, "Towards a unified sub-symbolic computational theory of cognition," *Frontiers in Psychology*, vol. 7, no. 925, 2016.
- [21] M. V. Butz, S. Swarup, and D. E. Goldberg, "Effective online detection of task-independent landmarks," in *Online Proceedings for the ICML'04 Workshop on Predictive Representations of World Knowledge*, R. S. Sutton and S. Singh, Eds. online, 2004, p. 10. [Online]. Available: <http://homepage.mac.com/rssutton/ICMLWorkshop.html>
- [22] E. von Holst and H. Mittelstaedt, "Das Reafferenzprinzip (Wechselwirkungen zwischen Zentralnervensystem und Peripherie.)," *Naturwissenschaften*, vol. 37, pp. 464–476, 1950.
- [23] M. Tomasello, *A Natural History of Human Thinking*. Harvard University Press, 2014.
- [24] R. L. Buckner and D. C. Carroll, "Self-projection and the brain," *Trends in Cognitive Sciences*, vol. 11, pp. 49–57, 2007.
- [25] N. Sebanz, H. Bekkering, and G. Knoblich, "Joint action: Bodies and minds moving together," *Trends in cognitive sciences*, vol. 10, pp. 70–76, 2006.
- [26] M. Tomasello, M. Carpenter, J. Call, T. Behne, and H. Moll, "Understanding and sharing intentions: The origins of cultural cognition," *Behavioral and Brain Sciences*, vol. 28, pp. 675–691, 2005.
- [27] K. Hamann, F. Warneken, J. R. Greenberg, and M. Tomasello, "Collaboration encourages equal sharing in children but not in chimpanzees," *Nature*, vol. 476, no. 7360, pp. 328–331, 2011.
- [28] C. Frith and U. Frith, "Theory of mind," *Current Biology*, vol. 15, no. 17, pp. R644–R645, 2005.