EUCOG

Proceedings of EUCognition 2016

Cognitive Robot Architectures

European Society for Cognitive Systems

<u>www.eucognition.org</u>

Vienna, 8-9 December, 2016

Edited by
Ron Chrisley[*]
Vincent C. Müller[**]
Yulia Sandamirskaya[***]
Markus Vincze[****]

* University of Sussex, Sackler Centre for Consciousness Science, Sussex House, Falmer, Brighton BN1 9RH, UK
** University of Leeds, Interdisciplinary Ethics Applied Centre (IDEA), 17 Blenheim Terrace, Leeds LS2 9JT, UK
*** University & ETH Zurich, Institute of Neuroinformatics, Neuroscience Center Zurich, Winterthurerstr. 190, 8057 Zurich, Switzerland
**** Technical University of Vienna, Institut für Automatisierungs- und Regelungstechnik, Gußhausstr. 27-29, 1040 Wien, Austria

Table of Contents

Section 3: Abstracts

Preface

The *European Association for Cognitive Systems* is the association resulting from the EUCog network, which has been active since 2006. It has ca. 1000 members and is currently chaired by Vincent C. Müller. We ran our annual conference on December 08-09 2016, kindly hosted by the Technical University of Vienna with Markus Vincze as local chair. The invited speakers were David Vernon and Paul F.M.J. Verschure.

Out of the 49 submissions for the meeting, we accepted 18 a papers and 25 as posters (after double-blind reviewing). Papers are published here as "full papers" or "short papers" while posters are published here as "short papers" or "abstracts".

Some of the papers presented at the conference will be published in a separate special volume on 'Cognitive Robot Architectures' with the journal *Cognitive Systems Research*.

RC, VCM, YS, MV

Section 1: Full Papers

Industrial Priorities for Cognitive Robotics

David Vernon[†]
Carnegie Mellon University Africa
Rwanda
Email: vernon@cmu.edu

Markus Vincze
Technische Universität Wien
Austria
Email: vincze@acin.tuwien.ac.at

Abstract—We present the results of a survey of industrial developers to determine what they and their customers require from a cognitive robot. These are cast as a series of eleven functional abilities:

1) **Safe, reliable, transparent operation.**
2) **High-level instruction and context-aware task execution.**
3) **Knowledge acquisition and generalization.**
4) **Adaptive planning.**
5) **Personalized interaction.**
6) **Self-assessment.**
7) **Learning from demonstration.**
8) **Evaluating the safety of actions.**
9) **Development and self-optimization.**
10) **Knowledge transfer.**
11) **Communicating intentions and collaborative action.**

I. Industrial Requirements

While cognitive robotics is still an evolving discipline and much research remains to be done, we nevertheless need to have a clear idea of what cognitive robots will be able to do if they are to be useful to industrial developers and end users. The RockEU2 project canvassed the views of thirteen developers to find out what they and their customers want. The results of this survey follow, cast as a series of eleven functional abilities.

A. Safe, reliable, transparent operation

Cognitive robots will be able to operate reliably and safely around humans and they will be able to explain the decisions they make, the actions they have taken, and the actions they are about to take. A cognitive robot will help people and prioritize their safety. Only reliable behaviour will build trust. It will explain decisions, i.e. why it acted the way it did. This is essential if the human is to develop a sense of trust in the robot.

A cognitive robot will have limited autonomy to set intermediate goals to when carrying out tasks set by users. However, in all cases it defers to the users preferences, apart from some exceptional circumstances, e.g. people with dementia can interact in unpredictable ways and the robot will be able to recognize these situations and adapt in some appropriate manner.

The freedom to act autonomously will have formal boundaries and the rules of engagement will be set on the basis of

three parameters: safety for people, safety for equipment, and safety of the robot system. The rules may change depending on the environment and a cognitive robot will not exceed the limits of safe operation. The limits may be application specific, e.g., the robot should not deviate further than a given specification/distance/etc. A cognitive robot will use this type of knowledge to act responsibly and will ask for assistance when necessary (e.g. before it encounters difficulties). In particular, in emergency situations, the robot will stop all tasks to follow some emergency procedure. Ideally, if the user is deliberately trying to misuse the robot, e.g. programming it to assist with some unethical task, a cognitive robot will cease operation.

B. High-level instruction and context-aware task execution

Cognitive robots will be given tasks using high-level instructions and they will factor in contextual constraints that are specific to the application scenario when carrying out these tasks, determining for themselves the priority of possible actions in case of competing or conflicting requirements.

Goals and tasks will be expressed using high-level instructions that will exploit the robots contextual knowledge of the task. This will allow the robot to pre-select the information that is important to effectively carry out the task. The goals will reflect the users perspective. This means that all skills which implicitly define the goals are tightly linked to real-world needs and to the solution of specific problems, e.g., "get me a hammer". The following guidelines will apply.

- Instructions will use natural language and gestures to specify the goals.
- Natural language will be relatively abstract but will be grounded in the codified organisational rules, regulations, and behavioural guidelines that apply to a given application environment. This grounding means that each abstract instruction is heavily loaded with constraints which should make it easier for the robot to understand and perform the task effectively.
- The goals should be specified in a formalised and structured way, where the designer defines them well and can verify them. For example, teach the robot the environment it is working in, follow a described route to reach each of the target locations and reach these positions to carry out the task. These clearly-specified tasks are tightly coupled with risks and costs, e.g. of incorrect execution.

- It should be possible for the robot to be given goals in non-specific terms (e.g. assist in alleviating the symptoms of dementia), guidelines on acceptable behaviour (or action policies), and relevant constraints, leaving it to the robot to identify the sub-goals that are needed to achieve these ultimate goals.
- A cognitive robot will learn ways of measuring the success of outcomes for the objectives that have been set, e.g., creating a metric such as the owners satisfaction related not only to the directly specified objective but also the manner in which the job was done). It should be learn from these metrics.

A cognitive robot will consider the contextual constraints that are specific to the application scenario. It will determine the priority of potential actions, e.g., in case of competing or conflicting needs.

For example, the robot might know the procedure to be followed but the locations to be visited or the objects to be manipulated need to be specified (or vice versa). For example, when an automated harvester encounters a bale of straw, it can deal with it as an obstacle or something to be harvested, depending on the current task. For example, the robot might engage in spoken interaction with older adults until the goal is communicated unambiguously, using context to disambiguate the message and allow for the difficulties in dealing with different accents, imprecise speech, and poor articulation.

A cognitive robot will know what is normal, i.e. expected, behaviour (possibly based on documented rules or practices) and it will be able to detect anomalous behaviour and then take appropriate action.

The following guidelines will apply.

- It will be possible to pre-load knowledge about the robots purpose and its operating environment, including any rules or constraints that apply to behaviour in that environment.
- It will be possible to utilize domain-specific skill pools (e.g. from shared databases) so that the robot is pre-configured to accomplish basic tasks without having to resort to learning or development.
- The robot will continually improve its skills (within limits of the goals and safety, see above) and share these with other robots.
- The robot might assist the user by proposing goals from what it understood and the user makes the final selection.

The level of detail in the description required by a cognitive robot will decrease over time as the robot gains experience, in the same way as someone new on the job is given very explicit instructions at first and less explicit instructions later on. One should need to demonstrate only the novel parts of the task, e.g., pouring liquid in a container, but not the entire process.

It will be possible to instruct the robot off-line if there is no access to the physical site; e.g., using a simulation tool, with the robot then being deployed in the real scenario.

C. Knowledge acquisition and generalization

Cognitive robots will continuously acquire new knowledge and generalize that knowledge so that they can undertake new tasks by generating novel action policies based on their history of decisions. This will allow the rigor and level of detail with which a human expresses the task specification to be relaxed on future occasions.

A cognitive robot will build and exploit experience so that its decisions incorporate current and long term data. For example, route planning in a factory, hospital, or hotel should take into account the history of rooms and previous paths taken, or it might take another look to overcome high uncertainty. In general, the robot will overcome uncertainty in a principled manner.

A cognitive robot will generalize knowledge to new task by understanding the context of a novel task and extrapolating from previous experience. For example, a care-giving robot will reuse knowledge of a rehabilitation exercise, customizing it to another person. A welding robot will weld a new instance of a family of parts. In general, a cognitive robot will extract useful meaning from an interaction for a future and more general use, with the same or another user. This may extend to learn cultural preferences and social norms.

For example, in a domestic environment, a cognitive robot will learn how to do simple household tasks, e.g. how to grasp different objects and them bring to a person that wants them. This will be continuously extended, allowing the robot to do more complex things, including cooking.

D. Adaptive planning

Cognitive robots will be able to anticipate events and prepare for them in advance. They will be able to cope with unforeseen situations, recognizing and handling errors, gracefully and effectively. This will also allow them to handle flexible objects or living creatures.

A cognitive robot will be able to recognize that circumstances have changed to avoid situations where progress is impossible. It will also be able to recognize errors and recover. This may include retrying with a slightly different strategy. The learning process will be fast, ideally learning from each error.

A cognitive robot will be able to learn how to handle errors, how to react to situations where, e.g., a human is doing something unexpected or parts are located in an unexpected place.

A cognitive robot will be able to anticipate events and compensate for future conditions. For example, an automated combine harvester will be able to apply a pre-emptive increase of power to compensate for the demands caused when an area of high yield is encountered.

A cognitive robot will be able to learn about the environment it is in and modify the its current information accordingly. That is, it will adapt to changes in the environment, verifying that the environment matches with what is known, or there is a change and updates. This may require an update of the task but only after asking the user.

A cognitive robot will be able to manipulate flexible or live objects, e.g. living creatures such as laboratory mice. To do so means that the robot must be able to construct a model of their behaviour and adapt its actions as required, continually refining the model.

E. Personalized interaction

Cognitive robots will personalize their interactions with humans, adapting their behaviour and interaction policy to the users preferences, needs, and emotional or psychological state. This personalization will include an understanding of the person's preferences for the degree of force used when interacting with the robot. A cognitive robot will be able to adapt its behaviour and interaction policy to accommodate the user's preferences, needs, and emotional state. It will learn the personal preferences of the person with whom it is interacting. For example, an autonomous car will learn the preferred driving style of the owner and adopt that style to engender trust.

A cognitive robot will understand nuances in tone to learn a person's voice, detecting signs of stress so that it can react to it and review what it is doing. In the particular case of interaction with older adults, the robot will be able to understand gestures to help disambiguate words.

A cognitive robot will able to extrapolate what has been taught to other situations. For example, it might remember that the user has certain preferences (e.g. to be served tea in the morning) and the robot will remember that preference. However, the robot will not allow these learned preferences to over-ride critical actions policies.

In cases where showing the robot what to do involves physical contact between the user and the robot, the robot will be able to learn the dynamics of the user, i.e. his or her personal preferred use of forces when interacting with objects in the environment.

A cognitive robot will be able to the psychological state of a user, e.g. based on the facial expressions, gestures, actions, movements. Based on this, it will be able to determine what they need by cross-referencing that with knowledge of the persons history.

A cognitive robot will be able to make decisions from a large body of observed data, thereby assisting people who typically make decisions based on learned heuristic knowledge but without a quantitative basis for this decision-making. For example, there is a need to provide farmers with a fact-based quantitative decision-making framework. A cognitive robot or machine would observe the physical environment and the farmer and provide a sound bases for making improved decisions.

F. Self-assessment

Cognitive robots will be able to reason about their own capabilities, being able to determine whether they can accomplish a given task. If they detect something is not working, they will be able to ask for help. They will be able to assess the quality of their decisions.

If a cognitive robot is asked to perform a certain task, it will be able to say whether it can do it or not. It will detect when something is not working and will be able to ask for help.

A cognitive robot will assess the quality of its decisions and apply some level of discrimination in the task at hand, e.g. being selective in its choice of fruit to harvest.

G. Learning from demonstration

Cognitive robots will be able to learn new actions from demonstration by humans and they will be able to link this learned knowledge to previously acquired knowledge of related tasks and entities.

Instructions will be communicated by demonstration, through examples, including showing the robot the final results, with the robot being able to merge prior know-how and knowledge with learning by demonstration. Some of this prior knowledge should be extracted from codified organisational rules, regulations, and behavioural guidelines.

The situation is analogous to training an intern or an apprentice: a trainer might ask"Has someone shown you how to do this? No? Okay, Ill show you how to do three, then you do 100 to practice (and to throw away afterwards). If you get stuck on one, call me, and Ill show you how to solve that problem".

A cognitive robot will learn and adapt the parameters to achieve the task. Today in the assembly of components, often robot assembly is not robotized because it requires too much engineering and it is too difficult for robots because it is based on traditional programming, tuning and frequent re-tuning of parameters.

Teaching will exploit natural language, gaze and pointing gestures, and by showing the robot what to do and helping it when necessary.

Actions will be expressed in high-level abstract terms, like a recipe, ideally by talking to it. For example, "go to hall 5 from hall 2 and pick up the hammer" or "open the valve".

When being taught, the robot should be anticipating what you are trying to teach it so that it predicts what you want it to do and then tries to do it effectively.

It will be possible to provide direct support for the robot, switching fluidly between full autonomy, partial autonomy, or manual control.

H. Evaluating the safety of actions

When they learn a new action, cognitive robots will take steps to verify the safety of carrying out this action. If a robot learns new action, it will be difficult to certify the new action. The process of generating a new action will involve interaction with the world and that may already be harmful. So, when learning a new action, there needs to be a step to verify the safety of carrying out this action. For example, showing a new action plus defining safety and success such that the robot can check if it achieved success.

I. Development and self-optimization

Cognitive robots will develop and self-optimize, learning in an open-ended manner from their own actions and those of others (humans or other robots), continually improving their abilities.

A cognitive robot will be able to use what it has learned to determine possible ways to improve its performance, e.g. through internal simulation at times when the robot is not working on a given task. It will also be able to learn from its mistakes, e.g., breaking china but learning from the effect of the action. A cognitive robot will learn to optimize the actions it performs (e.g. doing something faster) within the certified limits of safety and without increasing the risk of failure and associated costs.

J. Knowledge transfer

Cognitive robots will be able to transfer knowledge to other robots, even those having a different physical, kinematic, and dynamic configurations and they will be able to operate seamlessly in an environment that is configured as an internet of things (IoT).

A cognitive robot will be a crucial component of cyber-physical systems where the robot can be used, for example, as a way of collecting data from large experiments.

K. Communicating intentions and collaborative action

Cognitive robots will be able to communicate their intentions to people around them and, vice versa, they will be able to infer the intention of others, i.e. understanding what someone is doing and anticipating what they are about to do. Ultimately, Cognitive robots will be able to collaborate with people on some joint task with a minimal amount of instruction.

The need for people around a cognitive robot to be able to anticipate the robots actions is important because, if cognitive robots are to be deployed successfully, people need to believe the robot is trustworthy. A cognitive robot will be able to interact with people, collaborating with them on some joint task. This implies that the robot has an ability to understand what the person is doing and infer their intentions.

II. Conclusion

Establishing functional requirements is an essential prerequisite to developing useful systems. This is as true of cognitive robotics as it is for any other domain of information and communication technology. However, the effort to give robots a capacity for cognition is made more difficult by the fact that cognitive science, as a discipline in its own right, does not yet have many established normative models that lend themselves to realization in well-engineered systems. The goal of the work described in this short paper is to re-assert the priority of user requirements in the specification of cognitive robot systems. The motivation underpinning this goal is that, having identified these requirements, we can then proceed to determine the scientific and technological tools and techniques — drawn from the disciplines of artificial intelligence, autonomous systems, and cybernetics, among others — that can be deployed to satisfy these requirements in practical robots. It remains to complete this exercise.

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

Fabian Schrodt
Department of Computer Science
Eberhard Karls University of Tübingen
tobias-fabian.schrodt@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 unifed 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 environment[1][2][3]. Seeing that the game is in fact rather complex,

---

[1]https://www.youtube.com/watch?v=AplG6KnOr2Q
[2]https://www.youtube.com/watch?v=ltPj3RlN4Nw
[3]https://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 reaference 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
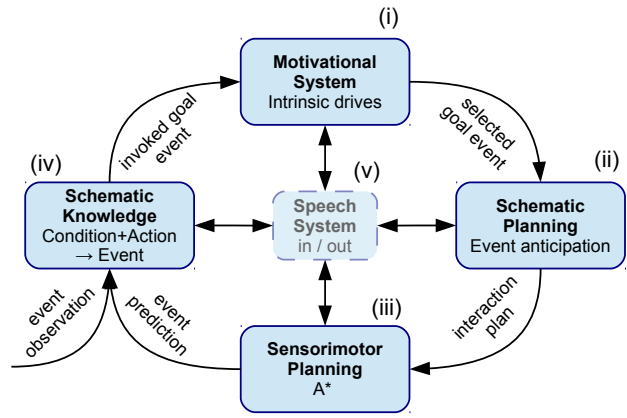

Fig. 1. Overview of the main modules and the cognitive control loop in the implementation of SEMLINCS.

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 reaference 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.
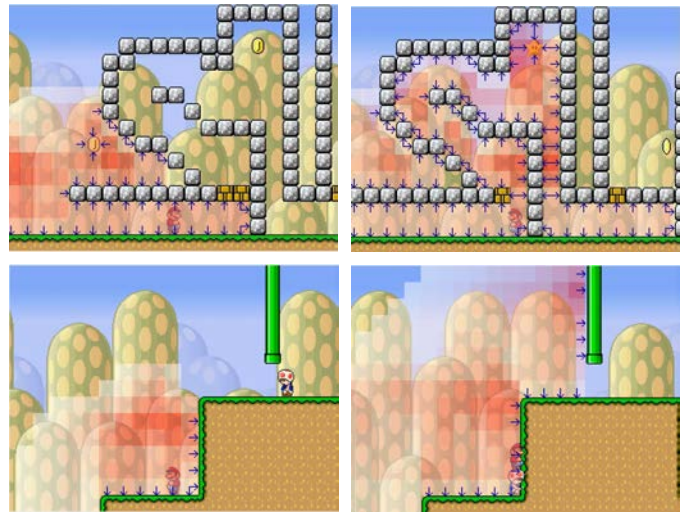

Fig. 2. Expansion of the scope of action by simulating environmental interactions. Red fields mark the reachable positions, while blue arrows denote the registered object interaction options, while simulating the scope of action. Top row: The scope of action is updated by simulating the destruction of 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 counter proposal that is always accepted by the initial planning agent. The process of negotiation is shown in Figure 3.
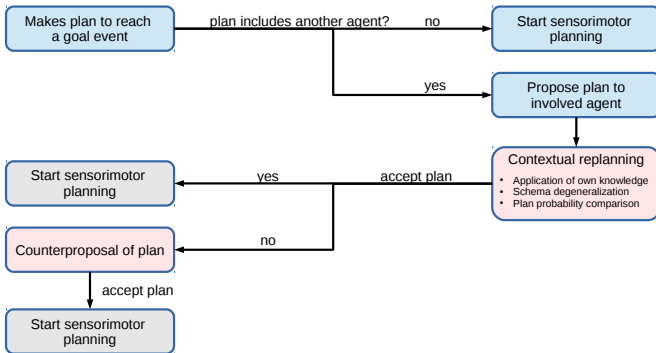

Fig. 3. Negotiation diagram for two agents. Blue boxes: Tasks of the planning agent. Red boxes: Tasks of an agent involved in the initial plan. Grey boxes: Both agents are planning.

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 SEMLINCS in various, similar scenarios and have observed the unfolding of similarly well-coordinated

behavior. Videos showcasing these scenarios are available online[4][5]. 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 here [6].

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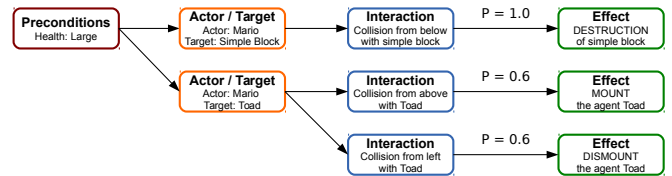
Fig. 4. Mario's schematic knowledge in scenario 1. The respective entries are put into context by the schematic planning module.

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

---

[4]Scenario 1: https://youtu.be/0zle8L6H-_4
[5]Scenario 2: https://youtu.be/WzOg_WcNDik
[6]Additional Scenario: https://youtu.be/7RV4QCwDK8U

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

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

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 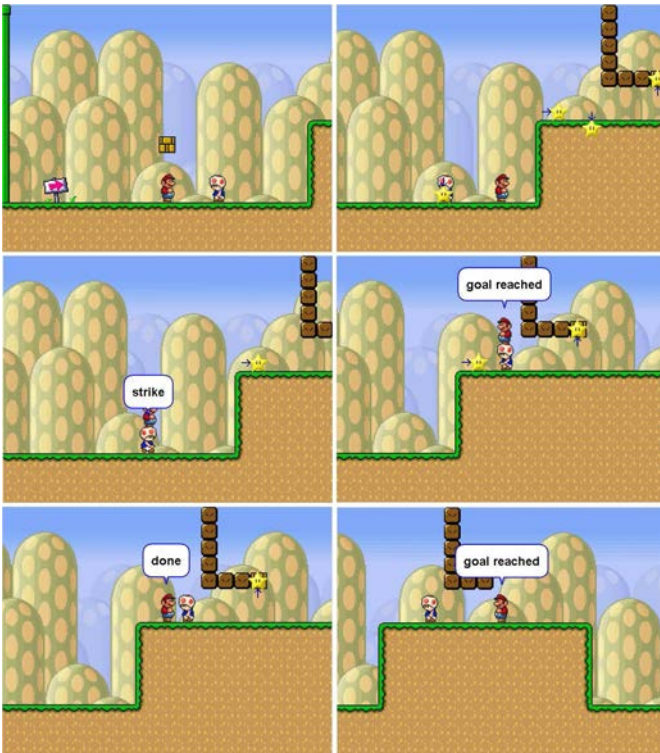execute its part when necessary. The resulting execution of the plan is shown in the following pictures: Mario mounting Toad; Toad transporting Mario to the elevated ground; Mario dismounting Toad and finally Mario moving to the simple block and destroying it.

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-

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. Müsseler, 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 corticostriatal 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. Schrodt, 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.

Representational Limits in Cognitive Architectures

Antonio Lieto

University of Turin, Department of Computer Science, Italy
ICAR-CNR, Palermo, Italy
lieto.antonio@gmail.com
http://www.antoniolieto.net

Abstract—**This paper proposes a focused analysis on some problematic aspects concerning the knowledge level in General Cognitive Architectures (CAs). In particular, it addresses the problems regarding both the limited size and the homogeneous typology of the encoded (and processed) conceptual knowledge. As a possible way out to face, jointly, these problems, this contribution discusses the possibility of integrating external, but architecturally compliant, cognitive systems into the knowledge representation and processing mechanisms of the CAs.**

Keywords—cognitive architectures, knowledge representation, knowledge level, common-sense reasoning.

I. INTRODUCTION

The research on Cognitive Architectures (CAs) is a wide and active area involving a plethora of disciplines such as Cognitive Science, Artificial Intelligence, Robotics and, more recently, the area of Computational Neuroscience. CAs have been historically introduced i) "to capture, at the computational level, the invariant mechanisms of human cognition, including those underlying the functions of control, learning, memory, adaptivity, perception and action" [1] and ii) to reach human level intelligence, also called General Artificial Intelligence, by means of the realization of artificial artifacts built upon them. During the last decades many cognitive architectures have been realized, - such as ACT-R [2], SOAR [3] etc. - and have been widely tested in several cognitive tasks involving learning, reasoning, selective attention, multimodal perception, recognition etc. Despite the recent developments, however, in the last decades the importance of the "knowledge level" [4] has been historically and systematically downsized by this research area, whose interests have been mainly based on the analysis and the development of mechanisms and the processes governing human and (artificial) cognition. The knowledge level in CAs, however, presents several problems that may affect the overall heuristic and epistemological value of such artificial general systems and therefore deserves more attention.

II. TWO PROBLEMS FOR THE KNOWLEDGE LEVEL IN CAs

Handling a huge amount of knowledge, and selectively retrieve it according to the needs emerging in different situational scenarios, represents an important aspect of human intelligence. For this task humans adopt a wide range of heuristics [5] due to their "bounded rationality" [6]. Currently, however, the Cognitive Architectures are not able, de facto, to deal with complex knowledge structures that can be even slightly comparable to the knowledge heuristically managed by humans. In other terms: CAs are general structures without a general content. This means that the knowledge embedded and processed in such architectures is usually very limited, ad-hoc built, domain specific, or based on the specific tasks they have to deal with. Thus, every evaluation of the artificial systems relying upon them, is necessarily task-specific and do not involve not even the minimum part of the full spectrum of processes involved in the human cognition when the "knowledge" comes to play a role. As a consequence, the structural mechanisms that the CAs implement concerning knowledge processing tasks (e.g. that ones of retrieval, learning, reasoning etc.) can be only loosely evaluated, and compared w.r.t. that ones used by humans in similar knowledge-intensive situations. In other words: from an epistemological perspective, the explanatory power of their computational simulation is strongly affected [7,8]. Such knowledge limitation, in our opinion, does not allow to obtain significant advancements in the cognitive science research about how the humans heuristically select and deal with the huge amount of knowledge that possess when they have to make decisions, reason about a given situation or, more in general, solve a particular cognitive task involving several dimensions of analysis. This problem, as a consequence, also limits the advancement of the research in the area of General Artificial Intelligence of cognitive inspiration.

The "content" limit of the cognitive architectures has been recently pointed out in literature [1] and some technical solutions for filling this "knowledge gap" have been proposed [9]. In particular the use of ontologies and of semantic formalisms and resources (such as DBPedia) has been seen as a possible solution for providing effective content to the structural knowledge modules of the cognitive architectures. Some initial efforts have been done in this sense but cover only part of the "knowledge problem" in CAs (i.e. the one concerning the limited "size" of the adopted knowledge bases). However, also these solutions, do not address another relevant aspect affecting the knowledge level of CAs: namely, the problem concerning the "knowledge homogeneity" issue. In other terms: the type of knowledge represented and manipulated by most CAs (including those provided with extended knowledge modules) is usually homogeneous in nature. It mainly covers, in fact, only the so called "classical" part of conceptual information (that one representing concepts in terms of necessary and sufficient information and compliant with ontological semantics (see [10]) on these aspects). On the other hand, the so called "common-sense" conceptual components of our knowledge (i.e. those that, based on the results from the cognitive science, allow to characterize concepts in terms of "prototypes", "exemplars" or "theories")

is largely absent in such computational frameworks. The possibility of representing and handling, in an integrated way, an heterogeneous amount of common sense conceptual representations (and the related reasoning mechanisms), in fact, is not sufficiently addressed both by the symbolic-based "chunk-structures" adopted by the most common general CAs (e.g. SOAR) and by fully connectionist architectures (e.g. LEABRA). This aspect is problematic also in the hybrid solutions adopted by CAs such as CLARION [11] or ACT-R (the different reasons leading to a non satisfactory treatment of this aspect are detailed in [12]). This type of knowledge, however, is exactly the type of "cognitive information" crucially used by humans for heuristic reasoning and decision making. This paper presents an analysis of the current situation by proposing a comparison of the representational level of SOAR, ACT-R, CLARION and Vector-LIDA. Finally, we suggest that a possible way out to deal with this problem could be represented by the integration of external cognitive systems into the knowledge representation and processing mechanisms of general cognitive architectures. Some initial efforts in this direction, have been proposed (see e.g. [13, 14]) and will be presented and discussed.

III. KNOWLEDGE REPRESENTATION IN CAS

In the following we provide a short overview of: SOAR [3], ACT-R [2], CLARION [11] and LIDA [15] (in its novel version known as Vector-LIDA [16]). The choice of these architecture has been based on the fact that they represent some of the most widely used systems (adopted in scenarios ranging from robotics to video-games) and their representational structures present some relevant differentiations that are interesting to investigate in the light of the issues raised in this paper. By analyzing, in brief, such architectures we will exclusively focus on the description of their representational frameworks since a more comprehensive review of their whole mechanisms is out of the scope of the present contribution (detailed reviews of their mechanisms are described in [17]; and [18]). We will show how all of them are affected, at different levels of granularity, by both the size and the knowledge homogeneity problems.

A. SOAR

SOAR is one of the oldest cognitive architectures. This system was considered by Newell a candidate for a Unified Theory of Cognition [19]. One of the main themes in SOAR is that all cognitive tasks can be represented by problem spaces that are searched by production rules grouped into operators. These production rules are red in parallel to produce reasoning cycles. From a representational perspective, SOAR exploits symbolic representations of knowledge (called chunks) and use pattern matching to select relevant knowledge elements. Basically, where a production match the contents of declarative (working) memory the rule fires and then the content from the declarative memory (called Semantic Memory in SOAR) is retrieved. This system adheres strictly to the Newell and Simon's physical symbol system hypothesis which assumes that symbolic processing is a necessary and sufficient condition for intelligent behavior. The SOAR system encounter, in general, the standard problems affecting symbolic formalisms at the representational level: it is not well equipped to deal with common-sense knowledge representation and reasoning (since approximate comparisons are hard and computationally

intensive to implement with graph-like representations), and, as a consequence, the typology of encoded knowledge is biased towards the ``classical'' (but unsatisfactory) representation of concepts in terms of necessary and sufficient conditions [10]. This characterization, however, is problematic for modelling real world concepts and, on the other hand, the so called common-sense knowledge components (i.e. those that, allow to characterize and process conceptual information in terms of *typicality* and involving, for example, prototypical and exemplar based representations and reasoning mechanisms) is largely absent. This problem arises despite the fact that the chunks in SOAR can be represented as a sort of frame-like structures containing some common-sense (e.g. prototypical) information [12]. W.r.t. to the size problem, the SOAR knowledge level is also problematic. SOAR agents, in fact, are not endowed with general knowledge and only process ad-hoc built (or task-specific learned) symbolic knowledge structures.

B. ACT-R

ACT-R is a cognitive architectures explicitly inspired by theories and experimental results coming from human cognition. Here the cognitive mechanisms concerning the knowledge level emerge from the interaction of two types of knowledge: declarative knowledge, that encodes explicit facts that the system knows, and procedural knowledge, that encodes rules for processing declarative knowledge. In particular, the declarative module is used to store and retrieve pieces of information (called chunks, featured by a type and a set of attribute-value pairs, similar to frame slots) in the declarative memory. ACT-R employs a wide range of sub-symbolic processes for the activation of symbolic conceptual chunks representing the encoded knowledge. Finally, the central production system connects these modules by using a set of IF-THEN production rules using a set of IF-THEN production rules. Differently from SOAR, ACT-R allows to represent the information in terms of prototypes and exemplars and allow to perform, selectively, either prototype or exemplar-based categorization. This means that this architecture allows the modeller to manually specify which kind of categorization strategy to employ according to his specific needs. Such architecture, however, only partially addresses the homogeneity problem since it does not allow to represent, jointly, these different types of common-sense representations for the same conceptual entity (i.e. it does not assume a heterogeneous perspective). As a consequence, it is also not able to autonomously decide which of the corresponding reasoning procedures to activate (e.g. prototypes or exemplars) and to provide a framework able to manage the interaction of such different reasoning strategies (however its overall architectural environment provides, at least in principle, the possibility of implementing cascade reasoning processes triggering one another). Even if, in such architecture, some attempts exist concerning the design of harmonization strategies between different types of common-sense conceptual categorizations (e.g. exemplars-based and rule based, see [20]) however they do not handle the problem concerning the interaction of the prototype or exemplars-based processes according to the results coming from the experimental cognitive science (for example: the *old item effect*, privileging exemplars w.r.t. prototypes is not modelled. See again [12] for a detailed analysis of this aspect). Summing up: w.r.t. the knowledge homogeneity problem, the components needed to

fully reconcile the Heterogeneity approach with ACT-R are present, however they have not been fully exploited yet. Regarding the size problem: as for SOAR, ACT-R agents are usually equipped with task-specic knowledge and not with general cross-domain knowledge. In this respect some relevant attempts to overcome this limitation have been recently done by extending the Declarative Memory of the architecture. They will be discussed in section *E* along with their current implications.

C. CLARION

CLARION is a hybrid cognitive architecture based on the dual-process theory of mind. From a representational perspective, processes are mainly subject to the activity of two sub-systems, the Action Centered Sub-system (ACS) and the Non-Action Centered Sub-system (NACS). Both sub-systems store information using a two-layered architecture, i.e., they both include an explicit and an implicit level of representation. Each top-level chunk node is represented by a set of (micro)features in the bottom level (i.e., a distributed representation). The (micro)features (in the bottom level) are connected to the chunk nodes (in the top level) so that they can be activated together through bottom-up or top-down activation. Therefore, in general, a chunk is represented by both levels: using a chunk node at the top level and distributed feature representation at the bottom level. W.r.t. to the knowledge size and homogeneity problems, CLARION, encounter problems with both these aspects since i) there are no available attempts aiming at endowing such architecture with a general and cross-domain knowledge ii) the dual-layered conceptual information does not provide the possibility of encoding (manually or automatically via learning cycles) the information in terms of the heterogeneous classes of representations presented in the section 2. In particular: the main problematic aspect concerns the representation of the common-sense knowledge components. As for SOAR and ACT-R, also in CLARION the possible co-existence of typical representations in terms of prototypes, exemplars and theories (and the interaction among them) is not treated. In terms of reasoning strategies, notwithstanding that the implicit knowledge layer based on neural network representations can provide forms of non monotonic reasoning (e.g. based on similarity), such kind of similarity-based reasoning is currently not grounded on the mechanisms guiding the decision choices followed, for example, by prototype or exemplars-based reasoning.

D. Vector-LIDA

Vector LIDA is a cognitive architecture employing, at the representational level, high-dimensional vectors and reduced descriptions. High-dimensional vector spaces have interesting properties that make them attractive for representations in cognitive models. The distribution of the distances between vectors in these spaces, and the huge number of possible vectors, allow noise-robust representations where the distance between vectors can be used to measure the similarity (or dissimilarity) of the concepts they represent. Moreover, these high-dimensional vectors can be used to represent complex structures, where each vector denotes an element in the structure. However, a single vector can also represent one of these same complex structures in its entirety by implementing a reduced description, a mechanism to encode complex

hierarchical structures in vectors or connectionist models. These reduced description vectors can be expanded to obtain the whole structure, and can be used directly for complex calculations and procedures, such as making analogies, logical inference, or structural comparison. Vectors in this framework are treated as symbol-like representations, thus enabling different kind of operations executed on them (e.g. simple forms of compositionality via vectors blending). Vector- LIDA, encounters the same limitations of the other CAs since i) its agents are not equipped with a general cross-domain knowledge and therefore can be only used in very narrow tasks (their knowledge structure is either ad hoc build or ad hoc learned). Additionally, this architecture does not address the problem concerning the heterogeneity of the knowledge typologies. In particular its knowledge level does not represent the common-sense knowledge components such as prototypes and exemplars (and the related reasoning strategies). In fact, as for CLARION, despite vector-representations allow to perform many kind of approximate comparisons and similarity-based reasoning (e.g. in tasks such as categorization), the peculiarity concerning prototype or exemplars based representations (along with the the design of the interaction between their different reasoning strategies) are not provided. In this respect, however an element that is worth-noting is represented by the fact that the Vector-LIDA representational structures are very close to the framework of Conceptual Spaces. Conceptual Spaces are a geometric knowledge representation framework proposed by Peter Gärdenfors [21]. They can be thought as a particular class of vector representations where knowledge is represented as a set of quality dimensions, and where a geometrical structure is associated to each quality dimension. They are discussed in more detail in section 5. The convergence of the Vector-LIDA representation towards Conceptual Spaces could enable, in such architecture, the possibility of dealing with at least prototype and exemplars-based representations and reasoning, thus overcoming the knowledge homogeneity problem.

E. Attempts to Overcome the Knowledge Limits

As mentioned above, some initial efforts to deal with the limited knowledge availability for agents endowed with cognitive architecture have been done. In particular, within Mind'sEye program (a DARPA founded project), the knowledge layers of ACT-R architecture have been semantically extended with an external ontological content coming from three integrated semantic resources composed by the lexical databases WordNet [22], FrameNet [23] and by a branch of the top level ontology DOLCE [24] related to the event modelling. In this case, the amount of semantic knowledge selected for the realization of the Cognitive Engine (one of the systems developed within the MindEye Program) and for its evaluation, despite by far larger w.r.t. the standard ad-hoc solutions, was tailored on the specific needs of the system itself. It, in fact, was aimed at solving a precise task of event recognition trough a video-surveillance intelligent machinery; therefore only the ontological knowledge about the events was selectively embedded in it. While this is a reasonable approach in an applicative context, still does not allow to test the general cognitive mechanisms of a Cognitive Architecture on a general, multi faceted and multi-domain, knowledge. Therefore it does not allow to evaluate *strictu sensu* to what extent the designed heuristics allowing to retrieve and process, from a massive and composite knowledge

base, conceptual knowledge can be considered *satisfycing* w.r.t. the human performances. More recent works have tried to completely overcome at least the size problem of the knowledge level. To this class of works belongs that one proposed by Salvucci [9] aiming at enriching the knowledge model of the Declarative Memory of ACT-R with a world-level knowledge base such as DBpedia (i.e. the semantic version of Wikipedia represented in terms of ontological formalisms) and a previous one proposed in [25] presenting an integration of the ACT-R Declarative and Procedural Memory with the Cyc ontology [26] (one of the widest ontological resources currently available containing more than 230,000 concepts). Both the wide-coverage integrated ontological resources, however, represents conceptual information in terms of symbolic structures and encounter the standard problems affecting this class of formalisms and discussed above. Some of these limitations can be, in principle, partially overcome by such works, since the integration of such wide-coverage ontological knowledge bases with the ACT-R Declarative Memory allows to preserve the possibility of using the common-sense conceptual processing mechanisms available in that architecture (e.g. prototype and exemplars based). Therefore, in principle, dealing with the size problem also allows to address some aspects concerning the heterogeneity problem. Still, however, remains the problem concerning the lack of the representation of common-sense information to which such common-sense architectural processes can be applied: e.g. a conceptual retrieval based on prototypical traits (i.e. a prototype-based categorization) cannot be performed on such integrated ontological knowledge bases since these symbolic systems do not represent at all the *typical* information associated to a given concept ([12] presents an experiment on this aspect). In addition, as already mentioned, it remains not yet addressed the problem concerning the interaction, in a general and principled way, of the different types of common-sense processes involving different representations of the same conceptual entity. In the light of the arguments presented above it can be argued, therefore, that the current proposed solutions for dealing with the knowledge problems in CAs are not completely satisfactory. In particular, the integrations with huge world-level ontological knowledge bases can be considered a necessary solution for solving size problem. It is, however, insufficient for dealing with the knowledge homogeneity problem and with the integration of the common-sense conceptual mechanisms activated on heterogeneous bodies of knowledge, as assumed in the heterogeneous representational perspective. In the next sections we outline a possible alternative solution that, despite being not yet fully developed is, in perspective, suitable to account for both for the heterogeneous aspects in conceptualization and for the size problems.

IV.   INTEGRATING EXTERNAL COGNITIVE SYSTEMS IN CA

Recently some available conceptual categorization systems, explicitly assuming the heterogeneous representational hypothesis and integrated with wide-coverage knowledge bases (such as Cyc) have been developed and integrated with the knowledge level of available CAs. For our purposes, we will consider here the DUAL PECCS system [13, 14]. We will not discuss the results obtained by such system in tasks of conceptual categorization, since they have been already presented elsewhere [14]. We shall briefly focus, in the following, on the representational level of the system.

The knowledge level of DUAL PECCS is heterogeneous in nature since it is explicitly based and designed on the assumption that concepts are "heterogeneous proxytypes" [27] and, as such, they are composed by heterogeneous knowledge components selectively and contextually activated in working memory. In particular, by following the proposal presented in [28, 29], the representational level of DUAL PECCS couples Conceptual Spaces representations and ontological knowledge (consisting in the Cyc ontology) for the same conceptual entity. Conceptual Spaces [21] is used to represent and process the common-sense conceptual information. In such framework, to each quality dimension is associated a geometrical (topological or metrical) structure. In some cases, such dimensions can be directly related to perceptual mechanisms; examples of this kind are temperature, weight, brightness, pitch. In other cases, dimensions can be more abstract in nature. In this setting, concepts correspond to convex regions, and regions with different geometrical properties correspond to different sorts of concepts [21]. Here, prototypes and prototypical reasoning have a natural geometrical interpretation: prototypes correspond to the geometrical centre of a convex region (the centroid). Also exemplars-based representation can be represented as points in a multidimensional space, and their similarity can be computed as the intervening distance between each two points, based on some suitable metrics (such as Euclidean and Manhattan distance etc.). The ontological component, on the other hand, is used to provide and process the "classical" knowledge component for the same conceptual entity.

The representational level of DUAL PECCS (and the corresponding knowledge processing mechanisms) has been successfully integrated with the representational counterpart of some available CAs [14, 30] by extending, *de facto*, the knowledge representation and processing capabilities of cognitive architectures based on diverse representational assumptions. One of the main novelties introduced by DUAL PECCS (and therefore one of the main advantages obtained by the CAs extended with such external cognitive system) consists in the fact that it is explicitly designed the flow of interaction between common-sense categorization processes (based on prototypes and exemplars and operating on conceptual spaces representations) and the standard deductive processes (operating on the ontological conceptual component). The harmonization regarding such different classes of mechanisms has been devised based on the tenets coming from the dual process theory of reasoning [31, 32]. Additionally, in DUAL PECCS, also the interaction of the categorization processes occurring *within* the class of non monotonic categorization mechanisms (i.e. prototypes and exemplars-based categorization) has been devised and is dealt with at the Conceptual Spaces level. This latter aspect is of particular interest in the light of the multifaceted problem concerning the heterogeneity of the encoded knowledge. In fact, since the design of the interaction of the the different processes operating with heterogeneous representations still represents, as seen before, a largely unaddressed problem in current CAs, this system shows the relative easiness that its knowledge framework (and, in particular, the Conceptual Spaces component) provides to naturally model the dynamics between prototype and exemplars-based processes. For what concerns the size problem, finally, the possibile grounding of the Conceptual Spaces representational component with symbolic structures enables the integration with wide-coverage

knowledge bases such as Cyc. Thus, the solution adopted in DUAL PECCS is, in principle, able to deal with both the size and the knowledge homogeneity problems affecting the CAs. In particular, the extension of the Declarative Memories of the current CAs with this external cognitive system allowed to empower the knowledge processing and categorization capabilities of such general architectures (an important role, in this respect, is played by the Conceptual Spaces component). Despite there is still room of improvements and further investigations, this seems a promising way to deal with the both the knowledge problems discussed in this paper.

REFERENCES

1. Oltramari A., Lebiere C., Pursuing Artificial General Intelligence By Leveraging the Knowledge Capabilities Of ACT-R, AGI 2012 (5th International Conference on "Artificial General Intelligence"), Oxford, 2012.

2. Anderson, J.R., Bothell, D., Byrne, M.D., Douglass, S., Lebiere, C., & Qin, Y., An integrated theory of the mind, Psychological Review, 111(4), 1036-1060, http://dx.doi.org/10.1037/0033-295X.111.4.1036, 2004.

3. Laird, John E., The Soar Cognitive Architecture, MIT Press, 2012.

4. Newell, A. The knowledge level, Artificial intelligence 18 (1), 87-127, 1982.

5. Gigerenzer, G., Todd, P., Simple Heuristics that make us smart", Oxford University Press, 1999.

6. Simon, H., A Behavioral Model of Rational Choice, in Mathematical Essays on Rational Human Behaviour in Social Setting. NY: Wiley, 1957.

7. Minkowski M., Explaining the Computational Mind, MIT Press, 2013.

8. Lieto, A., Radicioni, D.P. From Human to Artificial Cognition and back: Challenges and Perspectives of Cognitively-Inspired AI sistems, Cognitive Systems Research, 39 (2), pp. 1-3, 2016.

9. Salvucci., D., Endowing a Cognitive Architecture with World Knowledge, Proceedings of the 36th Annual Meeting of the CogSci Soc., 2014.

10. Frixione, M, Lieto A., Representing concepts in formal ontologies: Compositionality vs. typicality effects, Logic and Logical Philosophy 21 (4) (2012) 391–414, 2012.

11. Sun, R. The CLARION cognitive architecture: Extending cognitive modeling to social simulation. Cognition and multi-agent interaction pp. 79-99, 2006.

12. Lieto, A., Lebiere C., Oltramari, A. The Knowledge Level in Cognitive Architectures: Current Limitations and Possibile Developments, Cognitive Systems Research, Cognitive Systems Research, forthcoming.

13. Lieto, A., Radicioni, D.P., Rho, V., A Common Sense Conceptual Categorization System Integrating Heterogeneous Proxytypes and the Dual Process of Reasoning. In Proc. of IJCAI 2015, AAAI Press, 2015.

14. Lieto, A., D.P. Radicioni, V. Rho, Dual PECCS: A Cognitive System for Conceptual Representation and Categorization Journal of Experimental and Theoretical Artificial Intelligence, Taylor and Francis. doi: http://dx.doi.org/10.1080/0952813X.2016.1198934, 29 (2), 2017.

15. Franklin, S., F. Patterson, F., The Lida architecture: Adding new modes of learning to an intelligent, autonomous, software agent, 764-1004, 2006.

16. Snaider, S. Franklin, Vector Lida, Procedia Computer Science 4,188-203, 2014.

17. Vernon, D, Metta, G., Sandini, G,., A survey of artificial cognitive systems:Implications for the autonomous development of mental capabilities in computational agents, IEEE Transactions on Evolutionary Computation 11 (2), 2007.

18. Langley, P., Laird, J., Rogers, S., Cognitive architectures: Research issues and challenges, Cognitive Systems Research 10 (2), 141-160, 2009.

19. Newell, A. Unified theories of cognition. Cambridge, MA: Harvard University Press, 1990.

20. Anderson, J., Betz, J., A hybrid model of categorization, Psychonomic Bulletin & Review 8 (4) 629-647, 2001.

21. Gärdenfors, Conceptual spaces: The geometry of thought, MIT press, 2000.

22. Fellbaum, C. (ed.): WordNet - An Electronic Lexical Database. Cambridge, Massachusetts. MIT Press, 1998.

23. Fillmore, C.J.,The case for case. Bach, E., Harms, T. eds. Universals in Linguistic Theory. New York: Rinehart and Wiston, 1968.

24. Masolo, C., Borgo, S., Gangemi, A., Guarino, N., & Oltramari, A., Wonderweb deliverable d18, ontology library (final). ICT project, 33052, 2003.

25. Ball, J., Rodgers, S., Gluck, K., Integrating act-r and cyc in a large-scale model of language comprehension for use in intelligent agents, in: AAAI workshop, pp. 19-25, 2004.

26. Lenat, D., Cyc: A large-scale investment in knowledge infrastructure, Communications of the ACM 38 (11), 33-38, 1995.

27. Lieto A., A Computational Framework for Concept Representation in Cognitive Systems and Architectures: Concepts as Heterogeneous Proxytypes, Procedia Computer Science, 41, 6–14, http://dx.doi.org/10.1016/j.procs.2014.11.078, 2014.

28. Frixione M, Lieto A., Towards an Extended Model of Conceptual Representations in Formal Ontologies: A Typicality-Based Proposal, Journal of Universal Computer Science 20 (3) (2014) 257–276, 2014.

29. Lieto, A., Chella, A., Frixione, M., Conceptual Spaces for Cognitive Architectures: A Lingua Franca for Different Levels of Representation, In Biologically Inspired Cognitive Architectures, 19 (2), 1-9, 2017.

30. Lieto, A., Radicioni, D.P., Rho, V., Mensa, E., Towards a Unifying Framework for Conceptual Representation and Reasoning in Cognitive Systems, in Intelligenza Artificiale, forthcoming.

31. Evans, J., Frankish, K., In two minds: Dual processes and beyond, Oxford University Press, 2009.

32. Stanovich, K., West, R., Advancing the rationality debate, Behavioral and brain sciences, 23 (05), 701-717, 2000.

Behavioral Insights on Influence of Manual Action on Object Size Perception

Annalisa Bosco, Patrizia Fattori

Department of Pharmacy and Biotechnology
University of Bologna
Bologna, Italy
patrizia.fattori@unibo.it

Abstract— **Visual perception is one of the most advanced function of human brain. The study of different aspects of human perception currently contributes to machine vision applications. Humans estimate the size of objects to grasp them by perceptual mechanisms. However, the motor system is also able to influence the perception system. Here, we found modifications of object size perception after a reaching and a grasping action in different contextual information. This mechanism can be described by the Bayesian model where action provides the likelihood and this latter is integrated with the expected size (prior) derived from the stored object experience (Forward Dynamic Model). Beyond the action-modulation effect, the knowledge of subsequent action type modulates the perceptual responses shaping them according to relevant information required to recognize and interact with objects. Cognitive architectures can be improved on the basis of these processings in order to amplify relevant features of objects and allow to robot/agent an easy interaction with them.**

Keywords—visual perception, object recognition, motor output, human functions, context information.

I. INTRODUCTION

The majority of machine vision and object recognition systems today apply mechanistic or deterministic template matching, edge detection or color scanning approach for identifying different objects in the space and also to guide embodied artificial intelligent systems to interaction with them. However, fine disturbances in the workspace of a robot can lead to failures, and thus slow down their performance in identification, recognition, learning and adapting to noisy environment, compared to human brain. To go beyond these limitations robots with intelligent behavior must be provided with a processing architecture that allows them to learn and reason about responses to complex goals in a complex world.

The starting point for the development of such intelligent systems is the study of human behavior. Humans frequently estimate the size of objects to grasp them. In fact, when performing an action, our perception is focused towards object visual properties that enable us to execute the action successfully. However, the motor system is also able to influence perception, but only few studies reported evidence for action-induced visual perception modifications related to

hand movements [1–4]. For example, the orientation perception is enhanced during preparation of grasping action compared with a pointing for which object orientation is not important [5,6]. This "enhanced perception" is triggered by the intention to grasp and is important to examine objects with the maximum possible accuracy. If we consider the effects of action execution on visual perception of object features, there is ample evidence for visual perception changes in the oculomotor system, but little is known about the perceptual changes induced by different types of hand movements. In order to evaluate the influence of different hand movement on visual perception, we tested a feature-specific modulation on object size perception after a reaching and a grasping action in different contexts.

II. MATERIALS AND METHODS

A total of 16 right-handed subjects (11 females and 5 males, ages 21–40 years; with normal or corrected-to-normal vision) took part in the experiment. The experiment was performed by two groups of participants. One group of 8 subjects performed the Prior knowledge of action type experiment (PK condition) and the other group (8 participants) performed the No prior knowledge of action type (NPK condition). All subjects were naive to the experimental purpose of the study and gave informed consent to participate in the experiment. Procedures were approved by the Bioethical Committee of the University of Bologna and were in accordance with the Declaration of Helsinki.

A. Apparatus and Setup

Participants were seated in an environment with dim background lighting and viewed a touchscreen monitor (ELO IntelliTouch, 1939L), which displayed target stimuli within a visible display of 37.5 X 30.0 cm. To stabilize head position, the participants placed their heads on a chin rest located 43 cm from the screen, which resulted in a visual field of 50 x 40 deg. The display had a resolution of 1152 X 864 pixels and a frame rate of 60 Hz (15,500 touch points/cm^2). For stimulus presentation, we used MATLAB (The MathWorks) with the Psychophysics toolbox extension [7]. The stimuli were white, red and green dots with a radius of 1.5 mm and 10 differently

sized white, red and green bars all 9 mm large and whose length was: 30, 33.6, 37.2, 40.8, 44.4, 48, 51.6, 55.2, 58.8, 62.4 mm. Hand position was measured by a motion capture system (VICON, 460; frequency of acquisition 100 Hz), which follows the trajectory of the hand in three dimensions by recording infrared light reflection on passive markers.

Participants performed 10 blocks of 10 trials each. Each trial consisted of three successive phases: *Pre-size perception*, *Reaching or Grasping movement*, *Post-size perception* (Fig. 1). In Pre-size perception and Post-size perception phases (phases 1 and 3), a white or green central fixation target stayed on the screen for 1 s; then, a white or green bar was presented, for 1 s, 12 deg on the left or on the right side of the central fixation target and, after an acoustic signal, it disappeared. The participants were required to manually indicate the perceived horizontal size of the bar. All participants indicated the bar sizes by keeping the hand within the starting hand position square and the distance between subject eyes. In the Reaching or Grasping movement phase (phase 2), after 1 s, the white or green central fixation point was followed by a bar identical for position and size to that of phases 1 and 3. Participants were required to perform a reaching (closed fist) or grasping action (extension of thumb and index fingers to "grasp" the extremities of the bar) towards the bar after the acoustic signal, respectively. The type of actions was instructed by the colors of the stimuli (fixation point and bar). In fact, if the color of the stimuli was white, participants were required to perform a reaching movement whereas, if the color was green, they were required to perform a grasping movement. In PK condition, the color of fixation points and bars was white or green in all three phases of trial and in this way the participants knew in advance (from phase 1) which action type was required in the movement phase (phase 2). In the NPK condition, the sequence of the three phases was identically structured as in the PK condition, but we changed colors of fixation points and bars from white/green to red in phases 1 and 3. The color of stimuli during phase 2 remained white or green according to the movement type, reaching or grasping respectively. By this color manipulation, participants could not know in advance the successive action type.
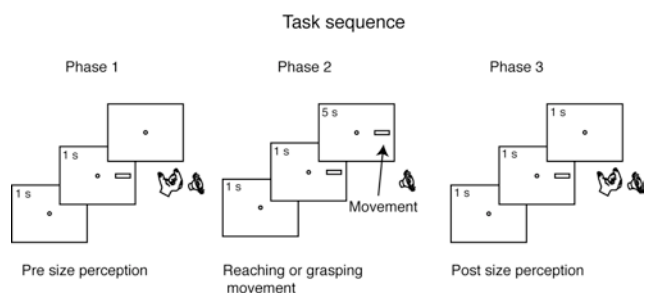

Fig. 1. *Task sequence. Circle = fixation point, Rectangle = stimulus, Hand = size indication by manual report, Speaker = acoustic signal to respond.*

B. Data analysis

After data collection, finger position data were interpolated at 1000 Hz, then data were run though a fifth-order Butterworth low-pass filter [8]. For data processing and analysis, we wrote custom software in MATLAB to compute the distance between index and thumb markers during the pre- and post-

manual estimation phases. Grip aperture was calculated considering trial intervals in which the velocities of the index and thumb markers remained <5 mm/s [8]. Grip aperture was defined as maximum distance within this interval. To evaluate the effect of different hand movement on size perception, we compared the manual perceptual responses before the movements with those after the movements by using two-tailed t-test with independent samples.

To evaluate the magnitude of the effect of NPK and PK conditions on perceptual responses before the movement we calculated the average difference between the two responses and we compared the responses between the two conditions by a t-test analysis. We extracted relevant features from the perceptual responses before the movement and we used them to predict the NPK and PK conditions. For this purpose, we performed a linear-discriminant analysis (LDA-based classifier), as implemented in Statistics and Machine Learning toolbox (Matlab). Pre movement manual responses of NPK and PK conditions were vertically concatenated to build the feature space composed by 958 trials. Fivefold cross-validation was performed by using the 80% of trials for training and the 20% for testing the data, so to ensure that the classifier was trained and tested on different data. Specifically, the classifier was trained on the training subset and the obtained optimal decision criteria was implemented on the testing subset. The prediction results were obtained for this testing subset. This procedure was repeated 5 times, so that all trials were tested and classified basing on models learned from the other trials. The prediction results for all the trials were taken together to give an averaged prediction result with standard deviation. We considered statistically significant the accuracies which standard deviations did not cross the theoretical chance level of 50%. We used a LDA classifier as decoder of the two conditions. LDA finds linear combination of features that characterizes or separates two or more classes of objects or event [9,10]. In fact, LDA explicitly attempts to model the difference between the classes of data. For all statistical analyses the significant criterion was set to P < 0.05.

III. RESULTS

We assessed the effects of action execution on perceptual responses comparing the single subject responses before the movement with those after the movement and calculating the difference between these. Fig. 2 shows these differences in grey color for reaching movement on the horizontal axis compared with those of grasping movement on vertical axis. Filled and empty circles are referred to PK and NPK condition, respectively. The majority of subjects fell below the diagonal suggesting that they corrected the perceptual estimation after the grasping movements with respect to the reaching movement. In particular, they perceived significantly smaller the bars after a grasping movement with respect to a reaching movement (P < 0.05). The averaged differences in PK and NPK conditions are reported in Fig. 2 as black and white dots, respectively. Both dots are below the diagonal suggesting that, globally, subjects perceived smaller after a grasping action compared with a reaching action.

To analyze the effect of the NPK and PK conditions on size perception, we focused the analyses on manual size reports before the movement execution (Pre size perception phase). We computed the difference between the Pre size perception reports in PK condition and the Pre size perception reports in NPK condition. This difference allowed to highlight the

amount of change in size perception in the two conditions tested. As it is shown in Fig. 3A, we found that the amount of change in reaching was -11.89 mm ±0.98 mm and in grasping -11.36 mm ±1.08 mm, and in both cases, they were significantly deviated from baseline (t-test, $P < 0.05$). Generally, the subjects tended to perceive smaller the sizes presented in the condition where they were aware about the subsequent action (PK condition) compared with the condition where they were uncertain about the successive movement (NPK condition). To evaluate whether the strength of this effect was due to a perceptual bias or to different neural processings, we used a LDA decoder to classify the manual responses according to the NPK and PK condition (see Material and Methods). In other words, we checked whether we were able to predict the PK and NPK conditions from perceptual responses before the movement execution, as this technique represents a powerful method to reconstruct experimental conditions and functional movements from neural responses using different types of classifiers [11,12]. Fig. 3A shows decoding results as confusion matrix and the corresponding mean accuracy expressed in percentage. We found a good correlation between the real conditions and the decoded conditions, as it is illustrated in Fig. 3B. The accuracies of decoding were significantly higher of 50% (66,8% for PK and 60.54% for NPK) as shown in Fig. 3C.
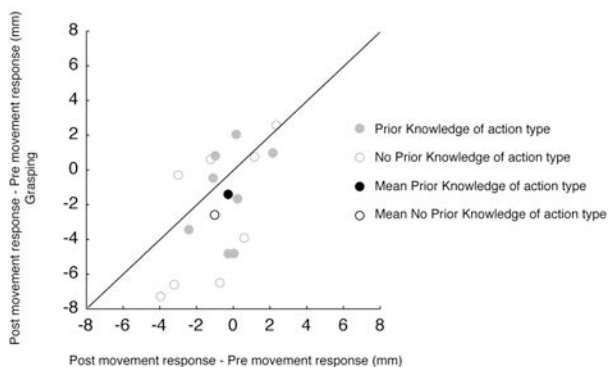

Fig. 2. *Differences between perceptual responses before and after the movement. Filled grey dots are differences in PK condition and empty grey dots are differences in NPK condition. Black and white dots are the mean differences in PK and NPK conditions, respectively.*

IV. DISCUSSION

In the present study, we found direct evidence for a perceptual modification of a relevant feature as object size before and after the execution of two types of hand movement. These changes depended on two factors: the knowledge of the subsequent action type and the type of action executed. Changes in perception were sharpened after a grasping action compared with a reaching. Specifically, subjects perceived objects smaller after a grasping movement than after a reaching movement. The study of action effects exerted by the skeletomotor system on perception has been focused on the evidence that relevant features of objects, such as size or orientation, prime the perceptual system in order to execute a more accurate subsequent grasping movement.

Indeed, Gutteling et al. [5] demonstrated an increased perceptual sensitivity to object orientation during a grasping preparation phase. The effect of action-modulated perception has also been shown to facilitate visual search for orientation.

Bekkering and Neggers [2] analysed the performance of subjects that were required to grasp or point to an object of a certain orientation and color among other objects. They demonstrated that fewer saccadic eye movements were made to wrong orientations when subjects had to grasp the object than point to it. Recently, Bayesian theory has been applied to formalize processes of cue and sensorimotor integration [13,14]. According to this view, the nervous system combines prior knowledge about object properties gained through former experience (prior) with current sensory cues (likelihood), to generate appropriate object properties estimations for action and perception. Hirsinger and coworkers [15], by application of a size-weight illusion paradigm, found that the combination of prior and likelihood for size perception were integrated in a Bayesian way. Their model consisted in a Forward Dynamic Model (FDM) that represented the stored object experience. The FDM output was the experience-based expected size and was referred as the prior. The prior then was integrated with the likelihood, which represented the afferent sensory information about object size. A feedback loop with a specified gain provides the FDM with the final estimate of size, which serves as learning signal for adapting object experience. In the present study, we can apply a similar model for size perception after an action execution. In our case, the objects were visual, not real objects and no haptic feedback was given after the execution of movement. So, the likelihood was represented by the matching of the fingers with the outer border of objects with/or the proprioceptive signals coming from the hand posture that are integrated with the prior.

We found that the knowledge of action type was a factor modulating size perception. In fact, subjects perceived smaller the bars during the condition where they knew the subsequent action (PK) compared with the other condition where they did not know the subsequent action (NPK) for both reaching and grasping. A further demonstration of that was related to the possibility to predict with significant accuracy (>50%) the two conditions from perceptual responses before movement (see Fig. 3B-C). This approach is typical for neural responses and represents a novelty for this type of behavioral variables. The significance of these results is in line with evidence from behavioral research suggesting that motor planning processes increase the weight of visual inputs.
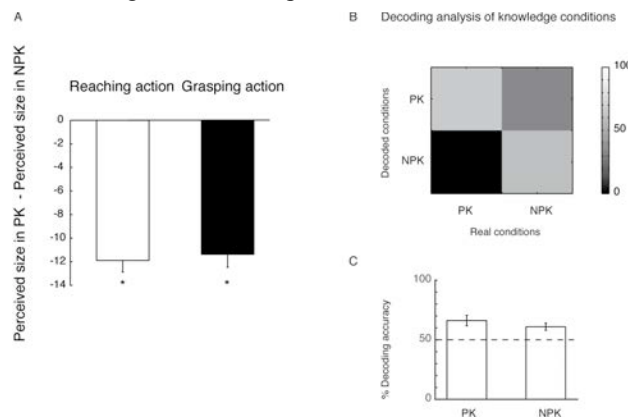

Fig.3. *A, Mean differences of perceptual responses between PK and NPK conditions in reaching and grasping. B, Confusion matrix of decoding results. C, Mean decoding accuracy for classification of NPK and PK conditions. Error bars are standard deviation. *P<0.05, significant level.*

Hand visual feedback has been found to have a greater impact on movement accuracy when subjects prepare their movements with the prior knowledge that vision will be available during their reaches [16,17]. More interestingly, motor preparation facilitates the processings of visual information related to the target of movement. Similarly to Gutteling et al. [5] for object orientation, Wykowska et al. [18] reported that the detection of target size was facilitated during the planning of grasping but not during the planning of pointing. All these studies show the capacity of the brain to modulate the weight of visual inputs and provide an illustration of the importance of the context in visual information processing. In line with all these studies, our findings suggest that the knowledge or not of subsequent movement type defines a context that modulates the perceptual system. When subjects knew the subsequent movement, the perceptual system was within a definite context and perceived object smaller, scaling the measures according to hand motor abilities. In the other case, subjects were in an uncertain context about the successive action, and the perceptual system used different rules to scale the size reports. In both cases, the defined and undefined context can be predicted. All the mechanisms described in the present study could implement models of cognitive architecture of vision-based reaching and grasping of objects located in the peripersonal space of a robot/agent. Additionally, the evidence that the perceptual system is dynamically modulated by contextual information about subsequent movement type can be used to improve cognitive architectures. For example one or multiple focus of attention signals can be sent to the object representation of robot/agent in order to amplify relevant features and at the same time inhibits distractors.

ACKNOWLEDGMENT

REFERENCES

[1] Craighero, L., Fadiga, L., Rizzolatti, G., Umiltà C. Action for Perception A Motor - Visual Attentional Effect. J Exp Psychol Hum Percept Perform. 1999;25: 1673–.

[2] Bekkering H, Neggers SFW. Visual search is modulated by action intentions. Psychol Sci a J Am Psychol Soc / APS. 2002;13: 370–374. doi:10.1111/j.0956-7976.2002.00466.x

[3] Hannus A, Cornelissen FW, Lindemann O, Bekkering H. Selection-for-action in visual search. Acta Psychol (Amst). 2005;118: 171–191. doi:10.1016/j.actpsy.2004.10.010

[4] Fagioli S, Hommel B, Schubotz RI. Intentional control of attention: Action planning primes action-related stimulus dimensions. Psychol Res. 2007;71: 22–29. doi:10.1007/s00426-005-0033-3

[5] Gutteling TP, Kenemans JL, Neggers SFW. Grasping preparation enhances orientation change detection. PLoS One. 2011;6. doi:10.1371/journal.pone.0017675

[6] Gutteling TP, Park SY, Kenemans JL, Neggers SFW. TMS of the anterior intraparietal area selectively modulates orientation change detection during action preparation. J Neurophysiol. 2013;110: 33–41. doi:10.1152/jn.00622.2012

[7] Brainard DH. The Psychophysics Toolbox. Spat Vis. 1997;10: 433–436. doi:10.1163/156856897X00357

[8] Bosco A, Lappe M, Fattori P. Adaptation of Saccades and Perceived Size after Trans-Saccadic Changes of Object Size. J Neurosci. 2015;35: 14448–14456. doi:10.1523/JNEUROSCI.0129-15.2015

[9] Fisher RA. The use of multiple measurements in taxonomic problems. Ann Eugen. 1936;7: 179–188.

[10] McLachlan GJ. Discriminant analysis and statistical pattern recognition. Wiley Intersci. 2004;. ISBN 0-4.

[11] Schaffelhofer S, Agudelo-Toro A, Scherberger H. Decoding a wide range of hand configurations from macaque motor, premotor, and parietal cortices. J Neurosci. 2015;35: 1068–81. doi:10.1523/JNEUROSCI.3594-14.2015

[12] Townsend BR, Subasi E, Scherberger H. Grasp movement decoding from premotor and parietal cortex. J Neurosci. 2011;31: 14386–98. doi:10.1523/JNEUROSCI.2451-11.2011

[13] Körding KP, Wolpert DM. Bayesian decision theory in sensorimotor control. Trends in Cognitive Sciences. 2006. pp. 319–326. doi:10.1016/j.tics.2006.05.003

[14] Van Beers RJ, Wolpert DM, Haggard P. When feeling is more important than seeing in sensorimotor adaptation. Curr Biol. 2002;12: 834–837. doi:10.1016/S0960-9822(02)00836-9

[15] Hirsiger S, Pickett K, Konczak J. The integration of size and weight cues for perception and action: Evidence for a weight-size illusion. Exp Brain Res. 2012;223: 137–147. doi:10.1007/s00221-012-3247-9

[16] Zelaznik HZ, Hawkins B, Kisselburgh L. Rapid visual feedback processing in single-aiming movements. J Mot Behav. 1983;15: 217–236. doi:10.1080/00222895.1983.10735298

[17] Elliott D, Allard F. The utilization of visual feedback information during rapid pointing movements. Q J Exp Psychol A Hum Exp Psychol. 1985;37: 407–425. doi:10.1080/14640748508400942

[18] Wykowska A, Schubö A, Hommel B. How you move is what you see: action planning biases selection in visual search. J Exp Psychol Hum Percept Perform. 2009;35: 1755–1769. doi:10.1037/a0016798

A Role for Action Selection in Consciousness: An Investigation of a Second-Order Darwinian Mind

Robert H. Wortham and Joanna J. Bryson
Dept of Computer Science, University of Bath
Claverton Down, Bath, BA2 7AY, UK
Email: {r.h.wortham, j.j.bryson}@bath.ac.uk

Abstract—We investigate a small footprint cognitive architecture comprised of two reactive planner instances. The first interacts with the world via sensor and behaviour interfaces. The second monitors the first, and dynamically adjusts its plan in accordance with some predefined objective function. We show that this configuration produces a Darwinian mind, yet aware of its own operation and performance, and able to maintain performance as the environment changes. We identify this architecture as a second-order Darwinian mind, and discuss the philosophical implications for the study of consciousness. We use the Instinct Robot World agent based modelling environment, which in turn uses the Instinct Planner for cognition.

BIOLOGICALLY INSPIRED COGNITIVE ARCHITECTURES

From the 1950's through to the 1980's the study of embodied AI assumed a cognitive symbolic planning model for robotic systems — SMPA (Sense Model Plan Act) — the most well known example of this being the Shakey robot project [1]. In this model the world is first sensed and a model of the world is constructed within the AI. Based on this model and the objectives of the AI, a plan is constructed to achieve the goals of the robot. Only then does the robot act. Although this idea seemed logical and initially attractive, it was found to be quite inadequate for complex, real world environments.

In the 1990's Rodney Brooks and others [2] introduced the then radical idea that it was possible to have intelligence without representation [3]. Brooks developed his subsumption architecture as a pattern for the design of intelligent embodied systems that have no internal representation of their environment, and minimal internal state. These autonomous agents could traverse difficult terrain on insect-like legs, appear to interact socially with humans through shared attention and gaze tracking, and in many ways appeared to possess behaviours similar to that observed in animals. However, the systems produced by Brooks and his colleagues could only respond immediately to stimuli from the world. They had no means of focusing attention on a specific goal or of executing complex sequences of actions to achieve more complex behaviours. Biologically inspired approaches are still favoured by many academics, although a wide gap exists between existing implementations and the capabilities of the human mind [4]. Today, the argument persists concerning whether symbolic, sub-symbolic or hybrid approaches are best suited for the creation of powerful cognitive systems [5]. Here we concern ourselves more specifically with action selection as a core component of any useful cognitive architecture.

From Ethology to Robots

Following in-depth studies of animals such as gulls in their natural environment, ideas of how animals perform action selection were originally formulated by Nico Tinbergen and other early ethologists [6], [7]. Reactions are based on pre-determined drives and competences, but depend also on the internal state of the organism [8]. Bryson [9] harnessed these ideas to achieve a major step forwards with the POSH (Parallel Ordered Slipstack Hierarchy) reactive planner and the BOD (Behaviour Oriented Design) methodology, both of which are strongly biologically inspired. A POSH plan consists of a *Drive Collection (DC)* containing one or more *Drives*. Each *Drive (D)* has a priority and a releaser. When the *Drive* is released as a result of sensory input, a hierarchical plan of *Competences*, *Action Patterns* and *Actions* follows. POSH plans are authored, or designed, by humans alongside the design of senses and behaviour modules. An iterative approach is defined within BOD for the design of intelligent artefacts — these are known as agents, or if they are physically embodied, robots.

Kinds of Minds

Daniel Dennett[10] elegantly outlines a high level ontology for the kind of minds that exist in the natural world. At the most basic level, the Darwinian mind produces 'hardwired' behaviours, or phenotypes, based on the genetic coding of the organism. The Skinnerian mind is plastic, and capable of 'ABC' learning — Associationism, Behaviourism, Connectionism. The Popperian mind runs simulations to predict the effect of planned actions, anticipating experience. It therefore permits hypotheses "to die in our head" rather than requiring them to be executed in the world before learning can take place. Finally the Gregorian mind (after the psychologist Richard Gregory) is able to import tools from the cultural environment, for example language and writing. Using these tools enables the Gregorian mind, for example the human mind, to be self-reflective.

However, perhaps the simple Darwinian mind might also be arranged to monitor itself, and in some small and limited sense to be aware of its own performance and act to correct it. Bryson suggests that consciousness might assist in action selection [11], and here we investigate whether action selection achieved through reactive planning might parallel one of the
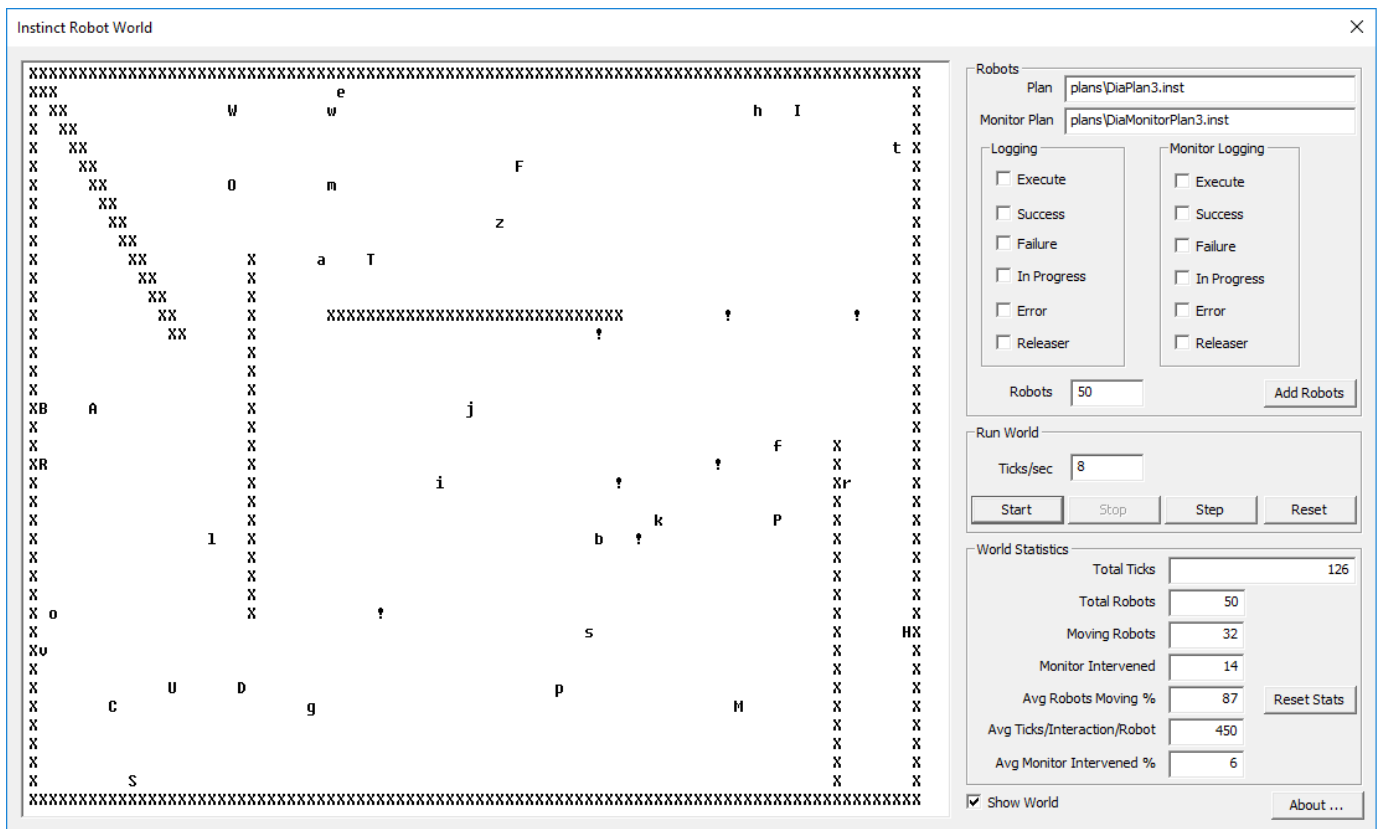
Fig. 1: Screen shot of the Instinct Robot World in operation. Each robot is represented as a single character within the display. Robots are labelled with letters and numbers to distinguish them. When a robot's monitor plan becomes active the robot representation changes to the shriek character (!). The top right section of the screen is used to control the robots and the plans they use. The bottom right section displays statistics about the world as it runs.

commonly accepted characteristics of consciousness; that is to be self-reflective and regulating [12].

Instinct and the Robot World

The *Instinct Planner* [13] is a biologically inspired reactive planner specifically designed for low power processors and embedded real-time AI environments. Written in C++, it runs efficiently on both ARDUINO and MICROSOFT VC++ environments and has been deployed within the R5 low cost maker robot to study AI Transparency [14].

It's unique features are its tiny memory footprint and efficient operation, meaning that it can operate on a low powered micro-controller environment such as ARDUINO. Alternatively, as in this experiment, many planners can run within one application on a laptop PC.

The *Instinct Robot World* is a new agent based modelling tool, shown in Figure 1. This is an open source project and all code and configuration files are available online [1]. Each virtual 'robot' within the Robot World uses an Instinct Planner to provide action selection. Strictly, since these virtual robots are not physically embodied, we should refer to them as agents. However, we have chosen to use 'robot' throughout, as intuitively these cognitive entities appear to be virtually embodied within the Robot World, and this choice of language seems more natural. In the final section of this paper we discuss

[1] http://www.robwortham.com/instinct-planner/

future work where we may realise physical embodiment of this architecture.

The Robot World allows many robots to be instantiated, each with the same reactive plan, or with a variety of plans. The robots each have senses to sense the 'walls' of the environment, and other robots. The reactive plan invokes simple behaviours to move the robot, adjust its speed and direction, or interact with robots that it encounters within the world as it moves. Most importantly for this investigation, each robot also has a second Instinct Planner instance. This planner monitors the first, and is able to modify its parameters based on a predefined plan.

The Instinct Robot World provides statistical monitoring to report on the overall activity of the robots within the world. These include the average percentage of robots that are moving at any one time, the average number of time units (ticks) between robot interactions, and the average amount of time that the monitor planner intervenes to modify the robot plan. We use the Instinct Robot World to investigate the idea of *Reflective Reactive Planning* — one reactive planner driving behaviour based on sensory input and predefined drives and competences, and another reactive planner monitoring performance and intervening to modify the predefined plan of the first, in accordance with some higher level objective. This simple combination of two Darwinian minds, one monitoring the other, might also be considered to be a second-order Darwinian mind.

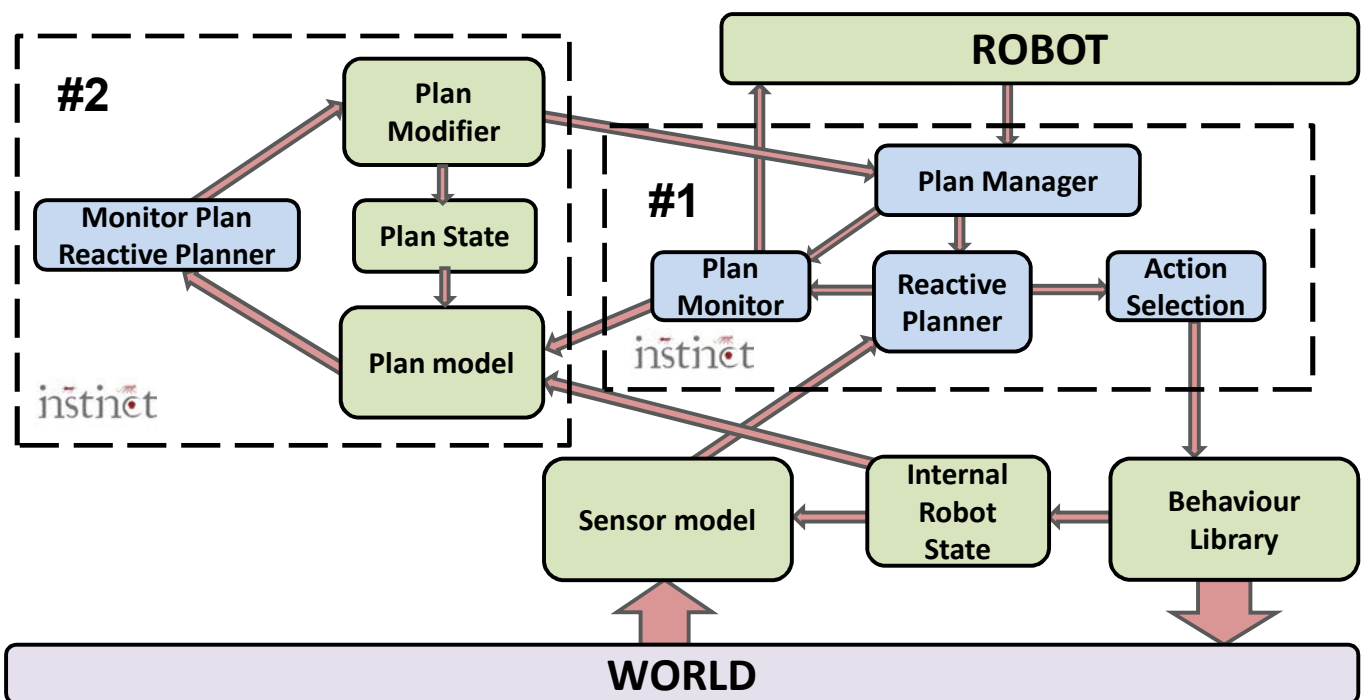# Reflective Reactive Planning - A 2nd Order Darwinian Mind

Fig. 2: Architecture of the second-order Darwinian mind. The robot is controlled by the Instinct Reactive Planner, as it interacts with the Sensor model and Behaviour Library. In turn, a second instance of Instinct monitors the first, together with the Internal robot state, and dynamically modifies parameters within the robot's planner.The overall effect is a robot that not only reacts to its environment according to a predefined set of goals, but is also to modify that interaction according to some performance measure calculated within the Plan model.

CONJECTURES

We expect that second-order Darwinian minds will outperform first order minds when the environment changes, because the monitor planner is concerned with achieving higher order objectives, and modifies the operation of the first planner to improve its performance. We also hypothesise that this architecture will remain stable over extended periods of time, because by restricting ourselves to the reactive planning paradigm we have reduced the number of degrees of freedom within which the architecture must operate, and previous work shows that first-order minds produce reliable control architectures [14]. Finally, we expect that such a second-order system should be relatively simple to design, being modular, well structured and conceptually straightforward.

METHODS

Figure 2 shows the Reflective Reactive Planning architecture implemented within the Instinct Robot World, and controlling the behaviour of each robot within that world. The robot plan has the following simple objectives, each implemented as an Instinct Drive.

- Move around in the environment so as to explore it.
- Avoid objects i.e. the walls marked as 'X' in Figure 1.
- Interact when another robot is 'encountered' i.e. when another robot is sensed as having the same coordinates within the grid of the Robot World. This interaction causes the robot to stop for 200 clock cycles or 'ticks'.

While the robot is in the 'Interacting' state it is shown as a shriek character (!) within the Robot World display. Once the robot has interacted its priority for interaction decreases, but ramps up over time. This may be likened to most natural drives, for example mating, feeding and the need for social interaction.

The Monitor Plan is designed to keep the robot exploring when it is overly diverted from social interactions. It achieves this by monitoring the time between interactions. If, over three interactions, the average time between interactions reduces below 1000 ticks, then the Monitor Planner reduces the priority of the interaction Drive. After 1000 ticks the priority is reset to its original level. We might use alternative intentional language here to say that the Monitor Planner 'notices' that the robot is being diverted by too many social interactions. It then reduces the priority of those interactions, so that the robot is diverted less frequently. After some time the Monitor Planner ceases to intervene until it next notices this situation re-occurring.

The Robot World is populated with varying numbers of robots (2, 3, 5, 10, 20, 50, 100, 200, 500, 1000), and for each number the experiment is run twice, once with a monitor plan, and once without. For each run, the environment is allowed to run for some time, typically about 10 minutes, until the reported statistics have settled and are seen to be no longer changing over time.

The results are most elegantly and succinctly presented as simple graphs. Firstly, the average number of robots moving at any one time within the world is shown in Figure 3. In both
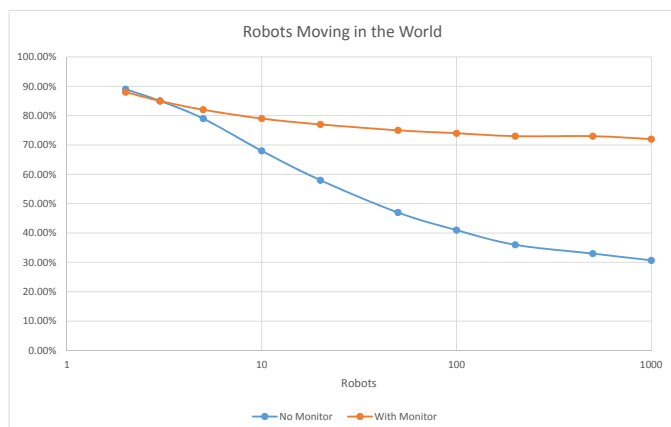

Fig. 3: This graph shows the average percentage number of robots that are moving at any one time within the world, for a given total number of robots in the world. It can be seen that the addition of the monitor plan maintains more robots moving as the number of robots increases. Note the log scale for robots in world.

cases, as the number of robots within the world increases, the amount of time that the robot spends moving reduces. However the Monitor Planner acts to reduce the extent of this reduction from 60% to less than 20% over the full range of two to a thousand robots within the world. Similarly, in Figure 4 we see that as more robots are introduced into the world, the average time between interactions naturally reduces. However, the action of the Monitor Planner progressively limits this reduction, so that with 1000 robots the time between interactions is almost trebled, from 310 to 885 ticks per interaction. Interestingly, in both these graphs we see smooth curves both with and without the action of the monitor plan.
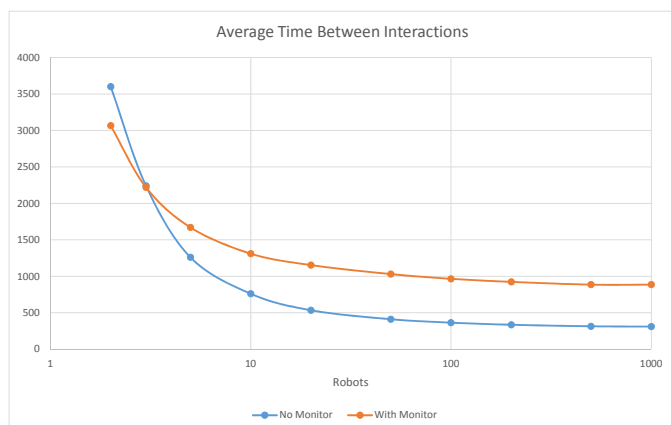

Fig. 4: This graph shows the average time between robot interactions, both with and without the monitor plan. The addition of the monitor plan reduces the variance in interaction time as robot numbers vary. Again, note the log scale.

The final graph, Figure 5 also shows a smooth, sigmoid like increase in activation of the Monitor Planner as the number of robots increases, plotted on a logarithmic scale.
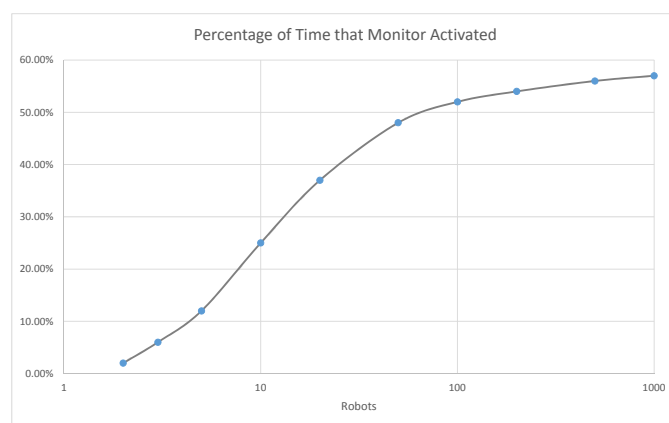

Fig. 5: This graph shows the average percentage number of robots whose monitor plan is activated at any one time, for a given number of total robots in the world. Note the log scale.

The Instinct Robot World was found to be a stable, reliable platform for our experiments, and the results it achieved were repeatable. The application is single threaded, and so uses only one core of the CPU on the laptop PC on which it was run. Nevertheless, it was possible to simulate 1000 robots with both reactive planners active operating in the world at the rate of 70 clock cycles (ticks) per second.

DISCUSSION

From the results we can see that by using a second Instinct instance to monitor the first, we can achieve real-time learning within a tiny-footprint yet nevertheless symbolic cognitive architecture. In addition, since this learning modifies parameters from a human designed plan, the learning can be well understood and is transparent in nature. This contrasts strongly with machine learning approaches such as neural networks that typically learn offline, are opaque, and require a much larger memory workspace. Despite the stochastic nature of the environment, the performance graphs show smooth curves over a wide range of robot populations.

This relatively simple experiment also provides further fuel for the fire concerning the philosophical discussion of the nature of consciousness. Critics may say that when we use the intentional stance [15] to describe the behaviour of the Monitor Planner as 'noticing' something, we are merely using metaphor. They might argue that there is in fact no sentience doing any noticing, and in fact the only 'noticing' that is happening here is us noticing the behaviour of this human designed mechanism, which itself is operating quite without any sentience and certainly without being conscious [16]. But that is to miss the point. We are not claiming that this architecture is conscious in the human or even significant sense of the word, merely that our architecture is inspired by one aspect of how biological consciousness appears to operate. However, having shown that this architecture can indeed provide adaptive control, and drawing on the knowledge that gene expression produces behaviours which can be modelled using reactive planning, we might also consider whether consciousness in animals and humans may indeed

arise from complex hierarchical mechanisms. These mechanisms are biologically pre-determined by genetics, and yet in combination yield flexible, adaptive systems able to respond to changing environments and optimise for objective functions unrelated to the immediate competences of preprogrammed behavioural responses. This is not to argue for some kind of emergence [17], spooky or otherwise, but more simply to add weight to the idea that the 'I' in consciousness is nothing more than an internal introspective narrative, and such a narrative may be generated by using hierarchical mechanisms that notice one another's internal states, decision processes and progress towards pre-defined (phenotypic) objectives.

We could certainly envisage a much grander architecture, assembled at the level of reactive planners, using maybe hundreds or thousands of planners each concerned with certain objectives. Many of these planners may be homeostatic in nature, whilst others would be concerned with the achievement of higher level objectives. We must remember that planners merely coordinate action selection, and say nothing about how sensor models may be formed, nor how complex behaviours themselves may be implemented. However, all dynamic architectures need some kind of decision centric 'glue' to bind them together, and reactive planning seems to be a useful candidate here, as evidenced by practical experiment and biological underpinning.

Machine transparency is a core element of our research. We have shown elsewhere [14] that reactive planners, particularly the Instinct Planner, are able to facilitate transparency. This is due to the human design of their plans, and the ability to gather meaningful symbolic information about internal system state and decision processes in real-time as the planner operates. This ability to inspect the operation of the architecture may assist designers in achieving larger scale cognitive implementations. Equally importantly, transparency is an important consideration for users and operators of intelligent systems, particularly robots, and this is highlighted in the EPSRC Principles of Robotics [18].

The human brain does not run by virtue of some elegant algorithm. It is a hack, built by the unseeing forces of evolution, without foresight or consideration for modularity, transparency or any other good design practice. If we are to build intelligent systems, the brain is not a good physical model from which we should proceed. Rather, we should look at the behaviours of intelligent organisms, model the way in which these organisms react, and then scale up these models to build useful, manageable intelligent systems.

Whilst our Reflective Reactive Planner is a very simple architecture, it does share many of the characteristics cited for architectures that are worthy of evaluation, such as efficiency and scalability, reactivity and persistence, improvability, and autonomy and extended operation [19]. We hope that our work with reactive planners might strengthen the case for their consideration in situations where decision centric 'glue' is required.

Conclusions and Further Work

We have shown that a second-order Darwinian mind may be constructed from two instances of the Instinct reactive planner. This architecture, which we call Reflective Reactive Planning, successfully controls the behaviour of a virtual robot within a simulated world, according to pre-defined goals and higher level objectives. We have shown how this architecture may provide both practical cognitive implementations, and inform philosophical discussion on the nature and purpose of consciousness.

The Instinct Robot World is an entirely open source platform, available online. We welcome those interested in agent based modelling, cognitive architectures generally, and reactive planning specifically, to investigate these technologies and offer suggestions for new applications and further work. One possibility might be to apply this architecture to the Small Loop Problem [20], a specific challenge for biologically inspired cognitive architectures.

We continue to develop robot applications for the Instinct Planner, together with the Instinct Robot World. We are investigating the use of a small robot swarm to build a physically embodied version of this experiment. To this end, we are currently working with the University of Manchester's Mona robot[2].

References

[1] N. J. Nilsson, "Shakey the Robot," SRI International, Technical Note 323, Tech. Rep., 1984.

[2] C. Breazeal and B. Scassellati, "Robots that imitate humans," *Trends in Cognitive Sciences*, vol. 6, no. 11, pp. 481–487, 2002.

[3] R. A. Brooks, "Intelligence Without Representation," *Artificial Intelligence*, vol. 47, no. 1, pp. 139–159, 1991.

[4] A. V. Samsonovich, "Extending cognitive architectures," *Advances in Intelligent Systems and Computing*, vol. 196 AISC, pp. 41–49, 2013.

[5] A. Lieto, A. Chella, and M. Frixione, "Conceptual Spaces for Cognitive Architectures : A lingua franca for different levels of representation," *Biologically Inspired Cognitive Architectures*, no. November, pp. 1–9, 2016. [Online]. Available: http://dx.doi.org/10.1016/j.bica.2016.10.005

[6] N. Tinbergen, *The Study of Instinct*. Oxford, UK: Oxford University Press, 1951. [Online]. Available: https://books.google.co.uk/books?id=WqZNkgEACAAJ

[7] N. Tinbergen and H. Falkus, *Signals for Survival*. Oxford: Clarendon Press, 1970. [Online]. Available: http://books.google.co.uk/books?id=5LHwAAAAMAAJ

[8] J. J. Bryson, "The study of sequential and hierarchical organisation of behaviour via artificial mechanisms of action selection," 2000, M.Phil. Thesis, University of Edinburgh.

[9] ——, "Intelligence by design: Principles of modularity and coordination for engineering complex adaptive agents," Ph.D. dissertation, MIT, Department of EECS, Cambridge, MA, June 2001, AI Technical Report 2001-003.

[10] D. C. Dennett, *Kinds of minds: Towards an understanding of consciousness*. Weidenfeld and Nicolson, 1996.

[11] J. J. Bryson, "A Role for Consciousness in Action Selection," in *Proceedings of the AISB 2011 Symposium: Machine Consciousness*, R. Chrisley, R. Clowes, and S. Torrance, Eds. York: SSAISB, 2011, pp. 15—-20.

[12] J. W. Sherman, B. Gawronski, and Y. Trope, *Dual-Process Theories of the Social Mind*. Guilford Publications, 2014. [Online]. Available: https://books.google.co.uk/books?id=prtaAwAAQBAJ

[13] R. H. Wortham, S. E. Gaudl, and J. J. Bryson, "Instinct : A Biologically Inspired Reactive Planner for Embedded Environments," in *Proceedings of ICAPS 2016 PlanRob Workshop*, London, UK, 2016. [Online]. Available: http://icaps16.icaps-conference.org/proceedings/planrob16.pdf

[14] R. H. Wortham, A. Theodorou, and J. J. Bryson, "Robot Transparency : Improving Understanding of Intelligent Behaviour for Designers and Users," in *Proceedings of TAROS 2017*. Guildford, UK: {accepted for publication}, 2017.

[15] D. C. Dennett, *The intentional stance*. MIT press, 1989.

---

[2] http://www.monarobot.uk/

[16] P. O. A. HAIKONEN, "Consciousness and Sentient Robots," *International Journal of Machine Consciousness*, vol. 05, no. 01, pp. 11–26, 2013. [Online]. Available: http://www.worldscientific.com/doi/abs/10.1142/S1793843013400027

[17] J. H. Holland, *Emergence: From Chaos to Order*, ser. Popular science / Oxford University Press. Oxford University Press, 2000. [Online]. Available: https://books.google.co.uk/books?id=VjKtpujRGuAC

[18] M. Boden, J. Bryson, D. Caldwell, K. Dautenhahn, L. Edwards, S. Kember, P. Newman, V. Parry, G. Pegman, T. Rodden, T. Sorell, M. Wallis, B. Whitby, and A. Winfield, "Principles of robotics," The United Kingdom's Engineering and Physical Sciences Research Council (EPSRC), April 2011, web publication.

[19] P. Langley, J. E. Laird, and S. Rogers, "Cognitive architectures: Research issues and challenges," *Cognitive Systems Research*, vol. 10, no. 2, pp. 141–160, jun 2009. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1389041708000557

[20] O. L. Georgeon and J. B. Marshall, "The small loop problem: A challenge for artificial emergent cognition," *Advances in Intelligent Systems and Computing*, vol. 196 AISC, pp. 137–144, 2013.

Architectural Requirements for Consciousness

Ron Chrisley
Centre for Cognitive Science (COGS),
Sackler Centre for Consciousness Science,
and Department of Informatics
University of Sussex
Brighton, United Kingdom
Email: ronc@sussex.ac.uk

Aaron Sloman
Department of Computer Science
University of Birmingham
Birmingham, United Kingdom
Email: axs@cs.bham.ac.uk

Abstract—This paper develops, in sections I-III, the virtual machine architecture approach to explaining certain features of consciousness first proposed in [1] and elaborated in [2], in which particular qualitative aspects of experiences (qualia) are proposed to be particular kinds of properties of components of virtual machine states of a cognitive architecture. Specifically, they are those properties of components of virtual machine states of an agent that make that agent prone to believe the kinds of things that are typically believed to be true of qualia (e.g., that they are ineffable, immediate, intrinsic, and private). Section IV aims to make it intelligible how the requirements identified in sections II and III could be realised in a grounded, sensorimotor, cognitive robotic architecture.

I. INTRODUCTION

Those who resist the idea of a computational, functional, or architectural explanation of consciousness will most likely concede that many aspects surrounding consciousness *are* so explicable (the so-called "easy problems" of consciousness [3]), but maintain that there are core aspects of consciousness having to do with phenomenality, subjectivity, etc. for which it is Hard to see how a computational explanation could proceed. A typical way of characterising this "Hard core" of consciousness employs the concept of *qualia*: "the introspectively accessible, phenomenal aspects of our mental lives" [4]. Surely there can be no computational explanation of qualia?

This paper develops the virtual machine architecture approach to explaining certain features of consciousness first proposed in [1] and elaborated in [2], in which qualia, understood as particular qualitative aspects of experiences, are proposed to be particular kinds of properties of components of virtual machine states of a cognitive architecture. Specifically, they are those properties of components of virtual machine states of agent A that make A prone to believe:

1) That A is in a state S, the aspects of which are knowable by A directly, without further evidence (immediacy);
2) That A's knowledge of these aspects is of a kind such that only A could have such knowledge of those aspects (privacy);
3) That these states have these aspects intrinsically, not by virtue of, e.g., their functional role (intrinsicness);
4) That these aspects of S cannot be completely communicated to an agent that is not A (ineffability).

Our emphasis on beliefs concerning these four properties (immediacy, privacy, intrinsicness and ineffability), follows the analysis in [5] in taking these properties to be central to the concept of *quale* or *qualia*. But whereas [5] understands this centrality to imply that the properties themselves are conditions for falling under the concept, we understand their centrality only in their role of *causally determining* the reference of the concept. Roughly, qualia are not whatever has those four properties; rather, qualia are whatever is (or was) the cause of our qualia talk. And if we do know anything about the cause of our qualia talk, it is this: it makes us prone to believe that we are in states that have those four properties.

A crucial component of our explanation, which we call the Virtual Machine Functionalism (VMF) account of qualia, is that the propositions 1-4 need not be true in order for qualia to make A prone to believe those propositions. In fact, it is arguable that nothing could possibly render all of 1-4 true simultaneously [5]. But on our view, this would not imply that there are no qualia, since for qualia to exist it is only required that that agents that have them be prone to believe 1-4, which can be the case even when some or all of 1-4 are false.

It is an open empirical question whether, in some or all humans, the properties underlying the dispositions to believe 1-4 have a unified, systematic structure that would make them a single cause, and that would thereby make reference to them a useful move in providing a causal explanation of such beliefs. Is "qualia" more like "gold", for which there was a well-defined substance that was the source of mistaken, alchemical talk and beliefs about gold? Or is "qualia" more like "phlogiston", in that there is no element that can be identified as the cause of the alchemists' mistaken talk and beliefs that they expressed using the world "phlogiston"? These are empirical questions; thus, according to the VMF account of qualia, it is an open empirical question whether qualia exist in any particular human. By the same token, however, it is an open *engineering* question whether, independently of the human case, it is possible or feasible to design an artificial system that a) is also prone to believe 1-4 and b) is so disposed because of a unified, single cause. Thus, it is an open engineering question whether an artificial system can be constructed to have qualia. This paper goes some way toward

getting clear on how one would determine the answer to that engineering question.

Section II notes the general requirements that must be in place for a system to believe 1-4, and then sketches very briefly, in section III, an abstract design in which the propensities to believe 1-4 can be traced to a unified virtual machine structure, underwriting talk of such a system having qualia.

II. GENERAL ARCHITECTURAL REQUIREMENTS FOR HAVING QUALIA

General requirements for meeting constraints 1-4 include being a system that can be said to have beliefs and propensities to believe, as well as what those properties themselves require. Further, having the propensities to believe 1-4 in particular requires the possibility of having beliefs about oneself, one's knowledge, about possibility/impossibility, and other minds. At a minimum, such constraints require a cognitive architecture with reactive, deliberative and meta-management components [1], with at least two layers of meta-cognition: (i) detection and use of various states of internal virtual machine components; and (ii) holding beliefs/theories about those components.

III. A QUALIA-SUPPORTING DESIGN

A little more can be said about the requirements that 1-4 might impose on a cognitive architecture.

1) A propensity to believe in immediacy (1) can be explained in part as the result of the meta-management layer of a deliberating/justifying but resource-bounded architecture needing a basis for terminating deliberation/justification in a way that doesn't itself prompt further deliberation or justification.

2) A propensity to believe in privacy (2) can be explained in part as the result of a propensity to believe in immediacy (1), along with a policy of *normally* conceiving of the beliefs of others as making evidential and justificatory impact on one's own beliefs. To permit the termination of deliberation and justification, some means must be found to discount, at some point, the relevance of others' beliefs, and privacy provides prima facie rational grounds for doing this.

3) A propensity to believe in intrinsicness (3) can also be explained in part as the result of a propensity to believe in immediacy, since states having the relevant aspects non-intrinsically (i.e., by virtue of relational or systemic facts) would be difficult to rectify with the belief that one's knowledge of these aspects does not require any (further) evidence.

4) An account of a propensity to believe in ineffability (4) requires some nuance, since unlike 1-3, 4 is in a sense true, given the causally indexical nature of some virtual machine states and their properties, as explained in [2]. However, properly appreciating the truth of 4 requires philosophical sophistication, and so its truth alone cannot explain the conceptually primitive

propensity to believe it; some alternative explanations must be offered, but it is not possible to do so here.

IV. COGNITIVE ARCHITECTURE, NOT COGNITIVIST ARCHITECTURE?

Given the anti-cognitivist, anti-representational, anti-symbolic, embodied, enactivist, etc. inclinations of many in the EUCognition community, the foregoing may be hard to accept given its free use of representational and computational notions such as belief, deliberation, justification, etc. The rest of this paper, then, is an attempt at an in-principle sketch of how one can have a grounded, dynamic, embodied, enactive(ish) cognitive architecture that nevertheless supports the notions of belief, inference, meta-belief, etc. that this paper has just maintained are necessary for the subjective, qualia aspect of consciousness, if not all aspects of consciousness.

This motivation is not strictly (that is, philosophically) required, for two reasons:

- First, our self-appointed philosophical opponents do not claim that the "easy problems" of consciousness cannot be solved physicalistically, or even computationally. Thus, in giving our explanation of the "Hard core" of consciousness, qualia, we can help ourselves to any of the capacities that are considered to fall under the "easy problems", which is the case for all of the requirements we identified in sections II and III.

- Second, an aspect a of a cognitive architecture A can be *of the same kind* as an aspect b of a distinct cognitive architecture B, even if B is capable of the sorts of beliefs mentioned in 1-4 because of possessing b, and A, despite having a, is not capable of having those sorts of beliefs. On our account, A might still have qualia by virtue of having a; this is why our account does not, despite appearances, over-intellectualize qualia, and is instead consistent with, e.g., the empirical possibility that animals and infants have qualia.

However, showing how architectures that *do* have the kinds of beliefs mentioned in 1-4 *can* be constructed out of grounded sensorimotor components is required if we are to achieve any understanding of what a system that is *incapable* of having those beliefs would have to be like for it to nevertheless warrant ascription of qualia.

This section (that is, the rest of this paper) will not have much to say about consciousness or qualia *per se*. Furthermore, the sketched architectures are likely not optimal, feasible, or even original. That there is some better way to solve the task that we use for illustrative purposes below is not to the point. The architectures and task are intended merely to act as a proof-of-concept, as a bridge between the kind of robotic systems that many in the EUCognition community are familiar or comfortable with, and the kind of robotic cognitive architecture that we have argued is required for qualia.

A. Robotic architecture, environment and task

Consider a robot that is static except that it can move its single camera to fixate on points in a 2D field. The result

R of fixating on point (x, y) is that the sensors take on a particular value s out of a range of possible values S. That is, $R(x, y) = s \in S$.

The visual environment is populated by simple coloured polygons, at most one (but perhaps none) at each fixation point (x, y). This visual environment is static during trials, although it may change from trial to trial.

The robot has learned a map M that is a discrete partition of S into a set of categories or features F (e.g., a self-organising feature map): $M(s) = f_i \in F$. In general, M is always applied to the current sensory input s, thus activating one of the feature nodes or vectors. For example, f_1 might be active in those situations in which the robot is fixating on a green circle, f_2 might be active in those situations in which the robot is fixating on a red triangle, etc.

Suppose also that the robot has the ability to detect the occurrence of a particular auditory tone. After the tone is heard, a varying visual cue (for example, a green circle) appears in some designated area of the field (the upper left corner say). The robot's task (for which it will be rewarded) is to perform some designated action (e.g. say "yes") if and only if there is something in the current visual environment (other than in the designated cue area) whose feature map classification matches that of the cue, that is: say "yes" iff $\exists (x, y) : M(R(x, y)) = M(cue)$.

There are, of course, many strategies the robot could use to perform this task. For illustrative reasons, we will consider three.

B. Strategy One: Exhaustive search of action space

The first strategy is an exhaustive search of action space. The robot performs a serial exhaustive search of the action space $R(x, y)$, stopping to say "yes" if at any point $M(R(x, y)) = M(cue)$. This requires motor activity, and is likely to take a relatively long time to perform, although it requires no "offline" preparation time. It is a "knowledge-free" solution.

C. Strategy Two: Exhaustive search of virtual action space

The second strategy is to perform an exhaustive search of a *virtual* action space.

1) Strategy Two, Version 1: Prior to hearing the tone, the robot learns a forward model E_w from points of fixation (x, y) to expected sensory input s at the fixated location: $E_w(x, y) = s \in S$. After the tone and presentation of the cue, the robot then performs a serial exhaustive search of the expectation space $E_w(x, y)$, stopping if at any point $M(E_w(i, j)) = M(cue)$. The robot then fixates on (i, j), and if $M(R(i, j)) = M(cue)$, then it says "yes". Otherwise, the search of the expectation space resumes. As this search is for the most part virtual, only occasionally requiring action (assuming E is reasonably accurate), this will be much faster than the first strategy.

2) Strategy Two, Version 2: If the idea of an exhaustive serial search of the expectation space is not considered neurally plausible enough, a a second version of the second strategy

could employ a kind of content-addressable search (following ideas first presented in [6]). The difference between cue and $E(x, y)$ (or between $M(cue)$ and $M(E(x, y))$; see below) can be used as a differentiable error signal, permitting gradient descent reduction of error not in weight w space, but in visual space (which is here the same as fixation space and action space). That is (hereafter re-writing (x, y) as u), the robot can apply the Delta rule, changing u proportionally to the partial derivative of the error with respect to u:

$$\Delta u = \mu \frac{\partial [\frac{1}{2} (cue - E(u))^2]}{\partial u} \qquad .$$

Since the task question is primarily about matching one of the cue categories f_i and not the cue itself, this process requires changing the robot's virtual fixation point u according to the above equation, and then checking to see if $M(E(u)) = M(cue)$. If not, u is again updated according to the Delta rule. Alternatively, one could measure the error directly in terms of differences in feature map (M) output; then the Delta rule would prescribe:

$$\Delta u = \mu \frac{\partial [\frac{1}{2} (M(cue) - M(E(u)))^2]}{\partial u} \qquad .$$

In either case, this process should eventually arrive at a value u' that is a minimum in error space, although the number of iterations of changes to u required to do so will depend on a number of factors, including μ, which itself is constrained by the "spikiness" of the error space with respect to fixation points. This could result in many changes to u, but as such changes are virtual, rather than actual changes in robot fixation point, they can be performed much faster than real-time.

Standard problems with local minima apply: the fixed point in u/error space where the derivative is zero may not only not be a point for which *actual* error is zero (that is, where $M(R(u')) = M(cue)$); it may not even be a point for which *expected* error is zero (that is, where $M(E(u')) = M(cue)$). Nonetheless, u' can serve as a plausible candidate solution, which can be checked by having the robot fixate on u' via $R(u')$. If a match ($M(R(u')) = M(cue)$) is not achieved, standard neural network methods for handling local minima can be applied, if desired, to see if a better result can be obtained.

This second version of the second strategy may in some cases be more efficient than the first variation, in that it is non-exhaustive. But both verisons of the second strategy buy online performance at the price of prior "offline" exploration of the action space, and the computational costs of learning and memory.

As an aside, we note that the second version of strategy two can be used in conjunction with strategy one (or even the first verison of strategy two), in that it can suggest a heuristically-derived first guess for a real-world (or virtual) search of points

in the vicinity of that guess. In the case of failure, it wouldn't be useful as it stands, it seems; since E is deterministic, when asked for a second guess after the failure of the first, strategy two would give the same recommendation again. However, it should be noted that the gradient descent method is dependent on an initial guess u, and derives candidate solutions as modifications to u. Therefore, it will give different u' answers if a different initial u is selected to seed the gradient descent process, with the new u' corresponding to the local error minimum that is closest to the new u seed chosen. Thus, search of the entire virtual (or actual) fixation point (u) space can be reduced, in theory, to a virtual search of the much smaller space of error basins in u-space. To prevent wasteful duplication of effort, there would have to be some way for the network to consider only previously-unconsidered seeds; perhaps inhibition of previously-considered seeds could achieve this.

D. Strategy Three: Learning a mapping from mappings to cues

A third strategy builds on the second strategy by employing a form of reflection or meta-cognition to guide search more efficiently. As with the second strategy, an expectational, forward model E_w is used. Note that for any given kind of cue (node or reference vector in the range of the feature map M), we can define the set P_{cue} to be all those parameter (weight) sets w for E that yield a forward model that contains at least one expectation to see that cue. That is, $P_{cue} = \forall w : \exists(x, y) : M(E_w(x, y)) = cue$.

With a network distinct from the one realising E, the robot can learn an approximation of P_{cue}. That is, the robot can learn a mapping F_{cue} from weight sets for E to $\{1,0\}$, such that $F_{cue}(w) = 1$ iff $w \in P_{cue}$. Generalising, the robot can learn a mapping F from cues and weight sets for E to $\{1,0\}$, such that $F(cue, w) = 1$ iff $w \in P_{cue}$. That is, F is a network that, given a vector w and a cue, outputs a 1 only if w parameterises a forward model E_w for which there is at least one fixation point (x, y) such that E_w "expects" cue as input after performing $R(x, y)$.

Given this, a third strategy for performing the task is to simply input the current E parameter configuration w and the cue into F, and say "yes" iff $F(w, cue) = 1$ (or, if one prefers, make the probability of saying "yes" proportional to $F(w, cue)$).

Like strategy two, strategy three spends considerable "off-line", pre-task resources for substantial reductions in the time expected to complete the online task. However, unlike both strategy one and strategy two, this third strategy answers the task question *directly*: it determines whether the existential condition of the task question holds without first finding a particular fixation point that satisfies the property that the task condition (existentially) quantifies over. A drawback of this is that the robot cannot, unlike with strategy two, check its answer in the real world (except by essentially performing strategy one). But as it is essentially a lookup computation, it is very fast: no search, even virtual, is required. Admittedly, this is only useful if F can be learned, and if the space

is not too spiky (nearby values for w should, in general, imply nearby values for $M(E(u))$). Nevertheless, the the third strategy would be useful for situations in which immediate, gist-based action is required.

E. Metamappings as metacognition

As explained at the beginning of this section, we have taken these efforts to incrementally motivate the architecture in strategy three in order to illustrate how a grounded, sensori-motor based system can merit ascription of the kinds of metacognitive abilities that we have proposed are necessary for crediting a system with qualia:

- In effect, the forward model E confers on the the system belief-like states, in the form of expectations of what sensor values will result from performing a given action. These (object, not meta) belief-like states are total in that a given state vector w yields an E_w that manifests a range of such expectational beliefs, each concerning a different action or point of fixation.
- Similarly, the forward model F confers on the the system meta-belief-like states, in that they indicate which total, object belief states have a particular content property. (Note that the meta beliefs are not of the form, for some particular w, u and cue: w manifests the belief that (or represents that) $M(R(u)) = M(cue)$. Rather, they are of the form, for some particular w and cue: $\exists u : w$ manifests the belief that $M(R(u)) = M(cue)$.)

Meta-belief is not only an explicit requirement for the kind of qualia-supporting architecture outlined in section II and III; it also opens to door to the further requirements of inference, deliberation and sensitivity to logical relations. To see how, consider one more addition to the architecture we arrived at when discussing strategy three. As with the individual nodes in the feature map, we can define the set P_{c_1,c_2} to be all those parameter sets w that yield a forward model that contains at least one expectation to see c_1 *and* one expectation to see c_2; that is, $P_{c_1,c_2} = \forall w : \exists(u_1)(u_2)$ such that:

- $M(E_w(u_1)) = c_1$; and
- $M(E_w(u_2)) = c_2$

With another network G distinct from E (and F), the robot can learn an approximation of P_{c_1,c_2}: $G(w, c_1, c_2) = 1$ iff $w \in P_{c_1,c_2}$. That is, G is a network that:

- takes the parameters w of E as input
- outputs a 1 only if those parameters realise a forward model E_w for which:
  - $\exists u_1 : M(E_w(u_1)) = c_1$; and
  - $\exists u_2 : M(E_w(u_2)) = c_2$;

Note that it is a logical truth that $w \in P_{c_1,c_2} \rightarrow w \in P_{c_1}$. It follows that there is a logical relation between G and F; specifically, it should be true that $G(w, c_1, c_2) = 1 \rightarrow F(w, c_1) = 1$. Assuming F and G are themselves reasonably accurate, the robot could observe and learn this regularity. But because F and G are only approximations, there might actually be cases (values of w) where they are inconsistent (where $G(w, c_1, c_2) = 1$ but $F(w, c_1) = 0$). That such a

mismatch constitutes error could be built into the architecture, yielding an error signal not between expected and empirical object-level states of affairs, but between a logical norm and the empirical relation between meta-belief states that should respect that norm.

How should the robot respond to this error signal, which indicates the violation of a logical norm? In the case of empirical, object-level error, the direction of fit is from model to world, so error should be reduced by changing the model (*pace* Friston and active inference[7]). But in this case, the error is not between model and world, but between two models of the world: should the robot modify F, or G, or both?

Although it seems unlikely that there is a general, situation-independent answer to this question, one could certainly imagine another iteration of reflection and complexity that would enable a robot to learn an effective way for handling such situations. For example, F and G could be part of a network of experts, in which a gating network learns the kinds of situations in which any F/G mismatch should be resolved in $F's$ favour, and which in $G's$. But there is also the possibility of a resolution due to *implicit* architectural features that do not constitute a semantic ascent. An interactive activation competition between F and G might, for example, always be resolved in $F's$ favour simply because F has fewer inputs and parameters than G – or vice versa. Such a system could be understood as having a belief, albeit an implicit one, that object-level beliefs manifested in F are always more reliable, justified, etc. than beliefs manifested in G. And again, a sophisticated architecture, although continuous with the kinds of systems considered so far, could observe instances of this regularity, and thus learn the regularity itself. It could thus come to know (or at least believe) that it always takes F-based judgements to be more reliable than (logically conflicting) G-based ones. From the error signal that is produced whenever they disagree the system could come to believe that G and F are logically related. The crucial point is that the robot has the essentials of a notion of logical justification and logical consistency of its own beliefs. It could use a systematic mismatch between G and F as evidence that G requires more learning, or indeed use that mismatch as a further error signal to guide learning in G, or even E itself.

One could ask: why go to all this trouble? Couldn't all of this have been motivated simply by considering a robot that contains two forward models, E and E', that are meant to have the same functionality, but which might contingently evolve in such a way that they disagree on some inputs? The answer is yes, and no. Yes, an instance of being a logically-constrained cognizer is that one eschews believing P and $\neg P$. But no: to start with such an architecturally unmotivated example would not serve to make a general case for how meta-beliefs as a whole could get going in a sensorimotor grounded architecture. For one thing, it doesn't suggest how sensitivity to logical relations between sub-networks could assist in inference. But with what has been presented concerning the conjunctive cue network G, it is possible to understand, for example, how there could be a *disjunctive* cue network H that maps weights w to

1 only if either one or the other of its associated cues c_1 and c_2 is in the range of E_w. Such a network having output of 1 for w, in the face of $F(w, c_1) = 0$, would allow the network to infer that $F(w, c_2)$ *should* be 1, and use that in place of computing $F(w, c_2)$ explicitly, or to generate an error signal if $F(w, c_2) \neq 1$, etc.

Further sophistication, conferring even more of the kinds of metacognitive abilities discussed in sections II and III, could be added by not just allowing the robot to observe the holding or not of various logical relations in its own beliefs, but by giving it the ability to take action on the meta-level, and allow such actions to be guided, as on the object level, by expectations realized in forward models on the meta-level. Such forward models would not manifest expectations about how sensory input would be transformed by performing this or that movement, but rather how object-level forward models such as E would change, if one were to perform this or that operation on their parameter sets w. To give a trivial example, there might be a primitive operation N that could be performed on a forward model's parameters that had the effect of normalizing those parameters. A network's understanding of this might be manifested in a network J such that $J(w_1, N) = norm(w_1)$, $J(w_2, N) = norm(w_2)$, etc., with J being consulted when normalization is being considered as a possible meta-action to perform.

V. Conclusion

The "Hard core" of consciousness is meant to be qualia, but sections I-III argue that qualia, understood as the underlying phenomenon (if any) that explains qualia-talk and qualia-beliefs, might be explicable in terms of phenomena that are considered to fall under the "easy problems" of consciousness. The speculations of section IV fall short of closing the loop started in sections II and III, but they hopefully give one an idea how a grounded sensorimotor robotic cognitive architecture could merit attribution of such features as having beliefs and having beliefs about beliefs. In particular, it is hoped that some substance has been given to the possibility of such an architecture being able to employ concepts such as justifcation, deliberation and consistency.

References

[1] A. Sloman and R. Chrisley, "Virtual machines and consciousness," *Journal of Consciousness Studies*, vol. 10, pp. 4–5, 2003.

[2] R. Chrisley and A. Sloman, "Functionalism, revisionism, and qualia," *APA Newsletter on Philosophy and Computers*, vol. 16, pp. 2–13, 2016.

[3] D. Chalmers, *The Conscious Mind: In Search of a Fundamental Theory*. Oxford: Oxford University Press, 1996.

[4] M. Tye, "Qualia," in *The Stanford Encyclopedia of Philosophy*, winter 2016 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2016.

[5] D. Dennett, "Quining qualia," in *Consciousness in Contemporary Science*, A. J. Marcel and E. Bisiach, Eds. Oxford: Oxford University Press, 1988, pp. 42–77.

[6] R. Chrisley, "Cognitive map construction and use: A parallel distributed processing approach," in *Connectionist Models: The Proceedings of the 1990 Connectionist Models Summer School*, D. Touretzky, J. Elman, T. Sejnowski, and G. Hinton, Eds. San Mateo: Morgan Kaufmann, 1990.

[7] K. Friston, T. FitzGerald, F. Rigoli, P. Schwartenbeck, J. O'Doherty, and G. Pezzulo, "Active inference and learning," *Neuroscience & Biobehavioral Reviews*, vol. 68, pp. 862 – 879, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0149763416301336

Section 2: Short Papers

Human-Aware Interaction:
A Memory-inspired Artificial Cognitive Architecture

Roel Pieters[1], Mattia Racca[1], Andrea Veronese[1] and Ville Kyrki[1]

Abstract— In this work we aim to develop a human-aware cognitive architecture to support human-robot interaction. Human-aware means that the robot needs to understand the complete state of the human (physical, intentional and emotional) and interacts (actions and goals) in a human-cognitive way. This is motivated by the fact that a human interacting with a robot tends to anthropomorphize the robotic partner. That is, humans project a (cognitive, emotional) mind to their interactive partner, and expect a human-like response. Therefore, we intend to include procedural and declarative memory, a knowledge base and reasoning (on knowledge base and actions) into the artificial cognitive architecture. Evaluation of the architecture is planned with a Care-O-Bot 4.

I. INTRODUCTION

As the western world is aging, solutions have to be found that ensure the current high-quality welfare state for the future. This research aims to assess the suitability of robotics for assistance and care. Such human-robot interaction should foremost be safe, intuitive and user-friendly. This implies that the robot must understand the person's tasks, intentions and actions, and must include a knowledge base for information storage and reasoning.

II. PERCEPTION: INTENTION AND TASK MODELING

In order to provide assistance, the general state of the human, as well as the task should be known. Human attention can be used to understand a person's intentions and the task he/she is engaged in. By detecting the head pose of the human and projecting this into a 3D point cloud of the environment, a weighted attention map can be generated (Fig. 1-left). Segmenting this map returns the object of interest and can be used to determine which task the person is engaged in [1]. Additionally, by actively gathering information (e.g., the robot asking questions) a model of the task can be learned (Fig. 1-right). This decision making problem under uncertain conditions can be modeled as a partially observable Markov decision process (POMDP). By solving the POMDP, the robot can refine the task model, supervise the task execution and provide assistance for the next phase [2].

III. COGNITIVE MODELING: MEMORY AND REASONING

The knowledge base is divided in declarative memory (semantic and episodic facts) and procedural memory (action library). Semantic facts is general knowledge to represent the beliefs, relations and intentions of the world, of humans and of objects. Episodic memory describes information about events and instances that occurred, e.g., what, where and when an event happened. The action library contains primitives and sequences of tasks available to the robot. For example, the task model is encoded as declarative knowledge and describes the intention and relation between states (phases) in a task. Moreover, it can also be described by an action sequence and event sequence (episodic knowledge). Reasoning over the knowledge base allows for fact checking, relation assessment and event comparison, and can be used for future predictions (internal simulation). Reasoning over the action library allows to reuse, adapt and augment actions and action sequences for different tasks.

IV. SYMBOLIC TASK PLANNING AND EXECUTION

The main function of the symbolic task planner is to generate a suitable plan by checking if the task was experienced in the past (episodic memory in the knowledge base) and how (procedural memory in the action library). Missing information for a generated plan is obtained from perception and reasoning over the knowledge base and the action library. For example, actions take arguments that apply to internal variables and functions (e.g., object pose, speech recognition). High level execution ensures that the planned task is executed appropriately and the instructed goal is achieved (Fig. 2).

V. ROSE AND CARE-O-BOT 4

The proposed developments are part of the interdisciplinary research project ROSE (Robots and the Future of Welfare Services[2]) which aims to study the social and psychological aspect of service robotics. In particular, one aim of
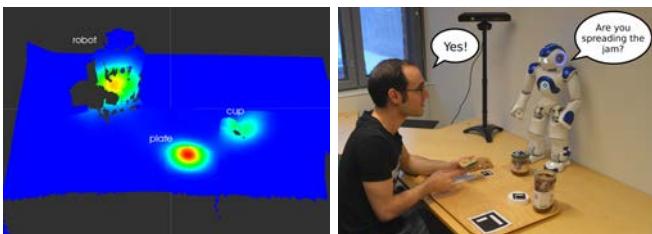

Fig. 1: Left: Weighted attention map that returns three objects of interest, the plate received most interest (red). Right: Task modeling scenario. A person is making a sandwich while a NAO robot observes and asks questions to build a task model for assistance.
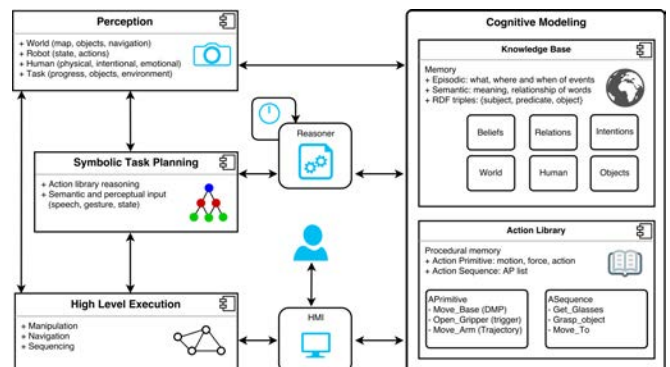

Fig. 2: Artificial cognitive architecture for human-aware interaction.

[1]All authors are with School of Electrical Engineering, Aalto University, Finland. Corresponding author: roel.pieters@aalto.fi

[2]http://roseproject.aalto.fi/en/

this project is to investigate the requirements for social HRI with elderly people and how these should be integrated in practice. This applies for both the technological requirements (i.e., what capabilities and algorithms are necessary) as well as the social requirements (i.e., what does the user want). The Care-O-Bot 4 will be used for human-robot interaction studies and evaluation of the proposed artificial cognitive architecture.

REFERENCES

[1] A. Veronese, M. Racca, R. Pieters, and V. Kyrki, "Action and intention recognition from head pose measurements," 2017 (in preparation).
[2] M. Racca, R. Pieters, and V. Kyrki, "Active information gathering for task modeling in hri," 2017 (in preparation).

The Role of the Sensorimotor Loop for Cognition

Bulcsú Sándor, Laura Martin, Claudius Gros

Institute for Theoretical Physics

Goethe University Frankfurt

Frankfurt a.M., Germany

Email: http://itp.uni-frankfurt.de/∼gros/

Abstract—**Locomotion is most of the time considered to be the result of top-down control commands produced by the nervous system in response to inputs received via sensory organs from the environment. Locomotion may arise alternatively when attracting states are stabilized in the combined dynamical space made up by the brain, the body and the environment. Cognition is embodied in this case within the sensorimotor loop, viz self-organized. Using a physics simulation environment we show that self-organized locomotion may result in complex phase spaces which include limit cycle corresponding to regular movements and both strong and partially predictable chaos describing explorative behavior.**

I. Introduction

We used the LPZRobots physics simulation environment [1] to investigate the occurrence of self-organized embodiment in robots for which sensation is confined to propio-sensation. The 'brain' of the robot, consisting of a single controlling neuron per actuator, receives sensory information only regarding the actual position $x_i^{(a)}$ of the actuators $i = 1, 2, 3$, which are in turn translated via

$$x_i^{(t)} = R \left[2y(x_i) - 1 \right] , \tag{1}$$

to a target position $x_i^{(t)}$ for the i-th actuator (compare Fig. 1). R denotes here the (rescaled) radius of the spherical robot and $y(x_i) = 1/(1 + \exp(-x_i))$ the firing rate of the controlling neuron. The membrane potential x_i is determined via

$$\dot{x}_i = -\Gamma x_i + \frac{w_0}{2R} \left(x_i^{(a)} + R \right) - z_0 \sum_{j \neq i} u_j \varphi_j y(x_j) \tag{2}$$

by the relaxation constant Γ, by the coupling $w_0 > 0$ to the proprio-sensory reading of $x_i^{(a)}$, and with $(-z_0) < 0$ by the inhibition it receives from the other two neurons. The interneural inhibition is dynamically modulated presynaptically by a mechanism known as short-term synaptic plasticity (STSP) [2], which we model as [3]:

$$
\begin{array}{llll}
\dot{u} & = & \frac{U(y) - u}{T_u} & U(y) & = & 1 + (U_{\max} - 1)y \\
\dot{\varphi} & = & \frac{\Phi(u,y) - \varphi}{T_\varphi} & \Phi(u,y) & = & 1 - \frac{uy}{U_{\max}} .
\end{array}
$$

Both the effective Ca^{2+} concentration u and the fraction of available vesicles φ of neurotransmitters relax to unity in the absence of a presynaptic input y, which, when present, tends to increase/decrease $u \to U_{\max}$ and $\varphi \to 0$ respectively.

We note that STSP is well known to change synaptic efficiencies transiently by up-to fifty percent on time scales of a few hundred milliseconds, as defined by T_u and T_φ. These are also the time scales which are relevant for locomotion.

STSP does not induce any long-lasting traces (modifications of the synaptic strength), being hence a fully transient form of plasticity which tends to destablize fixpoint attractors.

II. Autonomous mode switching

The here considered robot moves only, as an entity comprised of body and controlling neurons, when embedded within the environment. Locomotion corresponds then to self-stabilizing attractors in the combined phase space of the controlling neural network, of the body and of the environmental degrees of freedom it couples to [4].

Our robot may engage in a rich palette of regular motion patterns, as illustrated in Fig. 2, which are stable either for distinct sets of internal parameters, such as the bare synaptic weights w_0 and z_0, or simultaneously. Autonomous mode switching corresponding to a rollover from one to another basin of attraction occurs regularly in the latter case upon collision with either an external object, or with another robot. We note, importantly, that limit-cycles corresponding to regular motion, as shown in Fig. 2, are continuously degenerate with respect to the direction and/or to the center of propagation.

III. Explorative Chaos

Explorative behavior arises when the synaptic weights w_0 and z_0 are set such that chaotic attractors are formed within the
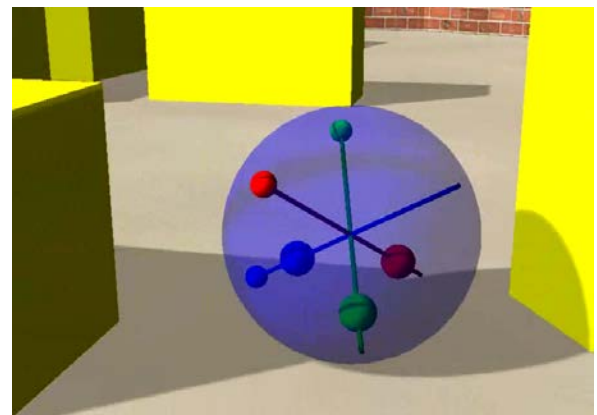

Fig. 1. The simulated robot contains three weights (red, green and blue) moving along perpendicular rods within a movable sphere. The position of the three weights is controlled respectively by a single neuron (see Eqs. (1) and (2)). The small balls at the end of the respective rods are guides to the eye. [video]
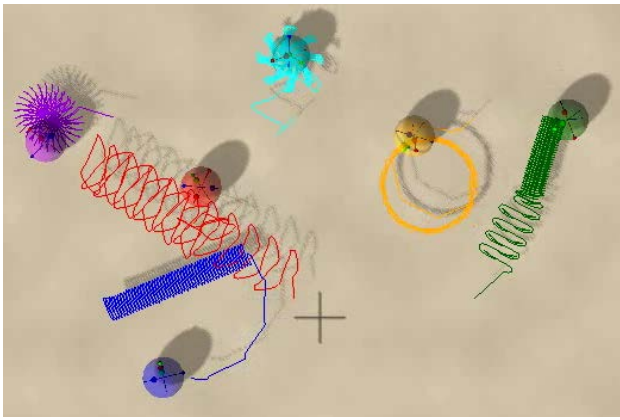
Fig. 2. Six color-coded copies of the sphere robot starting each with slightly different synaptic weights w_0 and z_0. The pink, cyan and yellow robots perform various types of circular and star-like motions, with the red, blue and green robots staring to move with finite translational velocities. For the parameters of the blue and of the green robot two limit cycles coexist. Both robots undergo collisions (the blue colliding with the red robot and the green with the yellow robot), which induce transitions from one to the other attracting state. A previous collision of the red with the pink robot resulted (just) in a direction reversal. [video]

Fig. 4. Within a structured environment the robot starts to push blocks around in a seemingly 'playful' manner. [video]

sensorimotor loop. We note that noise is absent for the simulations shown in Fig. 3, with the seemingly random wandering of the robot resulting exclusively from the chaotic nature of the underlying attractor. Two types of chaotic attractors may be stabilized in addition, denoted respectively as strong and as partially predictable chaos [5].

IV. PLAYFUL LOCOMOTION

Morphological computation [6], [7], [8] may occur when the body plays a central role in cognition. For a test of this concept we have situated the sphere robot in a structured environment, as shown in Fig. 4, containing movable blocks. One observes that our three-neuron robot starts to engage in

a seemingly 'playful' manner with its environment, pushing blocks around by bumping into individual objects repeatedly. This occurs, from a dynamical systems point of view, when the robot switches upon collisions back and forth between stable chaotic motion and another weakly unstable, or alternatively as in Fig. 3, stable coexisting limit-cycle attractor describing regular locomotion.

V. CONCLUSION

The sphere robot does neither perform any form of knowledge acquisition with its brain consisting of only three neurons, nor does its 'cognitive system' dispose of higher-level internal drives or motivations. The explorative behavior observed in Figs. 3 and 4 can be explained on the contrary fully in terms of dynamical systems theory. Taking a philosophical perspective our simulated robots hence demonstrate that it is in general impossible for an external observer to deduce reliably the internal settings and motivations of an acting cognitive system.

Fig. 3. Two color-coded copies of the sphere robot exploring a maze. The motions can be classified as strongly chaotic for the cyan robot and as partially predictable chaos for the blue robot [5]. The blue robot switches to another locomotion mode after colliding with the wall. The resulting radius of the circular mode is, however, too large for the maze and it can follow it hence only transiently. [video]
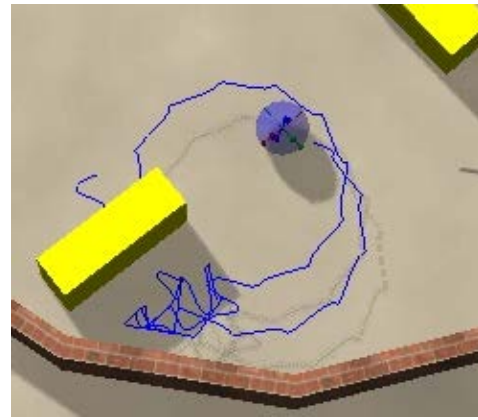
REFERENCES

[1] R. Der and G. Martius, *The Playful Machine: Theoretical Foundation and Practical Realization of Self-Organizing Robots.* Springer Science & Business Media, 2012, vol. 15.
[2] M. V. Tsodyks and H. Markram, "The neural code between neocortical pyramidal neurons depends on neurotransmitter release probability," *Proceedings of the National Academy of Sciences*, vol. 94, no. 2, pp. 719–723, 1997.
[3] L. Martin, B. Sándor, and C. Gros, "Closed-loop robots driven by short-term synaptic plasticity: Emergent explorative vs. limit-cycle locomotion," *Frontiers in Neurorobotics*, vol. 10, 2016.
[4] B. Sándor, T. Jahn, L. Martin, and C. Gros, "The sensorimotor loop as a dynamical system: How regular motion primitives may emerge from self-organized limit cycles," *Frontiers in Robotics and AI*, vol. 2, p. 31, 2015.
[5] H. Wernecke, B. Sándor, and C. Gros, "How to test for partially predictable chaos," *arXiv preprint arXiv:1605.05616*, 2016.
[6] V. C. Müller and M. Hoffmann, "What is morphological computation? on how the body contributes to cognition and control," *Artificial Life*, to be published.
[7] R. Pfeifer and J. Bongard, *How the body shapes the way we think: a new view of intelligence.* MIT press, 2006.
[8] R. Der and G. Martius, "Self-organized behavior generation for musculoskeletal robots," *Frontiers in Neurorobotics*, vol. 11, p. 8, 2017.

Two Ways (Not) To Design a Cognitive Architecture

David Vernon[†]

Carnegie Mellon University Africa

Rwanda

Email: vernon@cmu.edu

Abstract—In this short paper, we argue that there are two conflicting agendas at play in the design of cognitive architectures. One is principled: to create a model of cognition and gain an understanding of cognitive processes. The other is practical: to build useful systems that have a cognitive ability and thereby provide robust adaptive behaviour that can anticipate events and the need for action. The first is concerned with advancing science, the second is concerned with effective engineering. The main point we wish to make is that these two agendas are not necessarily complementary in the sense that success with one agenda may not necessarily lead, in the short term at least, to useful insights that lead to success with the other agenda.

I. INTRODUCTION

There are two aspects to the goal of building a cognitive robot [1]. One is to gain a better understanding of cognition in general — the so-called synthetic methodology — and the other is to build systems that have capabilities that are rarely found in technical artifacts (i.e. artificial systems) but are commonly found in humans and some animals. The motivation for the first is a principled one, the motivation for the second is a practical one. Which of these two aspects you choose to focus on has far-reaching effects on the approach you will end up taking in designing a cognitive architecture. One is about advancing science and the other is more about effective engineering. These two views are obviously different but they are not necessarily complementary. There is no guarantee that success in designing a practical cognitive architecture for an application-oriented cognitive robot will shed any light on the more general issues of cognitive science and it is not evident that efforts to date to design general cognitive architectures have been tremendously successful for practical applications.

The origins of cognitive architectures reflects the former principled synthetic methodology. In fact, the term cognitive architecture can be traced to pioneering research in cognitivist cognitive science by Allen Newell and his colleagues in their work on unified theories of cognition [2]. As such, a cognitive architecture represents any attempt to create a theory that addresses a broad range of cognitive issues, such as attention, memory, problem solving, decision making, and learning, covering these issues from several aspects including psychology, neuroscience, and computer science, among others. A cognitive architecture is, therefore, from this perspective at least, an over-arching theory (or model) of human cognition.

It continues today under the banner of artificial general intelligence, emphasizing human-level intelligence. The term cognitive architecture is employed in a slightly different way in the emergent paradigm of cognitive science where it is used to denote the framework that facilitates the development of a cognitive agent from a primitive state to a fully cognitive state. It is a way of dealing with the intrinsic complexity of a cognitive system by providing a structure within which to embed the mechanisms for perception, action, adaptation, anticipation, and motivation that enable development over the systems life-time. Nevertheless, even this slightly different usage reflects an endeavour to construct a viable model that sheds light on the natural phenomenon of cognition.

From these perspectives - cognitivist and emergent - a cognitive architecture is an abstract meta-theory of cognition and, as such, focusses on generality and completeness (e.g. see [3]). It reflects Krichmar's first aspect of the goal of building a cognitive robot: to gain a better understanding of cognition in general [1]. We draw from many sources in shaping these architectures. They are often encapsulated in lists of desirable features (sometimes referred to as desiderata) or design principles [4], [5], [6], [7]. A cognitive architecture schema is not a cognitive architecture: it is a blueprint for the design of a cognitive architecture, setting out the component functionality and mechanisms for specifying behaviour. It describes a cognitive architecture at a level of abstraction that is independent of the specific application niche that the architecture targets. It defines the necessary and sufficient software components and their organization for a complete cognitive system. The schema is then instantiated as a cognitive architecture in a particular environmental niche. This, then, is the first approach to designing a cognitive architecture (or a cognitive architecture schema). We refer to it as *design by desiderata*.

The second approach is more prosaic, focussing on the practical necessities of the cognitive architecture and designing on the basis of user requirements. We refer to this as *design by use case*. Here, the goal is to create an architecture that addresses the needs of an application without being concerned whether or not it is a faithful model of cognition. In this sense, it is effectively a conventional system architecture, rather than a cognitive architecture per se, but one where the system exhibits the required attributes and functionality, typically the ability to autonomously perceive, to anticipate the need for actions and the outcome of those actions, and
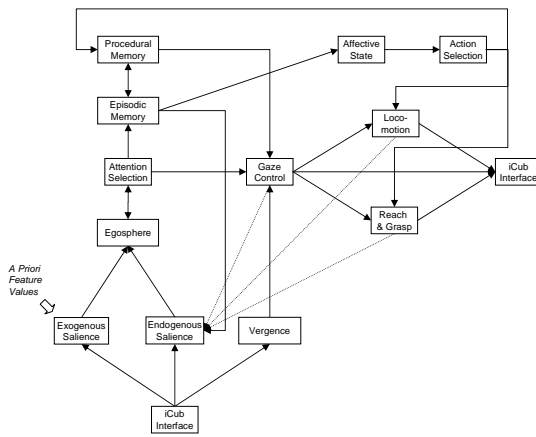
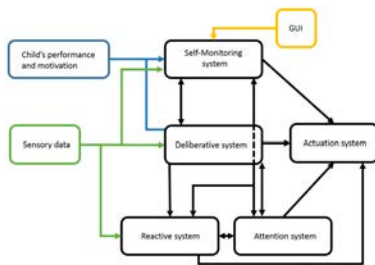Fig. 1. The iCub cognitive architecture (from [9]).

Fig. 2. Project DREAMs cognitive architecture (from [11]).

robot research platform, and the DREAM system architecture with its cognitive controller (2) [10], [11] which was designed by use case [12] for use in Robot-Enhanced Therapy targetted at children with autism spectrum disorder. The former comprises components that reflect generic properties of a cognitive system; the latter comprises several functional components that directly target the needs of therapists who can control the cognitive architecture through a GUI.

II. CONCLUSION

There are two ways not to design a cognitive architecture. If your focus is on creating a practical cognitive architecture for a specific application, you should probably not try to do so by attempting to instantiate a design guided by desiderata; you are probably better off proceeding in a conventional manner by designing a system architecture that is driven by user requirements, drawing on the available repertoire of AI and cognitive systems algorithms and data-structures. Conversely, if your focus is a unified theory of cognition — cognitivist or emergent — then you should probably not try to do so by developing use-cases and designing a matching system architecture. You are likely to miss some of the key considerations that make natural cognitive systems so flexible and adaptable, and it is unlikely that you will shed much light on the bigger questions of cognitive science.

REFERENCES

[1] J. L. Krichmar, "Design principles for biologically inspired cognitive architectures," *Biologically Inspired Cognitive Architectures*, vol. 1, pp. 73–81, 2012.
[2] A. Newell, *Unified Theories of Cognition*. Cambridge MA: Harvard University Press, 1990.
[3] J. E. Laird, C. Lebiere, and P. S. Rosenbloom, "A standard model of the mind: Toward a common computational framework across artificial intelligence, cognitive science, neuroscience, and robotics." *AI Magazine*, vol. In Press, 2017.
[4] R. Sun, "Desiderata for cognitive architectures," *Philosophical Psychology*, vol. 17, no. 3, pp. 341–373, 2004.
[5] J. L. Krichmar and G. M. Edelman, "Principles underlying the construction of brain-based devices," in *Proceedings of AISB '06 - Adaptation in Artificial and Biological Systems*, ser. Symposium on Grand Challenge 5: Architecture of Brain and Mind, T. Kovacs and J. A. R. Marshall, Eds., vol. 2. Bristol: University of Bristol, 2006, pp. 37–42.
[6] J. E. Laird, "Towards cognitive robotics," in *Proceedings of the SPIE — Unmanned Systems Technology XI*, G. R. Gerhart, D. W. Gage, and C. M. Shoemaker, Eds., vol. 7332, 2009, pp. 73 320Z–73 320Z–11.
[7] D. Vernon, C. von Hofsten, and L. Fadiga, "Desiderata for developmental cognitive architectures," *Biologically Inspired Cognitive Architectures*, vol. 18, pp. 116–127, 2016.
[8] D. Vernon, G. Sandini, and G. Metta, "The iCub cognitive architecture: Interactive development in a humanoid robot," in *Proceedings of IEEE International Conference on Development and Learning (ICDL)*, Imperial College, London, 2007.
[9] D. Vernon, C. von Hofsten, and L. Fadiga, *A Roadmap for Cognitive Development in Humanoid Robots*, ser. Cognitive Systems Monographs (COSMOS). Berlin: Springer, 2010, vol. 11.
[10] D. Vernon, E. Billing, P. Hemeren, S. Thill, and T. Ziemke, "An architecture-oriented approach to system integration in collaborative robotics research projects - an experience report," *Journal of Software Engineering for Robotics*, vol. 6, no. 1, pp. 15–32, 2015.
[11] P. Gomez Esteban, H. Cao, A. De Beir, G. Van De Perre, D. Lefeber, and B. Vanderborght, "A multilayer reactive system for robots interacting with children with autism," in *Proceedings of the Fifth International Symposium on New Frontiers in Human-Robot Interaction*, 2016.
[12] D. David, "Intervention definition," vol. DREAM Deliverable D1.1, 2014.

to act, learn, and adapt. In this case, the design principles, or desiderata, do not drive the cognitive architecture — the requirements do that — but it helps to be aware of them so that you know what capabilities are potentially available and might be deployed to good effect. Significantly, design by use case implies that it is not feasible to proceed by developing a cognitive architecture schema and then instantiating it as a specific cognitive architecture because routing the design through the meta-level schema tacitly abstracts away many of the particularities of the application that makes this approach useful.

We can recast the distinction between the two motivations for building cognitive robots and designing cognitive architectures by asking the following question. Should a cognitive architecture be a specific or a general framework? This is an important design question because a specific instance of a cognitive architecture derived from a general schema will inherit relevant elements but it may also inherit elements that are not strictly necessary for the specific application domain. Also, it is possible that it is not sufficient, i.e. that it does not have all the elements that are necessary for the specific application domain.

To illustrate this argument, consider two architectures that were designed in these two different manners: the iCub cognitive architecture (Figure 1) [8], [9] which was designed by desiderata [9], [7] for use in a general-purpose open cognitive

A System Layout for Cognitive Service Robots

Stefan Schiffer[1,2]
[1] Knowledge-Based
Systems Group (KBSG)
RWTH Aachen University

Alexander Ferrein[2]
[2] Mobile Autonomous Systems and
Cognitive Robotics Institute (MASCOR)
FH Aachen University of Applied Sciences

Abstract—In this paper we discuss a system layout for cognitive service robots. The goal is to sketch components and their interplay needed for cognitive robotics as introduced by Ray Reiter. We are particularly interested in applications in domestic service robotics where we focus on integrating qualitative reasoning and human-robot interaction. The overall objective is to build and maintain a knowledge-based system and agent specification.

I. Introduction

In this work, we are concerned with a system layout for what is often called *cognitive robotics*. Cognitive robotics as introduced by the late Ray Reiter is to be understood as "the study of the knowledge representation and reasoning problems faced by an autonomous robot (or agent) in a dynamic and incompletely known world" [5]. Our application domain is domestic service robotics [15]. It deals with socially assistive robots that perform helpful tasks for humans in and around the house. These robots must be able to engage in communication with the humans around them. What is more, when a robot needs to assist humans with complex and cognitively challenging tasks, it must be endowed with some form of reasoning that allows to take decisions on the course of action in complex scenarios. In addition, autonomous operation for extended periods of time is only possible if the robot can handle certain variations and unavoidable errors by itself. Also, it should be flexible in dealing with human fallibility. We refer to such a robot as a cognitive service robot system.

II. A Cognitive Service Robot System Layout

We now discuss a system layout for such a cognitive service robot in domestic applications. Figure 1 shows an overview of the elements that we think are necessary and useful for a cognitive robotic system. The particular focus here is on integrating qualitative reasoning and human-robot interaction [7], [8] for applications in domestic domains [11].

The blue elements are components that provide basic capabilities like collision avoidance and localization. The green boxes represent high-level components, that is, components featuring a sophisticated reasoning mechanism. We use a logic-based high-level language called ReadyLog [4] which, among other things, features decision-theoretic planning in the spirit of [2]. The orange components bridge between the high-level and the human or extend the high-level with mechanisms to facilitate intuitive interaction. The yellow box finally, is an optional but desirable component to enable
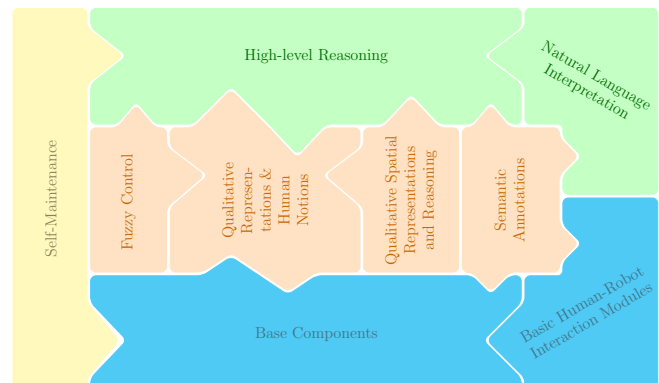

Fig. 1: A cognitive service robot system layout

enduring autonomy. It is an extension of the high-level control that has tight connections to the basic components.

A. Basic Human-Robot Interaction Modules

Our domestic service robot is supposed to interact with laymen. Hence, it needs to be operable by such laymen and the interaction between the human and the robot needs to be as *natural* and intuitive as possible. This is why we argue for extending the basic capabilities with modules for three important human-robot interaction components, namely speech, face, and gesture recognition. Examplary solutions for such components tailored for the particular application scenarios can be found in [3], [1], and [9] respectively. We consider these components since they represent (perhaps the most) important modalities in human-robot interaction. Human-robot interaction can be made even more natural and affective with additional components such as text-to-speech and an animated visual appearance.

B. High-level Reasoning

A domestic service robot that needs to assist humans with complex and cognitively challenging tasks, must be endowed with some form of reasoning that allows it to take decisions in such complex scenarios. This high-level reasoning abstracts from the details of lower levels and provides mechanisms to come up with a dedicated course of action for a robot to reach a particular goal. Our robot features a logic-based high-level reasoning component for that purpose. It allows for flexibly combining programming and planning in the behavior specification of the robot.

C. Qualitative Representations and Control

One of the issues in developing a robotic system that interacts with humans is the difference in representations with humans and machines. A technical system mostly uses numbers to represent things like speed, distance, and orientation while humans use imprecise linguistic notions. A robotic system that assists humans in their daily life, must be equipped with means to understand and to communicate with humans in terms and with notions that are natural to humans. The qualitative representations and reasoning with them should be available especially for positional information (e.g. as proposed in [12]) since these are very frequent in domestic settings, for example, with references to objects and places.

D. Semantic Annotations

Another building block to mediate between the raw sensor data and the numerical information that the base components of a cognitive robot work with are semantic annotations. In our cognitive robot system, for instance, we allow for generating semantically annotated maps [10]. This attaches semantic information to places like functions of a room or where it is likely to find people in an apartment. Another example could be part of the object recognition [6], where objects are described by a set of (semantic) attributes. This way, one can dynamically build classes of objects, for example, all objects with a specific color.

E. Natural Language Interpretation

Humans tend to be imprecise and imperfect in their natural spoken language. Therefore, when natural language is used to give instructions to a robot, the robot is potentially confronted with incomplete, ambiguous, or even incorrect commands. Aiming for a robust and flexible system a method for natural language interpretation that can account for handling such fallibility is beneficial. We present such a system [13] that uses decision-theoretic planning in the spirit of DT-Golog [2] to interpret the instruction given to the robot. It is able to account for imprecise and missing information and initiates steps for clarification accordingly.

F. Self-Maintenance

A robotic system that is capable of planning and executing complex tasks is a complex system itself. That is why such a system is itself vulnerable to errors. These errors are not restricted to action execution but span to internal system errors as well. As an additional component in the system layout we proposed a system for self-maintenance [14] that is able to detect and circumvent certain errors. Thus we increase the system's robustness and enable longer-term autonomous operation.

III. CONCLUSION

In this paper, we discussed the layout of a cognitive service robotic system that integrates qualitative reasoning and human-robot interaction for applications in domestic service robotics. The system layout features components that allow for implementing a capable service robotic system. The layout addresses bridging the gap between the robot and the human with several measures, making available the qualitative notions that humans commonly use in the robot system, in general, and in the high-level reasoning, in particular. This allows for natural interaction and with its advanced reasoning the robot can assist its human users with complex and cognitively challenging tasks. This is especially useful with disabled or elderly people.

REFERENCES

[1] Vaishak Belle, Thomas Deselaers, and Stefan Schiffer. Randomized trees for real-time one-step face detection and recognition. In *Proc. Int'l Conf. on Pattern Recognition (ICPR'08)*, pages 1–4. IEEE Computer Society, December 8-11 2008.

[2] Craig Boutilier, Ray Reiter, Mikhail Soutchanski, and Sebastian Thrun. Decision-theoretic, high-level agent programming in the situation calculus. In *Proc. Nat'l Conf. on Artificial Intelligence (AAAI-00)*, pages 355–362, Menlo Park, CA, July 30– 3 2000. AAAI Press.

[3] Masrur Doostdar, Stefan Schiffer, and Gerhard Lakemeyer. Robust speech recognition for service robotics applications. In *Proc. Int'l RoboCup Symposium (RoboCup 2008)*, volume 5399 of *LNCS*, pages 1–12. Springer, July 14-18 2008.

[4] Alexander Ferrein and Gerhard Lakemeyer. Logic-based robot control in highly dynamic domains. *Robotics and Autonomous Systems*, 56(11):980–991, 2008.

[5] Hector J. Levesque and Gerhard Lakemeyer. Cognitive Robotics. In Frank van Harmelen, Vladimir Lifschitz, and Bruce Porter, editors, *Handbook of Knowledge Representation*, chapter 23, pages 869–886. Elsevier, 2008.

[6] Tim Niemueller, Stefan Schiffer, Gerhard Lakemeyer, and Safoura Rezapour-Lakani. Life-long learning perception using cloud database technology. In *Proc. IROS Workshop on Cloud Robotics*, 2013.

[7] Stefan Schiffer. *Integrating Qualitative Reasoning and Human-Robot Interaction for Domestic Service Robots*. Dissertation, RWTH Aachen University, Department of Computer Science, Feb 2015.

[8] Stefan Schiffer. Integrating qualitative reasoning and human-robot interaction in domestic service robotics. *KI - Künstliche Intelligenz*, 30(3):257–265, 2016.

[9] Stefan Schiffer, Tobias Baumgartner, and Gerhard Lakemeyer. A modular approach to gesture recognition for interaction with a domestic service robot. In *Intelligent Robotics and Applications*, pages 348–357. Springer, 2011.

[10] Stefan Schiffer, Alexander Ferrein, and Gerhard Lakemeyer. Football is coming home. In *Proc. 2006 Int'l Symp. on Practical Cognitive Agents and Robots (PCAR'06)*, pages 39–50, New York, NY, USA, November 27-28 2006. ACM.

[11] Stefan Schiffer, Alexander Ferrein, and Gerhard Lakemeyer. CAESAR – An Intelligent Domestic Service Robot. *Journal of Intelligent Service Robotics*, pages 1–15, 2012.

[12] Stefan Schiffer, Alexander Ferrein, and Gerhard Lakemeyer. Reasoning with Qualitative Positional Information for Domestic Domains in the Situation Calculus. *Journal of Intelligent and Robotic Systems*, 66(1–2):273–300, 2012.

[13] Stefan Schiffer, Niklas Hoppe, and Gerhard Lakemeyer. Natural language interpretation for an interactive service robot in domestic domains. In *Agents and Artificial Intelligence*, volume 358, pages 39–53. Springer, 2013.

[14] Stefan Schiffer, Andreas Wortmann, and Gerhard Lakemeyer. Self-Maintenance for Autonomous Robots controlled by ReadyLog. In *Proc. IARP Workshop on Technical Challenges for Dependable Robots in Human Environments*, pages 101–107, Toulouse, France, June 16-17 2010.

[15] Thomas Wisspeintner, Tijn van der Zant, Luca Iocchi, and Stefan Schiffer. RoboCup@Home: Scientific Competition and Benchmarking for Domestic Service Robots. *Interaction Studies*, 10(3):392–426, 2009.

The Mirror Self-recognition for Robots

iCubSim at the mirror

Andrej Lucny

Department of Applied Informatics
FMFI, Comenius University
Bratislava, Slovakia
lucny@fmph.uniba.sk

Abstract—**We introduce a simple example of artificial system which aims to mimic the process of the mirror self-recognition - ability limited to very few species. We assume that evolution of the species is reflected in the structure of the underlying control mechanism and design modules of the mechanism, concerning its incremental development. On this example, we also demonstrate modular architecture suitable for such task. It is based on decentralization and massive parallelism and enables incremental building of control system which is running in real-time and easily combines modules operating at different pace.**

Keywords—the mirror self-recognition; robot iCubSim; Agent-space architecture

I. Introduction

Seeing in the mirror, we recognize ourselves. However child less than eighteen months old is rather looking for another person behind the mirror. In nature, the mirror self-recognition is present only in exceptional cases, for instance chimpanzees recognize themselves in the mirror, while cats do not. Can a robot recognize itself in the mirror? And what is the origin of the self-recognition ability? We follow Scassellati and Hart [3], who suppose that a robot should recognize itself in the mirror due to perfect correlation between body movement and the image seen in the mirror. Unlike them and like Takeno [8] we use a very simplified body model.

II. Testbed

Using the simulator of the humanoid robot iCub [7] we added a camera shooting the area in the front of the monitor with the robot rendered image, we create a control system on which we can examine how the mirror self-recognition emerges from basic mechanisms. This enables us to establish interaction between the virtual robot and human sitting at the monitor and later between the robot and its image reflected to the camera by the mirror. The simulator has almost perfect model of the robot body. However for our purposes we intend to simplify it, thus we move a single joint – the joint moving head from side to side. In this way the body model is simplified to a single number – the angle of the robot head inclination.

III. Architecture

For examination of mechanisms underlying the mirror self-recognition process, we employ a modular framework (Agent-Space Architecture [5]) which enables us to let the control system emerge from the parallel course of simpler modules and from their mutual interactions. Agent-Space Architecture is a software tool for building modular control systems of robots, running in real-time. It is strongly based on ideas of Brooks' subsumption architecture [1] and Minsky's society model of mind [6] [4]. Within the framework we define operation of individual modules and connect various module outputs to various module inputs (many:many) regardless the fact that each module can operate at different pace. Such combining of either really slow or really fast processes is possible due to all data produced by a module (producer) being written onto a blackboard (called *space*) and processed later, when another module (consumer) is ready to process them. These written data remain on the blackboard until they are rewritten by the same or another producer, or their time validity expires. Finally the data exchange supports also hierarchical control and its incremental development, e.g. module has to define sufficient priority for written data – otherwise it would not be able to rewrite their previous value already written on the blackboard.

IV. Mechanisms

Now we can try to implement the system, using which we are able to evaluate correlation between own body movement and the seen image movement. We aim to implement a working demo following biological relevancy of our approach, based on knowledge about the above-mentioned species.

A. Body model (proprioception)

The body model is provided by the iCubSim simulator, thus we need to implement just its control and monitoring from the blackboard. We implement a module *motor* which reads an intended joint position *intention* from the blackboard, communicates with the simulator (via yarp rpc protocol) and as a result, it writes *proprioception* to the blackboard which represents a real joint position (Fig. 1). In parallel, the added camera regularly provides a new image to the blackboard.

B. Mirroring

As we aim to compare the body model (i.e. value *proprioception*) with the *image* captured by the camera, we need to get an analogical model from the image. This approach is still biologically relevant since several species are proven to obtain such ability (mirror neurons). (How such ability has emerged and evolved, is not the subject of our exploration – it is only limited to the implementation of this ability.) Since we

take into account both interaction between the robot and human and interaction between the robot and its image in the mirror, we combine two specific methods which output the same data onto the backboard – the model of the seen image (the *model* value in Fig. 1).
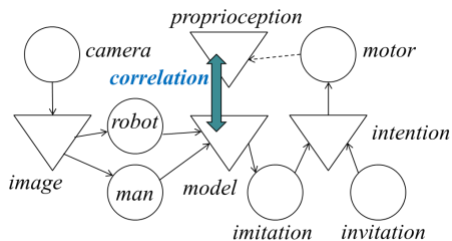

Fig. 1. The modular structure of the mirror self-recognition system

Fig. 2. The iCumSim robot following its own image in the mirror

For the iCubSim image processing (we rotate a printed picture of iCubSim in front of the camera) we employ the SURF method. SURF provides projection of given template to the image seen from which we can easily calculate inclination of the template. For the processing of human image (interacting person is moving his head in front of the camera) we use combination of Haar face detector and CamShift tracking: the Haar detector detects face in the upright position and sets up template for CamShift to track. The projection of the template to the seen image calculated by CamShift provides us with the required head inclination. All algorithms employed are available in OpenCV library (www.opencv.org).

C. Imitation

The correlation of these two comparable models can only be evaluated by their changes through time therefore both robot and human have to be put in motion. One of the mechanisms, that can provide it, is imitation. Due to the utilized mirroring, a passive imitation can be implemented by direct copying of values of the body model seen (i.e. the *model* value) into the robot's body model (i.e. the *intention* value). As a result, iCubSim imitates side to side movement of one's head when moving in front of the camera. Optionally we can also

implement the active part of imitation, which can be called the invitation to imitation and obtains a higher priority than the passive imitation and occasionally generates side to side movement of the robot's head when the body seen does not move. However we found out that even slight inaccuracies in the model evaluation are sufficient to induce the imitation process.

D. Social modelling

Imitation process provides us with the data of correlation between the body owned and the body seen. When an individual lives within a society, it is important for him to categorize the data and associate them with the image seen. When such individual meets the mirror, it possibly creates a new category and finds out that the corresponding correlation is unusually high. In fact, it is so high that the image seen can not only be associated with the body model seen, but also with its own body, meaning that it sees itself. Instead of modelling the society, for our purposes, it is sufficient to record the data produced by the two models since the aim of the study is to differentiate the data created when robot encounters human and when robot sees its own image in the mirror.

V. CONCLUSION

As a result of implementation of the above mentioned mechanisms, when iCubSim's image is reflected into the external camera by a mirror, robot invites himself to imitation and it stays in an interaction with itself for certain amount of time (Fig. 2). Thus correlation between the body model (*proprioception*) and the model of the image seen (*model*) can be evaluated and its unusually high value indicates that the robot sees itself in the mirror. Thus the designed structure of the mirror self-recognition process is partially verified.

As a teaser video can be viewed here:
http://www.agentspace.org/mirror/iCubSimAtTheMirror.mp4

REFERENCES

[1] Brooks, R. (1999). "Cambrian Intelligence". The MIT Press, Cambridge, MA

[2] Gallup, G. G., Jr. (1970). Chimpanzees: Self Recognition. Science, 167, 86-87.

[3] Hart, J. W. Scassellati, B. (2012). Mirror Perspective-Taking with a Humanoid Robot. In Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI-12). Toronto, Canada.

[4] Kelemen, J. (2001). "From statistics to emergence - exercises in systems modularity". In: Multi-Agent Systems and Applications, (Luck, M., Marik, V., Stepankova, O. Trappl, R.), Springer, Berlin, pp. 281-300

[5] Lucny, A. (2004). "Building complex systems with Agent-Space architecture". Computing and Informatics, Vol. 23, pp. 1001-1036

[6] Minsky, M. (1986). The Society of Mind. Simon&Schuster, New York

[7] Sandini, G. - Metta, G. - Vernon, D. (2007). The iCub cognitive humanoid robot: an open-system research platform for enactive cognition. In: 50 years of artificial intelligence, pp. 358-369, Springer-Verlag, Berlin

[8] Takeno, J. (2008). A Robot Succeeds in 100% Mirror Image Cognition. INTERNATIONAL JOURNAL ON SMART SENSING AND INTELLIGENT SYSTEMS, Vol. 1, No. 4, December 2008

Towards Incorporating Appraisal into Emotion Recognition: A Dynamic Architecture for Intensity Estimation from Physiological Signals

Robert Jenke[1], Angelika Peer[2]

Abstract— **Current approaches to emotion recognition do not address the fact that emotions are dynamic processes. This work concerns itself with the development of a gray-box framework for dynamic emotion intensity estimation that can incorporate findings from appraisal models, specifically Scherer's Component Process Model. It is based on Dynamic Field Theory which allows the combination of theoretical knowledge with data-driven experimental approaches. Further, we conducted an exemplary user study applying the proposed model to estimate intensity of negative emotions from physiological signals. Results show significant improvements of the proposed model to common methodology and baselines. The flexible cognitive architecture opens a wide field of experiments and directions to deepen the understanding of emotion processes as a whole.**

I. Introduction

Current efforts in Human-Robot-Interaction (HRI) aim at finding ways to make interaction more natural. In this, knowledge of the user's emotional state is considered an important factor. Methods of automatic estimation of affective states from various modalities, including physiological signals, have therefore received much attention lately.

Recent work in emotion theory, e.g. Scherer's Component Process Model (CPM), points out the dynamic nature of emotion processes which therefore, "require a dynamic computational architecture" [1]. To date, however, most work on emotion recognition concerns itself with static prediction of emotion labels from a window of time series data using machine learning methods (i.e. black-box approach).

Our main research objective is to design a gray-box model for emotion recognition from physiological signals, which is capable of combining theoretical knowledge incorporated in the CPM with experimental data to train parameters the model. In this paper, we address the hitherto neglected dynamic evolvement of the affective state by proposing an architecture for emotion intensity estimation based on the Dynamic Field Theory (DFT) [2].

II. Model

In the CPM, the subjective feeling (i.e. affective state) is characterized by the emotion intensity I of an emotion quality ϑ at time t and can generally be written as $I(\vartheta, t)$. The CPM provides detailed relations between the so-called stimulus evaluation checks (SECs) that happen in the appraisal process and their effects on physiology. For example, a novelty check can lead to an increase in skin conductance or the obstructiveness of an event changes the heart rate of a person.

[1]Chair of Automatic Control Engineering, Technische Universität München, Munich, Germany, www.lsr.ei.tum.de, E-mail: `rj@tum.de`
[2]Bristol Robotics Laboratory, University of the West of England, Bristol, UK, www.brl.ac.uk, E-mail: `angelika.peer@brl.ac.uk`
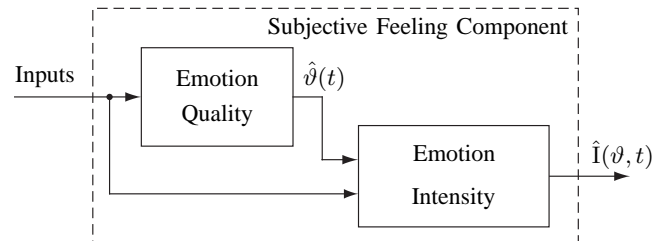
Fig. 1. The subjective feeling component is divided into consecutive estimation of emotion quality $\hat{\vartheta}(t)$ and emotion intensity $\hat{I}(\vartheta, t)$.

Similar to Bailenson et al. [3], we separate estimation of emotion quality and intensity (see Fig. 1). We control the former by experimental design, i.e. we assume $\hat{\vartheta}(t)$ to be a known input to our model. The architecture of our dynamic model is based on DFT. These fields usually span over physical dimensions such as space or angle and model dynamic changes along this dimension. Fields are governed by differential equations and can represent functionalities like memory (for details, see [4]).

For our model, we define the field over the emotion quality ϑ as shown in Fig. 2. The core part of the model is the intensity layer $i(\vartheta, t)$ together with a memory layer $m(\vartheta, t)$, which model the changes in the subjective feeling, i.e. the output $\hat{I}(\vartheta, t)$. The second part are the input layers, where we use one layer for each prediction from the SECs provided by the CPM, e.g. $u(\vartheta, t)$ in Fig. 2. For example, a change in skin response would be an input layer.
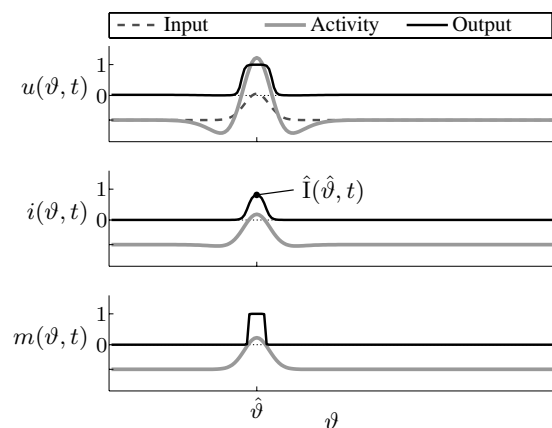

Fig. 2. Architecture of the proposed dynamic model: three-layer field spanned over the dimension of emotion quality ϑ.

III. Experimental Design

We control the emotion quality $\hat{\vartheta}$ in our experimental design by fixing it through choice of emotion induction. This results in a simplified dynamic model at one location of the fields, i.e. three neurons and their governing equations.

For the dataset, we recorded the galvanic skin response (GSR) of subjects. Additionally, we used a slider device interface to record the emotion intensity experienced by the subject. For emotion induction, we used standardized IAPS pictures of a fixed emotion quality, here, negative emotions [5]. After segmentation, we had collected 7 trials of 110 s recordings for each of three subjects.

The change in GSR is computed as a prediction of SECs and used as model input. The continuously recorded intensity measures of the slider served as ground truth. For training of the dynamic model, free parameters are determined by means of experimental data applying leave-one-out cross validation. In this, we minimize the error between the output of the dynamic model and the ground truth s.t. boundary conditions.

IV. Results

First, we compare the accuracy of our model with common static methods and baselines, i.e. linear regression and random regressors. We use the match of estimate with ground truth plus an acceptable error margin as accuracy measure. In summary, the dynamic model performs significantly better than common methodology and baselines. Limitations of the model become apparent for small error margins.

Secondly, capabilities and limitations of the model in its current version are exemplified in Fig. 3. In the upper graph, we see the changes in GSR, which characterize the onset as well as the increase of intensity well. The memory layer (bottom graph) helps to stabilize the decay at an appropriate rate. However, limitations of the current model are apparent, as the third change in GSR should not have any impact on the intensity. This points towards the need to include additional input layers where appropriate interaction can avoid this behavior.

V. Conclusion

For the first time, a dynamic gray-box model framework based on DFT has been proposed for emotion recognition, which allows to include theoretical knowledge into the model and learn free parameters from experimental results. We designed and carried out an exemplary study to estimate emotion intensity from physiological signals. In this, the dynamic model performed significantly better than baselines. We also identified current limitations and ways to improve the model. Future work includes several extension to the architecture as well as carrying out experiments to further evaluate the model.

References

[1] K. R. Scherer, "Emotions are emergent processes: they require a dynamic computational architecture." *Phil. Trans. of the Royal Society, Series B*, vol. 364, pp. 3459–74, Dec. 2009.
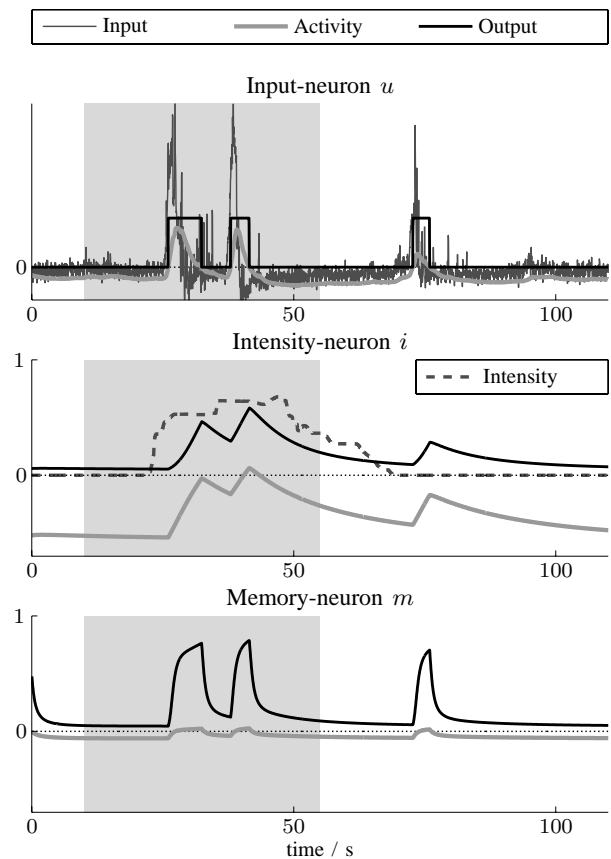
Fig. 3. Example of a single location of all layers over time.

[2] G. Schöner, "Dynamical systems approaches to cognition," in *Cambridge Handbook of Computational Psychology*, R. Sun, Ed. Cambridge University Press, 2008, pp. 101–126.
[3] J. Bailenson, E. Pontikakis, I. Mauss, J. Gross, M. Jabon, C. Hutcherson, C. Nass, and O. John, "Real-time classification of evoked emotions using facial feature tracking and physiological responses," *Int.Journal of Human-Computer Studies*, vol. 66, no. 5, pp. 303–317, May 2008.
[4] Y. Sandamirskaya, "Dynamic neural fields as a step toward cognitive neuromorphic architectures." *Frontiers in Neuroscience*, vol. 7, pp. 1–13. art. 276, Jan. 2014.
[5] P. Lang, M. Bradley, and B. Cuthbert, "International Affective Picture System (IAPS): Affective ratings of pictures and instruction manual," University of Florida, Tech. Rep., 2005.

A Needs-Driven Cognitive Architecture for Future 'Intelligent' Communicative Agents

Roger K. Moore

Dept. Computer Science, University of Sheffield, UK
Email: r.k.moore@sheffield.ac.uk

Abstract—Recent years have seen considerable progress in the deployment of 'intelligent' communicative agents such as Apple's *Siri*, Google *Now*, Microsoft's *Cortana* and Amazon's *Alexa*. Such speech-enabled assistants are distinguished from the previous generation of voice-based systems in that they claim to offer access to services and information via conversational interaction. In reality, interaction has limited depth and, after initial enthusiasm, users revert to more traditional interface technologies. This paper argues that the standard architecture for a contemporary communicative agent fails to capture the fundamental properties of human spoken language. So an alternative needs-driven cognitive architecture is proposed which models speech-based interaction as an emergent property of coupled hierarchical feedback control processes. The implications for future spoken language systems are discussed.

I. Introduction

The performance of spoken language systems has improved significantly in recent years, with corporate giants such as MicroSoft and IBM issuing claim and counter-claim as to who has the lowest word error rates. Such progress has contributed to the deployment of ever more sophisticated voice-based applications, from the earliest military 'Command and Control Systems' to the latest consumer 'Voice-Enabled Personal Assistants' (such as *Siri*) [1]. Research is now focussed on voice-based interaction with 'Embodied Conversational Agents (ECAs)' and 'Autonomous Social Agents' based on the assumption that spoken language will provide a 'natural' conversational interface between human beings and future (so-called) *intelligent* systems – see Fig. 1.
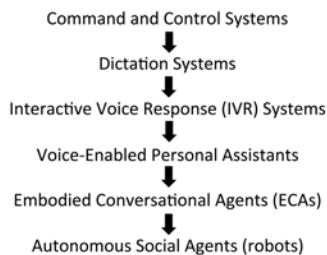

Fig. 1. The evolution of spoken language technology applications.

In reality, users' experiences with contemporary spoken language systems leaves a lot to be desired. After initial enthusiasm, users lose interest in talking to *Siri* or *Alexa*, and they revert to more traditional interface technologies [2]. One possible explanation for this state of affairs is that, while component technologies such as automatic speech recognition and text-to-speech synthesis are subject to continuous ongoing improvement, the overall architecture of a spoken language system has been standardised for some time [3] – see Fig. 2. Standardisation is helpful because it promotes interoperability and expands markets. However, it can also stifle innovation by prescribing sub-optimal solutions. So, what (if anything) might be wrong with the architecture illustrated in Fig. 2?
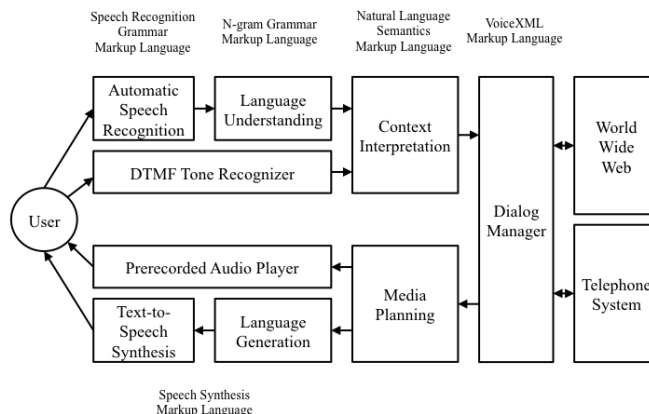

Fig. 2. Illustration of the W3C *Speech Interface Framework* [3].

In the context of spoken language, the main issue with the architecture illustrated in Fig. 2 is that it reflects a traditional stimulus–response ('behaviourist') view of interaction; the user utters a request, the system replies. This is the 'tennis match' analogy for language; a stance that is now regarded as restrictive and old-fashioned. Contemporary perspectives regard spoken language interaction as being more like a three-legged race than a tennis match [4]: continuous coordinated behaviour between coupled dynamical systems.

II. Towards a 'Cognitive' Architecture

What seems to be required is an architecture that replaces the traditional 'open-loop' stimulus-response arrangement with a 'closed-loop' dynamical framework; a framework in which needs/intentions lead to actions, actions lead to consequences, and perceived consequences are compared to intentions/needs (in a continuous cycle of synchronous
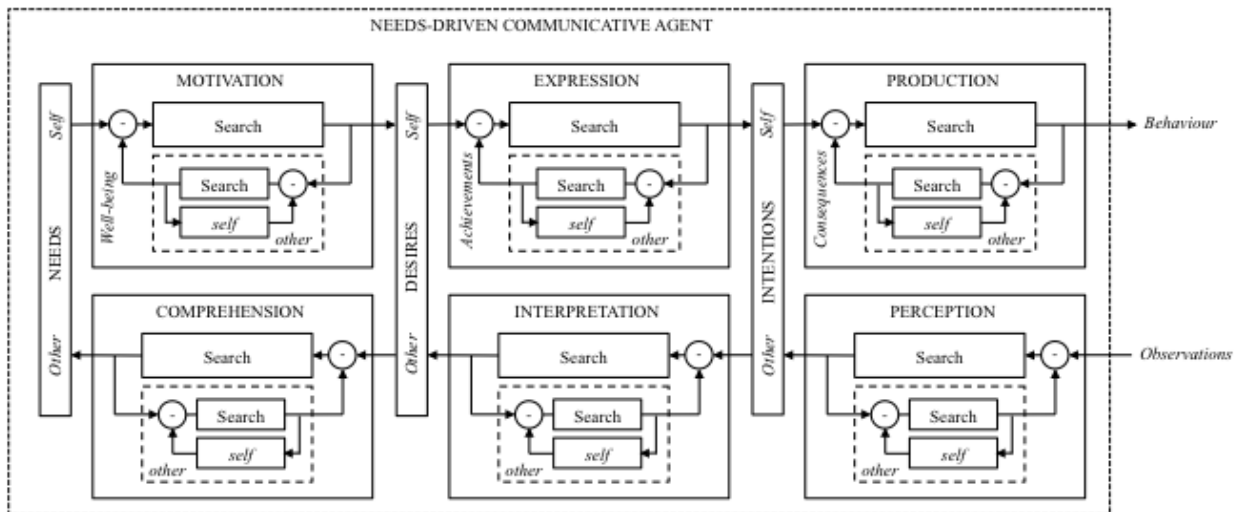
Fig. 3. Illustration of the proposed architecture for a needs-driven communicative agent [7].

behaviours). Such an architecture has been proposed by the author [5], [6], [7] – see Fig. 3.

One of the key concepts embedded in the architecture illustrated in Fig. 3 is the agent's ability to 'infer' (using search) the consequences of their actions when they cannot be observed directly. Another is the use of a *forward model* of 'self' to model 'other'. Both of these features align well with the contemporary view of language as "*ostensive inferential recursive mind-reading*" [8]. Also, the architecture makes an analogy between the depth of each search process and 'motivation/effort'. This is because it has been known for some time that speakers continuously trade effort against intelligibility [9], [10], and this maps very nicely into a hierarchical control-feedback process [11] which is capable of maintaining sufficient contrast at the highest *pragmatic* level of communication by means of suitable regulatory compensations at the lower semantic, syntactic, lexical, phonemic, phonetic and acoustic levels.

As a practical example, these ideas have been used to construct a new type of speech synthesiser (known as 'C2H') that adjusts its output as a function of its inferred communicative success [13], [14] – it listens to itself!

III. FINAL REMARKS

Whilst the proposed cognitive architecture successfully captures some of the key elements of language-based interaction, it is important to note that such interaction between human beings is founded on substantial shared priors. This means that there may be a fundamental limit to the language-based interaction that can take place between *mismatched* partners such as a human being and an autonomous social agent [15].

ACKNOWLEDGMENT

REFERENCES

[1] R. Pieraccini. *The Voice in the Machine*. MIT Press, Cambridge, 2012.
[2] R. K. Moore, H. Li, & S.-H. Liao. Progress and prospects for spoken language technology: what ordinary people think. In *INTERSPEECH* (pp. 3007–3011). San Francisco, CA, 2016.
[3] Introduction and Overview of W3C Speech Interface Framework, http://www.w3.org/TR/voice-intro/
[4] F. Cummins. Periodic and aperiodic synchronization in skilled action. *Frontiers in Human Neuroscience*, 5(170), 1–9, 2011.
[5] R. K. Moore. PRESENCE: A human-inspired architecture for speech-based human-machine interaction. *IEEE Trans. Computers*, 56(9), 1176–1188, 2007.
[6] R. K. Moore. Spoken language processing: time to look outside? In L. Besacier, A.-H. Dediu, & C. Martn-Vide (Eds.), *2nd International Conference on Statistical Language and Speech Processing (SLSP 2014), Lecture Notes in Computer Science* (Vol. 8791). Springer, 2014.
[7] R. K. Moore. PCT and Beyond: Towards a Computational Framework for "Intelligent" Systems. In A. McElhone & W. Mansell (Eds.), *Living Control Systems IV: Perceptual Control Theory and the Future of the Life and Social Sciences*. Benchmark Publications Inc. In Press (available at https://arxiv.org/abs/1611.05379).
[8] T. Scott-Phillips. *Speaking Our Minds: Why human communication is different, and how language evolved to make it special*. London, New York: Palgrave MacMillan, 2015.
[9] E. Lombard. Le sign de l?lvation de la voix. *Ann. Maladies Oreille, Larynx, Nez, Pharynx*, 37, 101–119, 1911.
[10] B. Lindblom. Explaining phonetic variation: a sketch of the H&H theory. In W. J. Hardcastle & A. Marchal (Eds.), *Speech Production and Speech Modelling* (pp. 403–439). Kluwer Academic Publishers, 1990.
[11] W. T. Powers. *Behavior: The Control of Perception*. NY: Aldine: Hawthorne, 1973.
[12] S. Hawkins. Roles and representations of systematic fine phonetic detail in speech understanding. *Journal of Phonetics*, 31, 373–405, 2003.
[13] R. K. Moore & M. Nicolao. Reactive speech synthesis: actively managing phonetic contrast along an H&H continuum, *17th International Congress of Phonetics Sciences (ICPhS)*. Hong Kong, 2011.
[14] M. Nicolao, J. Latorre & R. K. Moore. C2H: A computational model of H&H-based phonetic contrast in synthetic speech. *INTERSPEECH*. Portland, USA, 2012.
[15] R. K. Moore. Is spoken language all-or-nothing? Implications for future speech-based human-machine interaction. In K. Jokinen & G. Wilcock (Eds.), *Dialogues with Social Robots - Enablements, Analyses, and Evaluation*. Springer Lecture Notes in Electrical Engineering, 2016.

Artificial Spatial Cognition for Robotics and Mobile Systems: Brief Survey and Current Open Challenges

Paloma de la Puente and M. Guadalupe

Sánchez-Escribano Universidad Politécnica de
Madrid (UPM) Madrid, Spain

Abstract—**Remarkable and impressive advancements in the areas of perception, mapping and navigation of artificial mobile systems have been witnessed in the last decades. However, it is clear that important limitations remain regarding the spatial cognition capabilities of existing available implementations and the current practical functionality of high level cognitive models [1, 2]. For enhanced robustness and flexibility in different kinds of real world scenarios, a deeper understanding of the environment, the system, and their interactions -in general terms- is desired. This long abstract aims at outlining connections between recent contributions in the above mentioned areas and research in cognitive architectures and biological systems. We try to summarize, integrate and update previous reviews, highlighting the main open issues and aspects not yet unified or integrated in a common architectural framework.**

Keywords—spatial cognition; surveys; perception; navigation

I. BRIEF SURVEY

A. Initial models for spatial knowledge representation and main missing elements

Focusing on the spatial knowledge representation and management, the first contributions inspired by the human cognitive map combined metric local maps, as an Absolute Space Representation (ASR), and topological graphs [3]. As a related approach, the Spatial Semantic Hierarchy (SSH) [4] was the first fundamental cognitive model for large-scale space. It evolved into the Hybrid SSH [5], which also included knowledge about small-scale space. This fundamental work was undoubtedly groundbreaking, but it did not go beyond basic levels of information abstraction and conceptualization [6]. Moreover, the well-motivated dependencies among different types of knowledge (both declarative and procedural) were not further considered for general problem solving [7]. The SSH model was considered suitable for the popular schema of a "three layer architecture", without explicitly dealing with processes such as attention or forgetting mechanisms. This lack of principled forgetting mechanisms has been identified by the Simultaneous Localization and Mapping (SLAM) robotics community as a key missing feature of most existing mapping approaches [8, 9].

B. The role of cognitive architectures and their relation to other works in the robotics community

Cognitive architectures provide a solid approach for modeling general intelligent agents and their main commitments support the ambitious requirements of high level behavior in arbitrary situations for robotics [10]. A more recent model of spatial knowledge, the Spatial/Visual System (SVS) [11] designed as an extension of the Soar cognitive architecture, proposed a different multiplicity of representations, i.e. symbolic, quantitative spatial and visual depictive. The spatial scene is a hierarchy tree of entities and their constitutive parts, with intermediate nodes defining the transformation relations between parts and objects. Other works in robotics employ similar internal representation ideas [12-14], and other ones included the possibility to hypothesize geometric environment structure in order to build consistent maps [15]. While a complete implementation of this approach for all kind of objects requires solving the corresponding segmentation and recognition problems in a domain independent manner (which is far beyond the state of the art), keeping the perceptual level representations within the architecture enhances functionality. A very active research community address these difficult challenges.

The recognition process should not only use visual, spatial and motion data from the Perceptual LTM but also conceptual context information [7, 16] and episodic memories of remembered places [17], from Symbolic LTM. This should also apply to the navigation techniques for different situations [18, 19]. The existence of motion models for the objects can improve navigation in dynamic environments, which is one of the main problems in real world robotic applications [20, 21].

A novel cognitive architecture specifically designed for spatial knowledge processing is the Casimir architecture [22], which presents rich modeling capabilities pursuing human-like behavior. Navigation, however, has not been addressed, and this work has scarcely been discussed in the robotics domain.

One of the latest spatial models is the NavModel [23], designed and implemented for the ACT-R architecture. Besides considering multi-level representations, this model presents three navigation strategies with varying cognitive cost. The first developed implementation assumes known topological localization at room level, while a subsequent implementation incorporates a mental rotation model. This work focuses on the cognitive load and does not deal with lower level issues.

To point out how topics are addressed by the respective communities, we compiled Table I as a comparison. The contrast regarding memory management and uncertainty seems

to be relevant. The lack of approaches combining both allocentric and egocentric representations is also remarkable. To conclude, Table II shows a summary of surveys.

TABLE I.   COMPARISON OF TOPICS ADDRESSED BY THE COGNITIVE ARCHITECTURES AND ROBOTICS COMMUNITIES

Cognitive Architectures Community	← Topic →	Perception, Robotics, Vehicles Community
ACT-R/S, CLARION	Egocentric spatial models	[24, 25]
LIDA, SOAR-SVS	Allocentric spatial models	[9, 26]
Casimir, LIDA, SOAR-SVS	Object based/ semantic representations	[6, 12-14]
SOAR-SVS	Explicit motion models / dynamic information about the environment	[27, 28]
All	Memory management, forgetting mechanisms	[19]
Extended LIDA [29]	Uncertainty considerations	Most mapping and navigation approaches

TABLE II.   SUMMARY OF SURVEYS

Topic	References
Robotics and Cognitive Mapping	[1]
SLAM and Robust Perception	[8, 9]
Computational cognitive models of spatial memory	[2]
Object recognition	[30, 31]
Cognitive Architectures for Robotics	[10]
Spatial knowledge in brains	[17]

II.   CURRENT OPEN CHALLENGES

The big challenge is closing the gap between high level models and actual implementations in artificial mobile systems. To reduce this existing gap, we identify three main goals:

- Combination of allocentric and egocentric models using different levels of features/objects + topology/semantics.
- Acquisition and integration of motion models and dynamic information for the elements/objects.
- Integration of global mapping & loop closure capabilities with extensive declarative knowledge about features relevance and forgetting mechanisms with episodic memory.

ACKNOWLEDGMENT

REFERENCES

[1] G. Eason M. Jefferies and W.K. Yeap. Robotics and cognitive approaches to spatial mapping. Springer, 2008.

[2] T. Madl, K. Chen, D. Montaldi and R. Trappl. Computational cognitive models of spatial memory in navigation space: A review. Neural Networks, 2015.

[3] W.K. Yeap. Towards a computational theory of cognitive maps. Journal of Artificial Intelligence, 1988.

[4] B. Kuipers. The spatial semantic hierarchy. Artificial Intelligence. 2000.

[5] Kuipers, J. Modayil, P. Beeson, M.MacMahon and F. Savelli. Local metrical and global topological maps in the hybrid Spatial Semantic Hierarchy. ICRA, 2004.

[6] A. Pronobis and P. Jensfelt. Large-scale semantic mapping and reasoning with heterogeneous modalities. ICRA, 2012.

[7] S.D. Lathrop. Extending cognitive architectures with spatial and visual imagery mechanisms. PhD Thesis, 2008.

[8] J.A. Fernandez-Madrigal and J.L. Blanco. Simultaneous localization and mapping for mobile robots: iIntroduction and methods. IGI, 2012.

[9] C. Cadena et al. Past, present, and future of simultaneous localization and mapping: towards the robust-perception age. T-RO, 2016.

[10] U. Kurup and C. Lebiere. What can cognitive architectures do for robotics? Biologically Inspired Cognitive Architectures, 2012.

[11] S.D. Lathrop. Exploring the functional advantages of spatial and visual cognition from an architectural perspective. TopiCS 2011.

[12] R.F. Salas-Moreno, R.A: Newcombe, H. Strasdat, P.H.J Kelly and A.J. Davison. SLAM++: Simultaneous localisation and mapping at the Level of objects. CVPR, 2013.

[13] S. Eslami and C. Williams. A generative model for parts-based object segmentation. Advances Neural Information Processing Systems, 2012.

[14] A. Uckermann, C. Eibrechter, R. Haschke and H. Ritter. Real time hierarchical scene segmentation and classification. Humanoids, 2014.

[15] P. de la Puente and D. Rodriguez-Losada. Feature based graph SLAM in structured environments. Autonomous Robots, 2014.

[16] L. Kunze et al. Combining top-down spatial reasoning and bottom-up object class recognition for scene understanding. IROS, 2014.

[17] M.B Moser and E.I. Moser. The brain's GPS. Scientific American, 2016.

[18] G. Gunzelmann and D. Lyon (2007) Mechanisms for human spatial competence. Spatial Cognition V, LNAI-Springer, 2007.

[19] F. Dayoub, G. Cielniak and T. Duckett. Eight weeks of episodic visual navigation inside a non-stationary environment using adaptive spherical views. FSR, 2013.

[20] N. Hawes et al. The STRANDS project: long-term autonomy in everyday environments. Robotics and Automation Magazine, in press.

[21] P. de la Puente et al. Experiences with RGB-D navigation in real home robotic trials. ARW, 2016.

[22] H. Schultheis and T. Barkowsky. Casimir: an architecture for mental spatial knowledge processing. TopiCS, 2011.

[23] C. Zhao. Understanding human spatial navigation behaviors: A cognitive modeling. PhD Thesis, 2016.

[24] R. Drouilly, P. Rives and B. Morisset. Semantic representation for navigation in large-scale environments. ICRA, 2015.

[25] L.F. Posada, F. Hoffmann and T. Bertram. Visual semantic robot navigation in indoor environments. ISR, 2014.

[26] A. Richardson and E. Olson. Iterative path optimization for practical robot planning. IROS, 2011.

[27] R. Ambrus, N. Bore, J. Folkesson and P. Jensfelt. Meta-rooms: building and maintaining long term spatial models in a dynamic world. IROS, 2014.

[28] D. M. Rosen, J. Mason and J. J. Leonard. Towards lifelong feature-based mapping in semi-static environments. ICRA, 2016.

[29] T. Madl, S. Franklin, K. Chen, D. Montaldi and R. Trappl. Towards real-world capable spatial memory in the LIDA cognitive architecture. BICA, 2016.

[30] J. J. DiCarlo, D. Zoccolan and N. C. Rust. How does the brain solve visual object recognition? Neuron, 2012.

Combining Visual Learning with a Generic Cognitive Model for Appliance Representation

Kanishka Ganguly, Konstantinos Zampogiannis, Cornelia Fermüller and Yiannis Aloimonos
Institute for Advanced Computer Studies, University of Maryland
College Park, MD 20740
Email: {*kganguly,kzampog,fer,yiannis*}*@umiacs.umd.edu*

Abstract—**For robots to become ubiquitous in every home, it is imperative for them to be able to safely and intuitively interact with objects that humans use on a daily basis. To allow for such operation, we propose a cognitive model that will allow robots to recognize and work with common appliances, such as microwaves, refrigerators, dishwashers, and other similar equipment, found in everyday scenarios.**

I. Motivation

Robots are rapidly becoming a part of our everyday life and they will need to intelligently interact with complex, unknown and changing environments. Automatic modeling of various aspects of the environment based on visual perception is a fundamental challenge to be addressed. We humans are extremely adept at manipulating equipment with very little experience and are able to generalize to other similar equipments easily.

There has been prior work done on visual detection of appliances [1], [2] which use visual sensors such as cameras or barcode scanners to recognize features on the appliances and then matching them to a pre-populated database to identify the appliance and how it is to be operated. These approaches lack generality, since every appliance will have its own unique features and it is possible that such matches might not exist in their proposed database.

II. Generic Cognitive Model

We propose a generic cognitive model for an appliance-agnostic visual learning procedure that will allow the system to identify, understand and ultimately operate an appliance, such as a refrigerators or a microwave oven. The cognitive model approach allows a system to describe an appliance at a high level of abstraction, focusing on a hierarchical definition of the appliance under observation, and provides a general interface for describing all the possible interactions with the appliance. This approach has the additional benefit of allowing development of modular and generic software packages that can be utilized by any robotic system for performing similar tasks.

A. Cognitive Model Description

Our proposed cognitive model is organized as a hierarchy of schemas, arranged in a top-down fashion based on the level of abstraction and generalization of description. The main idea behind our model is the common-sense observation that every appliance has a box-like geometry. All other operational

aspects of this "box", such as the handle or the door, are positionally constrained to it in a very specific manner. This top-level box has a certain fixed size as a property and has "attributes" associated with it.

Every appliance, depicted as a box, has a set of "common" attributes associated with it, by virtue of its design and intended operation. These attributes include: *1)* an opening or a cavity, *2)* a door, *3)* a handle, and *4)* a control panel.

Each of these attributes have a fixed world location, specified relative to the box and has other task-specific properties associated with it. Correctly identifying these properties allow the system to generate a cognitive model that can specify, without explicitly requiring knowledge of the appliance itself, the possible tasks that can be performed with the appliance.
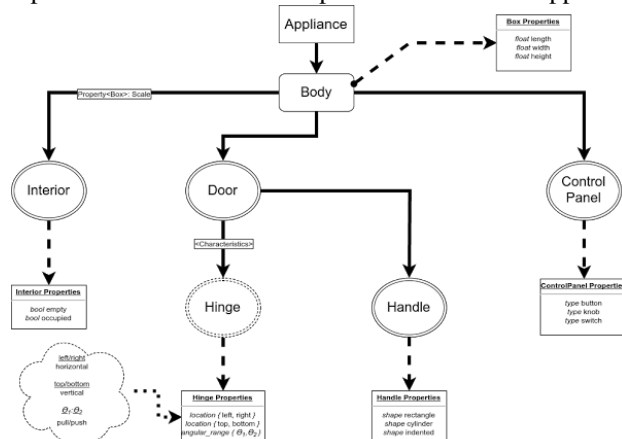

Fig. 1. Cognitive model hierarchy.

The cognitive model is generated according to the hierarchy specified in Fig. 1. From the figure, the topmost level of the hierarchy is populated by the box-like structure of the **Body** of the appliance itself. This level is characterized by the size of the appliance and its location in the world.

Following this, the hierarchy branches out into the **Interior**, also known as the internal cavity of the appliance. This interior cavity is also a box-like structure, usually having similar properties as the global box structure but having scaled-down property values. This **Interior** cavity is characterized by its relative location to the **Body** and has attributes of being `empty` or `occupied`.

Another child of the **Body**, we have the **Door**, having a relative positional constraint to the **Body**. The **Door** is character-

ized by the relative location of its sub-property, the **Hinge**. The **Door's** affordance, i.e. the property that allows the door to be manipulated, is decided by the properties of the **Hinge**, which has attributes of being located either at the `Top|Bottom` or `Left|Right` relative to the **Door**. This positional attribute dictates whether the **Door** will open horizontally or vertically and also determines the rotational/translational range of the **Door**.

The **Door** has another crucial sub-property, the **Handle**, which along with the **Hinge** also determines the affordance. The **Handle** is also relatively positioned to the **Door** and has attributes that describe the 'type' of handle present, namely a `rectangular` shape, a `cylindrical` shape or an `indented` handle.

The **Body** has another child, the **Control Panel**, which allows for operation of the appliance via its electronic control system. This control panel is characterized by the input it allows, such as via `knobs`, `switches` or `buttons`.

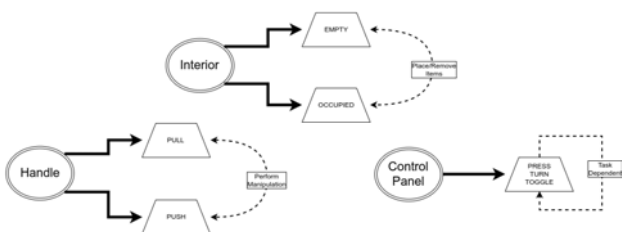### B. Using the Cognitive Model

Fig. 2. Possible state transitions for each of the appliance parts.

Once a cognitive model for an appliance is generated, we can use the inherent properties of each sub-part of the appliance to allow the robot to operate the appliance on a task-by-task basis. Assuming a proper implementation of the manipulation mechanism, it becomes possible for the robot to start using the appliance based on the cognitive model. This is facilitated by using the cognitive model as a state machine, as illustrated in Fig. 2, where every sub-part of the appliance body has a set of states associated with it, depending on the situation. For instance, the **Interior** cavity of a microwave can either be `occupied` or `empty`, which will consequently dictate whether an object can be placed inside the cavity or be removed from it. A **Handle** can be either `pulled` or `pushed` and a **Control Panel** can be operated by `pressing|turning|toggling`, depending on the task.

III. MODEL GROUNDING BY OBSERVATION

We implement a system that automatically grounds instances of the abstract cognitive model described above, given visual observations of humans operating the appliance under scrutiny. The input to our system is a recording, via a robot-mounted RGB-D camera, of a human demonstrating how to operate an appliance once. After several stages of offline processing, our system outputs a hierarchical structure that is a grounded instance of our proposed appliance cognitive model. Specifically, it both populates each individual part of the model with a suitable 3D geometric primitive in its correct position

relative to the object reference frame and encodes affordances of movable parts by providing hinge connection specifications (e.g., a hinge for a door). Processing stages can be summarized as follows:

1) Fusion of the depth maps into a single point cloud while the entire observed scene remains rigid. Frames involving independently moving areas are flagged as *dynamic* and involve moving appliance parts. Every non-dynamic interval will result in a 3D point cloud *reconstruction* of a different (static) state of the appliance under observation.

2) Fitting of geometric primitives for all reconstructed point clouds. The first reconstruction corresponds to the initial appliance state (with all doors closed) and is used to fit the appliance exterior box. Subsequent reconstructions will be registered to the same coordinate frame and used to fit primitives to the (now visible) appliance cavities.

3) Dynamic segments are used to model moving parts and their joint types. Each segment is expected to capture the movement of a single appliance part. The part's motion is classified as either rotational (e.g., for a fridge door) or translational (e.g., for a drawer) and the estimated hinge parameters are stored in the appliance model.
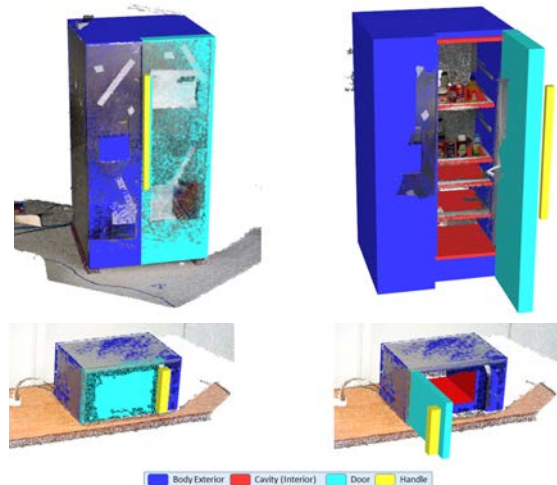

Fig. 3. Grounded model visualization for a fridge and a microwave oven.

IV. RESULTS

In Fig. 3, we depict grounded instances of our cognitive model for two appliances: a fridge and a microwave oven. The left column shows models in 'closed' state, overlaid to the exterior reconstructed point cloud. The right shows each appliance model rendered as 'open' with interior boxes becoming visible, showing detected affordances of the moving parts.

REFERENCES

[1] A. Hudnut and W. Gross, "Vision-enabled household appliances," Mar. 8 2011, uS Patent 7,903,838. [Online]. Available: https://www.google.com/patents/US7903838

[2] A. Enslin, "Vorrichtung und verfahren zum ansteuern eines haushaltsgeräts sowie ein haushaltsgerät apparatus and method for controlling a household appliance and a household appliance," Mar. 3 2016, dE Patent App. DE201,410,112,375. [Online]. Available: https://www.google.com/patents/DE102014112375A1?cl=en

Cognitive Control and Adaptive Attentional Regulations for Robotic Task Execution

Riccardo Caccavale and Alberto Finzi
DIETI, Università degli Studi di Napoli Federico II,
via Claudio 21, 80125, Naples, Italy.
{riccardo.caccavale, alberto.finzi}@unina.it

Abstract—We propose a robotic cognitive control framework that exploits supervisory attention and contention scheduling for flexible and adaptive orchestration of structured tasks. Specifically, in the proposed system, top-down and bottom-up attentional processes are exploited to modulate the execution of hierarchical robotic behaviors conciliating goal-oriented and reactive behaviors. In this context, we propose a learning method that allows us to suitably adapt task-based attentional regulations during the execution of structured activities.

I. Introduction

In this paper, we present a robotic cognitive control framework that permits flexible and adaptive orchestration of multiple structured tasks. Following a supervisory attentional system approach [12], [7], we propose an executive system that exploits top-down (task-based) and bottom-up (stimulus-based) attentional mechanisms to conciliate reactive and goal-oriented behaviors [4], [5]. In this paper, we describe adaptive mechanisms suitable for this framework. Specifically, we propose a learning method that allows us to regulate the top-down and bottom-up attentional influences according to the environmental state and the tasks to be accomplished. In contrast with typical task-learning approaches [11], [6], our aim here is to adapt and refine attentional parameters that affect the competition among active tasks and reactive processes. Learning methods for robotic supervisory attentional system have been proposed to enhance action execution automaticity and reduce the need of attentional control [8], instead here we are interested in flexible orchestration of hierarchical tasks.

In the following sections, we present the architecture of the executive system and briefly introduce the associated adaptive mechanisms.

II. System Architecture

The cognitive control framework presented in this paper is based on a supervisory attentional system that regulates the execution of hierarchical tasks and reactive behaviors. The system architecture is illustrated in Fig. 1. The attentional executive system is endowed with a long term memory (LTM) that contains the behavioral repertoire available to the system, including structured tasks and primitive actions; these tasks/behaviors are to be allocated and instantiated in the Working Memory (WM) for their actual execution. In particular, the cognitive control cycle is managed by the *alive* process that continuously updates the WM by allocating and deallocating hierarchical tasks/behaviors according to their denotations

in the LTM. We assume a hierarchical organization for tasks and activities [9], [12], [7] and this hierarchy is represented in the WM as a tree data structure that collects all the tasks currently executed or ready for the execution (see Fig. 2). More specifically, each node of the tree stands for a behavior with the edges representing parental relations among sub-behaviors. In this context, *abstract* behaviors identify complex activities to be hierarchically decomposed into different sub-activities, instead *concrete* behaviors are for sensorimotor processes that compete for the access to sensors and actuators. The allocated concrete behaviors are collected into the attentional behavior-based system illustrated in Fig. 1.
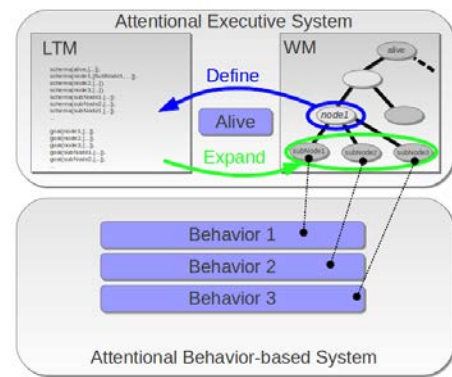

Fig. 1. System Architecture. The LTM provides the definitions of the available tasks, which can be allocated/deallocated in the WM by the *alive* behavior.

In this framework, each behavior is associated with an activation value, which is regulated by an adaptive clock [3], [2]. This clock represents a frequency-based attentional mechanism: the higher is the frequency, the higher is the resolution at which a process/behavior is monitored and controlled. The clock period is bottom-up and top-down regulated by a behavior-specific *monitoring function* $f(\sigma, \varepsilon) = \lambda$ according to the behavioral stimuli σ and the overall state of the WM ε. In particular, the bottom-up frequency $1/\lambda$ is directly affected by behavior-specific stimuli (e.g. distance of a target), while the top-down regulation is provided by a value μ that summarizes the overall top-down influences of the WM. In this context, bottom-up stimuli emphasize actions that are more accessible to the robot (e.g. object affordances), while top-down influences are affected by the task structures and facilitate

the activations of goal-oriented behaviors. In this framework, multiple tasks can be executed at the same time and several behaviors can compete in the WM generating conflicts, impasses, and crosstalk interferences [10], [1]. Contentions among alternative behaviors competing for mutually exclusive state variables (representing resources, e.g. sensors, actuators, etc.) are solved exploiting the attentional activations: following a winner-takes-all approach, the behaviors associated with the higher activations are selected with the exclusive access to mutually exclusive resources.

III. ADAPTIVE REGULATIONS

In the proposed framework, action selection depends on the combined effect of top-down and bottom-up attentional regulations. In order to set these regulations, we associate each edge of the WM with a weight $w_{j,i}$ that regulates the intensity of the attentional influence from the behavior j to the sub-behavior i (bottom-up for $i = j$, top-down otherwise). This way, the overall activation value associated with each node is obtained as the weighted sum $\sum_j w_{i,j} c_{i,j}$ of the contributions from the top-down and bottom-up sources. These weights are to be suitably adapted with respect to the tasks and the environment. For this purpose, we propose to deploy a neural network approach. Specifically, during the execution the WM tree is associated with a multi-layered neural network, while the weights associated with the nodes are refined exploiting error backpropagation. In this setting, the system can be trained by a user that takes the control of the robot to correct the execution of a task. The training session is associated with an adaptive process: the difference between the system behavior and the human correction is interpreted as an error to be backpropagated through the task hierarchy in order to adapt the associated weights.
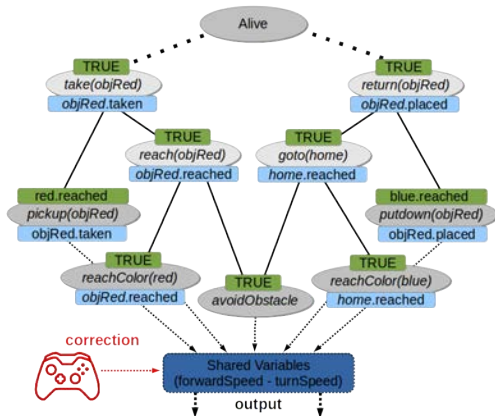

Fig. 2. WM configuration during the execution of a take-and-return task. The contending behaviors (leafs of the hierarchy) receive inputs from the upper nodes (black links) producing output values for the shared variables (dark blue box). The user can correct the execution (joypad) to train the system.

As an exemplification, we consider the instance of the WM illustrated in Fig. 2. In this case, a mobile robot has to take a colored object ($objRed$) and return it to the home position. Here, five concrete behaviors compete to acquire two

contended variables ($forwardSpeed$ and $turnSpeed$) which are used to control the robots movements. For instance, the $avoidObstacle$ behavior is affected by two top-down influences ($reach(objRed)$ and $goto(home)$ subtasks), while the bottom-up influence is inversely proportional to the distance of the closest obstacle. During the execution of the task, the system selects the most emphasized behavior and produces a vector of values $\vec{v}$ representing motor patterns for the shared variables. The robot navigation is monitored by the human, which is ready to change the robot trajectory using a joypad when a correction is needed. The user interventions generate a new set of values for the shared variables $\vec{v}^*$ that dominate and override the ones produced by the other behaviors. As long as the user drives the robot, the difference between the systems output $\vec{v}$ and the suggested values $\vec{v}^*$ is exploited to estimate the total error of the task execution. This error is backpropagated from the concrete behaviors to the rest of the hierarchy, in so adjusting the weights associated with the behavior and sub-behaviors which are active in the WM.

IV. CONCLUSIONS

We presented an adaptive mechanism suitable for a cognitive control framework based on a supervisory attentional system approach. The proposed method permits adaptive and interactive adaptation of top-down and bottom-up attentional regulations in order to execute structured hierarchical tasks.

REFERENCES

[1] M. M. Botvinick, T. S. Braver, D. M. Barch, C. S. Carter, and J. D. Cohen, "Conflict monitoring and cognitive control." *Psychological review*, vol. 108, no. 3, p. 624, 2001.

[2] X. Broquère, A. Finzi, J. Mainprice, S. Rossi, D. Sidobre, and M. Staffa, "An attentional approach to human-robot interactive manipulation," *I. J. Social Robotics*, vol. 6, no. 4, pp. 533–553, 2014.

[3] E. Burattini, S. Rossi, A. Finzi, and M. C. Staffa, "Attentional modulation of mutually dependent behaviors," in *Proc. of SAB 2010*, 2010, pp. 283–292.

[4] R. Caccavale and A. Finzi, "Plan execution and attentional regulations for flexible human-robot interaction," in *Proc. of SMC 2015*, 2015, pp. 2453–2458.

[5] ——, "Flexible task execution and attentional regulations in human-robot interaction," *IEEE Trans. Cognitive and Developmental Systems*, vol. 9, no. 1, pp. 68–79, 2017.

[6] G. Chang and D. Kulić, "Robot task learning from demonstration using petri nets," in *2013 IEEE RO-MAN*. IEEE, 2013, pp. 31–36.

[7] R. P. Cooper and T. Shallice, "Hierarchical schemas and goals in the control of sequential behavior," *Psychological Review*, vol. 113(4), pp. 887–916, 2006.

[8] J. Garforth, S. L. McHale, and A. Meehan, "Executive attention, task selection and attention-based learning in a neurally controlled simulated robot." *Neurocomputing*, vol. 69, no. 16-18, pp. 1923–1945, 2006.

[9] Lashley, "The problem of serial order in behavior," in *Cerebral Mechanisms in Behavior, Wiley, New York*, L. In Jeffress, Ed., 1951.

[10] M. C. Mozer and M. Sitton, "Computational modeling of spatial attention," *Attention*, vol. 9, pp. 341–393, 1998.

[11] M. N. Nicolescu and M. J. Mataric, "Natural methods for robot task learning: Instructive demonstrations, generalization and practice," in *Proc. of AAMAS 2003*. ACM, 2003, pp. 241–248.

[12] D. A. Norman and T. Shallice, "Attention to action: Willed and automatic control of behavior," in *Consciousness and self-regulation: Advances in research and theory*, 1986, vol. 4, pp. 1–18.

Solve Memory to Solve Cognition

Paul Baxter

Lincoln Centre for Autonomous Systems

School of Computer Science

University of Lincoln, U.K.

Email: pbaxter@lincoln.ac.uk

Abstract—The foundations of cognition and cognitive behaviour are consistently proposed to be built upon the capability to predict (at various levels of abstraction). For autonomous cognitive agents, this implicitly assumes a foundational role for memory, as a mechanism by which prior experience can be brought to bear in the service of present and future behaviour. In this contribution, this idea is extended to propose that an active process of memory provides the substrate for cognitive processing, particularly when considering it as fundamentally associative and from a developmental perspective. It is in this context that the claim is made that in order to solve the question of *cognition*, the role and function of *memory* must be fully resolved.

I. PREDICTION, COGNITION, AND MEMORY

There are a range of competencies that are involved in cognition: an ongoing challenge is to identify common functional and organisational principles of operation. This will facilitate both the understanding of natural cognition (particularly that of humans), and the creation of synthetic artefacts that can be of use to individuals and society. One such principle is that of prediction [1], prospection [2], or indeed simulation [3], as being fundamental to cognition. A further requirement is the need to incorporate an account of development [4] as a means of an individual to gain cognitive competencies through experience (of the physical and social world), rather than *a priori* programming.

It is suggested that one common dependency of these principles is a requirement for memory. At this point, the definition of memory provided is only in the broadest sense: i.e. memory is a process that acquires information through experience in the service of current and future behaviour [5]. While broad, it nevertheless commits to a fundamental function/role for memory in behaviour [6]. It is on this basis that the remainder of this contribution is focused: taking memory as fundamental, how can it be characterised such that it serves cognition (and the development thereof)?

In one particular perspective grounded in neuropsychological data, emphasis is placed on the associative and network nature of memory. This is apparent in the "Network Memory" framework for example [7], which proposes a hierarchical and heterarchical organisation of overlapping distributed associative networks that that formed through experience, and whose reactivation gives rise to the dynamics that instantiate cognition [8]. Such a perspective is not unusual, e.g. [1], despite the apparent contradiction to multi-system accounts of memory organisation, e.g. [9], [10], with it being also

consistent with more purely theoretical considerations, e.g. [11], that emphasise the dynamical process properties of memory over passive information storage.

By taking on this interpretation of memory, a more refined process definition memory may be ventured: memory is a distributed associative structure that is created through experience (the formation associations), and which forms the substrate for activation dynamics (through externally driven activity, and internal reactivation) that gives rise to cognitive processing [12], [5]. The creation of structure through experience is consistent with developmental accounts, and enforce the consideration of not only interaction with the environment, but also the social context of the learning agent (if human-like cognition is to be considered). Previous explorations have suggested how this framework can be used (in principle) to account for human-level cognitive competencies within a memory-centred cognitive architecture [13], although there remain many gaps in this account that require addressing before it can be considered definitive.

II. APPLICATION AND IMPLICATIONS

Following this definition, take for example the role that such a memory-centred cognitive architecture could play in facilitating social robot behaviour, as a prototypical example of a cognitive competence that needs to be fulfilled. It is uncontroversial to suggest that humans incrementally acquire social skills (though perhaps based on some inherently present mechanisms) over time and through development. The role of memory within this is therefore also not controversial, particularly when skills such as intent prediction (based on prior experience) are also considered [14]. Using an associative network that learns from the behaviour of the interaction partner [15], following the use of simple associative learning in [16], it has been found that a degree of behavioural alignment between a child and a robot is observed within real-time interactions - an effect readily seen in human-human interactions. While only a basic illustration of human-like competence, this nevertheless demonstrates the importance of memory for social HRI [17], and thus establishes associativity as a candidate foundational mechanism for a social cognitive architecture. Similarly, with associativity being considered sufficient for generating predictions as noted above, and prediction/anticipation being considered essential for sociality in terms of supporting coordination [18], then such an account of memory remains consistent.

An alternative implementation using similar principles of associativity and interactive learning has been applied to a range of embodied and developmental psychology models related to language. The Epigenetic Robotics Architecture (ERA) [19] emphasises associative learning, and is instantiated through linked self-organising maps (SOM), arranged through a "hub" SOM that learns from body posture. This structure, learning from a blank initial state, can provide an account of how aspects of language can extend cognitive processing [20], and of how word learning in infants is mediated by body posture [21]. In each of these examples, the computational instantiation of ERA is the same, but the functionality observed differs based on the interaction context of the experiment. Given the fundamentally associative nature of the learning process, this is consistent with the memory-centred account of social human-robot interaction competence described above.

In many principled but low-level approaches – including the those systems based on the developmental systems paradigm, as subscribed to here – there is often a gap between the theoretical consistency and the complexity of the applied resulting system, with simplified (or rather constrained) problems typically targeted. While the memory-centred approach advocated here suffers similarly, the range of applications outlined in the previous paragraphs cover a number of aspects of "higher level" (indeed, human-level) cognition that go beyond the typical domains for low-level associative systems. The efforts described here remain relatively sparse and currently lack a computational integration into a single coherent system that existing psychologically-derived cognitive architectures (such as SOAR, ACT-R, etc) attempt. Nevertheless, there appears to be a convergence of principles of operation that the present work seeks to extend: cognition founded on formation and manipulation of memory, and memory as associative and developmental. At the least, what is proposed here is a re-framing of the problem: not to look at cognition from the perspective of the 'computation' or the behavioural outcome as is typical, but rather to re-evaluate the problem from the perspective of memory.

III. THE SUFFICIENCY OF AN ACCOUNT OF MEMORY

The outcome of this discussion is a commitment to a fundamentally associative structure of memory, with this maintaining consistency with the developmental perspective, and as illustrated through the social human-robot interaction and language examples. The outline described in this abstract points to a framework within which the relationship between memory and cognition can be understood, although there remain a number of open questions that need to be resolved, such as reconciliation with empirical evidence supporting the multi-systems organisation of memory, e.g. [10], and the interplay of memory with non-memory mechanisms underlying cognition (such as affective processes, e.g. [22]). Nevertheless, the proposal is that even these aspects could be approached from the perspective of memory. In all, this leads to the view that in order to 'solve' cognition, the problem of memory must be fully resolved. Indeed, the suggestion of the present

contribution goes beyond this: that a full account of memory may be *sufficient* to provide an account of cognition.

REFERENCES

[1] M. Bar, "The proactive brain: using analogies and associations to generate predictions," *Trends in Cognitive Sciences*, vol. 11, no. 7, pp. 280–289, 2007.

[2] D. Vernon, M. Beetz, and G. Sandini, "Prospection in Cognition: The Case for Joint Episodic-Procedural Memory in Cognitive Robotics," *Frontiers in Robotics and AI*, vol. 2, no. July, pp. 1–14, 2015.

[3] G. Hesslow, "Conscious thought as simulation of behaviour and perception," *Trends in cognitive sciences*, vol. 6, no. 6, pp. 242–247, 2002.

[4] J. Weng, J. McClelland, A. Pentland, O. Sporns, I. Stockman, M. Sur, and E. Thelen, "Autonomous Mental Development by Robots and Animals," *Science*, vol. 291, pp. 599–600, 2001.

[5] R. Wood, P. Baxter, and T. Belpaeme, "A Review of long-term memory in natural and synthetic systems," *Adaptive Behavior*, vol. 20, no. 2, pp. 81–103, 2012.

[6] A. M. Glenberg, "What Memory is For," *The Behavioral and Brain Sciences*, vol. 20, no. 1, pp. 1–19; discussion 19–55, 1997.

[7] J. M. Fuster, "Network Memory," *Trends in Neurosciences*, vol. 20, no. 10, pp. 451–9, 1997.

[8] J. M. Fuster and S. L. Bressler, "Past Makes Future: Role of pFC in Prediction," *Journal of Cognitive Neuroscience*, vol. 27, no. 4, pp. 639–654, 2015.

[9] L. R. Squire, "Memory systems of the brain: a brief history and current perspective.," *Neurobiology of learning and memory*, vol. 82, no. 3, pp. 171–7, 2004.

[10] G. Repovs and A. Baddeley, "The multi-component model of working memory: explorations in experimental cognitive psychology," *Neuroscience*, vol. 139, no. 1, pp. 5–21, 2006.

[11] A. Riegler, "Constructive memory," *Kybernetes*, vol. 34, no. 1, pp. 89–104, 2005.

[12] P. Baxter and W. Browne, "Memory as the substrate of cognition: a developmental cognitive robotics perspective," in *Proceedings of the Tenth International Conference on Epigenetic Robotics* (B. Johansson, E. Sahin, and C. Balkenius, eds.), (Örenäs Slott, Sweden), pp. 19–26, 2010.

[13] P. Baxter, R. Wood, A. Morse, and T. Belpaeme, "Memory-Centred Architectures: Perspectives on Human-level Cognitive Competencies," in *Proceedings of the AAAI Fall 2011 symposium on Advances in Cognitive Systems* (P. Langley, ed.), (Arlington, Virginia, U.S.A.), pp. 26–33, AAAI Press, 2011.

[14] Y. Demiris, "Prediction of intent in robotics and multi-agent systems," *Cognitive Processing*, vol. 8, no. 3, pp. 151–8, 2007.

[15] P. E. Baxter, J. de Greeff, and T. Belpaeme, "Cognitive architecture for humanrobot interaction: Towards behavioural alignment," *Biologically Inspired Cognitive Architectures*, vol. 6, pp. 30–39, 2013.

[16] K. Dautenhahn and A. Billard, "Studying robot social cognition within a developmental psychology framework," in *Third European Workshop on Advanced Mobile Robots (Eurobot'99)*, (Zurich, Switzerland), pp. 187–194, 1999.

[17] P. Baxter, "Memory-Centred Cognitive Architectures for Robots Interacting Socially with Humans," in *2nd Workshop on Cognitive Architectures for Social Human-Robot Interaction at HRI'16*, (Christchurch, New Zealand), 2016.

[18] E. Di Paolo and H. De Jaegher, "The interactive brain hypothesis," *Frontiers in Human Neuroscience*, vol. 6, pp. 1–16, 2012.

[19] A. F. Morse, J. De Greeff, T. Belpaeme, and A. Cangelosi, "Epigenetic Robotics Architecture (ERA)," *IEEE Transactions on Autonomous Mental Development*, vol. 2, no. 4, pp. 325–339, 2010.

[20] A. F. Morse, P. Baxter, T. Belpaeme, L. B. Smith, and A. Cangelosi, "The Power of Words," in *Joint IEEE International Conference on Development and Learning and on Epigenetic Robotics*, (Frankfurt am Main, Germany), pp. 1–6, IEEE Press, 2011.

[21] A. F. Morse, V. L. Benitez, T. Belpaeme, A. Cangelosi, and L. B. Smith, "Posture affects how robots and infants map words to objects," *PLoS ONE*, vol. 10, no. 3, 2015.

[22] A. R. Damasio, "The somatic marker hypothesis and the possible functions of the prefrontal cortex," *Philosophical Transactions Of the Royal Society B*, vol. 351, pp. 1413–1420, 1996.

ABOD3:
A Graphical Visualisation and Real-Time Debugging Tool for BOD Agents

Andreas Theodorou
Department of Computer Science
University of Bath
Bath, BA2 7AY UK
Email: a.theodorou@bath.ac.uk

I. INTRODUCTION

Current software for AI development requires the use of programming languages to develop intelligent agents. This can be disadvantageous for AI designers, as their work needs to be debugged and treated as a generic piece of software code. Moreover, such approaches are designed for experts; often requiring a steep initial learning curve, as they are tailored for programmers. This can be also disadvantageous for implementing transparency to agents, an important ethical consideration [1], [2], as additional work is needed to expose and represent information to end users.

We are working towards the development of a new editor, ABOD3. It allows the graphical visualisation of Behaviour Oriented Design based plans [3], including its two major derivatives: Parallel-rooted, Ordered Slip-stack Hierarchical (POSH) and Instinct [4].

The new editor is designed to allow not only the development of reactive plans, but also to debug such plans in real time to reduce the time required to develop an agent. This allows the development and testing of plans from a same application.

II. BEHAVIOUR ORIENTED DESIGN

Behaviour Oriented Design (BOD) [5], [6] takes inspiration both from the well-established programming paradigm object-oriented design (ODD) and Behaviour-Base AI (BBAI) [7], to provide a concrete architecture for developing complete, complex agents (CCAs), with multiple conflicting goals and mutual-exclusive means of achieving those goals. BBAI was first introduced by Brooks [7], where intelligence is decomposed into simple, robust modules, each expressing capabilities, actions such as movement, rather than mental entities such as knowledge and thought.

Bryson's BOD is a cognitive architecture which promotes behaviour decomposition, code modularity and reuse, making the development of intelligent agents easier. BOD describes the agent's behaviour into multiple modules forming a behaviour library. Each module can have a set of expressed behaviours called acts, actions, perceptions, learning, and memory. Behaviour modules also store their own memories, i.e. sensory experiences.

Action selection is forced by competition for resources. If no such competition exists, the behaviour modules are able to work in parallel enforcing the long-term goals of the agent

A. POSH

POSH planning is the action selection for reactive planning derivative of BOD. POSH combines faster response times similar to reactive approaches for BBAI with goal-directed plans. A POSH plan consists of the following plan elements:

1) Drive Collection (DC): It contains a set of Drives and is responsible for giving attention to the highest priority Drive. To allow the agent to shift and focus attention, only one Drive can be active in any given cycle.
2) Drive (D): Allows for the design and pursuit of a specific behaviour as it maintains its execution state. The trigger is a precondition, a primitive plan element called *Sense*, determining if the drive should be executed by using a sensory input.
3) Competence (C): Contains one or more Competence Elements (CE), each of which has a priority and a releaser. A CE may trigger the execution of another Competence, an Action Pattern, or a single Action.
4) Action Pattern (AP): Used to reduce the computational complexity of search within the plan space and to allow a coordinated fixed sequential execution of a set of Actions.

B. Instinct

The Instinct Planner is a reactive planner based on the POSH planner. It includes several enhancements taken from more recent papers extending POSH [8]. In an Instinct plan, an AP contains one or more Action Pattern Elements (APE), each of which has a priority, and links to a specific Action, Competence, or another AP.

III. THE PLAN EDITOR

The editor provides a customisable user interface (UI) aimed at supporting both the development and debugging of agents. Plan elements, their subtrees, and debugging-related information can be hidden, to allow different levels of abstraction and

present only relevant information. The graphical representation of the plan can be generated automatically, and the user can override its default layout by moving elements to suit his needs and preferences. The simple UI and customisation allows the editor to be employed not only as a developer's tool, but also to present transparency related information to the end users, helping them to develop more accurate mental models of the agent.

Alpha testers have already used ABOD3 in experiments to determine the effects of transparency on the mental models formed by humans [9], [10]. Their experiments consisted of a non-humanoid robot, powered by the BOD-based Instinct reactive planner. They have demonstrated that subjects, if they also see an accompanying display of the robot's real-time decision making as provided by ABOD3, can show marked improvement in the accuracy of their mental model of a robot observed. They concluded that providing transparency information by using ABOD3 does help users to understand the behaviour of the robot, calibrating their expectations.

Plan elements flash as they are called by the planner and glow based on the number of recent invocations of that element. Plan elements without any recent invocations start dimming down, over a user-defined interval, until they return back to their initial state. This offers abstracted backtracking of the calls. Sense information and progress towards a goal are displayed. Finally, ABOD3 provides integration with videos of the agent in action, synchronised by the time signature within the recorded transparency feed.
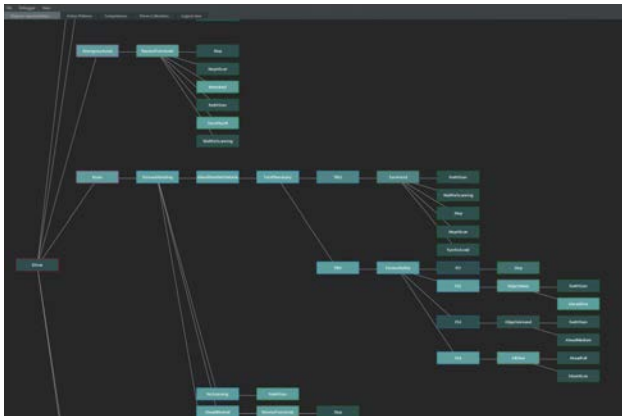

Fig. 1. The ABOD3 Graphical Transparency Tool displaying an Instinct plan in debugging mode. The highlighted elements are the ones recently called by the planner. The intensity of the glow indicates the number of recent calls.

ABODE3 provides an API that allows the editor to connect with planners, presenting debugging information in real time. For example, it can connect to the Instinct planner by using a built-in TCP/IP server, see Figure 2.

IV. CONCLUSION

We plan to continue developing this new editor, implementing debug functions such as "fast-forward" in pre-recorded log files and usage of breakpoints in real-time. A transparent agent, with an inspectable decision-making mechanism, could
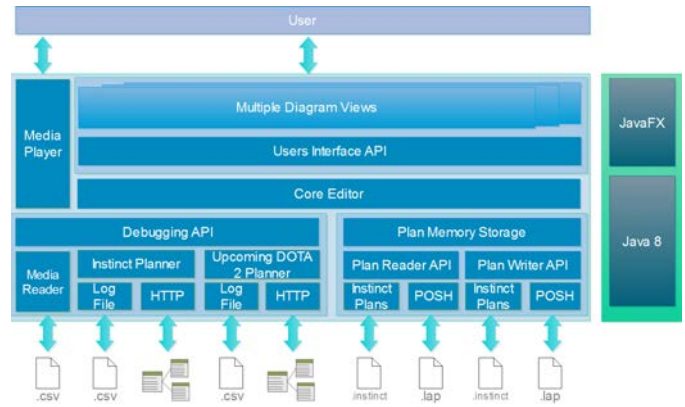

Fig. 2. System Architecture Diagram of ABOD3, showing its modular design. All of ABOD3 was written in Java, to ensure cross-platform compatibility. APIs allows the expansion of the software to support additional BOD planners for real-time debugging, BOD based plans, and User Interfaces. The editor aims, through personalisation, to support roboticists, games AI developers, and even end users.

also be debugged in a similar manner to the way in which traditional, non-intelligent software is commonly debugged. The developer would be able to see which actions the agent is selecting, why this is happening, and how it moves from one action to the other. This is similar to the way in which popular Integrated Development Environments (IDEs) provide options to follow different streams of code with debug points. Moreover, we will enhance its plan design capabilities by introducing new views, to view and edit specific types of plan-elements and through a public beta testing to gather feedback by both experienced and inexperienced AI developers.

REFERENCES

[1] A. Theodorou, R. H. Wortham, and J. J. Bryson, "Designing and implementing transparency for real time inspection of autonomous robots," *Connection Science*, vol. 29, 2017.

[2] R. H. Wortham, A. Theodorou, and J. J. Bryson, "Robot Transparency , Trust and Utility," in *ASIB 2016: EPSRC Principles of Robotics*, 2016.

[3] J. Bryson, "The behavior-oriented design of modular agent intelligence," in *System*, 2002, vol. 2592, pp. 61–76.

[4] R. H. Wortham, S. E. Gaudl, and J. J. Bryson, "Instinct : A Biologically Inspired Reactive Planner for Embedded Environments," in *Proceedings of ICAPS 2016 PlanRob Workshop*, 2016.

[5] J. Bryson and L. A. Stein, "Intelligence by Design : Principles of Modularity and Coordination for Engineering Complex Adaptive Agents by," no. September 2001, 2001.

[6] J. J. Bryson, "Action selection and individuation in agent based modelling," in *Proceedings of Agent 2003: Challenges in Social Simulation*, D. L. Sallach and C. Macal, Eds. Argonne, IL: Argonne National Laboratory, 2003, pp. 317–330.

[7] R. A. Brooks, "Intelligence Without Representation," *Artificial Intelligence*, vol. 47, no. 1, pp. 139–159, 1991.

[8] S. Gaudl and J. J. Bryson, "The Extended Ramp Goal Module: Low-Cost Behaviour Arbitration for Real-Time Controllers based on Biological Models of Dopamine Cells," *Computational Intelligence in Games 2014*, 2014. [Online]. Available: http://opus.bath.ac.uk/40056/

[9] R. H. Wortham, A. Theodorou, and J. J. Bryson, "What Does the Robot Think? Transparency as a Fundamental Design Requirement for Intelligent System," in *IJCAI-2016 Ethics for Artificial Intelligence Workshop*, 2016.

[10] ——, "Improving Robot Transparency : Real-Time Visualisation of Robot AI Substantially Improves Understanding in Naive Observers {submitted}," 2017.

Functional Design Methodology for Customized Anthropomorphic Artificial Hands

Muhammad Sayed, Lyuba Alboul and Jacques Penders
Materials and Engineering Research Institute, Sheffield Hallam University, UK
Sheffield Robotics, UK
muhammad.b.h.sayed@gmail.com; L.Alboul@shu.ac.uk; J.Penders@shu.ac.uk;

Abstract—**This short paper outlines a framework for an evaluation method that takes as an input a model of an anthropomorphic artificial hand and produces as output the set of tasks that it can perform. The framework is based on the anatomy and functionalities of the human hand and methods of implementing these functionalities in artificial systems and focuses on the evaluation of the intrinsic hardware of robot hands. The paper also presents a partial implementation of the framework: a method to evaluate anthropomorphic postures using Fuzzy logic and a method to evaluate anthropomorphic grasping abilities. The methods are applied on models of the human hand and the InMoov robot hand; results show the methods' ability to detect successful postures and grasps**.

Keywords— *Haptic Feedback, Haptic Rein, Navigation*

I. INTRODUCTION

The human hand is considered the most dexterous and sophisticated manipulator currently existing. Robotics developers naturally look towards the human hand for inspiration when designing robotic end-effectors. This inspiration varies from imitating its shape to attempting to replicate its functionality.

Hand *construction* (anatomy) gives rise to *capabilities.* Hand capabilities can be *motion* or *sensory*. Hand construction components can be categorised into *structure, [contact] surfaces, sensors,* and *actuation* components.

Consequently, functionalities can be categorized according to task aim into *information exchange (sensing), static grasping, within-hand manipulation, force exchange, or visual expression* [1].

We propose a framework for an evaluation method of functionalities of anthropomorphic artificial hands based on the anatomy and functionalities of a human hand.

II. THEORETIC CONSIDERATION

Anthropomorphic artificial hands "should" approximate the human hand *physically* and *functionally;* therefore, understanding *anthropomorphism* requires understanding the construction and operation of both human and artificial hands. By analysing the construction and tasks of human and artificial hands, a relation can be established between physical components of the hand and the tasks it can perform.

A simulation method must be used to evaluate the performance of the hand at each type of tasks (categorised according to the task aim).

The overall performance of the hand can be correlated to individual components by analysis or by repeating the evaluation while changing the component, therefore establishing a value representing the contribution of individual components to the overall performance allowing optimisation of hand construction.

The method should be able to describe generic tasks as well as specific ones (i.e. allow for arbitrary task modelling)
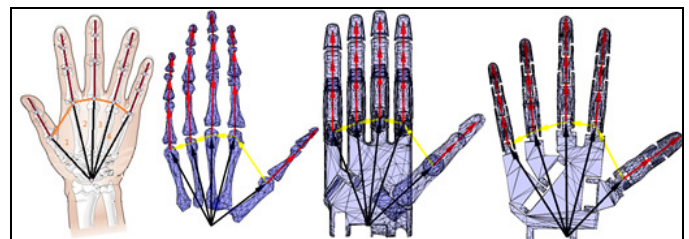

Fig. 1. Human and robotic hands "constructions"

III. PROTOTYPE DESIGN

Task description syntax is developed that describes the task in terms of

1) *Anthropomorphic postures* used to perform the task

2) *Objects* involved in the task and the *Interaction* with the objects (pose, information or force exchange, contact locations, prehension)

3) *Motion* required to perform the task

Postures are described using a syntax based on descriptions of British Sign Language (BSL) signs. A posture description takes the form of "hand/hand part(s) is/are at [state]" (for example: "The hand is [tightly closed] and the thumb is [across the fingers]") [2]

IV. EVALUATION

Evaluation of a posture is performed using Fuzzy logic, evaluation of prehension is performed using grasp quality metrics.

The evaluation process scans the configuration space of the hand and compares the posture to the reference postures (from the task model) using a mapping based on the human hand skeleton.

The two methods are implemented in MATLAB and tested on models of the human hand, Shadow robot hand, and InMoov robot hand using the postures of the seventeen basic handshapes of BSL [2] and thirty-one of the grasps of Feix grasp taxonomy [3].

V. RESULTS AND DISCUSSION

The methods were tested by evaluating the performance of a human hand model. The results showed, as expected, that the model can perform all the reference tasks (which are known to be possible to perform using the human hand). Robot hands scored less, for example, the InMoov hand was only able to perform fourteen grasps, five of which had very poor anthropomorphism.

The hand configuration space is very large, scanning the entire space is time consuming, especially when it must be sampled at a fine resolution to allow valid contact on hands with rigid surfaces. This is even a bigger problem when the object itself has a large configuration space (range of possible poses w.r.t. the hand).

Using grasp quality metrics and not using a separate step to verify closure conditions leads to situations where the ability of the hand to grasp objects cannot be correctly determined.

VI. FUTURE WORK

Future research will focus on developing methods to evaluate the remaining functionalities (dexterous manipulation, active sensing, and non-prehensile manipulation). In order to achieve this we plan to perform the correlation analysis between hand components and hand functional performance to obtain values associated with the contribution of each component. Then we proceed with constructing a database containing hardware components, each pre-analysed and assigned *functional performance*, *compatibility*, and *cost (monetary, computational,* and *energy)* values for every defined hand function and other components. Based on the data in the database, new hands can be designed using a selection process that aims to maximise performance and compatibility sums while minimising cost sum.

REFERENCES

[1] Jones, L.A., Lederman, S.J.: Human Hand Function. Oxford University Press, 2006.

[2] C. Smith. *Let's sign dictionary: everyday BSL for learners*. Co-Sign Communications, Stocktonon Tees, 2009.

[3] T. Feix, J. Romero, C. H. Ek, H.-B. Schmiedmayer, and D. Kragic, "A metric for Comparing the Anthropomorphic Motion Capability of Artificial Hands," Robotics, IEEE Transactions on, vol. 29, no. 1, pp. 82-93, Feb. 2013.

Development of an Intelligent Robotic Rein for Haptic Control and Interaction with Mobile Machines

Musstafa Elyounnss, Alan Holloway, Jacques Penders, and Lyuba Alboul

Materials and Engineering Research Institute, Sheffield Hallam University, UK
Sheffield Robotics, UK
b2051861@my.shu.ac.uk; A.Holloway@shu.ac.uk; J.Penders@shu.ac.uk; L.Alboul@shu.ac.uk

Abstract—**The rescue services face numerous challenges while entering and exploring dangerous environments in low or no visibility conditions and often without meaningful auditory and visual feedback. In such situations, firefighters may have to rely solely on their own immediate haptic feedback in order to make their way in and out of a burning building by running their hands along the wall as a mean of navigation. Consequently the development of technology and machinery (robots) to support exploration and aid navigation would provide a significant benefit to search and rescue operations; enhancing the capabilities of the fire and rescue personal and increasing their ability to exit safely. In our research, the design of a new intelligent haptic rein is proposed. The design is inspired by how guide dogs lead visually impaired people around obstacles. Due to complexity, the system design is separated into distinct prototype systems: sensors and monitoring, motion/feedback and the combined system with adaptive shared control.**

Keywords— Haptic Feedback, Haptic Rein, Navigation

I. INTRODUCTION

In the experiments conducted, a problem appeared when the robot made sharp movements; the human user had significant difficulty following the trajectory of the robot. In such cases, the system could be improved through pre-emptive indications to the user and a mutual adaption of both the robot and user's response (speed and turning rate) should be taken into consideration in order to maintain consistent fluid locomotion.

Following on from the previous research it has been proposed that an intelligent stiff rein system with the feedback of the environment and perceptual capabilities can enable and enhance navigation in complex environments. Additionally the use of haptic communication through force feedback guiding the user can be considered as a suitable approach to providing navigation information and is the least affected mode of communication in noisy environments [1].

II. WORK DEFINITION

The work focusses on investigating and building a prototype mobile robotic rein, which aims to emulate the natural and adaptable control relationship observed between a guide dog and a human user during navigation in new environments as Figure 1 shows. The research continues the work established in the REINS project [2].
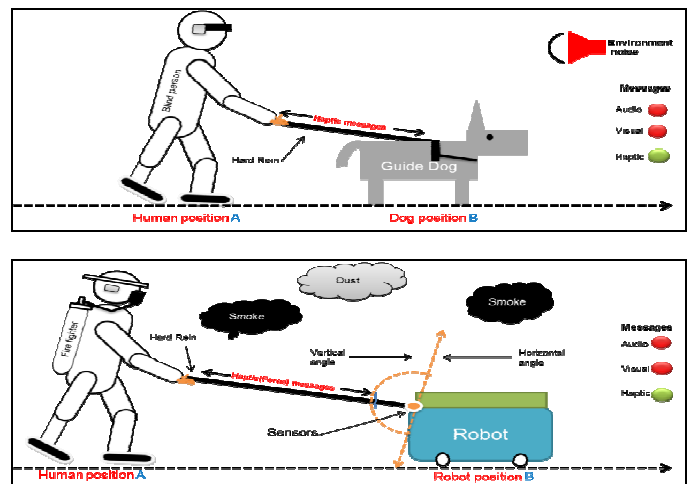

Fig. 1. Comparison between environments of a visually impaired and a firefighter

The proposed robotic rein will be designed and constructed to facilitate variable levels of haptic control and feedbacks allowing the user to either provide direct control over the proposed path of direction or the rein provide selective resistance/force. This will be achieved by using a high resolution stepper motor (position & torque control) in order to enforce the user response to the necessary change of route or direction determined by the mobile robot. In order to develop a prototype intelligent rein, detailed information about the relative positioning and compliance/resistance of the user to the robots responses must be known. This information is then be processed by the shared control system to provide adaptive control and force feedback. Figure 2 shows the stiff rein prototype with sensors and actuators (stepper motors) mounted on.

III. PROTOTYPE DESIGN

The sensor design measures the vertical and horizontal rein angle by using digital encoders. The data collected by the sensors is analyzed, processed and subsequently interpreted into movement of the rein to actively guide the human in the desired trajectory. Proportional levels of the torque are applied to describe the intensity and the amount of movement (haptic feedback) that the user must respond to. Monitoring the rein torque (user compliance and resistance) and their relative position to the robot will then provide feedback into the control

of the robot movement, adjusting the speed and rate of direction change accordingly. Both the sensor system and motor control is being implemented on a small embedded platform (National Instruments my RIO).
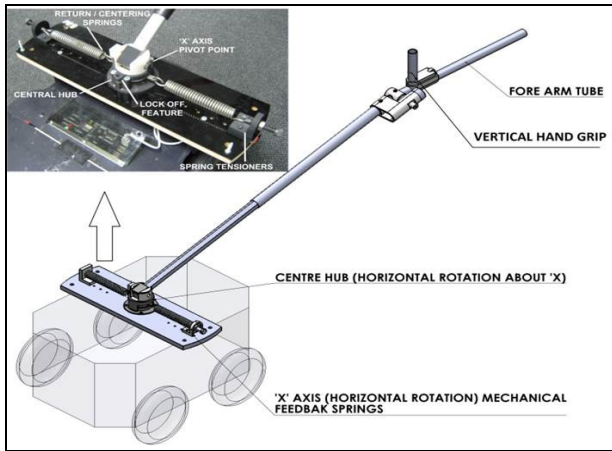

Fig. 2. Stiff rein prototype

IV. Current Work

The initial sensor system prototype has been completed and it provides detailed feedback on the human/rein interaction. Sensors have been mounted at all the flexing/moving joints and interfaced to a PC based logging system that will allow the capture of the rein kinematics during experiments. The work on the 2nd phase prototype, which aims to provide haptic force feedback, is nearly finished. Actuators are mounted and matched with the rein to give a specific movement as a haptic force feedback to user forearm with controlling the speed and angle. Figure 3 shows the structure of the prototype II. Test procedures are being developed to test the suitability of actuates to human sensing.
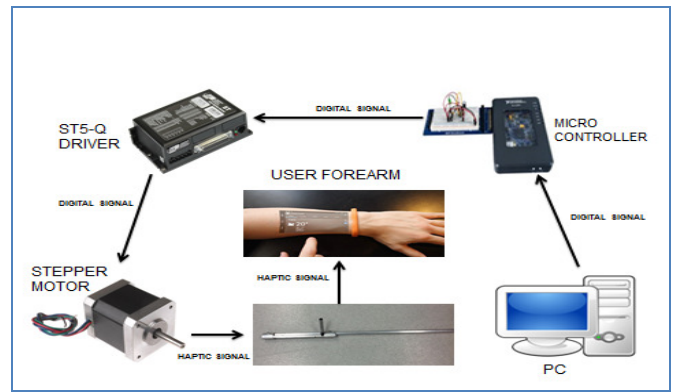

Fig. 3. Structure of prototype II

V. Conclusion

A first stage prototype system has been developed, which focuses on the deployment of suitable sensors to allow accurate and reliable measurement of the robot, rein and users relative positions. The data are required to enable for further stages of the intelligent robotic rein design. The stiff rein solves the issues of robot localization and orientation with respect to the user and provides a direct method of haptic feedback. The first prototype was tested and completed as the first part of the research. The majority of second prototype (motion /feedback system) has been finished and we are developing test procedures to insure the force haptic feedback is suitable for a human forearm. The overall system aims to mimic the complex shared control relationship observed between a guide dog and a human user.

References

[1] REINS EPSRC Reference: EP/I028757/1
http://gow.epsrc.ac.uk/NGBOViewGrant.aspx?GrantRef=EP/I028757/1

[2] Ghosh, A., Alboul, L., Penders, J., Jones, P. and Reed, H., Following a Robot using a Haptic Interface without Visual Feedback, In: Seventh International Conference on Advances in Computer-Human Interactions, Barcelona, Spain (2014).

Section 3: Abstracts

From Working Memory to Cognitive Control: Presenting a Model for their Integration in a Bio-inspired Architecture

Michele Persiani, Alessio Mauro Franchi, Giuseppina Gini
DEIB, Politecnico di Milano
Milano, Italy
michele.persiani@mail.polimi.it, alessiomauro.franchi@polimi.it,
giuseppina.gini@polimi.it

Abstract—The prefrontal cortex (PFC) in the brain is considered as the main responsible of cognitive processes. This brain area is adjacent to the sensorial and motor cortices, and most importantly, gets innervated by dopamine, the neurotransmitter associated to pleasure and reward. This setting allows neuronal ensembles belonging to the PFC to form associations between sensory cues, actions and reward, which is exactly what is needed for a control mechanism to emerge. In order to allow cognitive control, an agent must be able to both perceive and form associations between the perceived inputs and the available actions. These associations will form the experience of an individual, thus shaping his behaviour. A fundamental process supporting cognition is offered by the working memory (WM), that is a small, short-term memory containing and protecting from interference goal-relevant pieces of information. The WM exploits the dopamine activity for two functions: as a gating signal, which determines when useful information can enter, and as a learning function, which allows the memory to learn whether the currently stored information is good or not with respect to a certain situation and the undergoing task. Grounding our work on biological and neuroscientific studies, we extend our Intentional Distributed Robotic Architecture (IDRA)[1] with a more powerful model of the memory, in particular exploiting the capabilities of the WM. IDRA is a bioinspired modular intentional architecture shaped and acting as the amygdala-thalamo-cortical circuit in the human brain; the architecture deals mainly with two tasks, which are the storage of representations of the current situation in a way similar to what the visual cortex does, and the autonomous generation of goals, starting from a set of hard-coded instincts. Yet, IDRA relies on an external Reinforcement Learning (RL) agent to perform actions, but most important it lacks of a task-driven memory system. We defined a new IDRA core module, which is called Deliberative Module (DM), with the addition of a model of the WM. The DM can act as both WM storage and actions generator, thanks to the introduction of a powerful chunk selection mechanism. A chunk is an object containing arbitrary information that competes for retention in an active memory storage. Transforming the problem of selecting actions to that of retaining chunks, we are able to exploit the same exact mechanism for both retention of chunks and generation of actions, consequently dropping out the RL agent previously in charge of generating the actions. During each agent-environment interaction, the WM receives from sensors and inner processes the current state and a set of chunks of information proposed for retention. Its task is to select the best possible combination of them to maximize the future reward, estimated through a linear function approximator. The number of chunks that can be maintained in WM is small, 7 at maximum. Our WM model is composed by two modules, the first devoted to perception ad the second to choice. It receives in input the set of possible chunks, and outputs the content of the active memory, i.e., those chunks that are to be retained in memory. The perception stage builds a description of the currently perceived situation to obtain a sparse vector representing the state of the system it terms of percepts. The action selection selects the percepts to be kept as the WM content. This process is a form of context-sensitive learning as percepts are selected depending on both the current state and the context. The perception process is a cascade of feature extraction and clustering aimed at classifying the current input in an unsupervised fashion, obtaining their corresponding percepts. It first applies Principal Components Analysis (PCA) to reduce the dimensionality of the problem, then Independent Components Analysis (ICA) to extract the independent components, and finally K-Means to cluster data in the IC space. In this way the raw input is transformed into a set of perceivable classes represented in sparse coding. The active memory stage has to discard the percepts less useful keeping into consideration the limited capacity of its memory. After training, the experience is codified as "rules" determining the module's retention policy. We tested the WM model with available datasets to check whether the perception phase is able or not to create optimal features and clusters with respect to the input data, which can be produced by very heterogeneous sources. We compared our pipeline of sensor processing composed by PCA, ICA, and Softmax with the baseline being only Softmax on a heterogeneous dataset for classification, containing about 1500 entries coming from different sources (UCI repository), with nine classes. The result tells us that our pipeline outperforms the baseline, which is not able to distinguish at all some of the classes. In particular the addition of ICA is fundamental for dealing with heterogeneous data. Other experiments more relevant for robotics have been executed as well, demonstrating a good performance. Nevertheless, improvements are under way to integrate imitation learning in order to speed up the learning process.

---

[1] A. M. Franchi, F. Mutti, G. Gini, "From learning to new goal generation in a bioinspired robotic setup", Advanced Robotics, 2016, DOI 10.1080/01691864.2016.1172732

A Physical Architecture for Studying Embodiment and Compliance: The GummiArm

Martin F. Stoelen, Ricardo de Azambuja,
Angelo Cangelosi
Centre for Robotics and Neural Systems (CRNS),
Plymouth University
Plymouth, UK
martin.stoelen@plymouth.ac.uk

Fabio Bonsignorio
The BioRobotics Institute,
Scuola Superiore Sant'Anna, Pisa
and Heron Robots, Genova,
Italy

Abstract— **High bandwidth contacts and unforeseen deviations from planned actions are common in early human development. We here present the GummiArm, an open-source robot with characteristics that make it interesting for studying development, human motor control, and real-world applications that require robustness and safety. Joints with antagonist actuators and rubbery tendons provide passive compliance, where the stiffness can be adjusted in real-time through co-contraction. The robot structure is made printable on low-cost 3D printers, enabling researchers to quickly fix and improve broken parts. The arm has 7+3 Degrees of Freedom (DOF), of which 8 have variable stiffness. It is currently being replicated across 3 research groups, and we hope to establish a thriving and productive community around this replicable platform.**

Gagarin: A Cognitive Architecture Applied to a Russian-Language Interactive Humanoid Robot

Vadim Reutskiy
Innopolis University
v.reutsky@innopolis.ru

Nikolaos Mavridis
Interactive Robots and Media Lab
nmav@alum.mit.edu

ABSTRACT

Cognitive Architectures have been an active area of research for more than two decades, starting from well-known examples such as ACT-R. Beyond modeling human performance, one of the promising domains of application of cognitive architectures is in real-world embodied situated systems, such as robots. However, most of the existing systems have failed to be used widely, arguably for a number of reasons, the main being that they seem to have provide little added value to real-world complex interactive robot designers as compared to a totally ad-hoc approach. To address this situation, here we will present desiderata and an example of practical real-world cognitive architecture for the humanoid Gagarin, aiming to fill the gap between strongly defined systems and totally ad-hoc.