the social system analysis but is still required so that agents receive predictable sensor signals that are not disrupted by multiple collisions.

« 82 » As mentioned before, ICO learning is used to generate the steering of the left and right motors by using the anticipatory and reflex inputs:

$$\begin{aligned} ICO_{avoid,L}(t) = {} & \omega_{avoid,reflex,L} \cdot u_{avoid,reflex,L}(t) \\ & + \omega_{avoid,pred,L}(t) \cdot u_{avoid,pred,L}(t) \\ & + \omega_{self,L}(t) \cdot ICO_{avoid,self,L}(t-1) \\ & + \omega_{R2L}(t) \cdot ICO_{avoid,R2L}(t) \end{aligned} \quad (25)$$

$$\begin{aligned} ICO_{avoid,R}(t) = {} & \omega_{avoid,reflex,R} \cdot u_{avoid,reflex,R}(t) \\ & + \omega_{avoid,pred,R}(t) \cdot u_{avoid,pred,R}(t) \\ & + \omega_{self,R}(t) \cdot ICO_{avoid,self,R}(t-1) \\ & + \omega_{L2R}(t) \cdot ICO_{avoid,L2R}(t) \end{aligned} \quad (26)$$

with $\omega_{avoid,reflex,L} = -0.9$, $\omega_{avoid,reflex,R} = -1.0$ so that hitting an obstacle causes a retraction reaction. The slight differences in the weight cause the robot to turn slightly when hitting an object dead on. Here, the ICO neurons have recurrent synaptic connections and a push pull mechanism between left and right motor neuron as $\omega_{R2L} = -0.42$, $\omega_{L2R} = -0.3$, $\omega_{self,R} = 0.4$, $\omega_{self,L} = 0.4$ to implement a hysteresis effect. This effect causes the controller not to follow signals with a slight delay, as shown in Wischman Pasemann & Hülse (2004) and Hülse & Pasemann (2002). It means reactions on an incoming signal are time shifted. This is useful to enable agents to escape from corners: if there were no hysteresis mechanism, an agent would get stuck in a corner forever, turning left and right alternately.

« 83 » Learning generates anticipatory actions by using the information from the long range sensors $x_{avoid,pred,L/R}$ and updating their corresponding weights in such a way that the agent steers away from a wall before it crashes into it. This is achieved by ICO learning, which updates the weights $\omega_{avoid,pred,L/R}$:

$$\frac{d\omega_{avoid,pred,L}}{dt} = \mu \cdot u_{avoid,pred,L} \cdot \frac{du_{avoid,reflex,L}}{dt}$$

$$\frac{d\omega_{avoid,pred,R}}{dt} = \mu \cdot u_{avoid,pred,R} \cdot \frac{du_{avoid,reflex,R}}{dt}$$

Received: 5 October 2012
Accepted: 16 January 2014

Open Peer Commentaries

on Bernd Porr & Paolo Di Prodi's "Subsystem Formation Driven by Double Contingency"

Learning by Experiencing versus Learning by Registering

Olivier L. Georgeon
Université de Lyon, France
olivier.georgeon/at/liris.cnrs.fr

> Upshot • Agents that learn from perturbations of closed control loops are considered constructivist by virtue of the fact that their input (the perturbation) does not convey ontological information about the environment. That is, they learn by actively experiencing their environment through interaction, as opposed to learning by registering directly input data characterizing the environment. Generalizing this idea, the notion of learning by experiencing provides a broader conceptual framework than cybernetic control theory for studying the double contingency problem, and may yield more progress in constructivist agent design.

« 1 » Ernst von Glasersfeld differentiated radical constructivism from realist epistemology by the relation between knowledge and reality:

> "Whereas in the traditional view of epistemology, as well as of cognitive psychology, that relation is always seen as a more or less picture-like (iconic) correspondence or match, radical constructivism sees it as an adaptation in the functional sense." (Glasersfeld 1984: 20)

This suggests differentiating constructivist artificial agents from realist artificial agents by the relation between their input data and their environment. As illustrated by Bernd Porr and Paolo Di Prodi's implementation, in cybernetic theory, the agent's input (called perturbation) does not hold an "iconic correspondence" with the environment but rather consists of feedback from the agent's output (called action). In contrast, as we shall develop below, most machine-learning algorithms implement this iconic correspondence because they implement and exploit the agent's input as if it directly characterized the environment, thus representing a direct access to the ontological essence of reality.

« 2 » Here, we call *learning by experiencing* those learning mechanisms that implement and exploit input data as feedback from the agent's output, and *learning by registering* those learning mechanisms that

implement and exploit input data as a direct observation of the environment (either a simulated environment or the real world, in the case of robots). This formulation complies, for example, with Etienne Roesch et al.'s formulation that constructivist epistemology considers knowledge as resulting from experience of interaction with the environment, as opposed to existing "in an ontic reality […] available to registration from the physical world" (Roesch et al. 2013: 26).

« 3 » Partially Observable Markov Decision Process models (POMDP; Kaelbling, Littman & Cassandra 1998) well exemplify learning by registering because they typically formalize the agent's input as a function of the environment's state only. A similar argumentation can show that many other machine-learning approaches learn by registering, even supposedly constructivist approaches based upon schema mechanisms (e.g., Drescher 1991)[1] and many approaches based upon multi-agent systems, such as Roesch et al.'s (2013) agents, as we discussed in our open peer commentary (Georgeon & Hassas 2013).

« 4 » For the sake of argument, consider a POMDP in which the agent's input (called observation) is reduced to a single bit. A subset *S0* of the set *S* of all the environment's states are observed as "0", and the states in the complementary subset *S1* are observed as "1". Because of stochastic noise, some elements of *S0* may occasionally be observed as "1" and the other way around. Yet the observation statistically reflects the state of the environment, and the agent's policy generally exploits this assumption to try to construct an internal model of the agent's situation. To our knowledge, there is no POMDP implementation that would exhibit interesting behaviors with as little observation as a single bit when the number of states is great. This limitation is known as the perceptual aliasing problem (Whitehead & Ballard 1991), and is inherent to learning by registering.

« 5 » Note that some variations of POMDPs have been proposed in which the scope of the observation depends on the previous action, thus involving a form of active perception (e.g., McCallum 1996). However, the observation still reflects the state of the environment, as if the environment was observed through a filter that varied with the action.

« 6 » In contrast, mechanisms of learning by experiencing implement the agent's input such that it conveys information about the effect of an "experiment" performed by the agent. In the case of a single input bit, this bit indicates one out of two possible outcomes of the experiment. The same particular state of the environment induces different input bits depending on the experiment initiated by the agent. No partitioning of the set of states *S* can be made according to the input bit because all states may induce input "0" or "1," depending on the experiment. In this case, the learning algorithm must not exploit the agent's input as if it statistically and partially corresponded to the state of reality, because it does not. In contrast with learning by registering, there exist single-input-bit learning-by-experiencing agents that exhibit interesting learning behaviors (e.g., Georgeon & Hassas 2013; Georgeon & Marshall 2013).

« 7 » Besides cybernetic control theory, in §68, Porr and Di Prodi mention other examples of learning by experiencing: Richard Sutton et al.'s (2011) Horde architecture, and our work. Horde relies on a swarm of reinforcement-learning agents to learn hierarchical temporal regularities of interaction through experience. More broadly, learning by experiencing implements a form of conceptual inversion of the perception-action cycle recommended by some authors (e.g., Pfeifer & Scheier 1994; Tani & Nolfi 1999). In learning by experiencing, however, calling the input a perception or an observation is misleading because the input does not hold a direct correspondence with reality.

« 8 » Concerning our approach, we shall clarify that it does not only "act in discrete space," as Porr and Di Prodi wrote in §68. Instead, our agents are indifferent to the structure of their environment's space, which is precisely an advantage of learning by experiencing. We demonstrated that our algorithms could control agents in continuous two-dimensional simulated environments (Georgeon & Sakellariou 2012) and robots in the real world (Georgeon, Wolf & Gay 2013). It is true that our agent's set of possibilities of experience (the relational domain defined by the coupling between the agent and the environment, e.g., Froese & Ziemke 2009) is discrete, but this does not prevent the agent from learning interesting behaviors in continuous space.

« 9 » Since learning-by-experiencing (LbE) agents do not directly access the state of the environment, they incorporate no reward function or heuristics defined as a function of the state of the environment. This places LbE agents in sharp contrast with reinforcement-learning agents and problem solving agents. Notably, LbE agents even differ from reinforcement-learning agents with an intrinsic reward (e.g., Singh, Barto & Chentanez 2005), which consider some elements of the state of the world to be internal to the agent. As a generality, an LbE agent gives value to the mere fact of enacting interactive behaviors rather than to the state resulting from behaviors. We expect LbE agents to demonstrate that they learn to "master the laws of sensorimotor contingencies" (O'Regan & Noë 2001). Consequently, as some authors in the domain of intrinsic motivation also argued (e.g., Oudeyer, Kaplan & Hafner 2007), we recommend assessing LbE agent's learning through behavioral analysis rather than through a measure of their performance in reaching specific goals.

« 10 » In accordance with our view on LbE agent assessment, Porr and Di Prodi assess their agent's learning through behavioral analysis (Section 4). Their agents are motivated to interact with entities present in the environment by controlling sensorimotor loops (approaching food or other agents, §18). For each sensorimotor loop, Porr and Di Prodi define Prediction Utilization as a measure of the agent's commitment to control this loop. We wish to support their effort in specifying this kind of measure. This effort contributes to defining general quantifiers that could be used with other learning-by-experiencing approaches to characterize the agent's engagement in interactive behaviors.

« 11 » As Porr & Di Prodi noted in §68, simple linear control theory does not realize "the generation of more complex actions, the switching of actions and the sequencing of actions." However, other learning by experiencing approaches tackle these issues.

1| Gary Drescher (1991) modelled Piagetian schemes as triplets – <pre-observation, action, post-observation>. The argument that Drescher's agent's observation reflects the environment's state is similar to our argument about POMDPs.

Addressing the double contingency problem with approaches that generate such learning would allow more sophisticated subsystem organization because each subsystem could control more sophisticated interactions than a linear control loop. Therefore, we anticipate that addressing the problem of subsystem formation driven by double contingency within the general framework of learning by experiencing would allow more advances in constructivist agent design.

Olivier L. Georgeon is currently an associate researcher at the LIRIS Lab, with a fellowship from the French Government (ANR-RPDOC program). He received a Masters in computer engineering from Ecole Centrale de Marseille in 1988, and a PhD in psychology from the Université de Lyon in 2008.

Received: 19 February 2014
Accepted: 19 February 2014

Aligning Homeostatic and Heterostatic Perspectives

Patrick M. Pilarski
University of Alberta, Canada
pilarski/at/ualberta.ca

> Upshot • There is merit to the continuous-signal-space homeostatic viewpoint on subsystem formation presented by Bernd Porr and Paolo Di Prodi; many of their ideas also align well with a heterostatic constructivist perspective, and specifically developments in the field of reinforcement learning. This commentary therefore aims to identify and clarify some of the linkages made by the authors, and highlight ways in which these interdisciplinary connections may be leveraged to enable future progress.

« 1 » Learning to perceive, predict, and act based only on continuous-valued sensorimotor inputs and outputs is a challenging and important pursuit that deserves our focused attention. While subsystem formation and evaluation are the principal listed contributions of Bernd Porr and Paolo Di Prodi's target article, the nature of the signals and predictions in the paper's problem domain are crucial points that impact how we interpret the comparisons made in the paper and the ways its insights may be applied to work in other domains. I use Porr and Di Prodi's problem setting and agent formulation as a starting point for assessing some of the key statements made in their work, and build toward a specific look at prediction utilization as presented by the authors. This assessment is supplemented with comparisons to related work from the recent computational and biological literature.

Heterostasis and homeostasis

« 2 » Porr and Di Prodi's setting of agents interacting in a reflexive and predictive manner via continuous inputs and outputs is a natural one, albeit one that is often ignored in favour of the perceived clarity and mathematical benefits of discrete sensation and action spaces. Their specific setting is in fact a problem domain that resonates well with other robot-related constructivist demonstrations from the machine learning literature – e.g., learned multi-robot food foraging behaviour (Matarić 1997), robot learning applications as surveyed by Grondman et al. (2012), and robot knowledge acquisition as per Modayil, White & Sutton (2014) and Sutton et al. (2011, as cited by the authors). It is important to note, however, that many of these like-minded explorations are rooted in a rather different starting point: that of the learning system or systems attempting to maximize some aspect of its experience – in other words, an agent seeking to increase its long-term expected reward or learning progress, as in the intertwined fields of computational and biological reinforcement learning (Sutton & Barto 1998). This maximization, or *heterostatic* goal-seeking behaviour (after Harry Klopf's *The Hedonistic Neuron*, 1982) is at first glance in contrast with an agent's "task of restoring its desired state to homeostasis," as posed by the authors (§3). However, for our current discussion, it may be beneficial to explore the similarities between these viewpoints in terms of the authors' work, as opposed to the differences.

« 3 » Let us first examine the statements made in the authors' concluding remarks, suggesting that the homeostatic linear control approach in the paper "establishes an ongoing process that has no final goal but is rather driven by intrinsic motivations that are defined by desired states." (§68) After acknowledging the need and potential for more complex actions and action sequences, as potentially provided by techniques from reinforcement learning, the text of §68 continues by stating that the extrinsically defined reward used in standard reinforcement learning "usually means that the life of the animal is just directed toward this single moment in time but will not code an ongoing intrinsic motivation."[1] This sequence of text sets up a natural contrast between extrinsic and intrinsic reward – motivation or satisfaction derived from the world or from within the agent, respectively. At the same time, it reinforces a distinction between heterostatic and homeostatic optimization by an intelligent system.

« 4 » Intrinsic motivation is held to be a powerful way to drive exploration and potentially accelerate the learning of predictions, control behavior, and better representations (Schmidhuber 1991; Oudeyer, Kaplan & Hafner 2007). However, much like the actual boundary between an agent and its environment is often less of a boundary and more an opinion on the part of the system designer (or examiner), boundaries between what are considered intrinsic and extrinsic reward have been placed at different points by different authors. Is the distinction between these types of feedback actually useful to our discussion of the present paper, or does it further cloud the understanding of how Porr and Di Prodi's agents react to perturbations in their sensorimotor streams?

« 5 » One high-level view we could be inclined to take based on the statements made in §68 is that an intrinsic approach to motivation allows ongoing, life-long learning without the need for endpoints or imposed valuations of an agent's stream of experience (e.g., transient or final goals). However, it is interesting to refer again to the aforementioned text in §68 indicating

1| This statement seems to assume a terminal or discrete reward, and passes over the way that standard reinforcement learning often utilizes temporally extended expectations of future reward (e.g., *discounted future return;* Sutton & Barto 1998) or average reward (discussed below). However, a detailed discussion of all these points is best left outside the present commentary.