



Towards intrinsic autonomy through evolutionary computation

Yuri Lenon Barbosa Nogueira¹  · Carlos Eduardo Fisch de Brito¹ · Creto Augusto Vidal¹ · Joaquim Bento Cavalcante-Neto¹

Published online: 17 December 2019
© Springer Nature B.V. 2019

Abstract

This paper presents an embodied open-ended environment driven evolutionary algorithm capable of evolving behaviors of autonomous agents without any explicit description of objectives, evaluation metrics or cooperative dynamics. The main novelty of our technique is obtaining intrinsically motivated autonomy of a virtual robot in continuous activity, by internalizing evolutionary dynamics in order to achieve adaptation of a neural controller, and with no need to frequently restart the agent's initial conditions as in traditional training methods. Our work is grounded on ideas from the enactive artificial intelligence field and the biological concept of enaction, from which it is argued that what makes a living being “intentional” is the ability to autonomously, adaptively rearrange their microstructure to suit some function in order to maintain its own constitution. We bring an alternative understanding of intrinsic motivation than that traditionally approached by intrinsic motivated reinforcement learning, and so we also make a brief discussion of the relationship between both paradigms and the autonomy of an agent. We show the autonomous development of foraging and navigation behaviors of a virtual robot.

Keywords Autonomous agents · Embodied open-ended evolution · Reinforcement learning · Philosophical aspects of evolutionary computing · Enactive artificial intelligence · Artificial life

✉ Yuri Lenon Barbosa Nogueira
yuri@lia.ufc.br

Carlos Eduardo Fisch de Brito
carlos@lia.ufc.br

Creto Augusto Vidal
cvidal@lia.ufc.br

Joaquim Bento Cavalcante-Neto
joaquimb@lia.ufc.br

¹ Department of Computing, Federal University of Ceará, Campus do Pici - Bl. 910, Fortaleza, CE CEP: 60440-900, Brazil

1 Introduction

Evolutionary computation projects usually rely on some kind of prior goal description to evaluate and guide the development of artificial agents behaviors. Therefore, we can say that they follow externally defined objectives. In this paper, we argue that a genuinely autonomous behavior should only address the intrinsic needs of an agent and, in order to tackle such a problem, we propose the EMbodied Open-ended evoluTIONary ALgorithm (EMOTIONAL), capable of evolving behaviors of a single autonomous agent without any explicit description of objectives, evaluation metrics or cooperative dynamics, performing such process in continuous activity, i.e., with no need to restart the agent's initial conditions at any point during training.

Evolutionary computation (EC) has been a topic of considerable active development, and a widely applied approach to evolve behaviors of autonomous agents. We can mention several recent studies in the fields of robotics and virtual agents (Lehman and Miikkulainen 2014; Haasdijk et al. 2014; Trueba et al. 2015; Silva et al. 2016; Nogueira et al. 2016). Among these works, it is noticeable an increasing interest in open-ended evolution (OEE), out of the need for agents capable of evolving behaviors beyond a single predefined objective, and in embodied evolution (EE), particularly in robotics, in order to establish online adaptation.

Other machine learning (ML) techniques have also been used to address the problem of developing autonomous artificial agents in an open-ended fashion (Baldassarre and Mirolli 2013; Klyne and Merrick 2016; Gay et al. 2016; Merrick 2017). In this field, we can highlight the Intrinsically Motivated Reinforcement Learning (IMRL) approach (Barto 2013), that is based mainly on the concepts of novelty and curiosity (Barto et al. 2004; Schmidhuber 2010; Kompella et al. 2012, 2014). However, there is an apparent conflict between that definition of Intrinsic Motivation and the definition of autonomy as it has been approached by the enactive artificial intelligence (EAI) field (Froese and Ziemke 2009), whose ideas we are trying to advance with our technique. We will later discuss such relationship and its implications to agency.

In Sect. 2, we present a brief review of Open Ended Evolution, Embodied Evolution and other open ended approaches to develop autonomous agents. In Sect. 3, we discuss the relationship between objectless evolution, autonomy and intrinsic motivation. In Sect. 4, we present EMOTIONAL, an algorithm firmly grounded on the ideas discussed in Sect. 3. Next, in Sect. 5, we show what EMOTIONAL can do through a series of experiments, which are thoroughly discussed. In Sect. 6, we present our final remarks about this work.

2 Background

Open-ended evolution has the characteristic of continuously adapting the agent's behaviors to the conditions that surround it. As a consequence, it is possible to evolve increasingly complex dynamics, which are robust to environmental changes, and not limited to a simple well described objective. In fact, such an objective may be poorly described. In this sense, OEE seems a natural approach to use in the problem of behavior generation of autonomous agents.

One of the open issues in evolutionary algorithms is the deception problem, which happens when the solution is trapped into a local optimum of the fitness function (Silva

et al. 2016). Because that problem is very common when traditional optimization is employed, alternative approaches such as OEE have been sought. According to Lehman and Stanley (2011), “[...] sometimes, *opening up* the search, more in the spirit of artificial life than traditional optimization, can yield the surprising and paradoxical outcome that the more open-ended approach more effectively solves the problem than explicitly trying to solve it”.

An OEE approach known as Novelty Search (NS) (Lehman and Stanley 2011; Lehman and Miikkulainen 2014), instead of seeking an objective, simply searches for behavioral novelty. In order to do that, it is necessary to describe some metric of behavior differentiation, which is tuned to each specific application. Therefore, based on such novelty measure, the algorithm is guided to perform a search for a specific kind of planned behavior. In our opinion, sometimes, it is useful not to make that kind of predetermination, in order to allow the agent to develop a set of behaviors, instead of just one specific behavior. By doing so, some interesting unpredictable behaviors often appear.

Another OEE approach is based on Environment-Driven Evolution (EDE), i.e., evolution is promoted only by environmental pressures (Nogueira et al. 2016). In that case, there is no explicit description of fitness function, and the best-fit individuals flourish naturally, according to the dynamics of the whole system. It is also important to notice that any environmental change has the potential to modify the system’s dynamics, which affects the agent’s behavior as a response to the new imposed constraints. Thus, there is no search for a unique solution, and new solutions are frequently being tested.

The Environment-driven Distributed Adaptation algorithm (EDEA) (Bredeche and Montanier 2010; Bredeche et al. 2012) applied the idea of EDE to a swarm of real robots, which were able to evolve efficient survival behavior strategies, with no fitness function being ever formulated. The basic idea behind EDEA is the implicit nature of fitness function, i.e., the system’s dynamics enforces that “optimal genome should reach the point of equilibrium where genome spread is maximum (e.g. looking for mating opportunities) with regards to survival efficiency (e.g. ensuring energetic autonomy)” (Bredeche and Montanier 2010).

The EDE approach was also used to evolve emergent behaviors of autonomous virtual characters (Nogueira et al. 2013a). In that work, evolution’s dynamics was established through the simulation of sexual reproduction of the virtual agents. The “female” and “male” robots, are identical, except for the ability of the female robots to “get pregnant”. It is interesting to notice that the experiments show common emergent behaviors regarding navigation, foraging and mating for both genders, but also have generated gender-oriented behaviors.

The EE algorithms can be divided into those based on fitness functions and those that are completely evaluation-free. In the original EE algorithm (Watson et al. 2002), a robot computes the probability of reproduction of its genes based on its own energy level, so that such computation works as an explicit fitness function that tries to enforce the fittest genes. This very idea continues to be applied even in more recent studies (Trueba et al. 2015; Haasdijk et al. 2010a; Elfwing et al. 2011). On the other hand, the aforementioned EDEA (Bredeche and Montanier 2010) and mEDEA (minimal EDEA) (Bredeche et al. 2012), or the Simulated Sexual Reproduction (Nogueira et al. 2013a), try to insert the fitness function into the dynamics of the algorithm, where no explicit evaluation occurs. Also, it is possible to combine the better control allowed by task-driven optimization, in order to obtain useful behaviors, with the open-endedness allowed by environment-driven adaptation, which has the potential of achieving more complex behaviors. This hybrid idea is explored with the Multi-Objective and open-Ended Evolution (MONEE) algorithm (Haasdijk et al. 2014).

We can also classify the EE algorithms into Distributed EE (DEE) and Encapsulated EE (EEE) (Eiben et al. 2010). The DEE algorithms are built to run in a distributed fashion over a population of agents, and their dynamics depend on this fact. That is the case of the canonical EE algorithm. EDEA, mEDEA, and MONEE, are examples of DEE algorithms. Regarding EEE, evolution occurs within one agent only, usually employing a time-sharing strategy of genes (Bredeche et al. 2009; Haasdijk et al. 2010b, a; Elfving et al. 2011). Embodied Evolution is most commonly performed with DEE algorithms, i.e., multiple robots (Eiben et al. 2010; Bredeche et al. 2018), and known EEE algorithms are based on fitness functions evaluated during “gene lifetime” (Bredeche et al. 2009; Haasdijk et al. 2010b; Eiben et al. 2010).

Intrinsically Motivated Reinforcement Learning (Kaplan and Oudeyer 2006; Barto 2013) has also been a widespread approach to develop behaviors of autonomous agents capable of solving problems in a open-ended fashion, in order to achieve robustness to open worlds, multiple objectives, and behaviors that go beyond those that could be foreseen by a designer. Such works are grounded mainly on neuroscience findings of mechanisms of internal rewards in animal brain, as well as the psychological concepts that grounds curiosity (Barto et al. 2004; Oudeyer et al. 2007; Barto 2013; Gay et al. 2016). Basically, in curiosity-driven works, the agent is rewarded for discovering new patterns in the environment (Schmidhuber 2010), and it is always in search of novelty. This technique has been successfully used in works on learning of sensorimotor skills of artificial agents (Kompella et al. 2012, 2014).

Our algorithm is designed based on the evolutionary approaches discussed in this section. It is an Encapsulated Open-Ended Embodied Evolutionary algorithm, which is encapsulated to run within a single agent in continuous activity, and works without any explicit description of fitness function or objective metric.

3 Objectless evolution, intrinsic motivation and autonomy

3.1 The enactive approach to autonomy

In this section we present the relationship between objectless evolution, i.e., evolution without explicitly defined objective, intrinsic motivation and autonomy, and argue why objectless evolution better fits a particular and important definition of autonomy. Such definition is grounded on biological theories about intentionality of living beings (Varela 1992; Fitch 2008), the basis of the Enactive Artificial Intelligence field (Froese and Ziemke 2009), in which our technique is giving a practical contribution.

The enactive paradigm to artificial intelligence has emerged from the perception that “embodied artificial agents which are embedded in sensorimotor loops [has] not been sufficient to account for a meaningful perspective as it is enjoyed by (...) living beings” (Froese and Ziemke 2009). Researchers have found in biology, specifically in the concept of *autopoiesis* (Varela 1992), an explanation of how a living organism creates its own world of significance, or at least for what is “bad” or “good” from the point of view of the living agent itself.

Autopoiesis is defined as a network of processes that occur in living beings: (1) that are continuously regenerating and realizing the network that produces them; and (2) that constitute the system as a distinguishable unit in time and space. This action of self-construction and self-maintenance of an identity is pointed as the provider of a reference from

which the significance of the agent's interactions with the world can be derived. In fact, those processes have long been indicated as the basis on which intentionality and autonomy of living being are grounded (Varela 1979, 1992; Fitch 2008).

Influenced by the idea of living organisms as “autopoietic” systems, i.e., systems that produce their own identity through incessant endogenous activities, Barandiaran et al. (2009) proposed a strong definition of agent: “an autonomous organization capable of adaptively regulating its coupling with the environment according to the norms established by its own viability conditions”. Thus, the actions of a genuine agent maintain the agent in environmental conditions that are favorable to its self-constitution (or maintenance). Self-constitution enables the agent to continue exerting the necessary actions to maintain and individualize it. Such definition implies that the agent's actions are self-motivated, i.e., the agent's actions come from its internal dynamics of subsistence, and always seek the self-constitution of the agent's system. That is called “Constitutive Autonomy”, an essential property of life that has the potential to explain the intrinsic teleology of living beings, which are genuinely autonomous systems (Nogueira et al. 2016; Vernon et al. 2015).

Another important concept related to a living being's agency, is that of “precariousness”, which is the notion associated with metabolic or chemical systems that are not in thermodynamic equilibrium (Barandiaran and Moreno 2008; Egbert and Barandiaran 2011). Since a biological organism is in constant thermodynamical exchange with its environment in pursuit of thermodynamic equilibrium, the organism is always in precarious conditions, i.e., it has to regulate its interaction with the environment actively to keep itself alive. This precariousness, “is meant to form the basis of the normative character of behavior: the system must actively seek to compensate its inherently decaying organization” (Egbert and Barandiaran 2011).

3.2 Evolving enactive agents

Evolving Enactive Agents means that there is no other objective guiding evolution than that intrinsic of agency (maintain viability conditions), and, besides that, it cannot be explicitly defined (i.e., externally predefined), but it has to emerge as a consequence of the interactions between the agent and the environment. In pure EDE, no fitness function should be described, and evolution takes place only by environmental pressures. That is, there is no explicit evaluation metric to select, but the individuals that develop the best survival strategy of interaction between body and environment will naturally spread their characteristics to future generations. If we consider the agent-environment system, there is no external force to the system guiding its interactions in EDE, and so this approach has the potential to better fit the idea of autonomy in the enactive definition than traditional evolutionary computation.

EMOTIONAL can be classified as an EDE algorithm and, although there are no externally defined objectives, we still cannot talk about Constitutive Autonomy in this work, and thus we are not strictly following the precise agency definition aforementioned, since the artificial agent, a virtual robot, is always fully constituted. Nevertheless, we can simulate similar pressure based on the concept of precariousness, which generates the conditions to emerge what is called “Behavioral Autonomy” (Froese et al. 2007; Froese and Ziemke 2009; Nogueira et al. 2016; Vernon et al. 2015), i.e., the emergent behavior is what controls an “energetic” level that makes itself possible, in a dynamic that counteracts the simulated precariousness, otherwise it would lead to the impossibility of agent movement. We also approached that idea (Nogueira et al. 2013a) in our previous work about a simulated

reproduction method of virtual agents. That technique, however, depends on a population of virtual characters to work. EMOTIONAL allows such Behavioral Autonomy with only a single agent.

3.3 Advantages of objectless environment driven evolution

When evolution occurs as a result only of the interaction dynamics between the agent and its environment, any aspect of this system can offer some opportunity to improve adaptation, including those which we could not perceive a priori. A given characteristic of an agent cannot be judged in isolation to be good or bad, since its importance depends on the current dynamics of the system. In our previous work about simulated reproduction of virtual robots (Nogueira et al. 2013a), we noticed that, in some point during the simulation, the male robots were presenting primarily a foraging behavior, and, all of a sudden, they started to exhibit mainly a mating behavior. In that study, since the dynamics of the system was changing and so were the adaptation conditions, one could not say that a single implicit fitness function existed. In fact, the implicit fitness function may be seen as another emergent aspect of the system.

If the objectives of an agent change according to its environmental conditions, then it is possible to achieve not only one behavior, but a set of different behaviors. Moreover, since the agent's movements are gradually built from pressures and constraints of its surroundings, those pressures and constraints can incorporate aspects of the problem that possibly would be overlooked in a poorly described objective function, and also has the potential to lead to more complex or richer solutions (Nogueira et al. 2016). In fact, this is maybe the most relevant characteristic of an open-ended approach to development of learning agents.

3.4 Intrinsic motivation

Intrinsic motivation is a concept that came from psychology and has been adopted by machine learning practitioners due to its potential to produce open-ended learning agents (Barto et al. 2004; Kaplan and Oudeyer 2006; Oudeyer et al. 2007; Oudeyer and Kaplan 2008; Baldassarre 2011; Barto 2013; Oudeyer and Smith 2016). Baldassarre (2011) presented a biological perspective of intrinsic motivation that shows an apparent conflict with the enactive approach to autonomy, since the processes grounded on biological intentionality, and thus clearly intrinsic, are those ones that he claims as extrinsic motivations.

In that sense, extrinsically motivated behaviors are those related to food and water intake, for example, that lead to satisfaction of homeostatic needs. The reason behind such definition is that the behavior is not motivated only by the agent's brain activities itself, but it has the objective to fulfill the needs of the body, i.e., the external environment. What they call intrinsically motivated are those behaviors that don't show a clear biological function, such as ludic behaviors and curiosity (Barto 2013), and are a result only of the brain activity, i.e., the internal environment.

Such concept of intrinsic motivation found notably space within reinforcement learning works (Schmidhuber 2010; Kompella et al. 2012; Barto 2013; Kompella et al. 2014). In that context, what differs an internal motivated from an external motivated behavior is the origin of the reward signal. For example, a foraging behavior is triggered by rewards delivered by food (and so, it is an external reward), while curiosity is rewarded by hormones released by the brain (and so, it is an internal reward) when something new is discovered by the agent. Barto (2013), however, acknowledges: "the internal/external environment

dichotomy does not provide a way to cleanly distinguish between extrinsic and intrinsic reward signals”.

Another related concept proposed in reinforcement learning works is “Interactional Motivation” (Georgeon et al. 2012). In such paradigm, the reward is a function of the agent’s action and observation, rather than the state, i.e., the reward is a result of agency and not the environment, and thus “the agent enacts schemes for its own sake rather than for the value of the outcome that they produce”. This design puts the notion of interactional motivation to be in between extrinsic and intrinsic motivation. It is important to notice, however, that the reward is explicitly given to the reinforcement learning algorithm to compute the agent’s policy, and thus the behavior is an effect of an adequately predefined reward function.

In order to address the problems around the concept of motivation, we notice the obvious fact that any kind of motivation is essentially intentional. Thus, it doesn’t matter whether or not the object that triggered the motivation is internal or external to the agent’s brain. What leads a genuinely autonomous agent to perform some behavior is its intention to do so. Our technique tries to advance in the understanding about the mechanisms behind intentional behaviors, i.e., those ones that follow objectives that are not externally predefined, although they can be externally rewarded.

Intrinsically motivated reinforcement learning is mainly performed through what is called “artificial curiosity” (Oudeyer and Kaplan 2008). Such technique is based on rewarding the curiosity of agents, i.e., the search for novel states (Barto et al. 2004; Schmidhuber 2010; Kompella et al. 2012; Barto 2013; Gay et al. 2016). That method has been shown to be successful on simulated environments, however, its application to real robots was a concern to Oudeyer et al. (2007) who explored such concept of “novelty” to developmental robotics, and thus proposed an architecture based on rewarding the progress of prediction of consequences of actions taken by a robot (Kaplan and Oudeyer 2006; Oudeyer and Kaplan 2008). In fact, studies have shown that animal brains have similar mechanisms of curiosity rewarding (Barto et al. 2004). Developmental robotics takes additional inspiration from developmental psychology and infants’ development (Oudeyer et al. 2007).

As in enactive artificial intelligence community, Oudeyer et al. (2007) are also concerned with the problem of meaning in autonomous systems: “Can goals and means simply emerge out of subsymbolic dynamics? This is one of the most challenging issues that developmental approaches to cognition have to face”. What we are proposing here can thus be seen as an alternative solution to curiosity, addressing the problem in a lower level. In fact, curiosity seems to be an important element for explaining cognition, however, we believe that the question about meaning can only be answered if curiosity is achieved from emergence, and evolution seems to be the way, as noticed by Barto (2013) and Oudeyer and Smith (2016). However, it is not the aim of this work to deepen this discussion.

4 Emotional

4.1 Fundamentals

According to Egbert and Barandiaran (2011), in an evolutionary approach to explain agency, a behavior is considered normative if it has been selected by evolution. In this view, norm following is a result of natural selection, and defines whether a pattern of behavior is adaptive or maladaptive. As previously argued, the key for the normative character of behavior is

precariousness, i.e., the system must actively seek to compensate for its inherently decaying organization. That is the primary idea behind EMOTIONAL.

Before we present the algorithm, we analyze four fundamental aspects of environment-driven evolutionary algorithms:

1. Replication leads to evolution;
2. Replication must be easy;
3. Replication must be sensitive to the diversity of elements within the environment; and
4. Activities of individuals affect the environment, changing the possibilities of replication.

Evolutionary algorithms take place on a population of individuals that are candidate solutions to a certain problem. Replication of individuals plays the central role on evolution, i.e., the development of new individuals that describe better solutions (1).

Replication must occur in such a way that better individuals are always chosen. However, since the initial solutions are randomly generated, they are hardly good solutions. In traditional evolutionary algorithms, a fitness function guides the selection of the individuals that will reproduce. Now, the individuals themselves must do their own self-selections, and so the replication is part of the problem to be solved. Thus, at the beginning of the evolutionary process, replication must be easy (2) in order to allow the selection of weak solutions that will be gradually improved as the evolutionary process progresses.

As part of the problem and, at the same time, as a guide to evolution, replication must be related to the environment. For adequate behavior to be found, replication must be sensitive to the diversity of elements (3) at the agent's surroundings. Furthermore, the activities of individuals affect the environment, thus creating new conditions to guide novel paths of evolution, and, therefore, changing the possibilities of replication (4).

4.2 The algorithm

EMOTIONAL is designed to evolve a neurocontroller of an autonomous robot. Like any evolutionary algorithm, it works based on a population in which each individual encodes a controller. The population is organized in a queue of predetermined size. As we detail in Sect. 5, we use an indirect encoding of neural networks into arrays of integers, although any encoding scheme could be used, provided that some kind of crossover operation is defined.

An agent needs an internal energy variable whose value increases or decreases according to the agent's actions and to its relationship with the environment. Each controller is put to "live" within the robot with the energy variable set to an initial value. When that value gets to zero, the controller "dies" and is replaced by another controller with a reset energy variable. The new controller then starts to control the robot from the exact position where the last controller ended, as if the agent had "changed his mind". The Algorithm 1 shows the EMOTIONAL steps.

Algorithm 1: EMOTIONAL.

```

1 Create a queue  $Q$  of capacity  $N$ ;
2 while  $Q$  is not full do
3   Generate a random individual  $I$ ;
4   Set the controller encoded by  $I$  to the agent;
5   Reset agent's energy;
6    $Replicate(Q, I, t)$ ;
7 Create a queue  $S$  of capacity 2;
8 repeat
9   if  $S$  is empty then
10    Get two random individuals  $I_1$  and  $I_2$  of  $Q$ ;
11     $(I_{s1}, I_{s2}) \leftarrow Crossover(I_1, I_2)$ ;
12     $Mutation(I_{s1}, I_{s2}, p)$ ;
13     $S.enqueue(I_{s1})$ ;
14     $S.enqueue(I_{s2})$ ;
15   else
16     $I \leftarrow S.dequeue()$ ;
17    Set the controller encoded by  $I$  to the agent;
18    Reset agent's energy;
19     $Replicate(Q, I, t)$ ;
20 until forever;

```

As argued in the previous section, the main aspect of evolutionary algorithms is replication. Thus, the core procedure of EMOTIONAL is $Replicate(Q, I, t)$, responsible for the insertion of a copy of individual I into queue Q every t seconds until the agent's energy or a maximum predetermined lifespan end. When the queue is full, the first individual that entered the structure is removed, leaving room for a new one. The Algorithm 2 shows the $Replicate$ steps.

Algorithm 2: $Replicate$ procedure.

```

1 Procedure  $Replicate(Q, I, t)$ 
2    $start \leftarrow current\ time$ ;
3    $ti \leftarrow current\ time$ ;
4   while  $agent\ energy > 0$  or  $lifespan \geq (current\ time - start)$  do
5      $tf \leftarrow current\ time$ ;
6     if  $t = (tf - ti)$  then
7       if  $Q$  is full then
8          $Q.dequeue()$ ;
9        $Q.enqueue(I)$ ;
10       $ti \leftarrow tf$ ;

```

While the population queue Q is not full, a random individual is generated whenever the current individual “dies”. When Q is full, two random individuals are chosen from the population and crossover and mutation operations are performed in order to generate a new one.

The mutation and crossover operations are essential aspects of an evolutionary algorithm, allowing variation, avoiding local minima, and leading to a gradual improvement of the solution (behavior). In our crossover implementation, the breaking point of a

chromosome encoding an individual is randomly chosen, while the mutation changes the value of a gene with a probability p of occurrence. It is important to pay attention to the assignment of a low value to p , less than 1%, to achieve stability when some solution is found.

The parameters N , t , and *agent energy* in EMOTIONAL regulate the algorithm's pressure of selection. Notice that, since a copy of a "living" individual is enqueued every t seconds, and the queue has a limited capacity of size N , the controllers that develop a better behavior and are capable of sustaining the agent's energy levels are those that dominate the population. This fact leads them to a selective advantage, and makes them more susceptible to be chosen for crossover in line 8 of Algorithm 1.

EMOTIONAL implements the fundamentals of environment-driven evolution and closely follows the agency definition we have previously presented. Notice that a decaying energy simulates precariousness, the adaptive pressure element that leads to Behavioral Autonomy, since the agent needs to actively search ways to increase its own energy. It is also the energy variable that determines the replication of an individual, and thus the individual actions and its spreading within the population are intimately related.

5 Experiments

5.1 Description

In order to evaluate EMOTIONAL, we evolved a neurocontroller to a Khepera-like virtual robot. The environment consists of a room delimited by walls with randomly distributed fruits and poisons. The simulation was developed with the *Irrlicht 3D Engine*,¹ and physics provided by the *Bullet Physics Engine*.²

5.1.1 The controller

The controller we use in our experiments is the same we used in our previous works (Nogueira et al. 2016, 2013a, b). It is essentially a Continuous Time Recurrent Neural Network (CTRNN), whose neurons are modeled in the following general form (Beer 1995):

$$\frac{dy_i}{dt} = \frac{1}{\tau_i} \left(-y_i + \sum_{j=1}^n w_{ji} f(s_j) + I \right) \quad (1)$$

where t is time; y_i and τ_i are, respectively, the internal state and time constant for each neuron i ; w_{ji} is the weight of the j th input synapse of neuron i ; s_j is the state of the neuron linked to the j th input synapse; $f()$ is the activation function of a neuron, which we defined as $\tanh(x/2)$; and I represents an external input stimulus constant to neurons.

Furthermore, we use two types of neurons that do not have internal dynamics: the afferent and the efferent neurons. An afferent neuron, whose internal state is the value of one of the network's input, cannot receive input from another neuron. The afferent neurons constitute the network's input layer. An efferent neuron, on the other hand, is part of the network's output

¹ <http://irrlicht.sourceforge.net/>.

² <http://bulletphysics.org/>.

Table 1 Genes' identifiers

Id value	Meaning
$0 \leq id \leq 51$	Neuron (N)
$52 \leq id \leq 255$	Neuronic terminal (TR)

layer, and its internal state is the average of the internal states of all the neurons connected to it.

5.1.2 Genetic encoding

A controller is encoded according to the scheme we presented in our previous work (Nogueira et al. 2013a) on simulated reproduction. It is a simplified version of Analog Genetic Encoding (AGE) (Dürr et al. 2006), which allows indirect encoding of artificial neural networks (ANN) into a simple one-dimensional array and augmenting topologies. Each individual is represented by two chromosomes: the first chromosome encodes the stimulus I (Eq. 1), and the second, which we call the Network Chromosome (NC), holds the description of the ANN itself. Unlike our previous work, there is no need of a “gender” information in the current study, and thus we removed such a gene from the chromosomes.

A chromosome is an array of genes, where every gene can represent two types of elements: a Neuron (N) or a Neuronic Terminal (TR). To decode the ANN, we basically follow a two-step process:

1. Read the chromosome and extract the neurons and their respective input and output “ports”, i.e., the “Neuronic Terminals”; and
2. Create the synapses from the interaction between the input and output TRs.

Each gene is a 32-bit integer, where the first 8 bits (1 byte) encode an identifier, i.e., a code that tells whether the gene is representing a Neuron or a Neuronic Terminal, and the last 24 bits specify a value that indicates a parameter of the decoded element. Such parameters are internal time constant τ_i (Eq. 1) for neurons, or input/output values for terminals. In the decoding sequence, any TR gene that appears before the first N gene is ignored; and after each new neuron gene, only the first two TR genes are considered. The first of those valid TR genes determines its input terminal, while the second TR determines its output terminal.

The identifier (first 8 bits) of a gene is decoded according to Table 1, and the value part (last 24 bits) is linearly mapped into a floating-point number in the range $[-1, 1]$. The intervals shown in Table 1 imply a probability of approximately 20% of a gene being a neuron and 80% of a gene being a neuronic terminal. If the value is related to a neuron gene, the result is directly attributed to the time constant of the neuron. If, instead, the value is related to a TR, it is further used to calculate a synapse weight according to the equation:

$$w(i, o) = \frac{eb(i + o)}{2nb}, \quad (2)$$

where w is the weight of a synapse that links an output terminal of value o with an input terminal of value i . The symbol nb indicates the total number of bits that represent the value (24 bits), and eb is the number of equal bits in the binary representations of i and o . We also defined an existence condition to increase topological diversity: if $\lfloor eb/4 \rfloor \bmod 3 = 0$

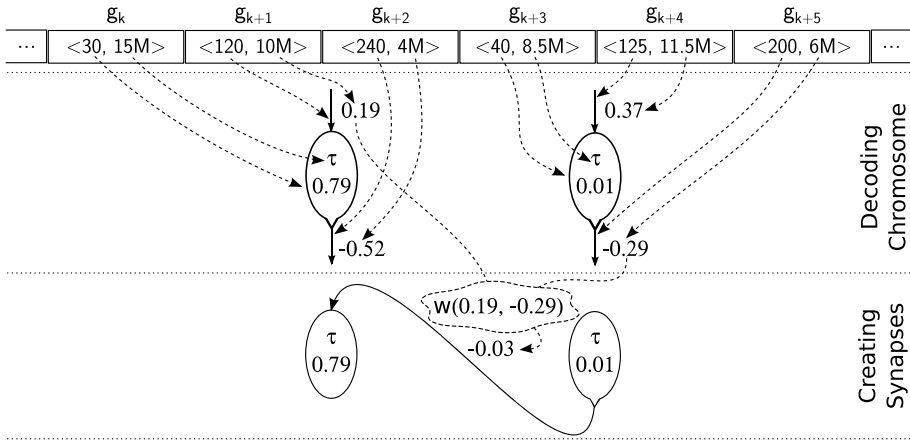


Fig. 1 Building the network (Nogueira et al. 2013a): first we decode the neurons and their respective terminals, then we apply Eq. 2 to each pair of terminals to create the synapses. Only one synapse was created due to the existence condition (see text)

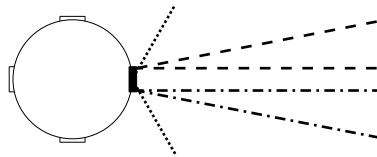


Fig. 2 The distribution of the three vision sensors. The dotted lines represent the FOS of the wall sensor. The dashed lines and the dashed-dotted lines represent, respectively, the left sensor and the right sensor of fruits and poisons

then $w(i, o) = 0$. The logic behind these equations is examined in detail in our previous work (Nogueira et al. 2013a). The whole process of network decoding is shown in Fig. 1.

Furthermore, if we want to obtain a network with q input neurons and r output neurons, the first q neuron genes in a chromosome are necessarily set to be input neurons, while the last r neuron genes are set to be output neurons. Input neurons don't have input terminals and output neurons don't have output terminals, and thus the respective neuron terminal genes are ignored. The number of hidden layers and their connections are completely defined by the evolutionary process and they also can be recurrent.

5.1.3 Robot and environment

The robot is shown in Fig. 2. It has a black box that plays the role of its eye and mouth, where three distance sensors are located. Each sensor is able to catch the normalized distance ($[0, 1]$) to the nearest object inside its “Field Of Sense” (arc) with respect to its range (the maximum detection distance of a sensor). The sensor that is located at the center of the eye is specialized in detecting walls only, it has a FOS of 120° , and a range of about $4r$, where r is the radius of the robot's body. The other two sensors, placed at each side of the eye, are able to sense fruits and poisons, thus generating two values each, and have a FOS of 10° , and a range of $14r$. Furthermore, there is a proprioceptive sense of energy, which

ranges from 0 (the robot is fully energized) to 1 (the robot is totally exhausted). Therefore, the strength of the signal allows the robot to perceive when its energy is finishing.

The robot has two motors, which are controlled by the efferent neurons respectively. When the first motor receives a signal from its efferent neuron, it moves the robot forward in case of positive value, and backward if negative. The other motor makes the robot turn right in the event of a positive signal, and turn left otherwise.

A robot starts with 50,000 energy units (eu). This value increases 10,000 eu whenever a fruit is eaten (up to a maximum value of 50,000 eu), and decreases in two situations: (1) when the robot is alive, in which case the loss of energy is continuous and directly proportion to the strength of the signal sent to a motor; and (2) when the robot eats a poison, in that case its energy is reduced to 10,000 eu. In the second situation, if the robot's energy level is less than or equal to 10,000 eu, the energy is zeroed. If the energy is exhausted, the controller is replaced. When a fruit or poison is eaten, a new one is placed randomly in the environment, and it could be of any type with 50% probability. The energy is also reduced every simulation step according to Eq. 3, where o_1 and o_2 are the output of the two efferent neurons of the controller, w is the sum of the activation values of all internal neurons in each timestep, and the constant value 10 mimics a "metabolic" energy waste. The quadratic term is related to the motors, simulating the energy used to move the robot.

$$C = (|100o_1| + |100o_2|)^2 + w + 10. \quad (3)$$

5.2 Results

As we show in this section, EMOTIONAL successfully evolved the behavior of a single agent, which has learned how to adequately move within its environment. The most interesting question about the results we got is the fact that variables whose behaviors would clearly reflect a good performance of a traditional evolutionary algorithm are not so obvious when analyzing EMOTIONAL. This fact is commonly noticed on embodied evolutionary algorithms analysis (Bredeche et al. 2018). At first glance, such variables do not seem to be evolving, which is apparently inconsistent with the agent's observed behavior, thus requiring a more careful analysis across several characteristics of the experiment. In fact, since our algorithm does not follow a well-defined objective, it was expected that we could not observe its performance through a single and isolated value.

The EMOTIONAL parameters were empirically chosen, and we simply repeated those common to our previous works on traditional genetic algorithm (Nogueira et al. 2013b) and distributed embodied evolution (Nogueira et al. 2013a), that we had already successfully applied. Thus, the results presented in this section were obtained with a queue of size N equal to 100, a Network Chromosome with 75 genes, a replication time t of 90 s, a mutation probability p of 0.1% and a maximum agent energy of 50,000 eu. Figure 3 shows the path traveled by the agent after learning during an experiment session.³ Notice that the robot successfully catches fruits while keeping its mouth away from poisons. The robot also learned how to deviate from walls.

Figure 4 shows the behaviors of an untrained (left) and a trained robot (right) in environments with (top) and without (bottom) fruits/poisons. Notice that the untrained behavior

³ Watch videos of other runs in <https://www.youtube.com/watch?v=X8DGK9ZIULA> and https://www.youtube.com/watch?v=Kil_MaAp64s.

Fig. 3 Robot path after learning. The red diamonds are fruits and the blue squares are poisons. The arrow A emphasizes a turn made by the robot when it realizes that it was heading toward a wall (the chart's boundaries). Note the detour taken by the robot in order to avoid a poison (arrow B). After that, the agent follows a straight path to catch a fruit (arrow C). (Color figure online)

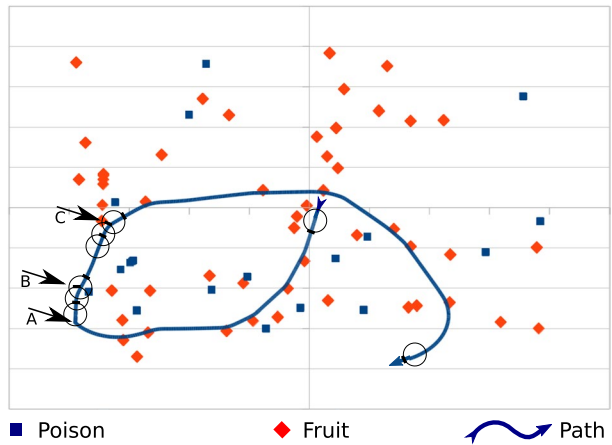
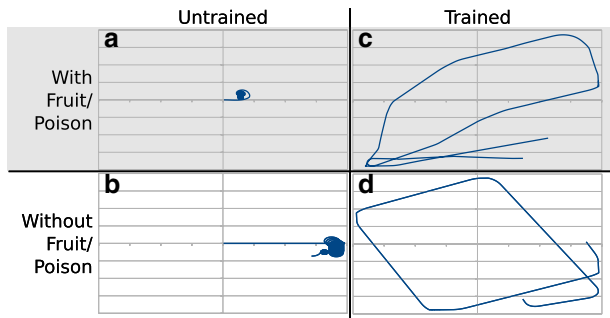


Fig. 4 Behaviors of untrained (left) and trained (right) robots in environments with fruits/poisons (top) and without fruits/poisons (bottom)



is not sensitive to other elements within the environment and the agent does not scan the room adequately, running blind and rotating around a small region no matter what is present. Since a random neural network is built when a new simulation is started, the behavior is not exactly the same when we run in an empty or in a full environment, which explains the difference between paths shown in Fig. 4a, b. The trained behavior, in turn, is more predictable and better covers the area, also changing the running direction in order to catch fruits and avoid poisons when they are present.

Figure 5 shows the network built by EMOTIONAL that produced the behavior described in Fig. 3. Since the Network Chromosome has 75 neurons and the probability of a gene being a neuron is 20%, then we have 15 neurons on average in each randomly generated individual, i.e., the individuals that are created while the EMOTIONAL's queue is not full. However, we have a simple environment exploited by a robot with simple sensors and motors. Thus, this leads to relatively simple problems that can be solved with a small neural network. In this case, the evolutionary process eliminated some neurons and only 10 were used in the end. We also have only three standard (processing) neurons with a single recurrent connection. Note that "right" sensors have no connections. This is reflected in the strategy the robot adopts: continuously turn right and correct the direction of movement turning left when necessary.

Since evolution is carried out without any explicit evaluation metric, traditional metrics to analyze evolutionary algorithms show to be inefficient in analyzing EMOTIONAL. In

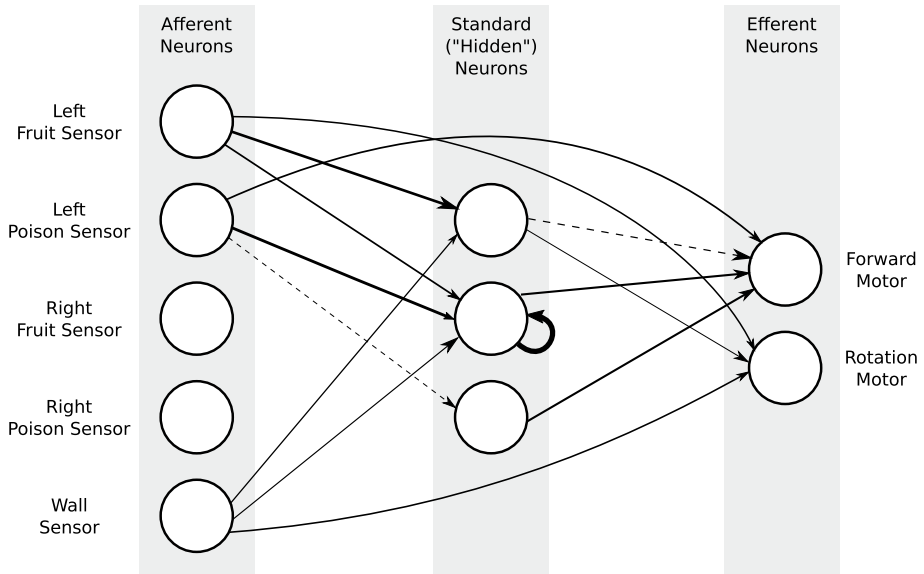


Fig. 5 Neural network built by EMOTIONAL that produced the behavior described in Fig. 3. There are ten neurons, and only three of them are processing units (standard neurons). The line thickness of a synapse is proportional to its weight. Dashed lines are inhibitory synapses, i.e., negative weights. Note that “right” sensors have no connections. This is reflected in the strategy the robot adopts: continuously turn right and correct the direction of movement turning left when necessary

fact, as we have shown, we have a clear evolution in the agent behavior. However, variables whose increasing values we expected to see as indication of “good behaviors”, such as the number of fruits collected in a given time interval, and the lifespan of a controller, counter-intuitively do not reflect exactly the agent’s behavior. This fact is due to the high sensitivity of those variables to environmental variations in the way the experiment is conducted, i.e., the simulation is continuously running, and a controller starts to work at the exact point and conditions where the last one had left off.

Figure 6a, line 1, shows the number of fruits collected by the agent each 10 min in a first simulation run. Notice that the values present a great deal of variation, but we can also observe, with the assistance of a trendline, a certain growth trend in the value. We will discuss later the environmental conditions that can cause a good controller to show a low value in the variable. Now, we want to emphasize that a traditional evolutionary algorithm using that variable as fitness function could lead to losing a good individual in such conditions. Our algorithm is robust to this case.

Figure 6b shows in row 1 the lifespan of each controller tested during the same simulation. Note that the lifespan substantially increased starting from approximately the controller number 7500, and this is reflected in the consistent increase of the average number of collected fruits, starting from approximately 6000 min as shown in Fig. 6a (row 1). As we will argue based on the other charts, the number of collected fruits and the lifespan do not directly determine each other, since the distribution of fruits into the environment changes, and can challenge the agent in different ways in distinct moments. Thus, a changing in the values of the first chart is not immediately reflected in the second one.

In Fig. 6a, row 2, we can see the number of collected fruits plotted against sampled time in a second simulation run. Note that after an initial increase, the number

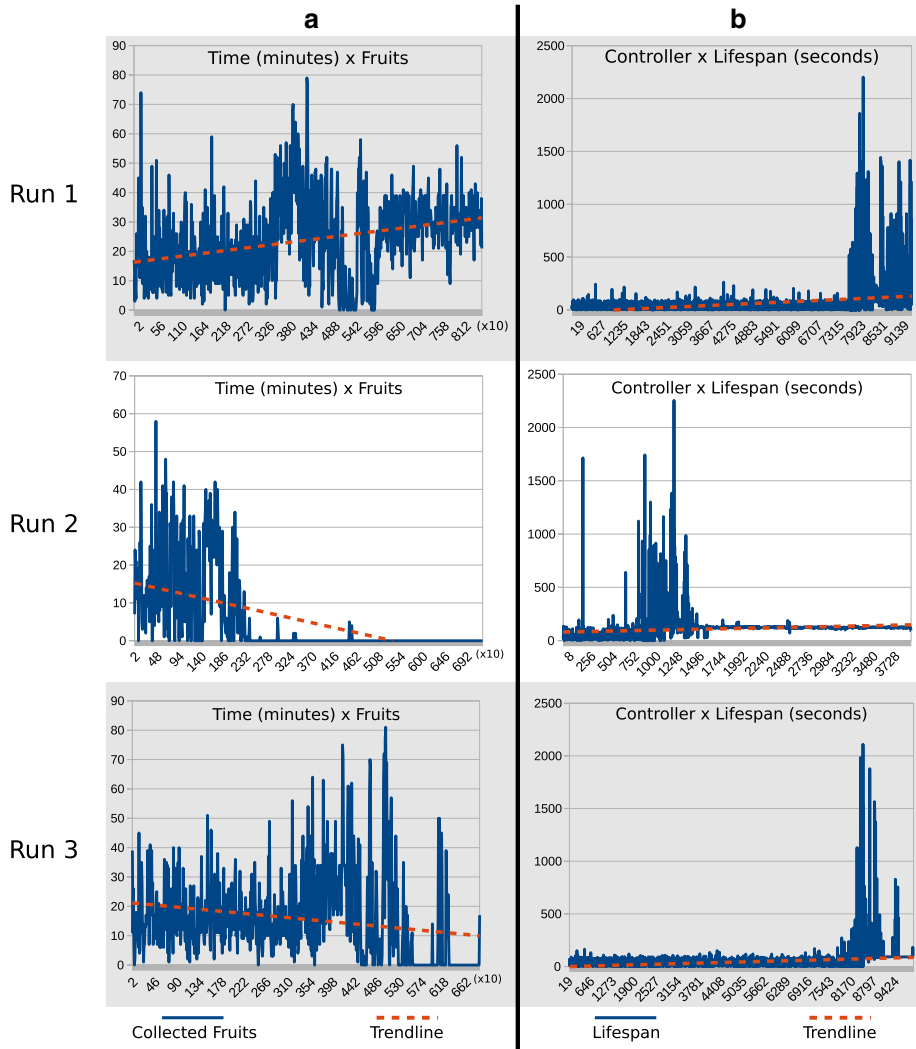


Fig. 6 **a** Number of fruits collected by the robot every 10 min in three different simulation runs. **b** Lifespan of each controller tested during each simulation run

of collected fruits drops. However, we cannot say that this is due to a loss of performance of the controller. In fact, as we can see in the trendline of lifespan in Fig. 6b, row 2, that value is constant at approximately the same time interval. Furthermore, the average lifespan after about 1000 controllers is greater than the initial values, which shows a better energy efficiency. The reason why almost no fruit is collected after about 2500 min is that the agent caught all the fruits located in the center of the room and the other fruits were scattered near the room’s boundaries, a region that is almost invisible to the kind of sensors the robot was equipped. Figure 7a shows that situation.

In row 3, Fig. 6a, b show the same type of values we have been analyzing in a third simulation run. Note a growth trend in Fig. 6a, row 3, until approximately 5000 min,

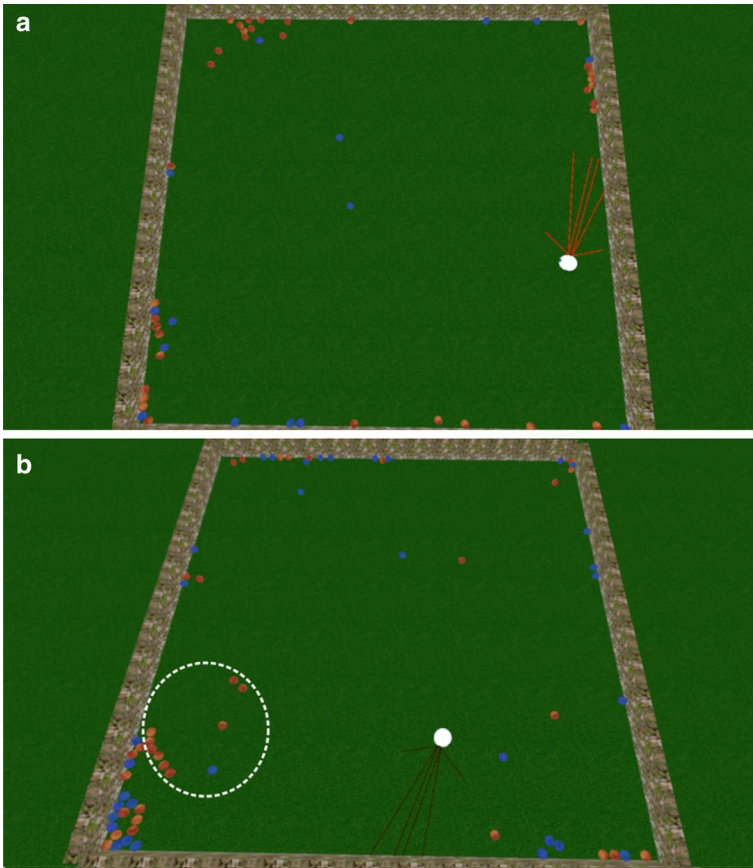


Fig. 7 **a** Environment with most fruits right close to the walls. **b** The dashed circle shows a group of fruits concentrated in a small area in another moment

followed by a drop in values (due to fruit shortage), leading to a decreasing trendline. In Fig. 6b, row 3, we can also observe the reflection of such decaying values after about 9000 controllers. Although some empty spaces were formed, there were places where food was still visible and, as soon as the agent detects them, the number of collected fruits substantially increases (samples from 600 to 650 in Fig. 6a, row 3, and from controller 9450 to 9600, approximately, in Fig. 6b, row 3), showing that the controllers have converged to a solution that is robust to food shortage, i.e., they do not diverge nor “forget” the solution when living in a scarce environment. Figure 7b shows such environmental condition.

In short, Figs. 8, 9 and 10 present the average performance of EMOTIONAL in 10 different runs of 30 h of simulation. They show the evolution of the number of fruits collected by the robot each 10 min, the rate of number of fruits collected to the number of fruit views each 10 min and the lifetime of each chromosome.

Finally, in order to evaluate the Open-Endedness of EMOTIONAL, we performed the following experiment: we put the robot to run for 30 h in an environment filled only with fruits. After that, we changed the environment, forcing a distribution of 3 fruits to

Fig. 8 Average amount of fruits collected by the robot each 10 min in 10 runs of 30 h of simulation

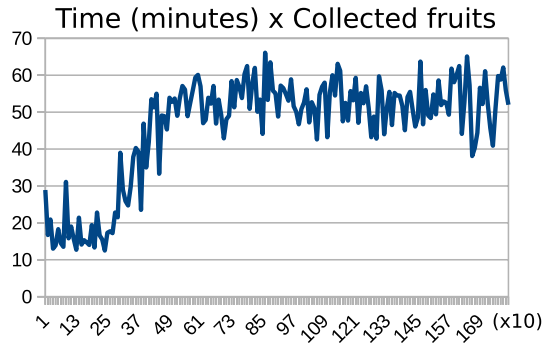


Fig. 9 Average rate of number of fruits collected to the number of fruit views each 10 min in 10 runs of 30 h of simulation

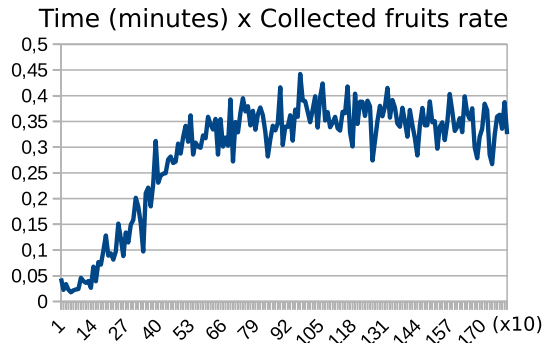
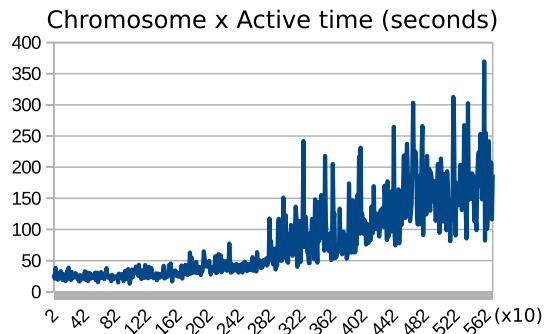


Fig. 10 Average activity time of each chromosome in 10 runs of 30 h of simulation



2 poisons. Figure 11 shows the sudden fall of the number of fruits collected by the robot each 10 min after 1800 min (30 h), which is a consequence of the fruit shortage and the added difficulty to catch them due to the presence of poisons. The total exhibited time is approximately 58 h of simulation. Figure 12, on the other hand, shows the robot's performance in avoiding poisons through the rate of the number of avoided poisons to the number of poison views each 10 min within the final 28 h of simulation. That figure also shows the trendline for the recorded data. Notice that there is a slight increasing tendency of the values, which indicates that even after a learning stabilization, when the environmental pressure was received, the robot has started to learn again. The slight

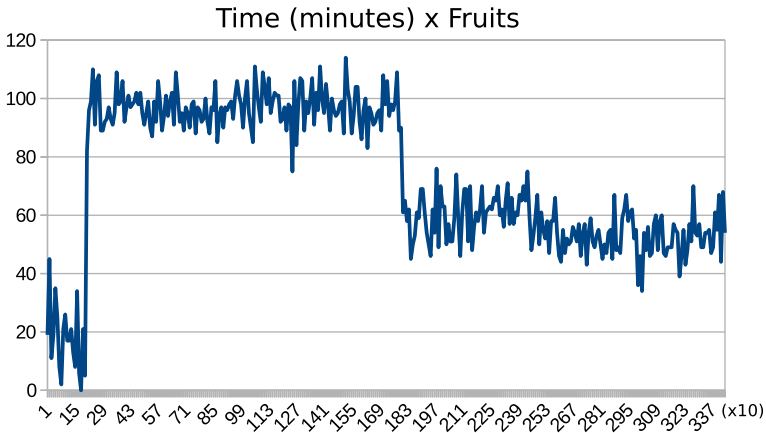


Fig. 11 Number of fruits collected each 10 min in 58 h of simulation. Notice the sudden drop in values from 1800 min (30 h) of simulation

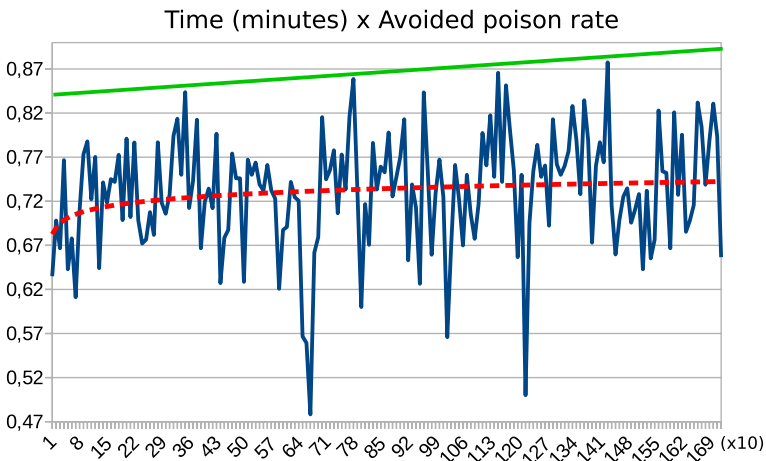


Fig. 12 Rate of number of poisons collected to the number of poison views each 10 min in the last 28 h of simulation. The dashed line is the trendline of data points. The topmost line is the trendline of maximum data values

value increase is due to the fact that, after convergence occurred during the first 30 h of training, the appearance of new features necessary to evolution is strongly determined by mutation, which occurs in a very low rate.

Notice that at the beginning of this last experiment, there is a jump to higher values in the number of collected fruits. Such behavior may seem odd to an evolutionary algorithm, where a gradual increase in agent's performance is expected. Thus, we might be wary of a solution found by chance with the initial randomly generated chromosomes. However, in EMOTIONAL, a new random chromosome is created only while the queue is not full, which occurs in about 150 min of simulation, since there are 100 "slots" and a chromosome is inserted every 1.5 min (90 s). That is, in the worst case, at 150 min of

simulation, the generation of new random chromosomes is stopped. Notice that, until that moment, none of the solutions reached the level shown after 300 min of simulation.

5.3 Evaluating without evaluation

Due to the absence of an objective function to watch, maybe the hardest aspect in analyzing our results is to find good metrics to evaluate them in comparison with traditional techniques. The agent's behavior clearly showed to be visually appropriate, as we can see in Figs. 3 and 4, or watching the animations, but variables traditionally used in evolutionary algorithms analysis seem not to fit with EMOTIONAL analysis. This fact, along with the absence of the idea of "generation" makes it difficult to compare our technique with those based on objective function.

As we argued in Sect. 3, it is not fair to an external observer to say that an autonomous agent does not have a good behavior if it is not performing exactly what the observer had in mind, since it is difficult to realize what the agent is actually experiencing from its point of view. Although we have some intuition about what we would like to see, the robot's actions are not always the same, since the environmental conditions are constantly changing and its behavior should be sensitive to that in ways that we could not foresee. Such a complexity leads to changes in the agent's objectives through time, and that is why some variables do not always work like we would expect.

Therefore, any variable we choose to watch does not reflect only the agent's performance. They also strongly show independent aspects of the environment, and the way the robot is experiencing it limited by its sensory capabilities. For example, due to vision sensors with a relatively small sensibility distance, sometimes the robot is completely blind, making it difficult to explore its surroundings.

However, we can manipulate the environment to be more stable and well distributed over time. By doing that, we can reduce the remarkable environmental noise observed in the number of collected fruit. That way, we clearly see the increase in value over time, as expected, since there are no more situations of food shortage. Figure 13 show the charts of such controlled experiment.

In order to get more controlled data, we also developed experiments based on test beds. Each 30 min, we got the current active chromosome and put it to run within an environment with a fixed distribution of fruits and poisons. Figure 14 shows the average evolution, in tests of 10 different runs, of the rate of the number of collected fruits to the number of fruit views for each chromosome, during 30 h of simulation in an environment with fruits and poisons. Figure 15 shows the same kind of data in an environment filled only with fruits. Notice the consistent growth of the values during the tests. Figure 16 shows the number of collected fruits and poisons. Notice that the number of collected poisons is always below 2 poisons, i.e., the robot does not develop any interest in poisons, unlike what happens with fruits. However, due to the weak sensorial apparatus, some poisons are inevitably collected.

6 Conclusion

In this paper, we addressed the problem of evolutionary computation without describing any objective or fitness functions. In our approach, we presented the Embodied Open-ended evoluTIONary ALgorithm (EMOTIONAL), an encapsulated environment-driven

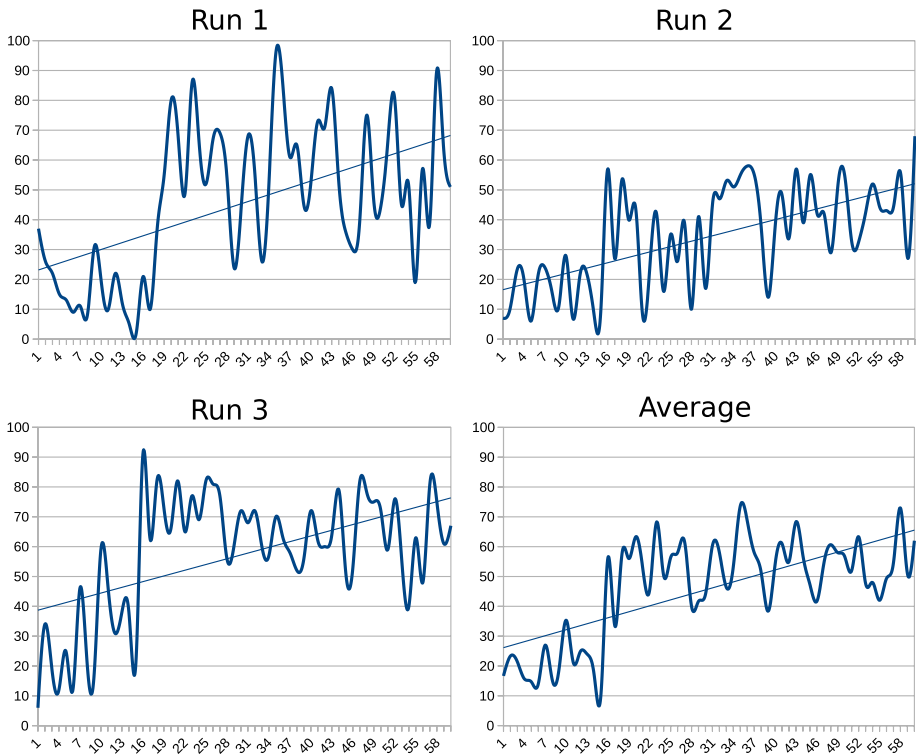
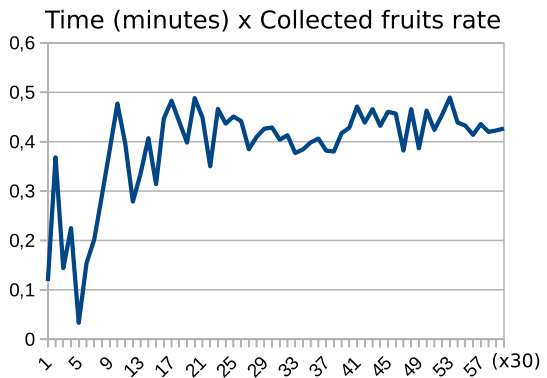


Fig. 13 Controlled environment experiments. The charts show the number of collect fruits each 10 min (approximately 10 h of simulation) in three different runs and the respective average

Fig. 14 Test bed experiments. The chart shows the average rate of collected fruits to fruit views for each chromosome in 30 h of simulation for 10 runs, testing in a fixed environment with fruits and poisons



algorithm, which was able to evolve behaviors of virtual autonomous robots without any explicit description of objectives or cooperative dynamics, performing it in continuous activity.

We argue that objectless evolution is a meaningful (in the agent’s point of view) way of dealing with artificial autonomous agents. Genuine agency is guided by internal goals and should not be limited to externally defined objectives. Moreover, the agent is free to

Fig. 15 Test bed experiments. The chart shows the average rate of collected fruits to fruit views for each chromosome in 30 h of simulation for 10 runs, testing in a fixed environment only with fruits

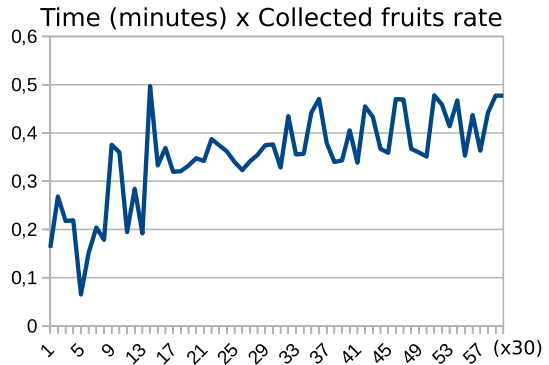
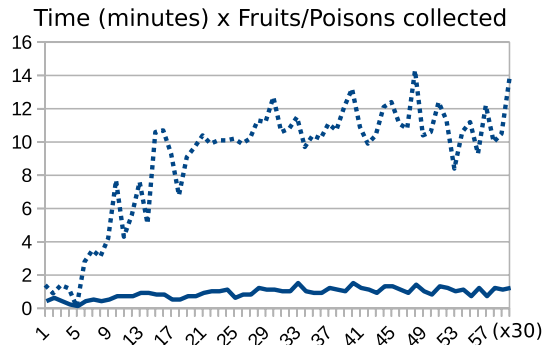


Fig. 16 Test bed experiments. The chart shows the average number of collected fruits and poisons for each chromosome in 30 h of simulation for 10 runs, testing in a fixed environment with fruits and poisons. Dotted line: fruits; Solid line: poisons



explore solutions to problems that it is actually facing, which could lead to a behavioral diversity.

Constitutive Autonomy has been identified in recent studies in the field of artificial intelligence as the key to the genuine agency of living beings. However, in artificial life, precariousness has shown to be a better standpoint than constitution in order to obtain actual agents. That factor is the fundamental basis of EMOTIONAL, evolving behaviors of agents by simulating precarious condition, i.e., the decreasing energy works as a selective pressure, leading to a “Behavioral Autonomy”.

EMOTIONAL showed to be capable of evolving a virtual robot’s controller based on a continuous time artificial neural network. The agent learned to “see” and to guide itself through a simplified vision apparatus based on distance sensors. Although such sensors do not directly provide a sense of direction, the robot proved capable of using its own movements in order to find where the environment’s elements are, catching fruits, avoiding poisons, and preventing collision with the walls. None of the robot’s actions were explicitly described in any way, and they were an exclusive result of EMOTIONAL’s dynamics as well as of the interactions between the agent and its environment. Also, the robot had to learn the “meaning” of the signals generated by the sensors, since there are no difference between them, i.e., the signals generated by a fruit sensor, a poison sensor, or a wall sensor are identical, i.e., values between 0 and 1 that represent the distance to a object, and so the robot had to learn how to distinguish them.

We can say that the algorithm's parameters queue size N , replication time t , mutation probability p , and the robot's maximum energy drive the emergence of behavior. Tuning those variables certainly enables us to constrain some evolutionary paths in order to guide the process of learning actions, and badly chosen values are critical, possibly leading to no useful behavior at all. However, that differs from an objective function description, since we are not determining what role the agent should play among all those that fit coherently with the environment. In fact, the evolutionary strategy we are proposing is also strongly guided by the environment, allowing a more open exploration of behavioral possibilities, and thus the agent can show unexpected good solutions that could be possibly avoided through a rigid predefined objective.

Our work also differs from traditional Reinforcement Learning techniques, since such algorithms are based on maximizing pre-interpreted rewards, while in our experiments the action of eating fruits only interferes with the agent's body. Thus, the evolutionary process has to "learn" that it is a reward, and only then the fruit acquires a meaning for the agent. We argue that such strategy is biologically plausible in a more fundamental way, and can lead to a better understanding of genuine autonomy. Besides that, the energy in our work is related to rewards in the context of traditional Reinforcement Learning techniques, as it is with objective functions of traditional Evolutionary Computation techniques, i.e., since the algorithm is not using it direct to make some objective counting, the agent can solve its problems in ways that goes beyond simply chasing such computation.

Since we do not have a fitness function we could watch in order to evaluate EMOTIONAL, it was necessary to analyze a set of observable data that are somewhat related to the expected behaviors. However, the chosen observable data were extremely sensitive to environmental conditions, as it would be expected in a goal-free evolution, and do not reflect just as well the performance of the agent in behaviors we would like to see. Studying them allowed us to observe several hypothesis about how an evolutionary algorithm without explicit objective description works, even though it is difficult to compare with a traditional approach.

In future work, a cautious study of the influence of the parameters of the algorithm, such as queue size, chromosome insertion frequency and mutation rate, is still necessary, mainly focusing on the reality gap. In the current state, our work shows a working instance of the concept of evolution without explicitly defined objective with a simulated environment, and here we are concentrating on the study of the environmental influence in directing evolution. However, we need to better understand the internal EMOTIONAL's parameters in order to make a generalization of the algorithm feasible, and to apply it to other cases, such as real robots. Since there is a strong integration between the algorithm and the environmental rules, such as fruits that recharge the robot's energy, whose real world reproduction is harder, the algorithm success in this case is more dependent of its internal work, since it is more feasible to make adjustments and adaptations to its parameters.

We also need to tune up the open-endedness feature of the algorithm in order to make it less dependent on mutation, which occurs at very low rates. We possibly need new methods to insert variation in the queue, such as creating new random individuals when the population is very similar, and hence we also need to study and adopt some similarity metrics in order to implement them.

Finally, in order to obtain genuine autonomous behavior, we certainly still need to continue our research on how to achieve "Constitutive Autonomy". The evolution of the agent-environment system we presented is limited, mainly because we have a "rigid" agent, i.e., the robot's body, motors and sensors do not change. Such evolution is necessary so that we could obtain the emergence of new possible goals within the system beyond those expected

and foreseen when designing it, and so we could state in a stronger and broader sense that the agent is capable of creating its own objectives.

References

- Baldassarre G (2011) What are intrinsic motivations? A biological perspective. In: 2011 IEEE international conference on development and learning (ICDL), vol 2, pp 1–8
- Baldassarre G, Mirolli M (eds) (2013) *Intrinsically motivated learning in natural and artificial systems*. Springer, Berlin
- Barandiaran X, Moreno A (2008) Adaptivity: from metabolism to behavior. *Adapt Behav Anim Animats Softw Agents Robots Adapt Syst* 16(5):325–344
- Barandiaran XE, Di Paolo E, Rohde M (2009) Defining agency: individuality, normativity, asymmetry, and spatio-temporality in action. *Adapt Behav Anim Animats Softw Agents Robots Adapt Syst* 17(5):367–386
- Barto AG (2013) Intrinsic motivation and reinforcement learning. In: Baldassarre G, Mirolli M (eds) *Intrinsically motivated learning in natural and artificial systems*. Springer, Berlin, pp 17–47
- Barto AG, Singh S, Chentanez N (2004) Intrinsically motivated learning of hierarchical collections of skills. In: *International conference on developmental learning*
- Beer RD (1995) On the dynamics of small continuous-time recurrent neural networks. *Adapt Behav* 3(4):469–509
- Bredeche N, Montanier J (2010) Environment-driven embodied evolution in a population of autonomous agents. In: *Parallel problem solving from nature*, pp 290–299
- Bredeche N, Haasdijk E, Eiben A (2009) On-line, on-board evolution of robot controllers. In: *Evolution artificielle/artificial evolution*, Strasbourg. <https://hal.inria.fr/inria-00413259>
- Bredeche N, Montanier J, Liu W, Winfield A (2012) Environment-driven distributed evolutionary adaptation in a population of autonomous robotic agents. *Math Comput Model Dyn Syst* 18(1):101–129
- Bredeche N, Haasdijk E, Prieto A (2018) Embodied evolution in collective robotics: a review. *Front Robot AI* 5:12. <https://doi.org/10.3389/frobt.2018.00012>
- Dürr P, Mattiussi C, Floreano D (2006) Neuroevolution with analog genetic encoding. In: Runarsson TP, Beyer HG, Burke E, Merelo-Guervós JJ, Whitley LD, Yao X (eds) *Parallel problem solving from nature—PPSN IX*. Springer, Berlin, pp 671–680
- Egbert MD, Barandiaran XE (2011) Quantifying normative behaviour and precariousness in adaptive agency. In: *Advances in artificial life, proceedings of the 11th European conference on artificial life, ECAL 11* pp 210–218
- Eiben A, Haasdijk E, Bredeche N (2010) Embodied, on-line, on-board evolution for autonomous robotics. In: Levi SKEP (ed) *Symbiotic multi-robot organisms: reliability, adaptability, evolution*. Cognitive systems monographs, vol 7. Springer, Berlin, pp 361–382. <https://hal.inria.fr/inria-00531455>
- Elfving S, Uchibe E, Doya K, Christensen HI (2011) Darwinian embodied evolution of the learning ability for survival. *Adapt Behav Anim Animats Softw Agents Robots Adapt Syst* 19(2):101–120
- Fitch WT (2008) Nano-intentionality: a defense of intrinsic intentionality. *Biol Philos* 23(2):157–177
- Froese T, Ziemke T (2009) Enactive artificial intelligence: investigating the systemic organization of life and mind. *Artif Intell* 173(3–4):466–500
- Froese T, Virgo N, Izquierdo E (2007) Autonomy: a review and a reappraisal. In: Almeida e Costa F, Rocha LM, Costa E, Harvey I, Coutinho A (eds) *Advances in artificial life*. Springer, Berlin, pp 455–464
- Gay S, Mille A, Georgeon O, Dutech A (2016) Autonomous construction and exploitation of a spatial memory by a self-motivated agent. *Cogn Syst Res* 41:1–35
- Georgeon OL, Marshall JB, Gay S (2012) Interactional motivation in artificial systems: between extrinsic and intrinsic motivation. In: 2012 IEEE international conference on development and learning and epigenetic robotics (ICDL), pp 1–2. <https://doi.org/10.1109/DevLrn.2012.6400833>
- Haasdijk E, Eiben AE, Karafotias G (2010a) On-line evolution of robot controllers by an encapsulated evolution strategy. In: *IEEE CEC 2010*. IEEE, New York, pp 1–7
- Haasdijk E, Eiben AE, Karafotias G (2010b) On-line evolution of robot controllers by an encapsulated evolution strategy. In: *Proceedings of the IEEE congress on evolutionary computation, CEC 2010*, Barcelona, Spain, 18–23 July 2010, pp 1–7. <https://doi.org/10.1109/CEC.2010.5585926>
- Haasdijk E, Bredeche N, Eiben A (2014) Combining environment-driven adaptation and task-driven optimization in evolutionary robotics. *PLoS ONE* 9(6):e98466

- Kaplan F, Oudeyer P (2006) Intrinsically motivated machines. In: 50 Years of artificial intelligence, essays dedicated to the 50th anniversary of artificial intelligence, pp 303–314. https://doi.org/10.1007/978-3-540-77296-5_27
- Klyne A, Merrick KE (2016) Intrinsically motivated particle swarm optimisation applied to task allocation for workplace hazard detection. *Adapt Behav* 24(4):219–236
- Kompella VR, Luciw MD, Stollenga MF, Pape L, Schmidhuber J (2012) Autonomous learning of abstractions using curiosity-driven modular incremental slow feature analysis. In: 2012 IEEE international conference on development and learning and epigenetic robotics, ICDL-EPIROB 2012, pp 1–8
- Kompella VR, Stollenga MF, Luciw MD, Schmidhuber J (2014) Explore to see, learn to perceive, get the actions for free: SKILLABILITY. In: 2014 international joint conference on neural networks, IJCNN 2014, pp 2705–2712
- Lehman J, Miikkulainen R (2014) Overcoming deception in evolution of cognitive behaviors. In: Proceedings of the 2014 annual conference on genetic and evolutionary computation, GECCO'14. ACM, New York, pp 185–192
- Lehman J, Stanley KO (2011) Abandoning objectives: evolution through the search for novelty alone. *Evol Comput* 19(2):189–223
- Merrick KE (2017) Value systems for developmental cognitive robotics: a survey. *Cogn Syst Res* 41:38–55
- Nogueira YLB, Fisch de Brito CE, Vidal CA, Cavalcante-Neto JB (2013a) Emergence of autonomous behaviors of virtual characters through simulated reproduction. In: Advances in artificial life: ECAL 2013. The MIT Press, Berlin, pp 750–757
- Nogueira YLB, Fisch de Brito CE, Vidal CA, Cavalcante-Neto JB (2013b) Evolving plastic neuromodulated networks for behavior emergence of autonomous virtual characters. In: Advances in artificial life, ECAL 2013. The MIT Press, Berlin, pp 577–584
- Nogueira YLB, de Brito CEF, Vidal CA, Neto JBC (2016) Emergent vision system of autonomous virtual characters. *Neurocomputing* 173:1851–1867
- Oudeyer PY, Kaplan F (2008) How can we define intrinsic motivation? In: the 8th international conference on epigenetic robotics: modeling cognitive development in robotic systems, Lund university cognitive studies. LUCS, Lund. <https://hal.inria.fr/inria-00420175>
- Oudeyer PY, Smith L (2016) How evolution may work through curiosity-driven developmental process. *Top Cogn Sci* 8. <https://doi.org/10.1111/tops.12196>. <https://hal.inria.fr/hal-01404334>
- Oudeyer PY, Kaplan F, Hafner V (2007) Intrinsic motivation systems for autonomous mental development. *IEEE Trans Evol Comput* 11(2):265–286. <https://doi.org/10.1109/TEVC.2006.890271>. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4141061
- Schmidhuber J (2010) Formal theory of creativity, fun, and intrinsic motivation (1990–2010). *IEEE Trans Auton Mental Dev* 2(3):230–247
- Silva F, Duarte M, Correia L, Oliveira SM, Christensen AL (2016) Open issues in evolutionary robotics. *Evol Comput* 24(2):205–236
- Trueba P, Prieto A, Bellas F, Duro RJ (2015) Applying the canonical distributed embodied evolution algorithm in a collective indoor navigation task. In: IJCNN. IEEE, New York, pp 1–8
- Varela FJ (1979) Principles of biological autonomy. North Holland, Amsterdam
- Varela FJ (1992) Autopoiesis and a biology of intentionality. In: McMullin B (ed) Autopoiesis and perception. Dublin City University, Dublin, pp 4–14
- Vernon D, Lowe R, Thill S, Ziemke T (2015) Embodied cognition and circular causality: on the role of constitutive autonomy in the reciprocal coupling of perception and action. *Front Psychol* 6:1660. <https://doi.org/10.3389/fpsyg.2015.01660>
- Watson RA, Ficici SG, Pollack JB (2002) Embodied evolution: distributing an evolutionary algorithm in a population of robots. *Robot Auton Syst* 39:1–18