GAIPS tech. rep. series, n. GAIPS-TR-003-13, October 2013

INTELLIGENT AGENTS AND SYNTHETIC CHARACTERS GROUP

Emergence of Emotion-Like Signals in Learning Agents

Pedro Sequeira, Francisco S. Melo and Ana Paiva
Intelligent Agents and Synthetic Characters Group
INESC-ID, Lisbon
Portugal

The positive impact of emotions in decision-making has long been established in both natural and artificial agents. Emotions complement the perceptual information acquired through the agent's sensors, coloring our sensations and thus guiding our decision-making. However, when designing autonomous agents, are *emotions* the best complement to the perceptions? Mechanisms investigated in affective neuroscience provide support for this hypothesis in biological agents. In this paper, we look for similar support in artificial systems. We adopt the intrinsically motivated reinforcement learning framework (IMRL) to investigate different sources of information that can guide decision-making in learning agents, and an evolutionary approach based on genetic programming to identify a small set of such sources that have the largest impact on the performance of the agent. We then show that these sources of information: (i) are applicable in a wider range of environments than those where the agents evolved; (ii) exhibit interesting correspondences to appraisal-like signals previously proposed in the literature, pointing towards our departing hypothesis that emotions might indeed provide essential information to complement perceptual capabilities and thus guide decision-making.

Contents

GAIPS/INESC-ID
TagusPark, Edifício IST
Av. Prof. Dr. Cavaco Silva
2780-990 Porto Salvo
Portugal

Tel.: +351 214 233 508
Fax: +351 214 233 290
http://gaips.inesc-id.pt/

Corresponding author:
Pedro Sequeira
Tel.: +351 214 233 553
E-mail: pedro.sequeira@gaips.inesc-id.pt
http://gaips.inesc-id.pt/~psequeira/site/

1 Introduction

Research on psychology, neuroscience and other related areas established emotions as a powerful adaptive mechanism that influences cognitive and perceptual processing [6, 8, 27]. Emotions indirectly drive behaviors that lead individuals to act, achieve goals and satisfy needs. Studies evidenced that damage to regions of the brain identified as responsible for emotional processing impact the human and animal ability to properly learn aversive stimuli, plan courses of action and, more generally, take decisions that are advantageous for their well-being [3, 7, 17].

In artificial systems, the area of affective computing (AC) also investigated the impact of emotional processing capabilities in the development of autonomous agents. "Emotional processing" was shown to improve the performance of artificial agents in terms of different metrics, such as robustness and efficiency [23, 29, 31, 33]. In very general terms, emotional architectures feature an emotional processing module that, together with the perceptual information acquired by the agent, guides its decision process—see Fig. 1 [22, 23]. The *emotional signals*[1] provided by such module "translate" information about the history of interaction of the agent with its environment that aid decision-making, complementing the perceptual information acquired through the agent's sensors. In other words, such emotional signals give "color" to the agent's raw perceptions indicating, for example, whether a perception is expected or not, or pleasant or not.

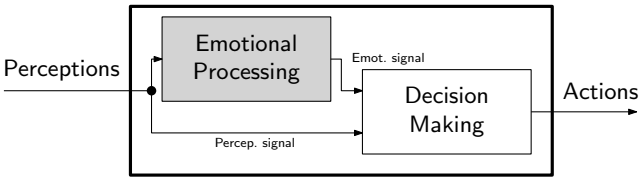

Figure 1: General architecture for an artificial agent with emotional processing [23].[2]

Although one of the driving motivations for the use of emotional agent architectures is the creation of "better agents" (*e.g.*, agents able to successfully perform more complex tasks) one fundamental question remains mostly unaddressed in the literature: in the search for information that may complement an agent's perceptual capabilities, are *emotions* the best candidate?

In this paper we contribute to this question, providing empirical evidence that emotion-like signals may arise as natural candidates when looking for sources of information to complement an agent's perceptual capabilities. Using an evolutionary approach, we show that emotion-related signals *emerge* as sources of information for artificial agents, providing evolutionary advantages. We thus contribute a computational parallel to the evidence observed in biological systems, where the organisms

[1] We adopt a rather broad definition of *signal*. Specifically, we refer to an emotional signal any emotional information received and processed, in this case, by the decision-making module.

[2] The diagram does not aim at providing an accurate representation of existing emotional architectures for autonomous agents, but instead to highlight the point that, in such architectures, the decision making process is driven by both *perceptual information* from the environment and also by some form of *emotional information*.

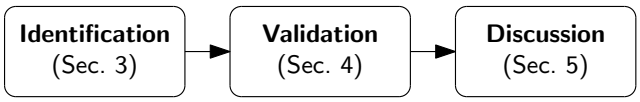

Figure 2: Roadmap for the study in the paper. We start by *identifying* optimal sources of information in Section 3. We validate these sources of information in Section 4 and conclude by discussing possible correspondences with appraisal dimensions of emotion in Section 5.

with the most complex emotional processing capabilities are arguably those most fit to their environment [17, 18, 26].

In our study, we rely on *intrinsically motivated reinforcement learning* agents [39]. The framework of intrinsically motivated reinforcement learning (IMRL) provides a principled manner to integrate multiple sources of information in the process of learning and decision-making of artificial agents [38].[3] As such, it is a framework naturally suited to our investigation.

Departing from an initial population of IMRL agents, each relying on different sources of information to guide their decisions, we use genetic programming to select those agents with maximal fitness. This evolutionary process allows us to identify a minimal set of informative signals that provide *general* and *useful* information for decision-making. Finally, we establish a correspondence between the identified sources of information and the information associated with appraisal variables usually identified in the specialized literature. This correspondence, although not formal, does provide some support to our hypothesis that emotion-like signals are natural sources of information to complement an agent's perceptual capabilities in the pursuit for more reliable artificial decision-makers.

The paper is organized according to the roadmap sketched in Fig. 2. Section 2 introduces the required background and notation on reinforcement learning. Section 3 identifies a minimal set of signals that provide the most useful information to guide IMRL agents. Section 4 analyzes the general applicability of the identified signals in a set of scenarios inspired by the game of Pac-Man. Finally, Section 5 analyzes the identified signals in light of the literature on emotions and summarizes our main findings.

2 Background

As discussed in Section 1, in our study we rely on reinforcement learning (RL) agents. This section reviews basic RL concepts and sets up the notation used throughout the paper. We refer to [13, 42] for a detailed overview of RL.

[3] These complementary sources of information endow the agent with a richer repertoire of behaviors that may successfully overcome agent limitations [33, 40]. In particular, emotion-like signals were studied as general but powerful sources of information and successfully applied in a variety of scenarios [33, 35].

2.1 Learning and Decision Making

At each time step, and depending on its perception of the environment, an RL agent must choose an action from its action repertoire, in order to meet some pre-specified optimality criterion. Actions determine how the state of the environment evolves in time and, depending on such state, different actions have different value for the agent. Typically, the RL agent knows neither the value nor the effect of its actions, and must thus *explore* its environment and action repertoire before it can adequately select its actions.

By *state of the environment* we refer to any feature of the environment that may be relevant for the agent to choose its actions optimally. Ideally, the agent should be able to unambiguously perceive all such features. Sometimes, however, the agent has limited sensing capabilities and is not able to completely determine the current state of the system. When this is the case, the agent has *partial observability*. Throughout the paper, most agents considered have partial observability.

RL agents can be modeled using the *partially observable Markov decision process* (POMDP) framework [14]. We denote a POMDP as a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{Z}, \mathsf{P}, \mathsf{O}, r, \gamma)$, where

- $\mathcal{S}$ is the set of all possible environment states;
- $\mathcal{A}$ is the action repertoire of the agent;
- $\mathcal{Z}$ is the set of all possible agent observations;
- $\mathsf{P}(s' \mid s, a)$ indicates the probability that the state at time step $t+1$, S_{t+1}, is s', given that the state at time step t, S_t, is s and the agent selected action $A_t = a$.
- $\mathsf{O}(z \mid s, a)$ indicates the probability that the observation of the agent at time step $t+1$, Z_{t+1}, is z, given that the state at time $t+1$ is s and the agent selected action a at time t.
- $r(s, a)$ represents the *average reward* that the agent expects to receive for performing action a in state s.
- $0 \leq \gamma < 1$ is some discount factor.

A POMDP evolves as follows. At every time step $t = 0, 1, 2, 3, \ldots$, the environment is in some state $S_t = s$. The agent selects some action $A_t = a$ from its action repertoire, $\mathcal{A}$, and the environment transitions to state $S_{t+1} = s'$ with probability $\mathsf{P}(s' \mid s, a)$. The agent receives a *reward* $r(s, a) \in \mathbb{R}$ and makes a new observation $Z_{t+1} = z$ with probability $\mathsf{O}(z \mid s', a)$, and the process repeats.[4]

The objective of the agent can be formalized as that of gathering as much reward as possible throughout its lifespan, usually discounted by the constant γ. This

[4]Typical RL scenarios assume that $\mathcal{Z} = \mathcal{S}$ and $\mathsf{O}(z \mid s, a) = \delta(z, s)$, where δ denotes the Kronecker delta [42]. When this is the case, parameters $\mathcal{Z}$ and O can be safely discarded and the simplified model thus obtained, represented as a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathsf{P}, r, \gamma)$, is referred to as a *Markov decision process* (MDP).

corresponds to maximizing the value

$$v = \mathbb{E}\left[\sum_t \gamma^t r(S_t, A_t)\right]. \tag{1}$$

The reward $r(s, a)$ thus evaluates the immediate utility of making action a in state s, in light of the underlying task that the agent must learn. In order to maximize the value in (1), the agent must learn a mapping that, depending on its history of observations and actions, determines the next action that the agent should take. Such mapping, denoted as π, is known as a *policy*, and is typically learned through a process of trial and error. In this paper we focus on policies that depend on the agent's current observation. In other words, our agents follow policies $\pi : \mathcal{Z} \to \mathcal{A}$ that map each observation $z \in \mathcal{Z}$ directly to an action $\pi(z) \in \mathcal{A}$. If the state is fully observable, then $\mathcal{Z} = \mathcal{S}$ and $Z_t = S_t$. In this case, there is a policy $\pi^* : \mathcal{S} \to \mathcal{A}$ referred to as the *optimal policy* maximizing the value in (1). We can associate with π^* a function $Q^* : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ that verifies the recursive relation:

$$Q^*(s, a) = r(s, a) + \gamma \sum_{s' \in \mathcal{S}} \mathsf{P}(s' \mid s, a) \max_{b \in \mathcal{A}} Q^*(s', b). \tag{2}$$

$Q^*(s, a)$ represents the *value* of executing action a in state s and henceforth following the optimal policy. We can use the recursion in (2) to iteratively compute Q^* for all pairs $(s, a) \in \mathcal{S} \times \mathcal{A}$. Additionally,

$$V^*(s) = \max_{a \in \mathcal{A}} Q^*(s, a)$$

represents the value obtained by an agent starting from state s and henceforth following π^*. From the above, it should be apparent that the goal of the RL agent can be restated as that of learning Q^*, since from the latter it is possible to derive the optimal policy. Since RL agents typically have no knowledge of either P or r, one possibility is to *explore* the environment—*i.e.*, select actions in some exploratory manner—building estimates for P and r, and then using these estimates to successively approximate Q^*. After exploring its environment, the agent can then *exploit* its knowledge and select the actions that maximize (its estimate of) Q^*. Throughout the paper, our RL agents follow a simple variation of this approach known as *prioritized sweeping* [24].

2.2 Partial Observability and IMRL

As discussed above, our RL agents use prioritized sweeping to build estimates of P and r from which they then approximate Q^*. Both P and r can be estimated by maintaining running averages of the corresponding values. For example,

$$\hat{\mathsf{P}}(s' \mid s, a) = \frac{n_t(s, a, s')}{n_t(s, a)},$$

models the probability of transition from state s to state s' by means of action a as the ratio between the number of times that, by time step t, the agent experienced a

transition from s to s' after selecting action a—$n_t(s, a, s')$—and the number of times agent selected action a in state s—$n_t(s, a)$.

However, as already mentioned, most agents considered in this paper have partial observability—*i.e.*, they are unable to unambiguously determine the state of their environment and are only able to perceive some features of this state. This is similar to what occurs in nature: individuals are only able to perceive the environment in their immediate surroundings. Such limited perception necessarily impacts their decision-making process. For example, while the optimal course of action for a hungry predator is to approach its prey, this actually requires the predator to be able to figure out the position of the prey. Similarly, partial observability also impacts the ability of our RL agents to select optimal actions.

In terms of their learning algorithm, our RL agents treat each observation Z_t as the full state of the environment. They thus build a transition model $\hat{\mathsf{P}}(z' \mid z, a)$ and $\hat{r}(z, a)$ that will generally provide inaccurate predictions. This model is then used to build a Q-function $\hat{Q} : \mathcal{Z} \times \mathcal{A} \to \mathbb{R}$ that the agent uses to guide its decision process. It is a well-established fact, in scenarios with partial observability, observations alone are not sufficient for the agent to accurately track the underlying state of the system. Therefore, policies computed by treating observations as states can lead to arbitrarily poor performance [37]. Moreover, computing the best such policy is generally hard [19]. In fact, creating robust RL agents that can overcome perceptual limitations often involves significant modeling effort and expert knowledge.

The *intrinsically motivated reinforcement learning* (IMRL) framework [38, 39] proposes the use of richer reward functions that implicitly encode information to potentially overcome the agents' perceptual limitations. And, in fact, this approach was shown useful both to facilitate reward design [25, 35] and to mitigate agent limitations [4, 40, 41]. In this framework, the performance of RL agents in the original task provides a measure of the *fitness* of those agents. Different agents, each with a different reward function accounting for multiple sources of information, are then compared in terms of their fitness, and the most fit agent is selected. This selection process allows to identify, for a given set of environments, which sources of information are most useful to maximize the fitness of RL agents in the task at hand, providing a natural framework for the study in this paper.

Formally, IMRL extends traditional RL and provides a framework to address the *optimal reward problem* (ORP), that we now describe [40]. Let H_t be a random variable representing the history of interaction of an agent with its environment up to time-step t, and let $h_t = \{z_1, a_1, \rho_1, \ldots, z_{t-1}, a_{t-1}, \rho_{t-1}, z_t\}$ denote a particular realization of H_t. Such history corresponds to all information perceived by the agent directly from the environment: sequence $\{z_\tau, \tau = 1, \ldots, t\}$ corresponds to observations about the environment state (according to the POMDP model described in Section 2); similarly, $\{a_\tau, \tau = 1, \ldots, t\}$ corresponds to the sequence of actions performed by the agent; finally, $\{\rho_\tau, \tau = 1, \ldots, t\}$ corresponds to an "external" evaluation signal that, at each time-step t, depends only on the underlying state S_t of the environment and the action A_t performed by the agent. This signal can be either environment feedback—for example, when an agent receives a monetary prize for performing some action—or physiological feedback—for example, when an agent feels satisfied after feeding.

Given a particular finite history h, we write $p_H(h \mid r, e)$ to denote the probability of an RL agent[5] observing history h in environment e when its reward function is r. We evaluate the agent's performance by means of some real-valued *fitness function* $f : \mathcal{H} \to \mathbb{R}$, where $\mathcal{H}$ is the space of all possible (finite) histories. Then, given a space $\mathcal{R}$ of possible reward functions, a set $\mathcal{E}$ of possible environments, and a distribution $p_E(\mathcal{E})$ over the environments in $\mathcal{E}$, the ORP seeks to determine the *optimal reward function*, denoted by r^*, maximizing the fitness over the set $\mathcal{E}$ according to

$$r^* = \operatorname*{argmax}_{r \in \mathcal{R}} \mathcal{F}(r), \tag{3}$$

where $\mathcal{F}(r)$ is the expected fitness of the RL agent using the reward function r, which is given by

$$\mathcal{F}(r) = \sum_{h,e} f(h) p_H(h \mid r, e) p_E(e), \tag{4}$$

where each e and h is sampled according to $p_E(e)$ and $p_H(h \mid r, e)$, respectively.

Throughout this paper, we specifically consider the fitness associated with a given history h_t is given by

$$f(h_t) = \sum_{\tau=1}^{t} \rho_t. \tag{5}$$

From the above, it should be apparent that the signal $\{\rho_t, t = 1, \ldots\}$ actually corresponds to a (external) reward signal that determines the fitness of the agent. We thus define the function $r^{\mathcal{F}} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ as

$$r^{\mathcal{F}}(s, a) = \mathbb{E}\left[\rho_t \mid S_t = s, A_t = a\right], \tag{6}$$

and henceforth refer to $r^{\mathcal{F}}$ as the *fitness-based reward function*. The function $r^{\mathcal{F}}$ can be seen as the sparsest representation of the task to be learned by the agent, as encoded by the signal $\{\rho_t\}$. We consider throughout the paper that $r^{\mathcal{F}} \in \mathcal{R}$.

The interest of considering the ORP problem instead of simple RL agents driven only by the reward $r^{\mathcal{F}}$ is that, in the presence of agents with limitations, the solution r^* to the ORP is often a better alternative than $r^{\mathcal{F}}$. In fact, the reward r^* obtained often leads to *faster learning* and induces behaviors that are more *robust* and *efficient* than those induced by $r^{\mathcal{F}}$ [33, 40].

3 Identification of Optimal Sources of Information

Referring back to the roadmap in Fig. 2, we now address the baseline question driving our study: *which information is (potentially) most useful to complement the perceptual capabilities of an autonomous learning agent*? In other words, and referring back to the diagram of Fig. 1, we investigate possible alternatives to the emotional processing module that may most significantly impact agent performance (see Fig. 3).

[5] Our RL agent all follow the prioritized sweeping algorithm and use the exploration policy detailed in Section 3.

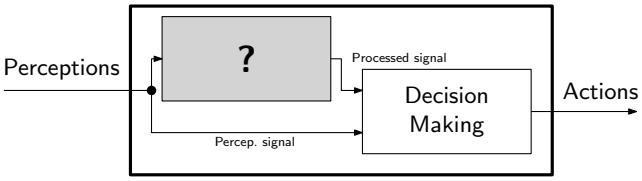

Figure 3: General architecture for an artificial agent.

To address this general question, we consider foraging scenarios where an IMRL agent acts as a predator in an environment such as those in Fig. 6. The perceptual limitations of the agent in the different environments pose challenges that directly impact its ability to capture its prey and, consequently, its *fitness*.

In order to identify possible sources of useful information to complement the agent's perceptual limitations, we depart from a primitive population of agents, each endowed with a reward structure containing information about different aspects of the agent's past interactions with its environment. The fittest agents (*i.e.*, those with greatest ability to capture preys) are used to successively improve the population. Upon convergence, we identify the set of agents able to attain the largest fitness. The analysis of the corresponding reward structure provides the required information about which signals are potentially most useful to complement the perceptual capabilities of our IMRL agents.

3.1 Methodology

In order to determine which reward functions—and, consequently, which information—best complements the agent's perceptions, we adopt the *genetic programming* (GP) approach proposed by Niekum et al. [25]. In that work, the authors used GP in the context of IMRL and the ORP as a possible approach to identify optimal rewards for RL agents. The procedure consisted in searching for reward functions represented by programs that combine different elements of the learning domain, such as the agent's position in the environment or its hunger status.

In the context of our work, there are some appealing features in the use of GP. Recall from Section 2.2 that the ORP involves the definition of a space of reward functions $\mathcal{R}$ and an optimization procedure to search for the optimal reward function r^*. GP facilitates the definition of the space of rewards by alleviating the need to specify an explicit parameterization. Instead, we implicitly define the space of possible rewards by specifying a set of operators and terminal nodes, the latter corresponding to constants or variables. Moreover, the optimization mechanism is implicitly defined by a selection method and mathematical operators that combine the terminal nodes, constructing richer, more complex and potentially more informative signals as the evolutionary procedure progresses. Another appealing feature of GP over other search methods (such as gradient descent [41]), in the context of our study, is its close parallel with natural evolution. In the continuation, we provide a detailed description of the setup and procedure used in this first experiment.

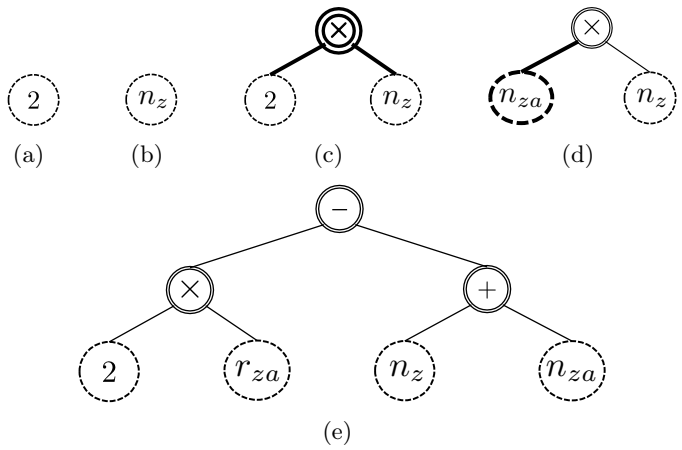

Figure 4: Defining reward functions as genetic programs. Some examples: (a) a constant GP node; (b) a variable GP node; (c) a GP tree obtained by a crossover operation between the nodes in (a) and (b); (d) a GP tree obtained by a mutation operation made to the tree in (c); (e) possible evolved GP tree. Bold nodes and lines indicate changes in the tree induced by the several operations. See text for detailed explanation.

3.1.1 Genetic Programming

In general terms, GP aims to find a *program* that maximizes some measure of *fitness* [15]. Programs are represented as syntax trees, where nodes correspond to either *operators* or *terminal nodes* representing *primitive quantities*. In our case, we use as terminal nodes quantities that summarize aspects of the history of interaction of the agent with its environment. The GP approach allows for the discovery of interesting mathematical relations between such primitive quantities. Fig. 4 shows the basic elements and operations involved in the approach of using GP to represent and evolve reward functions within IMRL.

Non-operator (terminal) nodes are selected from a set $\mathcal{T}$ of possible terminal nodes, and represent either numerical variables or constants. Fig. 4(a) shows an example of a GP tree with a single *constant terminal node* representing the reward function $r = 2$. Fig. 4(b) shows an example of a tree with a single *variable terminal node* representing a reward function $r = n_z$ that rewards visits to state z according to the number of times it was observed.

Operators are selected from a set $\mathcal{O}$ of possible operators, and its arguments are represented as their descendants in the tree. GP iteratively explores possible solutions by maintaining a population of candidate programs, producing new generations of programs by means of *selection*, *mutation* and *crossover*. The crossover function randomly replaces some sub-tree (a node and all of its descendants) of a parent program by another sub-tree from another parent upon reproduction. Fig. 4(c) shows an example of a GP tree that could be obtained through a crossover operation between the nodes in Figs. 4(a) and 4(b), where the multiplication operator node was introduced. The resulting tree represents the reward function $r = 2n_z$. The mutation operator replaces some node by another randomly selected one. For example,

Fig. 4(d) depicts a possible GP tree obtained by a mutation operation made to the tree in Fig. 4(c), where the left node was replaced. The resulting tree represents the reward function $r = n_{za}n_z$.[6]

In our experiments, we used for primitive quantities the set $\mathcal{T} = \mathcal{C} \cup \mathcal{V}$, with $\mathcal{C}$ corresponding to the set of constants, $\mathcal{C} = \{0, 1, 2, 3, 5\}$, and $\mathcal{V}$ to the set of basic variables, $\mathcal{V} = \{r_{za}, n_z, n_{za}, v_z, q_{za}, d_z, e_{za}, p_{zaz'}\}$, where

- $r_{za} = \hat{r}_t^{\mathcal{F}}(z, a)$ is the agent's estimate at time t of the fitness-based reward function for performing action a after observing z. This basic variable essentially informs the agent of its performance in respect to the external signal ρ provided by its environment/designer.[7] It is a function of z, a, and the agent's history up to time t, H_t.

- $n_z = n_t(z)$ is the number of times that z was observed up to time-step t. This signal informs the agent about the frequency of observations. When compared globally across observations it can be used by the agent *e.g.*, to determine which states were observed more often or which may need further exploration. It is a function of z and the agent's history up to time t, H_t.

- $n_{za} = n_t(z, a)$ is the number of times the agent executed action a after observing z up to time-step t. Similarly to n_z, this signal informs the agent about how frequent some action was executed after observing some state. It is a function of z, a, and the agent's history up to time t, H_t.

- $v_z = V_t^{\mathcal{F}}(z)$ is the value function associated with the reward function estimate $\hat{r}_t^{\mathcal{F}}$. As we have seen in Section 2.1 this function indicates the expected value (relating fitness attainment) of having observed z and following the current policy being learned henceforth. This signal can be used to inform the agent about the fitness-based "long-term utility" associated with some observation. It is a function of z and the agent's history up to time t, H_t.

- $q_{za} = Q_t^{\mathcal{F}}(z, a)$ is the Q-function associated with the reward function estimate $\hat{r}_t^{\mathcal{F}}$. Likewise v_z, it can be used to indicate the "long-term impact" for the agent's fitness of executing some action given some observation. It is a function of z, a, and the agent's history up to time t, H_t.

- $d_z = \hat{d}_t(z)$ corresponds to an estimate of the number of actions needed to reach a *goal* after observing z. Goals correspond to those observations that maximize $\hat{r}_t^{\mathcal{F}}$ and therefore this variable denotes observations that are close/far away from experienced situations providing maximal *immediate* fitness. This signal can be used by the agent in its planning mechanism to pursue courses of action that will lead to greater degrees of fitness in the long-run. It is a function of z and the agent's history up to time t, H_t.

[6] More details on GP can be found in [15].

[7] Recall that $r^{\mathcal{F}}(s, a)$ rewards the agent in accordance with the increase/decrease of fitness caused by executing each a in each state s.

- $e_{za} = \mathbb{E}\left[\Delta Q_t^{\mathcal{F}}(z,a)\right]$ is the expected *Bellman error* associated with $Q_t^{\mathcal{F}}$ at (z,a). Given an observed transition (z,a,r,z'), the Bellman error associated with $Q_t^{\mathcal{F}}$ is given by

$$\Delta Q_t^{\mathcal{F}}(z,a) = \hat{r}_t^{\mathcal{F}}(z,a) + \gamma \max_{b \in \mathcal{A}} Q_t^{\mathcal{F}}(z',b) - Q_t^{\mathcal{F}}(z,a).$$

  This signal essentially indicates the prediction error associated with some transition. If the agent receives a reward and observes a situation which value greatly differs from the previous value attributed by $Q^{\mathcal{F}}(z,a)$ then this transition will denote a *discrepancy* between what was observed and the agent's previous model of the world. The agent can use this basic variable to *e.g.*, identify situations changing very often or choose actions leading to more stable outcomes. It is a function of z, a, and the agent's history up to time t, H_t.

- $p_{zaz'} = \hat{\mathsf{P}}_t(z' \mid z,a)$ corresponds to the estimated probability of observing z' when executing action a after observing z. Since the learning algorithm used by the agent averages the perceived reward function, $p_{zaz'}$ is actually equivalent to

$$\mathbb{E}\left[\hat{\mathsf{P}}(z' \mid z,a)\right] = \sum_{z' \in \mathcal{Z}} \hat{\mathsf{P}}_t(z' \mid z,a) \mathbb{P}\left[Z_{t+1} = z' \mid Z_t = z, A_t = a\right].$$

  Similarly to e_{za}, this signal can be used by the agent to identify the execution of actions leading to more (un)stable outcomes, *i.e.*, the greater the number of transitions z' observed so far after executing a in z, the smaller the value of $p_{zaz'}$, hence the more "unreliable" or "erratic" pair z,a will be. $p_{zaz'}$ is a function of z, a, and the agent's history up to time t, H_t.

The variables above include all elements stored and/or computed by the learning agent, and therefore summarize the agent's history of interaction with its environment. As for the operators used by the GP algorithm, we considered the set $\mathcal{O} = \{+, -, \times, /, \sqrt{\cdot}, \exp, \log\}$.

Throughout time, and according to the fitness obtained for each reward function, the GP procedure applies the aforementioned operations to evolve relations between the primitive variables and constants in set $\mathcal{T}$ and the mathematical operators in set $\mathcal{O}$. For example, the GP tree depicted in Fig. 4(e) represents a more complex reward function expressed by the program $2r_{za} - (n_z + n_{za})$ that could be obtained after a few iterations of the GP algorithm. This evolved function rewards the agent for fitness-inducing behaviors by means of the relation $2r_{za}$ and punishes the agent as it becomes more and more "familiarized" with z and a, as given by $-(n_z + n_{za})$.

3.1.2 Evolutionary Procedure

Figure 5 outlines the optimization scheme for the ORP using GP, for a specific set of environments $\mathcal{E}$. At each generation j, a *reward function population* $\mathcal{R}_j$ of size K contains a set of *candidate reward functions* $r_k, k = 1, \ldots, K$. Each $r_k \in \mathcal{R}_j$ is evaluated according to the fitness function $\mathcal{F}(r_k)$. When all the reward functions have been evaluated, the evolutionary procedure takes place by applying the *mutation*

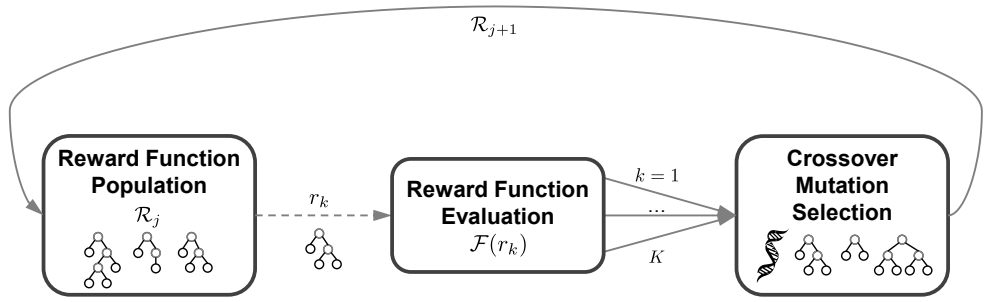

Figure 5: The GP approach to the ORP, as proposed in [25]. In each generation j, a population $\mathcal{R}_j$ contains a set of candidate reward functions $r_k, k = 1, \ldots, K$. All are evaluated according to a fitness function $\mathcal{F}(r_{k)}$ and evolve according to crossover, mutation and selection.

and *crossover* operations defined earlier and applying *selection* over the population in order to produce the new generation of reward functions, corresponding to population $\mathcal{R}_{j+1}$. The process repeats for a number J of generations.[8]

In our experiments, to run the evolutionary procedure we generate a total of 50 independent initial populations, each containing $K = 100$ elements, and run the evolutionary procedure for $J = 50$ generations for each population. For the selection method we use a *steady-state procedure* [43] that, in each generation j, maintains the 10 most fit elements—the reward functions with highest fitness—and generates 10 new random elements. The remaining 80 elements are generated either by mutating one element or through crossover by pairing elements of the previous population according to a *rank selection* that chooses parents with a probability that is proportional to their fitness, *i.e.*, reward functions with a greater fitness have a higher probability of being mutated or paired with another reward function.

Recall that resolving the ORP implies the definition of a *space of reward functions* and the determination of the *optimal reward function* r^* for a specific scenario. By space of reward functions we refer to the set of all reward functions that can (potentially) be generated by the GP algorithm. In particular, any possible combination of the primitive quantities in $\mathcal{T}$ and the operators in $\mathcal{O}$ that may be generated throughout time by the evolutionary procedure corresponds to a possible reward function and, as such, to an element of our so-called space of reward functions. The parameterization is therefore implicitly defined by the sets $\mathcal{T}$ and $\mathcal{O}$. The evolved optimal reward function is determined by (3) for all $r_k \in \mathcal{R}_j, j = 1, \ldots, J$, *i.e.*, it corresponds to the reward function with highest fitness considering all generations of all the populations that were initialized.

As an effect of mutation and crossover, reward functions might gain sub-expressions that do not contribute to the overall fitness attained by the agent as time evolves. Because we are interested in identifying only the interesting sources of information from the optimal reward functions, in a *post hoc* procedure r^* is parsed for sub-expressions that may have no effect on the computed fitness. This is done by first generating all possible sub-combinations of the tree representing r^*. For example,

[8] The first generation, corresponding to the population $\mathcal{R}_1$, is randomly generated.

an optimal reward function defined by the program $2r_{za} - (n_z + n_{za})$, depicted in Fig. 4(e), would generate the following sub-expressions: 2, $2r_{za}$, $2r_{za} - n_z$, $2r_{za} - n_{za}$, $2 - (n_z + n_{za})$, $2 - n_z$, $2 - n_{za}$, r_{za}, $r_{za} - (n_z + n_{za})$, $r_{za} - n_z$, $r_{za} - n_{za}$, $-n_z$, $-(n_z + n_{za})$ and $-n_{za}$. Each sub-expression is used to form a new reward function and its fitness is estimated. The "simplified" optimal reward function is then selected as the shortest sub-expression (in number of nodes) which difference in fitness in relation to the evolved optimal reward function is not statistically significant.[9] Many simplifications involve operations with the constants 0 and 1 as they sometimes cancel or offer no effect of the associated nodes to the overall reward, *e.g.*, expression $0r_{za} - 1(v_z - (\exp(0)q_{za})) + \log(1)$ would automatically simplify to $q_{za} - v_z$. In general, depending on the results for each scenario, other sub-expressions, possibly involving variable nodes, may be removed from r^*.

3.1.3 Estimating the Reward Function Fitness

It is a computationally demanding endeavor to explicitly compute $\mathcal{F}(r)$, since it involves computing the expectation of f over p_H and p_E, as seen in (4). As such, in order to estimate the value $\mathcal{F}(r)$, corresponding to the "reward function evaluation" stage in Fig. 5, we run $N = 200$ independent Monte-Carlo trials of $100,000$ time-steps each, where in each trial we simulate an RL agent driven by reward r in an environment selected randomly from the corresponding environment set, $\mathcal{E}$.[10] We then approximate $\mathcal{F}(r)$ as the mean fitness across all observed histories, *i.e.*,

$$\mathcal{F}(r) \approx \frac{1}{N} \sum_{i=1}^{N} f(h^i), \tag{7}$$

where h^i is the sampled history in the ith trial.

3.1.4 Scenarios

We used a total of six scenarios (see Fig. 6), either from the IMRL literature or modifications thereof [35, 39, 40]. We refer to [35] for a more detailed description of each environment and associated challenges.

Hungry-Thirsty scenario: The environment is depicted in Fig. 6(a). It contains two inexhaustible resources, corresponding to food and water. Resources can be positioned in any of the environment corners (positions $(1:1)$, $(5:1)$, $(1:5)$, and $(5:5)$), leading to a total of 12 possible configurations of food and water. The agent's fitness is defined as the amount of food consumed. However, the agent can only consume food if it is not *thirsty*, a condition that the agent can achieve by consuming the water resource (drinking). At each time-step after drinking, the agent becomes thirsty again with a probability of 0.2. The agent observes its position and thirsty status.

[9] We resorted to a simple unpaired t test to determine this statistical significance.

[10] The set $\mathcal{E}$ is scenario-specific. For example, in the *Hungry-Thirsty* scenario, $\mathcal{E}$ includes all possible configurations of food and water.

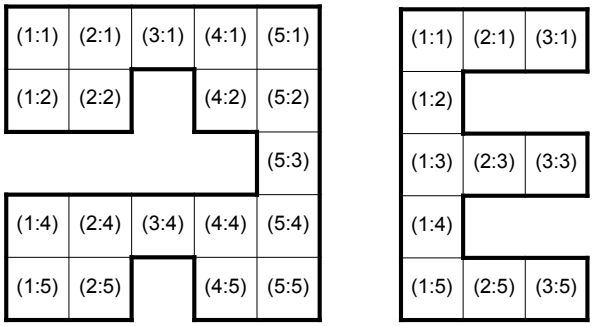

Figure 6: Structure of the foraging environments used in the first set of experiments. The pairs $(x : y)$ indicate the possible locations for the agent.

Lairs scenario: In this scenario, the layout of the environment corresponds again to Fig. 6(a). In it, the agent is a predator and there are two prey lairs positioned in different corners of the environment, resulting in 6 possible configurations. The fitness of the agent is defined as the number of preys captured. Whenever a lair is `occupied` by a prey, the agent can drive the prey out by means of a *Pull* action. The state of the lair transitions to `prey outside`, and the agent has exactly one time-step to capture the prey with a *Capture* action, before the prey runs away. In either case, the state of the lair transitions to `empty`. At every time-step there is a 0.1 probability that a prey will appear in an `empty` lair. In this scenario, $\mathcal{A} = \{N, S, E, W, P, C\}$, where N, S, E and W move the agent in the corresponding direction, and P and C correspond to the *Pull* and *Capture* actions. The agent is able to observe its position in the environment and the state of both lairs.

Exploration scenario: The environment for this scenario is depicted in Fig. 6(b). In this scenario, the agent is a predator and, at any time-step, there is *exactly one prey* available, located in one of the end-of-corridor locations (positions $(3 : 1)$, $(3 : 3)$ or $(3 : 5)$). The agent's fitness is again defined as the number of preys captured. Whenever the agent captures a prey, the latter disappears from the current location and a new prey randomly appears in one of the two other possible prey locations.

Persistence scenario: The environment again corresponds to the one in Fig. 6(b). In this scenario, the environment contains two types of prey always available. *Hares* are located in position $(3 : 1)$ and contribute to the fitness of the agent with a value of 1. *Rabbits* are located in position $(3 : 5)$ and contribute with a value of 0.01 to the agent's fitness. Whenever it captures a prey, the agent's position is reset to the initial position (position $(3 : 3)$). The environment also contains a *fence*, located in position $(1 : 2)$, that prevents the agent from easily capturing hares. In order for the agent to cross over the fence toward the hare location at time t, it must persistently perform the action N for N_t consecutive time-steps.[11] Every time the agent crosses the fence

[11] The fence is only an obstacle when the agent is moving upward from position $(1 : 2)$.

upwards, the fence is reinforced, requiring an increasing number of N actions to be crossed.[12]

Seasons scenario: The environment again corresponds to the one in Fig. 6(b). In this scenario the environment contains two possible types of prey. *Hares* are appear in position $(3:1)$ and contribute to the agent's fitness with a value of 1. *Rabbits* appear in position $(3:5)$ and contribute to the fitness of the agent with a value of 0.1. As with the *Persistence* scenario, the agent's position is reset to $(3:3)$ upon capturing any prey. However, unlike the *Persistence* scenario, in this scenario only one prey is available at each time-step, depending on the *season*, which changes every $5,000$ time steps. The initial season is randomly selected as either *Hare Season* or *Rabbit Season* with equal probability. Additionally, in the rabbit season, for every 10 rabbits that it captures, the agent is attacked by the rabbit farmer, which negatively impacts its fitness by a value of -1.

Poisoned prey scenario: This scenario is a variation of the the *Seasons* scenario. The scenario layout and prey positions are the same, but both rabbits and hares are always available to the agent. Rabbits contribute to the fitness of the agent with a value of 0.1. Hares, when *healthy* contribute positively to the agent's fitness by an amount of 1. When *poisoned* they contribute negatively to the fitness of the agent with a value if -1. As in the *Seasons* scenario, the health status of hares changes every $5,000$ steps.

3.1.5 Agent Description

In all scenarios, the agent is modeled as a POMDP whose state dynamics follow from the descriptions above. In all but the *Lairs* scenario, the agent has 4 actions available, $\mathcal{A} = \{N, S, E, W\}$ that deterministically move it in the corresponding direction; preys are captured automatically whenever co-located with the agent. In all but the *Hungry-Thirsty* and *Lairs* scenarios, the agent is only able to observe its current $(x:y)$ position, and whether it is collocated with a *prey*. In all experiments, we consider $\gamma = 0.9$.

All scenarios use prioritized sweeping RL agents [24] to learn a policy that treats observations as states (see Section 2). In our experiments, prioritized sweeping updates the Q-value of up to 10 state-action pairs in each iteration, using a learning rate of $\alpha = 0.3$. During its life-time, the agent uses an ε-greedy exploration strategy with a decaying exploration parameter $\varepsilon_t = \lambda^t$, where $\lambda = 0.999$.

3.2 Results

The results of the GP experiment are summarized in Table 1. We present the average fitness estimated according to (7) and the expression, simplified using the procedure described in Section 3.1.2, obtained by the agent using the evolved optimal reward function r^* selected using GP in each of the test scenarios. As a straightforward

[12] Denoting by $n_t(\text{fence})$ the number of times that the agent crossed the fence upwards up to time-step t, N_t is given by $N_t = \min\{n_t(\text{fence}) + 1; 30\}$.

Table 1: Mean fitness and evolved optimal reward function r^* for each scenario. For each scenario, we also include the performance of the fitness-based reward function $r^{\mathcal{F}}$. The results correspond to averages over 200 independent Monte-Carlo trials.

Scenario	Reward function	Mean Fitness
Hungry-Thirsty	$r^* = q_{za} - v_z - 2$	$10,252.1 \pm 6,773.1$
	$r^{\mathcal{F}} = r_{za}$	$7,129.4 \pm 6,603.2$
Lairs	$r^* = q_{za} - v_z$	$8,136.5 \pm 1,457.5$
	$r^{\mathcal{F}} = r_{za}$	$7,478.3 \pm 791.6$
Exploration	$r^* = -n_z^2$	$2,452.6 \pm 45.4$
	$r^{\mathcal{F}} = r_{za}$	381.1 ± 18.0
Persistence	$r^* = q_{za} - v_z$	$1,877.4 \pm 11.6$
	$r^{\mathcal{F}} = r_{za}$	136.1 ± 1.5
Seasons	$r^* = r_{za} + q_{za} - p_{zaz'}$	$6,426.1 \pm 149.1$
	$r^{\mathcal{F}} = r_{za}$	$4,936.4 \pm 1,900.9$
Poisoned prey	$r^* = 5r_{za} - q_{za}$	$5,233.7 \pm 715.3$
	$r^{\mathcal{F}} = r_{za}$	$1,284.3 \pm 4.1$

baseline for comparison, we also present the fitness obtained by an agent driven by the fitness-based reward function $r^{\mathcal{F}} = r_{za}$. We note that the agents compared are similar in all aspects except the reward function. In particular, the dimension of the transition function and Q-function learned are the same.

One first observation is that, in all scenarios, the evolved reward function clearly outperforms the fitness-based reward function. Our results are in accordance with findings in previous works on the advantages of allowing additional sources of information to guide the agent decision-making [4, 33, 39, 40].

Our results also confirm previous findings on the usefulness of an evolutionary approach to search for optimal reward functions [25]. There is, however, one key difference between our approach and that in [25]: we provide the evolutionary approach with *domain-independent* sources of information relating to the agent's history of interaction with the environment. We expect that the reward functions thus evolved can be applied in domains other than those used in our experiments and described in Section 3.1.

By analyzing the several simplified expressions that emerged from the evolutionary procedure in Table 1 we observe the presence of a particular sub-expression, that given by $q_{za} - v_z$. Aside from the fact that 3 out of the 6 rewards can be reconstructed directly from this quantity, it is a well-known quantity in the RL literature (known as the *advantage function* [2]). It proved to be crucial in scenarios having a great diversity of environment configurations, such as the hungry-thirsty and lairs scenario, and also in the persistence scenario. In this scenario, it was important for the agent to ignore sub-optimal decisions when facing the obstacle in the environment, *i.e.*, where choosing actions other than N (the one with higher advantage) was prejudicial in terms of the future gains provided by capturing the hare. The result of

the exploration scenario is given by the expression $-n_z^2$ and is quite obvious as it was important for the agent to explore the environment—by choosing states with a low number of visits—in order to capture the "moving" preys. In the seasons scenario, the resulting expression gives importance both to the fitness-based reward by means of $r_{za} + q_{za}$ and also to state-action pairs that provide low probability transitions as indicated by $-p_{zaz'}$. Such sub-expression proved useful for the agent to continue to go to the hares even when the seasons changed, thus avoiding the negative penalties from the rabbits. In the poisoned prey scenario, a greater importance was given to the fitness-based reward by means of the sub-expression $5r_{za}$ and the value provided by $-q_{za}$ ensured that the agent kept capturing hares, eventually gaining advantage in the *Healthy Season*.

3.3 Discussion

We recall that the goal of our first experiment was to identify possible sources of information that could improve the agent's performance if taken into consideration in the process of decision-making. Given the simplification process used to remove unnecessary sub-expressions from the optimal reward functions evolved through GP, each sub-expression indicated in Table 1 can be interpreted as possible "signal" that can drive the agent's decision process, allowing it to maximize its fitness. Discarding additive and multiplicative constants, we can distill from Table 1 a set of five signals, $\Phi = \{\phi_{\mathfrak{fit}}, \phi_{\mathfrak{adv}}, \phi_{\mathfrak{rel}}, \phi_{\mathfrak{prd}}, \phi_{\mathfrak{frq}}\}$, given by

- $\phi_{\mathfrak{fit}} = r_{za}$ corresponds to the agent's estimate of the *fitness-based reward function*. It evaluates the immediate impact on *fitness* associated with performing action a after observing z.
- $\phi_{\mathfrak{rel}} = q_{za}$ corresponds to the estimated Q-function associated with r_{za}. This function assesses the *value* of executing action a after observing z in terms of long-term impact on fitness, corresponding to the long-run counterpart to $\phi_{\mathfrak{fit}}$.
- $\phi_{\mathfrak{adv}} = q_{za} - v_z$ corresponds to the estimated advantage function associated with r_{za} [2]. This function evaluates how good action a is in state s *relatively to the best action* (its *advantage*). While $\phi_{\mathfrak{rel}}$ evaluates the absolute value of actions, $\phi_{\mathfrak{adv}}$ evaluates their relative value.
- $\phi_{\mathfrak{prd}} = p_{zaz'}$ corresponds to the agent's estimate of the transition probabilities. As discussed in Section 3.1, it provides a measure of how *predictable* the observation at time $t+1$ is given that the agent performed action a after observing z.
- Finally, $\phi_{\mathfrak{frq}} = -n_z^2$ provides a (negative) measure of how *novel* z is given the agent's observations.

The signals ϕ_k defined above correspond to the minimal set of sub-expressions from which we can form all the optimal reward functions for each scenario by combining them with the constants in $\mathcal{C}$ and using the different operators in $\mathcal{O}$.[13] As noted

[13] In our distillation process we are focused on extracting a *minimal* set domain-independent informative signals. As will become clearer in the next section, apart from additive constants (which

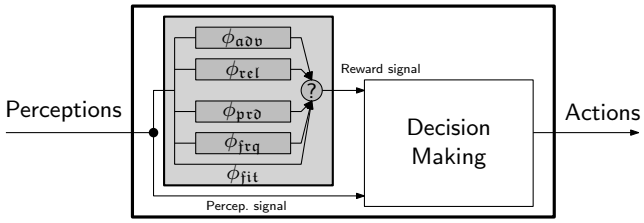

Figure 7: Architecture for an agent using the identified sources of information.

earlier, the expression of the advantage automatically emerged as a natural candidate for our optimal sources information due to representing the whole optimal reward function (discarding additive constants) in 3 of the 6 tested scenarios. The expression for novelty also emerged as a natural candidate. As for the remaining signals we opted by breaking down the two optimal reward functions $r_{za}+q_{za}-p_{zaz'}$ and $5r_{za}-q_{za}$ into their smallest terms, thus ensuring that a wide range of rewards can be reconstructed. It was particularly of interest to consider r_{za} as an independent signal given that, unlike the other basic variables, it is not learned and does not depend on the agent's experience, corresponding to an external evaluative signal. However, we note that this partitioning option is by no means unique, *e.g.*, the reward features $r_{za} - q_{za}$ and $r_{za} + q_{za} - p_{zaz'}$ are also a possibilities that could be used while still assuring a minimal set of information signals. Each of the emerged signals is a function mapping observation-action-history triplets to a real-value, and will henceforth be used as a *source of information* guiding the decision process of the agent. The updated agent architecture is depicted in Fig. 7.

Two observations are in order. First of all, in obtaining these signals, we considered a specific class of agents: our agents run prioritized-sweeping with ε-greedy exploration. Had a different learning algorithm or exploratory strategy been used, it is possible that variations of the identified features could be observed. However, as long as a reasonable exploration-exploitation trade-off is ensured, we would not expect the learning algorithm or exploration strategy to have a dramatic impact on our results. In particular, we would not expect these variations to dramatically change the sort of information required by the agent and provided by such features—and the consequent relation with the information in emotional processes.

Secondly, we would expect GP to yield different (and eventually more complex) signals, had we considered more elaborate domains. However, one interesting aspect of our results arises precisely from the fact that the features used throughout the paper were evolved in such simple scenarios. In spite of their simplicity, and as will soon become apparent, they yield significant improvements in performance in significantly more complex settings (that even include other agents). This, in our view, is indicative that, even though simple, they are extremely informative.

have minimum impact on the policy and can therefore be safely discarded), it will be possible to reconstruct the reward functions (and attain comparable degrees of fitness) in Table 1 as a linear combination of these signals.

4 Validation of Identified Sources

Section 3 focused on identifying general-purpose sources of information that can guide the decision process of an IMRL agent and impact positively its performance. These different sources of information emerged from the interaction of agents with several different environments and, as such, should be applicable in scenarios other than those in Section 3. This section investigates whether this is indeed so, *i.e.*, whether the sources of information identified in Section 3 can be of use in a broader range of scenarios than those considered so far. In particular,

- We show that the set of "signals" $\Phi = \{\phi_{\mathfrak{fit}}, \phi_{\mathfrak{adv}}, \phi_{\mathfrak{rel}}, \phi_{\mathfrak{prd}}, \phi_{\mathfrak{frq}}\}$ can be used to construct reward functions other than those in Table 1, establishing them as *general-purpose* sources of information for IMRL agents;
- We show that it is generally advantageous to indeed include one or more of the signals in Φ to construct the reward function driving our IMRL agents, establishing them as *universal* sources of information for IMRL agents.

The agent architecture considered in this Section specializes that in Fig. 3, specifically accounting for the sources of information in Φ (see Fig. 7). In this architecture, the reward signal driving the decision-making process is a *linear combination* of the different signals in Φ. We perform an initial validation, where we replicate the results reported in Table 1 using this architecture. We then perform a more challenging validation of the proposed architecture, by testing it in several significantly harder domains built from the well-known Pac-Man game.

4.1 Methodology

The linear formulation of the ORP adopted in this Section has been explored in the IMRL literature by different authors [38, 39, 40, 41]. In the context of this paper, the linear formulation has two appealing properties. First, by comparing the parameters associated with each source of information, we are able to perceive their relative importance in each scenario: signals for which the corresponding parameter has only a residual value have little weight in the agent's reward and, consequently, in the decision process of the agent. This is useful to assess whether the sources of information in Φ indeed provide useful information to guide the decisions of the agent.

A second appealing aspect of this formulation is that it allows a relatively general agent architecture, where all the signals in Φ are provided to the agent. The particular environment with which the agent interacts will condition *how* the agent uses these different signals, paralleling the evolutionary process by which natural organisms are conditioned to act differently in face of similar stimuli. We refer to [35, 39], where a more detailed discussion on the evolutionary interpretation of this framework is provided.

4.1.1 Linearly Parameterized ORP

In the linear formulation of the ORP, the space of possible rewards, $\mathcal{R}$, is considered as the linear span of some set Φ of real-valued *reward features*, $\Phi = \{\phi_1, \ldots, \phi_p\}$. In

other words, each reward $r \in \mathcal{R}$ is a linear combination of the features in Φ,

$$r(s,a,h) = \sum_{k=1}^{p} \phi_k(s,a,h)\theta_k = \boldsymbol{\phi}^\top(s,a,h)\boldsymbol{\theta}, \tag{8}$$

where the $\theta_k, k = 1,\ldots,p$, correspond to the parameters of the linear combination [39]. We henceforth write $r(\boldsymbol{\theta})$ to explicitly denote the reward function corresponding to the parameter vector $\boldsymbol{\theta}$. The ORP then reduces to finding the parameter vector $\boldsymbol{\theta}^*$ such that the corresponding reward function, $r(\boldsymbol{\theta}^*)$, has maximal fitness, *i.e.*,

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\operatorname{argmax}}\, \mathcal{F}(r(\boldsymbol{\theta})).$$

The above optimization can be conducted using different techniques [4, 39, 40, 41]. For our purposes, the particular algorithm by which $\mathcal{F}(r(\boldsymbol{\theta}))$ is optimized is irrelevant, and we therefore adopt the simple search approach used in [39].

In our experiments, we use as reward features the set of signals

$$\Phi = \{\phi_{\mathfrak{frq}}, \phi_{\mathfrak{rel}}, \phi_{\mathfrak{prd}}, \phi_{\mathfrak{adv}}, \phi_{\mathfrak{fit}}\}$$

identified Section 3. By optimizing the associated parameters (henceforth denoted $\{\theta_{\mathfrak{frq}}, \theta_{\mathfrak{rel}}, \theta_{\mathfrak{prd}}, \theta_{\mathfrak{adv}}, \theta_{\mathfrak{fit}}\}$), in a broad set of scenarios, we can analyze the relative importance of the different signals and draw conclusions on their general usefulness and applicability.

4.1.2 Scenarios

To validate the applicability of the signals identified in Section 3, we conducted two sets of experiments. In a first set of experiments we conduct an initial validation, again resorting to the foraging scenarios described in Section 3.1. This first set of experiments is essentially equivalent to those in Section 3, now using the linear formulation of the ORP instead of the GP approach. The goal is merely to replicate the results reported in Table 1 with the linear ORP formulation.

The second set of experiments is the central purpose of this section, and aims at providing a more extensive validation of the applicability of the signals in Φ as useful sources of information to complement the agent's perceptions. To this purpose, we consider several different scenarios inspired by the traditional computer game of Pac-Man. We use a total of four scenarios (see Fig. 8), each with different goals and posing different challenges to the agent.

Power-pellet scenario: In this scenario, our agent corresponds to the Pac-Man, and co-exists in the environment with two *ghosts*, the *smart ghost* and the *keeper ghost*. One pellet is available per episode (the *power-pellet*), and is located in the central cell of the environment. The power-pellet is *consumed* and removed from the environment as soon as Pac-Man reaches its position, and contributes to the fitness of Pac-Man with a value of 0.8.

In each episode, Pac-Man departs from the position depicted in Fig. 8 and the two ghosts depart from the central position. When a ghost and Pac-Man stand in

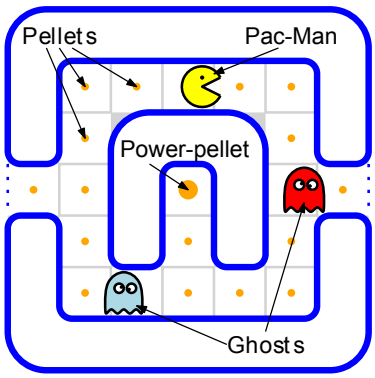

Figure 8: Structure and elements of the Pac-Man environment used in the second set of experiments.

the same cell, the ghost *captures* Pac-Man if the latter has not yet consumed the power-pellet, and *is consumed* by Pac-Man otherwise. The episode terminates as soon as one of the following conditions is met:

- Pac-Man consumes both ghosts (which contributes to its fitness with a value of 1).
- Pac-Man is captured by a ghost (which contributes to its fitness with a value of -1).
- 20 time-steps have elapsed after the power-pellet was consumed.

When an episode terminates, the environment is reset to its initial configuration and a new episode starts.

Eat-all-pellets scenario: In this scenario, our agent again corresponds to the Pac-Man, and co-exists with only the *smart ghost*. The environment has available a total of 20 pellets (one in each cell), which are consumed and removed from the environment whenever Pac-Man visits the corresponding cell. The Pac-Man and ghost initial configuration is the same as in the *power-pellet* scenario. In this scenario, consuming the power-pellet contributes to the fitness of the agent with a value of 0.5, but does not enable Pac-Man to consume the ghost. Instead, episodes terminate as soon as one of the following conditions occurs:

- Pac-Man consumes all 20 pellets (which contributes to its fitness with a value of 1).
- Pac-Man is captured 3 times by the ghost before all pellets are consumed (which contributes to its fitness with a value of -0.5).

When the ghost captures Pac-Man, their positions are reset. When an episode terminates, the whole environment (including existing pellets) is reset to its initial configuration and a new episode starts.

Rewarding-pellets scenario: In this scenario, our agent again corresponds to the Pac-Man, and co-exists with both the *smart ghost* and the *keeper ghost*. The environment has available a total of 20 pellets (one in each cell), which are consumed and removed from the environment whenever Pac-Man visits the corresponding cell. Each consumed pellet contributes to the fitness of the agent with a value of 0.1, in the case of a regular pellet, or 0.8, in the case of the power-pellet. The Pac-Man and ghost initial configuration is the same as in the previous scenarios. As before, consuming the power-pellet does not enable Pac-Man to consume the ghosts. Instead, an episode terminates as soon as one of the following conditions is met:

- Pac-Man consumes all 20 pellets (which contributes to its fitness with a value of 1).
- Pac-Man is captured by a ghost before all pellets are consumed (which contributes to its fitness with a value of -1).

When an episode terminates, the whole environment (including existing pellets) is reset to its initial configuration and a new episode starts.

Pac-Man scenario: This scenario is a combination of all previous scenarios, and is the one closest to the original game of Pac-Man. In this scenario, our agent again corresponds to the Pac-Man, and co-exists with only the *smart ghost*. The environment has available a total of 20 pellets (one in each cell), which are consumed and removed from the environment whenever Pac-Man visits the corresponding cell. The Pac-Man and ghost initial configuration is the same as in the *power-pellet* scenario. In this scenario, consuming the power-pellet does not contribute to the fitness of the agent, but does enable Pac-Man to consume the ghost. An episode terminates as soon as one of the following conditions is met:

- Pac-Man consumes all 20 pellets (which contributes to its fitness with a value of 1).
- Pac-Man is captured 3 times before all pellets are consumed (with no impact in fitness).

When the ghost captures Pac-Man, the fitness of the latter is decreased by a value of -0.1, and their positions are reset. When an episode terminates, the whole environment (including existing pellets) is reset to its initial configuration and a new episode starts.

4.1.3 Agent description

We refer to Section 3.1 for a description of the agent used in the foraging scenarios.

In the Pac-Man scenarios, our agent has 4 actions available, $\mathcal{A} = \{Up, Down, Left, Right\}$, that deterministically move it in the corresponding direction. The regions delimited by solid blue lines in Fig. 8 correspond to obstacles that cannot be traversed. Moving $Right$ in the leftmost/$Left$ in the rightmost cell moves Pac-Man to the rightmost/leftmost cell, respectively.

The motion of the ghosts respects the same restrictions as the motion of Pac-Man (*e.g.*, they cannot traverse obstacles). At every time-step, the *smart ghost* moves toward Pac-Man with probability 0.6. However, once Pac-Man consumes the power-pellet, the *smart ghost* instead moves *away* from Pac-Man (in the *power-pellet* and *pac-man* scenarios). With a probability 0.4 it moves in a random direction. The *keeper ghost* moves towards the *smart ghost* with probability 0.5 and towards one of the bottommost cells otherwise.[14]

In each scenario, our agent is modeled as a POMDP whose state-dynamics follow from the description above. In this POMDP model, the Pac-Man agent is able to observe, at each time step,

- Its current position $(x : y)$ in the environment;
- Whether a ghost exists in the same corridor as the agent, in each of the 4 possible directions;
- Whether a pellet exists in the same corridor as the agent, in each of the 4 possible directions;
- When co-located with the *smart ghost*;
- When co-located with the *keeper ghost*;
- When co-located with a pellet;
- When co-located with the power-pellet.

As in previous experiments, the Pac-Man scenarios use RL agents using prioritized sweeping [24] to learn a policy that treats observations as states. Prioritized sweeping updates the Q-value of up to 10 state-action pairs in each iteration, using a learning rate of $\alpha = 0.3$. During its life-time, the agent uses an ε-greedy exploration strategy with a decaying exploration parameter $\varepsilon_t = \lambda^t$, where $\lambda = 0.999$.

4.2 Results

We evaluate $\mathcal{F}(r(\boldsymbol{\theta}))$ by running $N = 200$ independent Monte-Carlo trials of $100,000$ time-steps each, where in each trial we simulate an RL agent driven by reward $r(\boldsymbol{\theta})$ in an environment selected randomly from the corresponding environment set, $\mathcal{E}$. $\mathcal{F}(r(\boldsymbol{\theta}))$ is then approximated as in (7).

4.2.1 Foraging Scenarios

The results corresponding to the foraging scenarios are summarized in Table 2. Comparing these results with those in Table 1, it is clear that the linear architecture is able to attain similar fitness values by combining the reward features in Φ. In some scenarios, the obtained fitness is, inclusively, slightly superior, although overall the differences are not statistically significant. It is also interesting to compare the

[14] The *keeper ghost*, when present, makes it difficult for Pac-Man to reach the central cell, essential for the completion of most scenarios.

Table 2: Mean cumulative fitness and obtained parameter vectors for each foraging scenario. The fitness results correspond to averages over 200 independent Monte-Carlo trials.

Scenario	Parameter Vector $[\theta_{\mathfrak{frq}}, \theta_{\mathfrak{rel}}, \theta_{\mathfrak{prd}}, \theta_{\mathfrak{adv}}, \theta_{\mathfrak{fit}}]^\top$	Mean Fitness
Hungry-Thirsty	$[0.0, 0.0, -0.2, 0.5, 0.3]^\top$	$10,718.8 \pm 7,226.5$
Lairs	$[0.1, 0.0, 0.3, 0.4, 0.2]^\top$	$9,598.6 \pm 1,543.4$
Exploration	$[1.0, 0.0, 0.0, 0.0, 0.0]^\top$	$2,394.5 \pm 48.8$
Persistence	$[-0.1, 0.0, 0.0, 0.5, 0.4]^\top$	$1,879.9 \pm 10.7$
Seasons	$[0.0, 0.2, -0.5, 0.1, 0.2]^\top$	$6,473.6 \pm 136.5$
Poisoned prey	$[0.0, 0.2, 0.0, -0.1, 0.7]^\top$	$5,297.9 \pm 529.4$

Table 3: Mean cumulative fitness and parameter vector determined in each of the Pac-Man scenarios. The results correspond to averages over 200 independent Monte-Carlo trials.

Scenario	Parameter vector $[\theta_{\mathfrak{frq}}, \theta_{\mathfrak{rel}}, \theta_{\mathfrak{prd}}, \theta_{\mathfrak{adv}}, \theta_{\mathfrak{fit}}]^\top$	Mean Fitness
Power-pellet	$[-0.2, 0.2, 0.1, 0.5, 0.0]^\top$	$1,265.0 \pm 424.9$
	$[0.0, 0.0, 0.0, 0.0, 1.0]^\top$	$-1,902.6 \pm 183.5$
Eat-all-pellets	$[0.1, 0.1, 0.1, 0.6, 0.1]^\top$	$1,005.5 \pm 207.1$
	$[0.0, 0.0, 0.0, 0.0, 1.0]^\top$	25.3 ± 215.5
Rewarding-pellets	$[0.5, 0.0, 0.1, 0.2, 0.2]^\top$	$4,343.7 \pm 210.1$
	$[0.0, 0.0, 0.0, 0.0, 1.0]^\top$	$3,060.8 \pm 208.6$
Pac-Man	$[0.2, 0.1, 0.2, 0.2, 0.3]^\top$	$1,223.6 \pm 117.5$
	$[0.0, 0.0, 0.0, 0.0, 1.0]^\top$	862.2 ± 95.7

weights associated with each of the signals in Φ, noting that the resulting signals generally match those found in Section 3.2. The results in Table 2 thus validate the signals in Φ as those responsible for the performance reported in Section 3.

4.2.2 Pac-Man Scenarios

The results of the Pac-Man experiment are summarized in Table 3. We present the average fitness obtained by the agent using the optimized reward parameters in each of the test scenarios, as well as the corresponding parameter vector. As a baseline for comparison, we also present the fitness obtained by an agent driven by the fitness-based reward function, corresponding to the parameter vector $\boldsymbol{\theta} = [0, 0, 0, 0, 1]^\top$.

As with the scenarios in Section 3, our results show that the Pac-Man agents that use only the fitness-based reward function are clearly inferior to those agents that use additional sources of information, namely those identified in Section 3.

This observation settles the main issue addressed in this section, on the general applicability of the sources of information identified in Section 3: not only using these signals is advantageous in terms of the performance of the agent but also, as seen from the weights in Table 3, these signals are generally more informative than the fitness-based reward function in most scenarios.

4.3 Discussion

We conclude this section by analyzing in greater detail some of the main challenges that the Pac-Man agent must face in the scenarios used in the experiments. These challenges allow us to evaluate the benefit of the reward features in Φ in guiding the agent.

First of all, the Pac-Man scenarios are significantly larger and more complex than the foraging scenarios of Section 3. For example, the *Rewarding Pellets* scenario has over 8×10^9 states. Additionally, the Pac-Man agent is lacking information regarding significant elements of the game, necessary to act optimally in the Pac-Man scenarios—for example, the agent generally cannot tell the position of the ghosts.

Considering some scenarios in more detail, in the *power-pellet* scenario, the observations of the agent were not sufficient to distinguish the different behavior of the ghosts before and after the power-pellet was consumed. As such, the fitness-based agent ended up suffering a significant number of captures, attaining significantly negative fitness. Our agent instead learned to avoid the ghosts in a first instance, and then to wait for the ghosts just bellow the power-pellet, which allowed it to capture them both after eating the power-pellet.[15]

In the *eat-all-pellets* scenario, the fitness-based agent learned a very conservative strategy, avoiding being eaten by the ghost and aiming only at the fitness increment provided by eating the power-pellet. Because the small pellets did not provide any direct fitness increment, it never got to eat all pellets and get the largest fitness increment. Our agent, guided by a balanced consideration of all available sources of information, was able to partly disregard the largest reward provided by the power-pellet and instead learned to avoiding the ghost throughout the environment, eating all small pellets in the process and, when possible, eating the power-pellet.

5 Discussion

We now analyze the *nature* of the signals from the perspective of emotion theories, particularly that of *appraisal theories*.

5.1 The Perspective of Appraisal Theories of Emotion

For the bodily signals and behavioral reactions that arise with emotions to be generated, stimuli perceived from the environment have to be somehow evaluated having

[15] Illustrative videos of the observed behaviors in different stages of the learning process in all the Pac-Man scenarios have been provided along with the submission (and are also available online at `http://gaips.inesc-id.pt/~psequeira/emot-emerg/`).

into account the organism's previous interactions with its environment [9, 28]. Appraisal theories propose that emotions are elicited by evaluations (appraisals) of the significance of a situation for an individual's well-being or goals [9]. The outcome of the emotional processing is a set of cognitive and behavioral responses to the eliciting event that have the objective of *coping* with the situation at hand [16, 18].

The most common frameworks of appraisal proposed in the literature model the elicitation of emotions as the result of a set of *appraisal variables*, each evaluating a particular aspect of the individual-environment relationship [9, 28]. Each variable can be conceptualized as a *dimension* varying continuously and characterizing several aspects of the situation in relation to the individual's goals [28]. While many theories differ in the specific number and types of appraisal variables and dimensions, there are some appraisal "themes" or "groups" of dimensions that all theories seem to agree to being part of the appraisal process. The work of Ellsworth and Scherer [9] overviewed several appraisal theories postulated within the psychology literature [10, 16, 18, 28, 32] and identified such appraisal "themes", which the authors refer to as "major appraisal dimensions". Each major dimension refers to the criteria and kind of information that is used to perform a certain appraisal, and each can be associated with particular aspects of the subject-environment relationship. We now summarize the major appraisal dimensions identified in [9].

Novelty: one of the most basic and low-level dimensions proposed by many appraisal theorists has to do with the *novelty* or *matching* between the perceived stimuli and the subject's acquired knowledge, usually referred to as the dimension of *familiarity* [10, 18]. In nature, the objective of this dimension is to focus the organisms' attention to changes perceived within the environment that might be relevant to its survival [9, 10].

Intrinsic Pleasantness: like with the novelty dimension, the major dimension of intrinsic pleasantness or *valence* is considered as a basic evaluation of a stimulus by determining the fundamental reaction of the organism—attraction versus aversion—according to whether the event is seen as "good" or "bad" to the subject. In nature, valence is thus considered an evaluation of the *value* of stimuli according to what the individual currently believes is the impact of a situation for its fitness [18]. Also, the criteria used in this kind of evaluations relates to innate rather than acquired feature detectors within the organism, some of which may even be universal or species specific [9]. Despite that, intrinsic pleasantness is also subject to learning and conditioning throughout the organism's lifetime, giving origin to acquired tastes and dislikes even for stimuli never experienced before [6, 9, 18].

Motivational Bases: this set of dimensions asserts the *motivational significance* or *conduciveness* of a situation in relation to the individual's long-term goals or the satisfaction of its needs [9, 16, 18]. In nature, an evaluation of the *goal relevance* of a situation is essential for the adaptation of an individual to its environment, promoting behaviors that seem to improve its goals and desires and disfavoring threatening situations [9]. Moreover, unlike intrinsic pleasantness providing general guidance on whether or not a stimulus should

be approached, this group of dimensions provides information about specific adaptive responses [9].

Power and Coping: according to many appraisal theorists of emotions, one important group of appraisal variables, referred to as *power and coping*, assesses the ability that an individual believes to possess in order to deal with some emotion-eliciting situation [9, 10, 16, 18]. The subject's *coping potential* usually refers to the *power* (physical, financial, cognitive, etc.) it has to asses the probability of possible outcomes and change the situation and its consequences [9]. Several aspects of the subject-environment relationship contribute to this major dimension, including the attribution of *causal agency*—the responsibility and intention behind the event—the assessment of *control*—whether some situation can be altered to favor the subject's objectives—and an *adjustment* evaluation—determining the ability to adapt to changing situations [9, 10, 16, 18, 32]. Such assessments often involve determining the degree of *predictability* or *likelihood* of the events being considered. The rationale is that situations which outcome is more predictable are more easy to cope with than those with more uncertain results [9].

Social Dimensions: another factor influencing emotional appraisal within natural organisms is its social context. The social dimensions make a subject take into account the beliefs, goals and actions of other members of its social group when taking its individual decisions [9]. One important factor influencing the appraisal variables within this group is the existence of shared rules or norms specifying status hierarchies and (un)acceptable behaviors within the social context [9]. Several appraisal theorists suggest the existence of "moral" dimensions related to *legitimacy*, *compatibility to standards* and *justice* with the purpose of promoting socialization and maintaining social order [9, 10, 18, 32].

For the purposes of our work, we now examine each of the sources of information described in Section 3 from the perspective of appraisal theories of emotions. Given the nature of our emerged features, the abstracted "major dimensions" identified in [9] and summarized above aid our analysis of the dynamical and structural properties of the signals and the type of information relating the agent's history of interaction with the environment that they evaluate. We note that we do not intend to match any of the emerged features to a specific variable or dimension suggested by the appraisal theories, but rather assess the emotional tone related to the evaluations they allow the agent to perform. Therefore, many of our reward features may be associated with several appraisal variables in the literature, but as will become clearer, each captures a distinct "theme" of the appraisal process. For each source of information we analyze the characteristics of the signals and compare them with the criteria defined by the major appraisal dimensions listed above.

5.1.1 Fitness

The expression for this source of information $\phi_{\mathfrak{fit}} = r_{za}$ signals behaviors that directly enhance or reduce fitness. As we have seen, computationally this feature corresponds

to the agent designer's reward function that directly ascribes *preferences* over the behavior of the agent. As such, fitness does not correspond to a subjective evaluation of the situation by the agent, a condition necessary for the process of emotional appraisal. This feature rather corresponds to externally ascribed, innate preferences over certain characteristics of the environment that, depending on the scenario, the agent may be conditioned to ignore by means of the optimization procedure. The most flagrant example of this occurred in the exploration scenario, where the best strategy focused only on the frequency of occurrence of stimuli, independently of their impact on fitness (see Table 2). As such, this feature is related to the major appraisal dimension of intrinsic pleasantness by evaluating preferences over the environment, but departs from it by not being subject to learning or depending on the agent's experience [9]—the "fitness value" of a situation is externally ascribed and depends on specific elements of the environment, *e.g.*, the hares and rabbits.[16]

5.1.2 Relevance

This source of information denotes the impact of executing actions in some states for the agent's fitness in the long-run, as given by the expression $\phi_{\mathfrak{rel}} = q_{za}$ indicating the expected return of executing actions in states according to the fitness-based action-value function. As described in [9], the major appraisal dimension assessing the motivational bases of a situation is fundamental for the adaptation of an individual by making it engage in goal-enhancing behaviors. As we have seen, within our learning framework the goal of the agent is to maximize its fitness in some environments of interest. Consequently, the relevance source of information has similar evaluative properties of appraisal variables such as goal-relevance/conduciveness by ascribing preference over actions that seem to lead to higher degrees of fitness in the long-run in a particular situation, and by denoting the contribution and future consequences of a particular behavior for the agent's goal.

In our experiments, when taken positively, the emerged feature of relevance fostered behaviors that the agent believed conducive for its goals, especially in scenarios where the environment (and the source of reward) changed constantly, as is the case of the prey season scenarios (see Table 2) and the power-pellets scenario (see Table 3). On the other hand, a negative weight associated with relevance might be useful in situations where relying in lower fitness-based rewards is beneficial compared to aiming at high-valued but possibly more irregular states.

5.1.3 Advantage

This source of information, expressed by $\phi_{\mathfrak{adv}} = q_{za} - v_z$, denotes the (dis)advantage of executing actions in some states considering their future impact on fitness. It thus gives origin to learned, acquired *preferences* by the agent towards behaviors it currently believes will lead to future high degrees of fitness in the environment. As described earlier, the major appraisal dimension of valence is considered an evaluation of the "value" of a situation in terms of being liked and therefore approached or

[16] Recall that in the seasons environments the value of preys depends on the current season, but this still is an external factor that the agent cannot act upon and change.

disliked and thus avoided. As we have seen in Section 3.1.1, within our framework the *value* of observing z in relation to the long-term fitness-based reward is precisely indicated by the function estimate v_z. Similarly, q_{za} indicates the expected fitness-based value of executing action a having observed z. They both therefore denote acquired preferences, *e.g.*, the fence in the Persistence scenario, unlike a rabbit or a hare, has no implicit or innate value to the agent in relation to its fitness, it's just an object in the environment. However, throughout time, the agent gained an acquired "taste" for the fence as it allowed the access to higher fitness-based rewards provided by the hare, and the evolved advantage feature allowed our agent to achieve a superior degree of fitness when compared to a standard agent, as indicated in Table 1.

As such, the advantage feature is in accordance with the perspective that the *implicit value* of things changes as a consequence of experience and associative learning processes [6, 9, 18]. Unlike the *static* external preferences attributed by the fitness signal, this signal relates to the emotional valence of stimuli acquired through experience [6]. It captures the essence of valence by rewarding (and thus promoting approach to) fitness-inducing states, and punishing (which leads to aversion towards) fitness-hindering situations, and also by being a continuous process biased by learning. While relevance corresponds to an estimate of future fitness gain, the advantage feature denotes the relative loss of executing some actions in certain states, which is sometimes important in situations where the value of stimuli changes throughout time, as occurred in the is the case of the ghosts in the power-pellet scenario, which impact on fitness changed depending of whether the power-pellet had been consumed by the agent (see Table 3).

5.1.4 Prediction

As stated earlier, this source of information, expressed through $\phi_{\mathfrak{prd}} = p_{zaz'}$, indicates how *predictable* the transition to some state is after the execution of an action in a previous state. Recall that the determination of the *predictability* or *likelihood* of an event is one of the mechanisms behind some of the appraisal variables within the major dimension of power and coping [9]. Namely, the *control* and *adjustment* evaluations depend on the subject's prediction power to assess whether the situation can be changed and how the environment is changing, respectively, in order to determine the appropriate response to an event [9]. As noted in [9, p.580], "control is not the same as predictability, although it often implies predictability, particularly as far as offset of a stimulus is concerned." In our analysis, we focus on the evaluation of predictability in aiding a subject of determining stable or more uncertain situations. The control the agent has is reflected on the action chosen in a given state that, according to the specific scenario, may be influenced by the value of the prediction feature.

In our framework, this emerged feature indicates the level of *certainty* within the agent's transition model. The higher the level of $p_{zaz'}$ for some state z', the transition from state z by using action a will be more certain and thus the greater the agent's coping potential will be. Therefore, this feature will disfavor the execution of actions that do not provide guarantees in terms of the future state of environment, favoring behaviors leading to more certain and expected situations. If taken

positively, prediction is useful for achieving a faster learning in environments where more uncertain situations are detrimental when compared to more controllable ones. On the contrary, in scenarios where fitness enhancement may come from states and actions changing very often, *i.e.*, in transitions in which $p_{zaz'}$ is low, a "negative prediction" value might be advantageous, as occurred in the seasons scenario (see Table 2), where the reward provided by the preys varied at certain time intervals.

5.1.5 Frequency

As the expression $\phi_{\texttt{frq}} = -{n_z}^2$ indicates, this source of information punishes visits to states to which the agent is more accustomed to, somehow favoring states that have been visited less often. As described earlier, the major dimension of novelty is responsible for assessing the familiarity of perceived stimuli, motivating behavior towards the search for potentially significant situations [9]. Similarly, the information provided by the frequency signal mainly punishes visits to states regularly encountered. As such, if associated with a positive weight, it is a feature that fosters exploratory behaviors, necessary to deal with always changing, unpredictable environments, such as the exploration scenario in Experiment I (see Table 2), or situations where the agent has to continuously traverse its environment to achieve optimal performance, as was the case of the rewarding-pellets scenario, in which it was advantageous for the agent to eat *all* the pellets in the environment as opposed to eating only one small pellet or even the power-pellet (see Table 3). On the contrary, a negative frequency weight can be useful in scenarios where well-known states and actions are better and thus where familiar behavior "routines" are preferable, as occurred in the persistence scenario, where action N in position $(1:2)$ lead to a better outcome despite the fact that it becomes more and more difficult to cross the fence throughout time.

5.2 Other Dimensions

As can be seen from the description in the previous section of the major appraisal dimensions proposed in [9], not all appraisal dimensions (or variables) are covered by the sources of information emerged in the first experiment in Section3. Although our main objective was to inspect the emotional properties of the emerged signals, we believe there are two main reasons behind the absence of some of the appraisal variables commonly proposed in the emotions literature in our results.

The main reason has, perhaps, to do with the particular characteristics of the environments used to search for the optimal sources of information in Experiment I, and not the primitive variables in set F provided to the evolutionary genetic algorithm. On one hand, the dynamics of the environments promoted the appearance of strategies that favored some (combinations of) specific sources of information in order to tackle with the challenges offered by each environment. On the other hand, different environments would possibly favor different (combinations of) sources of information. Nevertheless, the results of both experiments demonstrate the general-purpose and domain-independent character of the emerged signals, an attribute commonly associated with emotions in nature, where such mechanism seems to be necessary for the adaptation of organisms to ever-changing and sometimes unpredictable habitats.

Another reason for the non-emergence of some of the appraisal variables proposed in the literature has to do with the characteristics of the agents themselves, which makes that the sources of information they have access to are of a very statistical nature, which in terms of appraisal corresponds to being made at a rather low-level, *i.e.*, requiring little cognitive processing [9, 18]. For example, evaluations such as the causal agency within the power and coping group determine the responsibility or agency for the occurrence of some event [9, 32], which is difficult to assess in our experiments, where each new state is the consequence of the agent's actions and the environment's dynamics which are dependent, *e.g.*, upon time. Also, we assume that the agent is always capable of performing a given action in some state according to its decision-making. Because of that, an evaluation of the agent's power does not make much sense in our framework, whereas the notion of control, depending as we have seen upon an evaluation of the predictability of action execution, can be easily assessed by the kinds of learning agents we model. We also note that the appraisal dimensions related to the motivational bases of a situation usually involve a much more complex analysis of a situation than our emerged feature of relevance does, *e.g.*, determining the pertinence of the event, the several motives that are affected, the consistency with the current motivational state, etc. [9]. However, the sole goal ascribed to our agents is to attain as much fitness as possible, and therefore the relevance signal only relates to this aspect of its motivational bases. Finally, the absence of the so-called social dimensions is easy to explain, as we test the agents in single-agent scenarios. Appraisals like evaluating the compatibility of some behavior against socially-defined norms or moral, individual values would require the interaction of several agents within the same environment and a notion of expected social norms or values to be accessible to the agent, for example through its reward function.[17]

5.3 Emotional Tone of the Learning Framework

An important aspect of our analysis has to do with the level at which appraisal takes place and the kinds of emotional states can we derive with our learning framework. In that respect we follow the perspective that appraisal occurs at several levels [9, 16, 18]–for example, the more basic fight-or-flight kind of evaluation observed in humans and mostly in other animals when facing a dangerous situation is different from the more cognitive assessments that we make when dealing with a complex task like building a shelter. Also, many theorists distinguish between *primary appraisal*, providing a rather crude evaluation but a fast, almost automatic response to events, and *secondary appraisal*, allowing a more deep and cognitive analysis of the situation and leading to more complex patterns of response throughout time by means of associative learning processes [6, 7, 16, 26]. We also distinguish between motivations directed at reducing some internal biological drive, and emotions, that operate over motivations, focus attention on important aspects of the environment and influence memory and learning to achieve a greater adaptation therein [6, 27].

[17] We refer to [34] for a work in which the reward functions included information about whether the agent was being considerate about other agents of the same population in the context of resource sharing scenarios.

Therefore, our evaluative mechanism, by means of the emerged reward features, is based on low-level information signals relating the agent's history of interaction with its environment, as occurs with appraisal in nature. We can also assess the rewarding mechanism of our agents as lying between a primary and secondary appraisal system: on one hand, it provides a fast evaluation of the perceived stimuli (the state) and provides responses (the actions) based on a single signal—the learned Q-value function; on the other hand, as time progresses during the agent's lifetime, the resulting rewards will reflect more what was learned in the previous interactions thus providing a more accurate evaluation over the environment. Our framework can therefore emerge both primary and secondary emotions in the sense of [7], discounting the social emotions as discussed earlier.[18] Our mechanism also departs from a simple drive-like motivational system in that it operates over the subject-environment representations, a central tenet for appraisal theory [9, 16, 28].

5.4 Related Work

In this section we analyze our resulting emotional-toned framework in light of other works that explicitly leveraged emotions to improve the adaptive capacity of autonomous learning agents. We also compare the signals emerged by means of GP with other reward features proposed within (IM)RL to further assess the computational usefulness of our approach.

In [36], the authors propose a computational model of emotional appraisal in a multiagent framework using POMDP agents, where five key appraisal dimensions are derived to aid decision-making in a lookahead process that calculates the next action depending on the agent's predicted states. Although the authors do not use the dimensions' values as rewards, this system, like ours, shows how appraisal can be derived from the agent's relationship with the environment and how can it be tightly integrated with the decision-making mechanism.

Within emotion-based RL, the work in [11] proposes a bottom-up approach to emotion elicitation by using artificial neural networks to determine a dominant *emotional state* from a set of four basic emotions, namely *happiness*, *sadness*, *fear* and *anger*. A traditional RL mechanism is used to reinforce state-behavior associations, where the rewards are calculated by the intensity of the current dominant emotion. In [30], three basic emotions control the behavior of an agent in an RL task, *happiness*, *sadness* and *fear*, where the reward is calculated according to a temporal difference of a measure of the agent's *well-being*. The work in [21] proposes an intrinsic reward signal based on the appraisal of *conduciveness*, which valence (positive/negative) determines the sign of the reward value, while its magnitude is determined by the intensity of the agent's current *feeling*. All these systems derive an emotional state to influence reward and guide the agent's behavior towards more "beneficial" situations. As with our framework, such rewards show to be useful in a

[18] We don't make use of emotion labels like "angry", "sad" or "happy" to describe the emotional state of our agents as we focus on the power of the emerged features in evaluating the agents' current state of affairs, as occurs with appraisal in nature. Nevertheless, we could devise a mechanism that would correspond the vector of reward features at each time step, $\boldsymbol{\phi} = [\phi_{\mathtt{frq}}, \phi_{\mathtt{rel}}, \phi_{\mathtt{prd}}, \phi_{\mathtt{adv}}, \phi_{\mathtt{fit}}]$, to a point in an 5-dimensional emotional state space, for which a specific label would be associated, as suggested by many appraisal theorists [9, 28].

variety of different scenarios. However, unlike our framework, they rely either on a set of discrete emotions or positive/negative evaluations of the emotional state of the agent. In our framework the emerged signals correspond to domain-independent features which, together, create a multidimensional emotional experience space capable of generating a multitude of distinct emotional states.

There are yet other approaches that can be related to our own in that they also propose some emotion-based reward features derived from the agent's history of interaction with its environment. The work in [1] proposes a model for *affective anticipatory (intrinsic) reward* based on *valence* and *arousal* levels. Interestingly, valence is calculated according to the expected reward for some action in relation to the average reward, which is very similar to our emerged feature of advantage, denoting the usefulness of executing some action in a given state. The arousal dimension is calculated according to an uncertainty model, which in our framework is provided by the prediction feature asserting the likelihood of state-action transitions. As another example, the idea behind the approach in [5] is that associating *positive* affective states with *exploitation* and *negative* affect with *exploration* strategies provides adaptive benefits in some RL scenarios. The reward with which the agent learns and its affective state is calculated by measuring the window-limited short-term running average of the (fitness-based) reinforcement against its long-term running average, which is similar to our feature of advantage.

A fundamental difference between the abovementioned works and our approach is that we did not modify the learning architecture to include emotion-related information—the intrinsic rewards provided to the agent were emerged through a GP procedure and are the only source of emotional information. Also, in our approach, instead of using predefined rules relating particular emotion states and action strategies (exploration vs. exploitation), emotions influence the choice of actions indirectly, *i.e.*, actions are chosen so as to maximize the "emotional benefit" of the current situation, as ascribed by the appraisal-related reward mechanism.

Within the realm of IMRL, our emerged signals bare also some dynamical and structural similarities with already proposed reward features. For example, our frequency feature is similar in effect with *recency-based* features [39, 40, 41], rewarding interactions with state-action pairs (not) visited recently thus encouraging dynamic behaviors, although the frequency feature seems more useful to efficiently exploring the environment in that it relies on the number of visits to some state. The emerged expression resembles more the *inverse-frequency* reward feature in [4]. Our prediction feature is also in line with reward features accounting for discrepancies in the state transition model perceived throughout time, *e.g.*, as proposed by the *quality of model* feature in [40]. The *variance* reward in [12] follows a similar approach by measuring the variance in the predictions of decision trees used to model the dynamics of the learning domain. In [20], a measure for *model accuracy* and *learning progress* related to prediction is presented where the likelihood of recently-visited state-action data-sets are evaluated in order to determine whether particular state-actions need further exploration or not. Although similar in spirit, the rewards used in [20, 40**?**] were used in a factored state context, and as such our emerged features deal with different kinds of information.

6 Conclusions and Future Work

In this paper we addressed the question relating the impact of emotion-based signals for the performance of autonomous agents by using a novel approach that is inspired by the way emotions have developed in nature throughout evolution as a mechanism that allows individuals of better adapting to their environment. We used an evolutionary computation algorithm guided by a measure of the agent's fitness to its environment within IMRL to emerge a set of basic reward signals allowing optimal performances in a set of foraging scenarios of interest. We verified the generality and applicability of these signals by testing their use in a set of scenarios different from those in which they were evolved. In order to assess if these emerged signals have an emotional tone, we analyzed them from the perspective of appraisal theories of emotions. Indeed, we found that the kinds of evaluations they make about the agent's relationship with its environment in fact shared some properties with common dimensions of emotional appraisal that, according to appraisal theorists, are used by individuals in nature to evaluate the impact of some situation for their goals and needs and to respond accordingly.

Some conclusions stem from our experimental study. First, the results from the experiments show that the reward features emerging from the GP optimization procedure exhibit dynamics and properties that can be related to the way natural agents evaluate their environment, according to appraisal theories of emotions. The emerged features result from reward functions that provided optimal performance as measured by a fitness measure that nothing has to do with emotions. Moreover, these features, much like emotions in nature, also proved to be useful in different environments, providing a general-purpose reward mechanism for artificial learning agents.

In this paper we contribute for research within IMRL by emerging four domain-independent reward features that could be applied in different scenarios with distinct purposes. This enables not only the construction of more robust, autonomous and adaptive agents, but also reduces the need for agent designers within RL to hand-code reward functions for a specific scenario. We also used a novel method for assessing the significance of embedding emotions into artificial agents that, unlike previous approaches, uses an evolutionary computation mechanism to search for optimal sources of information that can be compared to the way humans and other animals relate to their environment according to appraisal theories of emotions. All the findings resulting from our study thus point towards the idea that emotions might have a greater impact for the adaptation of artificial agents to their environments than thought before.

References

[1] H. Ahn and R. Picard. Affective cognitive learning and decision making: The role of emotions. In *Proc. 18th Eur. Meeting on Cybernetics and Systems Research*, pages 1–6, 2006.

[2] L. Baird. Advantage updating. Technical Report WL-TR-93-1146, Wright Laboratory, Wright-Patterson Air Force Base, 1993.

[3] A. Bechara, H. Damasio, and A. Damasio. Emotion, decision making and the orbitofrontal cortex. *Cerebral Cortex*, 10(3):295–307, 2000.

[4] J. Bratman, S. Singh, R. Lewis, and J. Sorg. Strong mitigation: Nesting search for good policies within search for good reward. In *Proc. 11th Int. Joint Conf. Autonomous Agents and Multiagent Systems*, 2012.

[5] D. Broekens, W. Kosters, and F. Verbeek. On affect and self-adaptation: Potential benefits of valence-controlled action-selection. In *Bio-inspired Modeling of Cognitive Tasks: Proc. 2nd Int. Conf. Interplay Between Natural and Artificial Computation*, pages 357–366, 2007.

[6] R. Cardinal, J. Parkinson, J. Hall, and B. Everitt. Emotion and motivation: The role of the amygdala, ventral striatum, and prefrontal cortex. *Neuroscience and Biobehavioral Reviews*, 26(3):321–352, 2002.

[7] A. Damasio. *Descartes' Error: Emotion, Reason, and the Human Brain*. Putnam Publishing, 1994.

[8] M. Dawkins. Animal minds and animal emotions. *American Zoologist*, 40(6): 883–888, 2000.

[9] P. Ellsworth and K. Scherer. Appraisal processes in emotion. In R. Davidson, K. Scherer, and H. Goldsmith, editors, *Handbook of the Affective Sciences*. Oxford University Press, 2003.

[10] N. Frijda and B. Mesquita. The analysis of emotions: Dimensions of variation. In M. Mascolo and S. Griffin, editors, *What Develops in Emotional Development? (Emotions, Personality, and Psychotherapy)*. Springer, 1998.

[11] S. Gadanho and J. Hallam. Robot learning driven by emotions. *Adaptive Behavior*, 9(1):42–64, 2001.

[12] Todd Hester and Peter Stone. Intrinsically motivated model learning for a developing curious agent. In *Proc. IEEE Int. Conf. on Development and Learning and Epigenetic Robotics*, pages 1–6. ICDL 2013, 2012.

[13] L. Kaelbling, M. Littman, and A. Moore. Reinforcement learning: A survey. *J. Artificial Intelligence Res.*, 4:237–285, 1996.

[14] L. Kaelbling, M. Littman, and A. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101:99–134, 1998.

[15] J. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.

[16] R. Lazarus. Relational meaning and discrete emotions. In K. Scherer, A. Schorr, and T. Johnstone, editors, *Appraisal Processes in Emotion: Theory, Methods, Research*. Oxford University Press, 2001.

[17] J. LeDoux. Emotion circuits in the brain. *Annual Review of Neuroscience*, 23 (1):155–184, 2000.

[18] H. Leventhal and K. Scherer. The relationship of emotion to cognition: A functional approach to a semantic controversy. *Cognition & Emotion*, 1(1): 3–28, 1987.

[19] M. Littman. Memoryless policies: Theoretical limitations and practical results. In *Proc. 3rd Int. Conf. Simulation of Adaptive Behavior - From Animals to Animats 3*, 1994.

[20] Manuel Lopes, Tobias Lang, Marc Toussaint, and Pierre-Yves Oudeyer. Exploration in model-based reinforcement learning by empirically estimating learning progress. In *Advances in Neural Information Processing Systems 25*, pages 206–214. 2012.

[21] R. Marinier. *A computational unification of cognitive control, emotion, and learning*. Phd thesis, University of Michigan, Ann Arbor, MI, 2008.

[22] S. Marsella and J. Gratch. EMA: A process model of appraisal dynamics. *Cognitive Systems Research*, 10(1):70–90, 2009.

[23] S. Marsella, J. Gratch, and P. Petta. Computational models of emotion. In K. Scherer, T. Bänziger, and E. Roesch, editors, *Blueprint for Affective Computing*. Oxford University Press, 2010.

[24] A. Moore and C. Atkeson. Prioritized sweeping: Reinforcement learning with less data and less real time. *Machine Learning*, 13:103–130, 1993.

[25] Scott Niekum, Andrew G. Barto, and Lee Spector. Genetic programming for reward function search. *IEEE Trans. Autonomous Mental Development*, 2(2): 83–90, 2010.

[26] K. Oatley and J. Jenkins. *Understanding Emotions*. Wiley-Blackwell, 2006.

[27] E. Phelps and J. LeDoux. Contributions of the amygdala to emotion processing: From animal models to human behavior. *Neuron*, 48(2):175–187, 2005.

[28] I. Roseman and C. Smith. Appraisal theory: Overview, assumptions, varieties, controversies. In K. Scherer, A. Schorr, and T. Johnstone, editors, *Appraisal Processes in Emotion: Theory, Methods, Research*. Oxford University Press, 2001.

[29] T. Rumbell, J. Barnden, S. Denham, and T. Wennekers. Emotions in autonomous agents: Comparative analysis of mechanisms and functions. *Autonomous Agents and Multiagent Systems*, 25(1):1–45, 2011.

[30] M. Salichs and M. Malfaz. Using emotions on autonomous agents: The role of happiness, sadness and fear. In *Proc. Annual Conf. on Ambient Intelligence and Simulated Behavior*, pages 157–164, 2006.

[31] M. Salichs and M. Malfaz. A new approach to modeling emotions and their use on a decision-making system for artificial agents. *IEEE Trans. Affective Computing*, 3(1):56–68, 2012.

[32] K. Scherer. Appraisal considered as a process of multilevel sequential checking. In K. Scherer, A. Schorr, and T. Johnstone, editors, *Appraisal Processes in Emotion: Theory, Methods, Research*. Oxford University Press, 2001.

[33] P. Sequeira, F.S. Melo, and A. Paiva. Emotion-based intrinsic motivation for reinforcement learning agents. In *Proc. 4th Int. Conf. Affective Computing and Intelligent Interaction*, pages 326–336, 2011.

[34] P. Sequeira, F.S. Melo, R. Prada, and A. Paiva. Emerging social awareness: Exploring intrinsic motivation in multiagent learning. In *Proc. 1st IEEE Int. Joint Conf. Development and Learning and Epigenetic Robotics*, volume 2, pages 1–6, 2011.

[35] P. Sequeira, F.S. Melo, and A. Paiva. Learning by appraising: An emotion-based approach for intrinsic reward design. Technical Report GAIPS-TR-001-12, INESC-ID, 2012.

[36] M. Si, S. Marsella, and D. Pynadath. Modeling appraisal in theory of mind reasoning. *Autonomous Agents and Multi-Agent Systems*, 20(1):14–31, 2010.

[37] S. Singh, T. Jaakkola, and M. Jordan. Learning without state estimation in partially observable Markovian decision processes. In *Proc. 11th Int. Conf. Machine Learning*, pages 284–292, 1994.

[38] S. Singh, R. Lewis, and A. Barto. Where do rewards come from? In *Proc. Annual Conf. Cognitive Science Society*, pages 2601–2606, 2009.

[39] S. Singh, R. Lewis, A. Barto, and J. Sorg. Intrinsically motivated reinforcement learning: An evolutionary perspective. *IEEE Trans. Autonomous Mental Development*, 2(2):70–82, 2010.

[40] J. Sorg, S. Singh, and R. Lewis. Internal rewards mitigate agent boundedness. In *Proc. 27th Int. Conf. Machine Learning*, pages 1007–1014, 2010.

[41] J. Sorg, S. Singh, and R. Lewis. Reward design via online gradient ascent. *Adv. Neural Information Proc. Systems*, 23:1–9, 2010.

[42] R. Sutton and A. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, 1998.

[43] G. Syswerda. Uniform crossover in genetic algorithms. In *Proc. 3rd Int. Conf. Genetic Algorithms*, pages 2–9, San Francisco, CA, USA, 1989.

Acknowledgements

This work was partially supported by the Portuguese Fundação para a Ciência e a Tecnologia (FCT) under project PEst-OE/EEI/LA0021/2013 and the EU project SEMIRA through the grant ERA-Compl/0002/2009. P. Sequeira acknowledges grant SFRH/BD/38681/2007 from FCT.

GAIPS Technical Report Series

This report is part of the GAIPS Technical Report Series. The reports currently available are:

[a] P. Sequeira, F.S. Melo, A. Paiva. Learning by appraising: An emotion-based approach for intrinsic reward design Tech. Rep. GAIPS-TR-001-12, GAIPS/INESC-ID, March 2012.

[b] P. Sequeira, F.S. Melo, A. Paiva. Associative metric for learning in factored MDPs based on classical conditioning. Tech. Rep. GAIPS-TR-002-12, GAIPS/INESC-ID, June 2012.

[c] S. Mascarenhas, R. Prada, A. Paiva. Social importance dynamics: A model for culturally-adaptive agents. Tech. Rep. GAIPS-TR-001-13, GAIPS/INESC-ID, April 2013.

[d] S. Mascarenhas, R. Prada, A. Paiva. Planning culturally-appropriate conversations using adjacency pairs. Tech. Rep. GAIPS-TR-002-13, GAIPS/INESC-ID, April 2013.

[e] P. Sequeira, F.S. Melo, A. Paiva. Emergence of emotion-like signals in learning agents. Tech. Rep. GAIPS-TR-003-13, GAIPS/INESC-ID, October 2013.

You may obtain any of the GAIPS technical reports from the corresponding author or on the group's web page (`http://gaips.inesc-id.pt/`).

INTELLIGENT AGENTS AND SYNTHETIC CHARACTERS GROUP