Intrinsically Motivated Reinforcement Learning: An Evolutionary Perspective

Satinder Singh, Richard L. Lewis, Andrew G. Barto, *Fellow, IEEE*, and Jonathan Sorg

***Abstract*—There is great interest in building intrinsic motivation into artificial systems using the reinforcement learning framework. Yet, what intrinsic motivation may mean computationally, and how it may differ from extrinsic motivation, remains a murky and controversial subject. In this article, we adopt an evolutionary perspective and define a new optimal reward framework that captures the pressure to design good primary reward functions that lead to evolutionary success across environments. The results of two computational experiments show that optimal primary reward signals may yield both emergent intrinsic and extrinsic motivation. The evolutionary perspective and the associated optimal reward framework thus lead to the conclusion that there are no hard and fast features distinguishing intrinsic and extrinsic reward computationally. Rather, the directness of the relationship between rewarding behavior and evolutionary success varies along a continuum.**

***Index Terms*—intrinsic motivation, reinforcement learning**

I. Introduction

The term "intrinsically motivated" first appeared (according to Deci and Ryan [9]) in a 1950 paper by Harlow [12] on the manipulation behavior of rhesus monkeys. Harlow argued that an intrinsic *manipulation drive* is needed to explain why monkeys will energetically and persistently work for hours to solve complicated mechanical puzzles without any extrinsic rewards. Intrinsic motivation plays a wide role in human development and learning, and researchers in many areas of cognitive science have emphasized that intrinsically motivated behavior is vital for intellectual growth.

This article addresses the question of how processes analogous to intrinsic motivation can be implemented in artificial systems, with specific attention to the factors that may or may not distinguish intrinsic motivation from extrinsic motivation, where the latter refers to motivation generated by specific rewarding consequences of behavior, rather than by the behavior itself.

There is a substantial history of research directed toward creating artificial systems that employ processes analogous to intrinsic motivation. Lenat's AM system [18], for example, focused on heuristic definitions of "interestingness," and Schmidhuber [32]–[37] introduced methods for implementing forms of curiosity using the framework of computational reinforcement learning (RL)[1] [47]. More recently, research in this tradition has expanded, with contributions based on a variety of more-or-less formal conceptions of how intrinsic motivation might be rendered in computational terms. Reviews of much of this literature are provided by Oudeyer and Kaplan [25], [26] and Merrick and Maher [22].

Despite this recent attention, what intrinsic motivation may mean computationally, and how it may differ from extrinsic motivation, remains a murky and controversial subject. Singh et al. [41] introduced an evolutionary framework for addressing these questions, along with the results of computational experiments that help to clarify some of these issues. They formulated a notion of an *optimal reward function* given a *fitness function*, where the latter is analogous to what in nature represents the degree of an animal's reproductive success. The present article describes this framework and some of those experimental results, while discussing more fully the notions of extrinsic and intrinsic rewards and presenting other experimental results that involve model-based learning and non-Markovian environments. In addition to emphasizing the generality of the approach, these results illuminate some additional issues surrounding the intrinsic/extrinsic reward dichotomy. In our opinion, the evolutionary perspective we adopt resolves what have been some of the most problematic issues surrounding the topic of intrinsic motivation, including the relationship of intrinsic and extrinsic motivation to primary and secondary reward signals, and the ultimate source of both forms of motivation.

Other researchers have reported interesting results of computational experiments involving evolutionary search for RL reward functions [1], [8], [19], [31], [43], but they did not directly address the motivational issues on which we focus. Uchibe and Doya [51] do address intrinsic reward in an evolutionary context, but their aim and approach differ significantly from ours. Following their earlier work [50], these authors treat extrinsic rewards as constraints on learning, while intrinsic rewards set the learning objective. This concept of the relationship between extrinsic and intrinsic rewards is technically interesting, but its relationship to the meanings of these terms in psychology is not clear. The study closest to ours is that of Elfwing et al. [11] in which a genetic algorithm is used to search for shaping rewards [23] and other learning algorithm parameters that improve an RL learning system's performance. We discuss how our approach is related to this

Satinder Singh is with the Division of Computer Science & Engineering, University of Michigan, Ann Arbor, email: baveja@umich.edu.

Richard Lewis is with the Department of Psychology, University of Michigan, Ann Arbor, email: rickl@umich.edu.

Andrew G. Barto is with the Department of Computer Science, University of Massachusetts, Amherst, email: barto@cs.umass.edu.

Jonathan Sorg is with the Division of Computer Science & Engineering, University of Michigan, Ann Arbor, email: jdsorg@umich.edu.

[1]We use the phrase *computational RL* because this framework is not a theory of biological RL despite what it borrows from, and suggests about, biological RL. However, in the following text we use just RL to refer to computational RL.

study and others in Section VII.

II. Computational Reinforcement Learning

Rewards—more specifically, reward functions—in RL determine the problem the learning agent is trying to solve. RL algorithms address the problem of how a behaving agent can learn to approximate an optimal behavioral strategy, called a *policy*, while interacting directly with its environment. Roughly speaking, an optimal policy is one that maximizes a measure of the total amount of reward the agent expects to accumulate over its lifetime, where reward is delivered to the agent over time via a scalar-valued signal.

In RL, rewards are thought of as the output of a "critic" that evaluates the RL agent's behavior. In the usual view of an RL agent interacting with its environment (left panel of Figure 1), rewards come from the agent's environment, where the critic resides. Some RL systems form *value functions* using, for example, Temporal Difference (TD) algorithms [45], to assign a value to each state that is an estimate of the amount of reward expected over the future after that state is visited. For some RL systems that use value functions, such as systems in the form of an "actor-critic architecture" [4], the phrase "adaptive critic" has been used to refer to the component that estimates values for evaluating on-going behavior. It is important not to confuse the adaptive critic with the critic in Figure 1. The former resides within the RL agent and is not shown in the figure.

The following correspondences to animal reward processes underly the RL framework. Rewards in an RL system correspond to *primary rewards*, i.e., rewards that for animals exert their effects through processes hard-wired by evolution due to their relevance to reproductive success. Value functions are the basis of *secondary* (or *conditoned* or *higher-order*) rewards, whereby learned predictions of reward act as reward themselves. The value function implemented by an adaptive critic therefore corresponds to a secondary, or learned, reward function. As we shall see, one should not equate this with an intrinsic reward function. The local landscape of a value function gives direction to an RL agent's preferred behavior: decisions are made to cause transitions to higher-valued states. A close parallel can be drawn between the gradient of a value function and *incentive salience* [20].

III. The Place of Intrinsic Motivation in Reinforcement Learning

How is intrinsic motivation currently thought to fit into the standard RL framework?[2] Barto et al. [3] used the term *intrinsic reward* to refer to rewards that produce analogs of intrinsic motivation in RL agents, and *extrinsic reward* to refer to rewards that define a specific task or rewarding outcome as in standard RL applications. Most of the current approaches to creating intrinsically motivated agents are based on defining special types of reward functions and then employing standard RL learning procedures, an approach first suggested by Schmidhuber [32] as a way to create an artificial analog of curiosity.

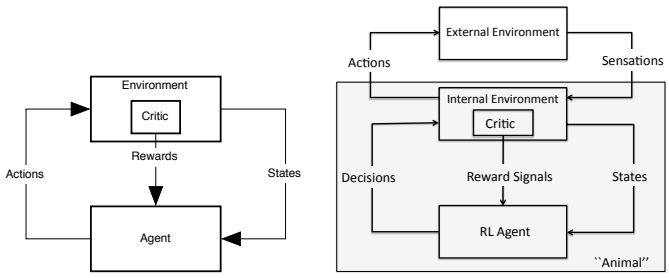

Fig. 1. Agent-environment interactions in reinforcement learning; adapted from [3]. Left panel: Primary reward is supplied to the agent from its environment. Right panel: A refinement in which the environment is factored into and internal and external environment, with all reward coming form the former. See text for further discussion

But let us step back and reconsider how intrinsic motivation and RL might be related. As Sutton and Barto [47] point out (also see [3], [40]), the standard view of the RL agent, and its associated terminology—as represented in the left panel of Figure 1—is seriously misleading if one wishes to relate this framework to animal reward systems and to the psychologist's notions of reward and motivation. First, psychologists distinguish between *rewards* and *reward signals*. For example, Schultz [38], [39] writes that "Rewards are objects or events that make us come back for more" whereas reward signals are produced by reward neurons in the brain. What in RL are called rewards would better be called reward signals. Rewards in RL are abstract signals whose source and meaning are irrelevant to RL theory and algorithms; they are not objects or events, though they can sometimes be the result of perceiving objects or events.

Second, the environment of an RL agent should not be identified with the external environment of an animal. A less misleading view requires dividing the environment into an *external environment* and an *internal environment*. In terms of animals, the internal environment consists of the systems that are internal to the animal while still being parts of the RL agent's environment. The right panel of Figure 1 refines the usual RL picture by showing the environment's two components and adjusting terminology by using the labels "RL Agent" and "Reward Signals." Further, we label the RL Agent's output "Decisions" instead of "Actions," reflecting the fact that actions that effect the external environment are generated by an animal's internal environment, for example, by its muscles, while the RL Agent makes decisions, such as the decision to move in a certain way. In this article, however, we retain the usual RL terms agent, reward, and action, but it is important not to interpret them incorrectly. Similarly, an "environment" in what follows should be understood to consist of internal and external components. Note that these refinements do not materially change the RL framework; they merely make it less abstract and less likely to encourage misunderstanding.

This refined view better reflects the fact that *the sources of*

[2]While we acknowledge the limitation of the RL approach in dealing with many aspects of motivation, this article nevertheless focuses on the sources and nature of reward functions for RL systems. We believe this focus allows us to clarify issues facing not only the computational community but other communities as well that are concerned with motivation in biological systems.

all of an animal's reward signals are internal to the animal. Therefore, the distinction between the internal and external environments is not useful for distinguishing between rewards that underlie intrinsically and extrinsically motivated behavior, a point also emphasized by Oudeyer and Kaplan [25]. It is clear that rewards underlying both intrinsically and extrinsically motivated behavior depend in essential ways on information originating in both the internal and external environments. For example, the motivational valence of the manipulation experiences of Harlow's monkeys was clearly derived, at least in part, from properties of the monkeys' external environments, and the motivational influence of extrinsic food reward depends on an animal's internal state of satiety.

If the distinction between internal and external environments is not useful for distinguishing intrinsic and extrinsic motivation, we are still left with the question: What does it mean *in the computational RL framework* to do something "for its own sake" or because "it is inherently interesting or enjoyable" [28]? One possibility, which has a long history in psychology, is that extrinsic and intrinsic motivation map onto primary and secondary reward signals, respectively. We consider this view next, before introducing our alternative evolutionary perspective.

IV. Do Extrinsic and Intrinsic Motivation Map Onto Primary and Secondary Reward?

Among the most influential theories of motivation in psychology is the drive theory of Hull [13]–[15]. According to Hull's theory, all behavior is motivated either by an organism's survival and reproductive needs giving rise to primary drives (such as hunger, thirst, sex, and the avoidance of pain) or by derivative drives that have acquired their motivational significance through learning. Primary drives are the result of physiological deficits—"tissue needs"—, and they energize behavior whose result is to reduce the deficit. A key additional feature of Hull's theory is that a need reduction, and hence a drive reduction, acts as a *primary reinforcer* for learning: behavior that reduces a primary drive is reinforced. Additionally, through the process of secondary reinforcement in which a neutral stimulus is paired with a primary reinforcer, the formerly neutral stimulus becomes a *secondary reinforcer*, i.e., acquires the reinforcing power of the primary reinforcer. In this way, stimuli that predict primary reward, i.e., predict a reduction in a primary drive, become rewarding themselves. According to this influential theory (in its several variants), all behavior is energized and directed by its relevance to primal drives, either directly or as the result of learning through secondary reinforcement.

Hull's theory followed the principles of physiological homeostasis that maintains bodily conditions in approximate equilibrium despite external perturbations. Homeostasis is achieved by processes that trigger compensatory reactions when the value of a critical physiological variable departs from the range required to keep the animal alive [6]. Many other theories of motivation also incorporate the idea that behavior is motivated to counteract disturbances to an equilibrium condition. These theories have been influential in the design of motivational systems for artificial agents, as discussed in Savage's review of artificial motivational systems [30]. Hull's idea that reward is generated by drive reduction is commonly used to connect RL to a motivational system. Often this mechanism consists of monitoring a collection of important variables, such as power or fuel level, temperature, etc., and triggering appropriate behavior when certain thresholds are reached. Drive reduction is directly translated into a reward signal delivered to an RL algorithm.

Among other motivational theories are those based on the everyday experience that we engage in activities because we enjoy doing them: we seek pleasurable experiences and avoid unpleasant ones. This is the ancient principle of hedonism. These theories of motivation hold that it is necessary to refer to affective mental states to explain behavior, such as a "feeling" of pleasantness or unpleasantness. Hedonic theories are supported by many observations about food preferences which suggest that "palatability" might offer a more parsimonious account of food preferences than tissue needs [55]. Animals will enthusiastically eat food that has no apparent positive influence on tissue needs; characteristics of food such as temperature and texture influence how much is eaten; animals that are not hungry still have preferences for different foods; animals have taste preferences from early infancy [7]. In addition, non-deprived animals will work enthusiastically for electrical brain stimulation [24]. Although it is clear that biologically-primal needs have motivational significance, facts such as these showed that factors other than primary biological needs exert strong motivational effects, and that these factors do not derive their motivational potency as a result of learning processes involving secondary reinforcement.

In addition to observations about animal food preferences and responses to electrical brain stimulation, other observations showed that something important was missing from drive-reduction theories of motivation. Under certain conditions, for example, hungry rats would rather explore unfamiliar spaces than eat; they will endure the pain of crossing electrified grids to explore novel spaces; monkeys will bar-press for a chance to look out of a window. Moreover, the opportunity to explore can be used to reinforce other behavior. Deci and Ryan [9] chronicle these and a collection of similar findings under the heading of *intrinsic motivation*.

Why did most psychologists reject the view that exploration, manipulation, and other curiosity-related behaviors derived their motivational potency only through secondary reinforcement, as would be required by a theory like Hull's? There are clear experimental results showing that such behavior is motivationally energizing and rewarding on its own and not because it predicts the satisfaction of a primary biological need. Children spontaneously explore very soon after birth, so there is little opportunity for them to experience the extensive pairing of this behavior with the reduction of a biologically primary drive that would be required to account for their zeal for exploratory behavior. In addition, experimental results show that the opportunity to explore retains its energizing effect without needing to be re-paired with a primary reinforcer, whereas a secondary reinforcer will extinguish, that is, will lose its reinforcing quality, unless often re-paired with

the primary reinforcer it predicts. Berlyne summarized the situation as follows:

> As knowledge accumulated about the conditions that govern exploratory behavior and about how quickly it appears after birth, it seemed less and less likely that this behavior could be a derivative of hunger, thirst, sexual appetite, pain, fear of pain, and the like, or that stimuli sought through exploration are welcomed because they have previously accompanied satisfaction of these drives. (p. 26, Berlyne [5])

Note that the issue was not whether exploration, manipulation, and other curiosity-related behaviors are important for an animal's survival and reproductive success. Clearly they are if deployed in the right way. Appropriately cautious exploration, for example, clearly has survival value because it can enable efficient foraging and successful escape when those needs arise. The issue was whether an animal is motivated to perform these behaviors because previously *in its own lifetime* behaving this way predicted decreases in biologically-primary drives, or whether this motivation is built-in by the evolutionary process. The preponderance of evidence supports the view that the motivational forces driving these behaviors are built-in by the evolutionary process.

V. Evolutionary Perspective

It is therefore natural to investigate what an evolutionary perspective might tell us about the nature of intrinsic reward signals and how they might differ from extrinsic reward signals. We adopt the view discussed above that intrinsic reward is not the same as secondary reward. It is likely that the evolutionary process gave exploration, play, discovery, etc., positive hedonic valence because these behaviors contributed to reproductive success throughout evolution. Consequently, we regard intrinsic rewards in the RL framework as primary rewards, hard-wired from the start of the agent's life. Like any other primary reward in RL, they come to be predicted by the value-function learning system. These predictions can support secondary reinforcement so that predictors of intrinsically rewarding events can acquire rewarding qualities through learning just as predictors of extrinsically rewarding events can.

The evolutionary perspective thus leads to an approach in which adaptive agents, and *therefore their reward functions*, are evaluated according to their *expected fitness* given an explicit fitness function and some distribution of environments of interest. The fitness function maps trajectories of agent-environment interactions to scalar fitness values, and may take any form (including functions that are similar in form to discounted sums of extrinsic rewards). In our approach, we search a space of primary reward functions for the one that maximizes the expected fitness of an RL agent that learns using that reward function. Features of such an *optimal reward function*[3] and how these features relate to the environments in which agent lifetimes are evaluated provide insight into the relationship between extrinsic and intrinsic rewards (as discussed in Section VI and thereafter).

We turn next to a formal framework that captures the requisite abstract properties of agents, environments, and fitness functions and defines the evolutionary search for good reward functions as an optimization problem.

A. Optimal Reward Functions

As shown in the right panel of Figure 1, an agent $\mathcal{A}$ in some (external) environment E receives an observation and takes an action at each time step. The agent has an internal environment that computes a state, a summary of history, at every time step (e.g., in Markovian environments the last observation is a perfect summary of history and thus state can be just the last observation). The agent's action is contingent on the state. The reward function can in general depend on the entire history of states or equivalently on the entire history of observations and actions. Agent $\mathcal{A}$'s goal or objective is to attempt to maximize the cumulative reward it receives over its lifetime. In general, defining agent $\mathcal{A}$ includes making very specific commitments to particular learning architectures, representations, and algorithms as well as all parameters. Our evolutionary framework abstracts away from these details to define a notion of optimal reward function as follows.

For every agent $\mathcal{A}$, there is a space of reward functions $R_{\mathcal{A}}$ that maps features of the history of observation-action pairs to scalar primary reward values (the specific choice of features is determined in defining $\mathcal{A}$). There is a distribution P over sequential decision making environments in some set $\mathcal{E}$ in which we want our agent to perform well. A specific reward function $r_{\mathcal{A}} \in R_{\mathcal{A}}$ and a sampled environment $E \sim P(\mathcal{E})$ produces h, the history of agent $\mathcal{A}$ adapting to environment E over its lifetime using the reward function $r_{\mathcal{A}}$, i.e., $h \sim \langle \mathcal{A}(r_{\mathcal{A}}), E \rangle$, where $\langle \mathcal{A}(r_{\mathcal{A}}), E \rangle$ makes explicit that agent $\mathcal{A}$ is using reward function $r_{\mathcal{A}}$ to interact with environment E and $h \sim \langle \cdot \rangle$ makes explicit that history h is sampled from the distribution produced by the interaction $\langle \cdot \rangle$. A given fitness function $\mathcal{F}$ produces a scalar evaluation $\mathcal{F}(h)$ for each such history h. An optimal reward function $r^*_{\mathcal{A}} \in R_{\mathcal{A}}$ is the reward function that maximizes the expected fitness over the distribution of environments, i.e.,

$$r^*_{\mathcal{A}} = \arg \max_{r_{\mathcal{A}} \in R_{\mathcal{A}}} \mathbb{E}_{E \sim P(\mathcal{E})} \mathbb{E}_{h \sim \langle A(r_{\mathcal{A}}), E \rangle} \{ \mathcal{F}(h) \}, \qquad (1)$$

where $\mathbb{E}$ denotes the expectation operator. A special reward function in $R_{\mathcal{A}}$ is the fitness-based reward function, denoted $r_{\mathcal{F}}$, that most directly translates fitness $\mathcal{F}$ into an RL reward function, i.e., the fitness value of a lifetime-length history is the cumulative fitness-based reward for that history. For example, if the fitness value of a history was the number of children

[3] We use this term despite the fact that none of our arguments depend on our search procedure finding true globally-optimal reward functions. We are concerned with reward functions that confer advantages over others and not with absolute optimality. Similarly, the fact that optimization is at the core of the RL framework does not imply that what an RL system learns is optimal. What matters is the process of improving, not the final result.

produced, then a corresponding fitness-based reward function could assign unit reward to the state resulting from the birth of a child and zero otherwise (additional concrete examples are in our experimental results reported below).

Our formulation of optimal rewards is very general because the constraints on $\mathcal{A}$, $R_{\mathcal{A}}$, $\mathcal{F}$, and $\mathcal{E}$ are minimal. Agent $\mathcal{A}$ is constrained only to be an agent that uses a reward function $r_{\mathcal{A}} \in R_{\mathcal{A}}$ to drive its search for good behavior policies. The space $R_{\mathcal{A}}$ is constrained to be representable by the internal architecture of agent $\mathcal{A}$ as well as to contain the fitness-based reward $r_{\mathcal{F}}$. Fitness $\mathcal{F}$ is constrained only to be a function that maps (lifetime-length) histories of agent-environment interactions to scalar fitness values. The space $\mathcal{E}$ is constrained only to be a (finite or infinite) set of discrete-time decision making environments (Markovian or non-Markovian[4], and indeed our empirical results will use both). Finally, the evolutionary or fitness pressure that defines optimal rewards is represented by an optimization or search problem (Equation 1) unconstrained by a commitment to any specific evolutionary process.[5]

Note an immediate consequence of Equation 1: in terms of the expected fitness achieved, the agent with the optimal reward function will by definition outperform (in general, and never do worse than) the same agent with the fitness-based reward function. Crucially, it is this possibility of outperforming the fitness-based reward in the amount of fitness achieved that produces the evolutionary pressure to reward not just actions that directly enhance fitness— what might be termed extrinsically motivated behavior—but actions that intermediate evolutionary success—what might be termed intrinsically motivated behaviors.

B. Regularities Within and Across Environments

The above formulation of Equation 1 defines a search problem—the search for $r^*_{\mathcal{A}}$. This search is for a primary reward function and is to be contrasted with the search problem faced by an agent during its lifetime, that of learning a good value function (and hence a good policy) specific to its environment leading to history $h \sim \langle \mathcal{A}(r_{\mathcal{A}}), E \rangle$ (cf. Equation 1). These two (nested) searches are at the heart of our evolutionary perspective on reward in this article. Specifically, our concrete hypotheses are (1) the optimal reward $r^*_{\mathcal{A}}$ derived from search will capture regularities across environments in $\mathcal{E}$ as well as complex interactions between $\mathcal{E}$ and specific structural properties of the agent $\mathcal{A}$ (note that the agent $\mathcal{A}$ is part of its environment and is constant across all environments in $\mathcal{E}$), and (2) the value functions learned by an agent during its lifetime will capture regularities present within its specific environment that are not necessarily shared across environments. It is the first hypothesis, that of the primary reward capturing regularities across environments and between environments and agents, that should lead to the emergence of both extrinsic and intrinsic rewards, the former from objects or other sources of primal needs present across environments and the latter from behaviors such as play and exploration that serve the agents well across environments in terms of expected fitness.

Next we describe experiments designed to test our hypotheses as well as to illustrate the emergence of both extrinsic and intrinsic rewards in agents through search for optimal reward functions.

VI. Computational Experiments

We now describe two sets of computational experiments in which we directly specify the agent $\mathcal{A}$ with associated space of reward functions $R_{\mathcal{A}}$, a fitness function $\mathcal{F}$, and a set of environments $\mathcal{E}$, and derive $\hat{r}^*_{\mathcal{A}}$ via (approximately) exhaustive search. These experiments are designed to serve three purposes. First, they will provide concrete and transparent illustrations of the basic optimal reward framework above. Second, they will demonstrate the *emergence* of interesting reward function properties that are not direct reflections of the fitness function—including features that might be intuitively recognizable as candidates for plausible intrinsic and extrinsic rewards in natural agents. Third, they will demonstrate the *emergence* of interesting reward functions that capture regularities across environments, and similarly demonstrate that value function learning by the agent captures regularities within single environments.

A. Experiment 1: Emergent Intrinsic Reward for Play and Manipulation

This first experiment was designed to illustrate how our optimal reward framework can lead to the emergence of an intrinsic reward for actions such as playing with and manipulating objects in the external environment, actions that do not directly meet any primal needs (i.e., are not fitness inducing) and thus are not extrinsically motivating.

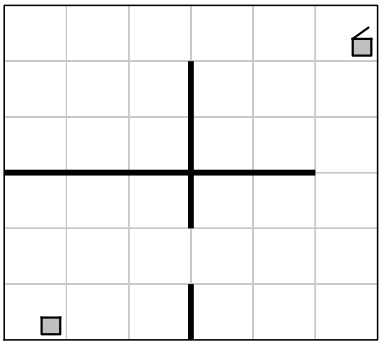

Fig. 2. Boxes environments used in Experiment 1. Each boxes environment is a 6×6 grid with two boxes that can contain food. The two boxes can be in any two of the four corners of the grid; the locations are chosen randomly for each environment. The agent has four (stochastic) movement actions in the four cardinal directions, as well actions to open closed boxes and eat food from the boxes when available. See text for further details.

(Boxes) Environments. We use a simulated physical space shown by the 6×6 grid in Figure 2. It consists of four

[4]Specifically, we allow both for Markov decision processes, or MDPs, as well as for partially observable MDPs, or POMDPs. See Sutton and Barto [47] and Kaelbling et.al. [16] for a discussion of the different mathematical formalisms of RL problems.

[5]However, in many cases the space of reward functions will have structure that can be exploited to gain computational efficiency, and many classes of optimization algorithms might prove useful in a practical methodology for creating reward functions for artificial agents.

subspaces (of size 3×3). There are four movement actions, `North`, `South`, `East` and `West`, that if successful move the agent probabilistically in the direction implied, and if they fail leave the agent in place. Actions fail if they would move the agent into an outer bound of the grid or across a barrier, which are represented by the thick black lines in the figure. Consequently, the agent has to navigate through gaps in the barriers to move to adjacent subspaces. In each sampled environment two boxes are placed in randomly chosen special locations (from among the four corners and held fixed throughout the lifetime of the agent). This makes a uniform distribution over a space of six environments (the six possible locations of two indistinguishable boxes in the four corners). In addition to the usual movement actions, the agent has two special actions: `open`, which opens a box if it is closed and the agent is at the location of the box and has no effect otherwise (when a closed box is opened it transitions first to a half-open state for one time step and then automatically to an open state at the next time step regardless of the action by the agent), and `eat`, which has no effect unless the agent is at a box location, the box at that location is half-open, and there happens to be food (prey) in that box, in which case the agent consumes that food.

An open box closes with probability 0.1 at every time step.[6] A closed box always contains food. The prey always escapes when the box is open. Thus to consume food, the agent has to find a closed box, open it, and eat immediately in the next time step when the box is half-open. When the agent consumes food it feels *satiated* for one time step. The agent is *hungry* at all other time steps. The agent-environment interaction is not divided into trials or episodes. The agent's observation is 6 dimensional: the x and y coordinates of the agent's location, the agent's hunger-status, the open/half-open/closed status of both boxes, as well the presence/absence of food in the square where the agent is located. These environments are Markovian because the agent senses the status of both boxes regardless of location and because closed boxes always contain food; hence each immediate observation is a state.

Fitness. Each time the agent eats food its fitness is incremented by one. This is a surrogate for what in biology would be reproductive success (we could just as well have replaced the consumption of food event with a procreation event in our abstract problem description). The fitness objective, then, is to maximize the amount of food eaten over the agent's lifetime. Recall that when the agent eats it becomes satiated for one time step, and thus a direct translation of fitness into reward would assign a reward of $c > 0$ to all states in which the agent is satiated and a reward of $d < c$ to all other states. Thus, there is a space of fitness-based reward functions. We will refer to fitness-based reward functions in which d is constrained to be exactly 0 as *simple* fitness-based reward functions. Note that our definition of fitness is incremental or cumulative and thus we can talk about the cumulative fitness of even a partial (less than lifetime) history.

[6]A memoryless distribution for box-closing was chosen to keep the environment Markovian for the agent; otherwise, there would be information about the probability of a box closing from the history of observations based on the amount of time the box had been open.

Agent. Our agent ($\mathcal{A}$) uses the lookup-table ϵ-greedy Q-learning [52] algorithm with the following choices for its parameters: 1) Q_0, the initial Q-function (we use small values chosen uniformly randomly for each state-action pair from the range $[-0.001, 0.001]$) that maps state-action pairs to their expected discounted sum of future rewards, 2) α, the step-size, or learning-rate parameter, and 3) ϵ, the exploration parameter (at each time step the agent executes a random action with probability ϵ and the greedy action with respect to the current Q-function with probability $(1 - \epsilon)$).

For each time step t, the current state is denoted s_t, the current Q-function is denoted Q_t, the agent executes an action a_t, and the Q-learning update is as follows:
$Q_{t+1}(s_t, a_t) = (1-\alpha)Q_t(s_t, a_t)+\alpha[r_t+\gamma \max_b(Q_t(s_{t+1}, b)]$, where r_t is the reward specified by reward function $r_{\mathcal{A}}$ for the state s_t, and γ is a discount factor that makes immediate reward more valuable than later reward (we use $\gamma = 0.99$ throughout).

We emphasize that the discount factor is an agent parameter that does not enter into the fitness calculation. That is, the fitness measure of a history remains the total amount of food eaten in that history for any value of γ the agent uses in its learning algorithm. It is well known that the form of Q-learning used above will converge asymptotically to the optimal Q-function[7] and hence the optimal policy [53]. Thus, our agent uses its experience to continually adapt its action selection policy to improve the discounted sum of rewards, as specified by $r_{\mathcal{A}}$, that it will obtain over its future (remaining in its lifetime). Note that the reward function is distinct from the fitness function $\mathcal{F}$.

Space of Possible Rewards Functions. To make the search for an optimal reward function tractable, each reward function in the search space maps abstract features of each immediate observation to a scalar value. Specifically, we considered reward functions that ignore agent location and map each possible combination of the status of the two boxes and the agent's hunger-status to values chosen in the range $[-1.0, 1.0]$. This range does not unduly restrict generality because one can always add a constant to any reward function without changing optimal behavior. Including the box-status features allows the reward function to potentially encourage "playing with" boxes while the hunger-status feature is required to express the fitness-based reward functions that differentiate only between states in which the agent is satiated from all other states (disregarding box-status and agent location).

Finding a Good Reward Function. The psuedo-code below describes how we use simulation to estimate the *mean cumulative fitness* for a reward function $r_{\mathcal{A}}$ given a particular setting of agent (Q-learning) parameters (α, ϵ).

set (α, ϵ)
for $i = 1$ **to** N **do**
  Sample an environment E_i from $\mathcal{E}$
  In $\mathcal{A}$, intialize Q-function

[7]Strictly speaking, convergence with probability one requires the step-size parameter α to decrease appropriately over time, but for our purposes it suffices to keep it fixed at a small value.

Generate a history h_i over lifetime for $\mathcal{A}$ and E_i
Compute fitness $F(h_i)$
end for
return average of $\{F(h_1), \ldots, F(h_N)\}$

In the experiments we report below, we estimate the mean cumulative fitness of $r_{\mathcal{A}}$ as the maximum estimate obtained (using the pseudo-code above) over a coarse discretization of the space of feasible (α, ϵ) pairs. Finding good reward functions for a given fitness function thus amounts to a large search problem. We discretized the range $[-1.0, 1.0]$ for each feasible setting of the three reward features such that we evaluated $54,000$ reward functions in the reward function space. We chose the discretized values based on experimental experience with the boxes environments with various reward functions.

Note that our focus is on demonstrating the generality of our framework and the nature of the reward functions found rather than on developing efficient algorithms for finding good reward functions. Thus, we attempt to find a good reward function $\hat{r}^*_{\mathcal{A}}$ instead of attempting the usually intractable task of finding the optimal reward function $r^*_{\mathcal{A}}$, and we are not concerned with the efficiency of the search process.

Results. Recall the importance of regularities within and across environments to our hypotheses. In this experiment, what is unchanged across environments is the presence of two boxes and the rules governing food. What changes across environments—but held fixed within a single environment—are the locations of the boxes.

We ran this experiment under two conditions. In the first, called the *constant condition*, the food always appears in closed boxes throughout each agent's lifetime of $10,000$ steps. In the second, called the *step condition*, each agent's lifetime is $20,000$ steps, and food appears only in the *second half* of the agent's lifetime, i.e., there is never food in any of the boxes for the first half of the agent's lifetime, after which food always appears in a closed box. Thus in the step condition, it is impossible to increase fitness above zero until after the $10,000^{th}$ time step.

The step condition simulates (in extreme form) a developmental process in which the agent is allowed to "play" in its environment for a period of time in the absence of any fitness-inducing events (in this case, the fitness-inducing events are positive, but in general there could also be negative ones that risk physical harm). Thus, a reward function that confers advantage through exposure to this first phase must reward events that have only a distal relationship to fitness. Through the agent's learning processes, these rewards give rise to the agent's intrinsic motivation. Notice that this should happen in both the step and constant conditions; we simply expect it to be more striking in the step condition.

The left and middle panels of Figure 3 show the mean (over 200 sampled environments) cumulative fitness as a function of time within an agent's lifetime under the two conditions. As expected, in the step condition, fitness remains zero under any reward function for the first $10,000$ steps. Also as expected, the best reward function outperforms the best fitness-based reward function over the agent's lifetime. The best fitness-based reward function is the best reward function in the reward function space that satisfies the definition of a fitness-based reward function for this class of environments. We note that the best fitness-based reward function assigns a negative value to states in which the agent is hungry (this makes the agent's initial Q-values optimistic and leads to efficient exploration; see Sutton and Barto [47] for an explanation of this effect). The best reward function outperforms the best simple fitness-based reward by a large margin (presumably because the latter cannot make the initial Q-values optimistic).

Table I shows the best reward functions and best fitness-based reward functions for the two conditions of the experiment (e.g., the best reward function for the Step condition is as follows: being satiated has a positive reward of 0.5 when both boxes are open and 0.3 when one box is open, being hungry with one box half-open has a small negative reward of -0.01, and otherwise being hungry has a reward of -0.05. Note that the agent will spend most of its time in this last situation.) Of course, as expected and like the best fitness-based reward function, the best reward function has a high positive reward for states in which the agent is satiated. More interestingly, the best reward function in our reward function space rewards opening boxes (by making their half-open state rewarding relative to other states when the agent is hungry). This makes the agent "play" with the boxes and as a result learn the environment-specific policy to optimally navigate to the location of the boxes and then open them during the first half of the step condition so that when food appears in the second half, the agent is immediately ready to exploit that situation.

The policy learned under the best reward function has an interesting subtle aspect: it makes the agent run back and forth between the two boxes, eating from both boxes, because this leads to higher fitness (in most environments)[8] than staying at, and taking food from, only one box. This can be seen indirectly in the rightmost panel where the mean cumulative number of times both boxes are open is plotted as a function of time. It is clear that an agent learning with the overall best reward function keeps both boxes open far more often than one learning from the best fitness-based reward function. Indeed the behavior in the latter case is mainly to loiter near (an arbitrary) one of the boxes and repeatedly wait for it to close and then eat.

Finally, it is also noteworthy that there are other reward functions that keep both boxes open even more often than the best reward function (this is seen in the rightmost panel), but this occurs at the expense of the agent not taking the time to actually eat the food after opening a box. This suggests that there is a fine balance in the best reward function between intrinsically motivating "playing" with and manipulating the boxes and extrinsically motivating eating.

Summary. This experiment demonstrates that the evolutionary pressure to optimize fitness captured in the optimal reward

[8]The agent could hang out at one box and repeatedly wait for it to close randomly and then open it to eat, but the probability of an open box closing was specifically (experimentally) chosen so that it is better for the agent in the distribution over environments to repeatedly move between boxes to eat from both. Specifically, an open box closes with probability 0.1 and thus on average in 10 time steps, while the average number of time steps to optimally travel between boxes across the 6 environments is less than 10 time steps.

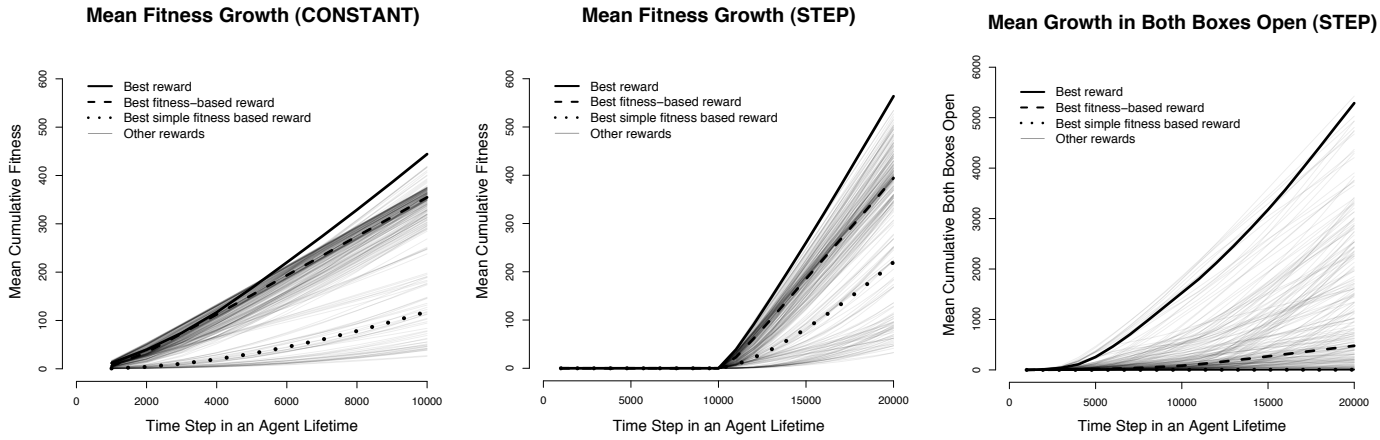

Fig. 3. Results from Boxes environments. The leftmost panel shows for the *constant* condition the mean cumulative (over agent lifetime) fitness achieved by all the reward functions sampled in our search for good reward functions. The middle panel shows the same results but for the *step* condition. The rightmost panel shows for the *step* condition the mean cumulative growth in the number of time steps both boxes were open for all the reward functions explored. In each panel, the curves for the best reward function, for the best fitness-based reward function, and for the best simple fitness-based reward functions are distingusihed. See text for further details.

TABLE I
RESULTS FOR THE *step* AND *constant* CONDITIONS OF EXPERIMENT 1. EACH ROW OF PARAMETER VALUES DEFINES A REWARD FUNCTION BY SPECIFYING REWARD VALUES FOR EACH OF SEVEN FEASIBLE COMBINATIONS OF STATE FEATURES. THE COLUMN HEADINGS O, NOT-O, AND HALF-O, ARE SHORT FOR OPEN, NOT-OPEN AND HALF-OPEN RESPECTIVELY. SEE TEXT FOR FURTHER DETAILS.

CONDITION	REWARD TYPE	REWARD AS A FUNCTION OF STATE						
		Satiated		Hungry				
		o/o	o/not-o	o/o	o/not-o	o/half-o	not-o/half-o	not-o/not-o
Constant	Best	0.7	0.3	-0.01	-0.05	0.2	0.1	-0.02
	Best fitness-based	0.7	0.7	-0.005	-0.005	-0.005	-0.005	-0.005
Step	Best	0.5	0.3	-0.05	-0.05	-0.01	-0.01	-0.05
	Best fitness-based	0.5	0.5	-0.01	-0.01	-0.01	-0.01	-0.01

framework can lead to the emergence of reward functions that assign positive *primary* reward to activities that are not directly associated with fitness. This was especially evident in the step condition of the Boxes experiment: during the first half of the agent's lifetime, no fitness-producing activities are possible, but intrinsically rewarding activities (running between boxes to keep both boxes open) are pursued that have fitness payoff later. The best (primary) reward captures the regularity of needing to open boxes to eat across all environments, while leaving the learning of the environment-specific navigation policy for the agent to accomplish within its lifetime by learning the (secondary reward) Q-value function.

B. Experiment 2: Emergent Intrinsic Reward Based on Internal Environment State

This second experiment was designed with two aims in mind. The first is to emphasize the generality of our optimal reward framework by using a model-based learning agent in non-Markovian environments instead of the model-free Q-learning agent in the Markovian environments of Experiment 1. The second is to demonstrate the emergence of optimal reward functions that are contingent on features of the internal environment (cf. Figure 1) of the agent rather than features of the external environment (as, for example, boxes and their status in Experiment 1).

(Foraging) Environments. We use the foraging environment illustrated in Figure 4. It consists of a 3×3 grid with three dead-end corridors (as rows) separated by impassable walls. The agent, represented by the bird, has four movement actions available in every location which deterministically move the agent in each of the cardinal directions. If the intended direction is blocked by a wall or the boundary, the action results in no movement. There is a food source, represented by the worm, randomly located in one of the three right-most locations at the end of each corridor. The agent has an `eat` action, which consumes the worm when the agent is at the worm's location. The agent is *hungry* except when it consumes a worm, which causes the agent to become *satiated* for one time step. Immediately, the consumed worm disappears and a new worm appears randomly in one of the other two potential worm locations. This creates a distribution over foraging environments based on random sequences of worm appearances.

The agent observations are four-dimensional: the agent's x and y coordinates and whether it is hungry (binary), and whether or not it is co-located with the worm (binary). The agent *cannot* see the worm unless it is co-located with it. In the environments of Experiment 1 the agent could also not see the food unless it was co-located with it, but the food locations were fixed throughout an agent's lifetime. Crucially, in the

foraging environments here, the location of every new worm within an agent's lifetime is chosen randomly. Thus, unlike the environments of Experiment 1, the foraging environments here are non-Markovian because the agent's past observations predict where the worm cannot be (specifically, the worm cannot be at any end-of-corridor location that the agent has visited since the last time it ate the worm), and this information is not available from just the current observation.

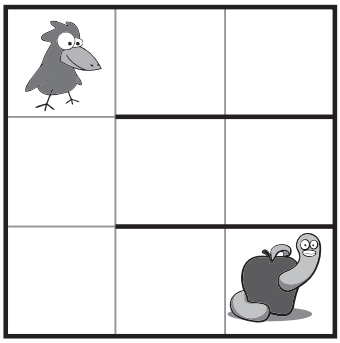

Fig. 4. Foraging environments used in Experiment 2. Each foraging environment is a 3×3 grid arranged in (row) corridors. The food represented by a worm appears at the rightmost end of a corridor. The agent represented by a bird has the usual movement actions in the four cardinal directions as well as an eat action when co-located with the worm. Crucially, once the agent eats a worm, a new worm appears at a random corridor-end location and the agent cannot see the worm unless co-located with it. These foraging environments are non-Markovian unlike the boxes environments of Experiment 1. See text for further details.

Fitness. Each time the agent eats a worm, its fitness is incremented by one. The fitness objective is to maximize the number of worms eaten over an agent lifetime of $10,000$ time steps. When the agent eats, it becomes satiated for one time step, and thus a direct translation of fitness into reward would assign a positive reward to all states in which the agent is satiated and a strictly lower reward to all other states. In Experiment 1, because of the interaction of the choice of reward values with the initial Q-value function, we needed to consider a space of possible fitness-based rewards. In this experiment the agent does complete estimated-model-based planning via dynamic programming at each time step and it is easily seen that all fitness-based rewards yield exactly the same policy, and thus we define $r_{\mathcal{F}}$ to map all satiated states to 1.0 and all other states to 0.0.

Agent. We used a standard model-based learning agent for this experiment. Specifically, the agent updates an estimated model of its environment after each time step and always acts greedily according to a (certainty equivalent) policy optimal with respect to its latest estimated model. The transition-dynamics of the environment are estimated assuming that the agent's observations (x and y coordinates, hunger-status, co-located-with-worm-status) are Markovian, i.e., assuming that these observations comprise a state.

Specifically, let $n_{s,a}$ be the number of times that action a was taken in state s. Let $n_{s,a,s'}$ be the number of times a transition to state s' occurred after action a was taken in state s. The agent models the probability of a transition to s' after taking a in state s as $\hat{T}(s'|s,a) = \frac{n_{s,a,s'}}{n_{s,a}}$.[9] The optimal policy with respect to the current model is computed at every time step via repeated Q-value iteration: for all (s,a),

$$Q_d(s,a) = r_{\mathcal{A}}(s,a) + \gamma \sum_{s' \in S} \hat{T}(s'|s,a) \max_{a'} Q_{d-1}(s',a'),$$

where $Q_0(s,a) \stackrel{\text{def}}{=} 0$, $\gamma = 0.99$ is the discount factor,[10] and iteration is performed until the maximal (across state-action pairs) absolute change in Q-values is less than a very small threshold. If, after convergence, the Q-values of multiple actions in a current state are equal, the agent selects randomly among those equal-valued actions.

Space of Reward Functions. We selected a reward function space consisting of linear combinations of the features of the state of the internal environment, i.e., of the history, h, of the observations and actions. This is another departure from Experiment 1, where we used a tabular representation of reward functions with features based solely on the immediate observations from the external environment.

Our choice of reward-features for this domain is driven by the following intuition. With a fitness-based reward function that only distinguishes satiated states from hungry states, even the policy found via infinite Q-value iteration on the estimated model cannot, from most locations, take the agent to the worm (and make it eat). This is because the agent cannot see the worm's location when it is not co-located with it. Indeed there is little guidance from a fitness-based reward unless the agent is co-located with the worm. Reward functions that encourage systematic exploration of the grid locations could be far more effective in expected fitness than the fitness-based reward function. In fact, unlike most applications of RL wherein exploration serves a transient purpose to be eliminated as soon as possible, here it is essential that the agent explore persistently throughout its lifetime.

What kind of reward function could generate systematic and persistent exploration? We consider the reward function space $r_{\mathcal{A}}(s,a) = \beta_{\mathcal{F}} \phi_{\mathcal{F}}(s) + \beta_c \phi_c(s,a,h)$, where $\beta_{\mathcal{F}}$ and β_c are parameters of a linear reward function, feature $\phi_{\mathcal{F}}(s)$ is 1 when the agent is satiated in state s and 0 otherwise, and feature $\phi_c(s,a,h) = 1 - \frac{1}{c(s,a,h)}$, where $c(s,a,h)$ is the number of time steps since the agent previously executed action a in state s within current history h[11] (see Sutton [46] for an earlier use of a similar feature with the similar goal of encouraging exploration). Feature $\phi_c(s,a,h)$ captures inverse-recency: the feature's value is high when the agent has not experienced the indicated state-action pair recently in history h, and is low when the agent has experienced it recently. Note that it is a feature of the history of the agent's interaction with the external environment and not a feature of the state of the external environment. It can be thought of as a feature

[9] Before an observation-action pair is experienced (i.e., when $n_{s,a} = 0$) the transition model is initialized to the identity function: $\hat{T}(s'|s,a) = 1$ iff $s' = s$.

[10] A discount factor is used to ensure convergence of Q-value iteration used for planning. As for Experiment 1, we emphasize that the discount factor is an agent parameter and does not effect the calculation of fitness for a history.

[11] We encoded the feature in this way to normalize its value in the range $(0,1]$.

maintained by the internal environment of the agent. When the parameter β_c is positive, the agent is rewarded for taking actions that it has not taken recently from the current state. Such a reward is not a stationary function of the external environment's state. Finally, feature $\phi_{\mathcal{F}}(s)$ is a hunger-status feature, and thus when $\beta_{\mathcal{F}} = 1$ and $\beta_c = 0$, the reward function is the fitness-based reward function.

Finding a Good Reward Function. Our optimization procedure adaptively samples reward vectors on the unit sphere, as it can be shown that for the (linear) form of the reward functions and for the agent presented here, searching this subset is equivalent to searching the entire space. More specifically, multiplying the linear reward function parameters by a positive scalar preserves the relative magnitude and signs of the rewards and thus we only need to search over the possible directions of the parameter vector ($\beta_{\mathcal{F}}$ and β_c as a 2d vector) and not its magnitude. Our optimization procedure samples reward vectors on the unit sphere using an adaptive approach that samples more finely where needed; we test the origin $\beta_c = \beta_{\mathcal{F}} = \mathbf{0}$ separately (for the agents presented here, this reward function results in random behavior).

TABLE II
RESULTS FROM THE FORAGING ENVIRONMENTS. THE FIRST COLUMN PRESENTS THE DIFFERENT REWARD FUNCTION TYPES OF INTEREST IN THIS EXPERIMENT. THE SECOND COLUMN SPECIFIES THE SETTING OF THE TWO LINEAR REWARD FUNCTION PARAMETERS FOR EACH TYPE OF REWARD FUNCTION. THE THIRD COLUMN PRESENTS THE MEAN CUMULATIVE (OVER LIFETIME) FITNESS AND THE STANDARD DEVIATION (OVER 200 RANDOMLY SAMPLED ENVIRONMENTS) ACHIEVED BY THE AGENT WITH EACH REWARD FUNCTION TYPE. SEE TEXT FOR FURTHER DETAILS.

reward function type	$\beta_{\mathcal{F}}$	β_c	mean cumulative fitness
Random	0	0	60.51 $\pm$ 0.868
Fitness-based	1	0	1.086 $\pm$ 0.037
Best	0.147	0.989	408.70 $\pm$ 13.685

Results. In this experiment, unchanged across foraging environments are the motion dynamics and the action needed to consume food when the agent is co-located with it. Changing across environments is the sequence of food-appearance locations.

In Table II, we compare the agent using the fitness-based reward function $r_{\mathcal{F}}$ with the agent using the (approximately) best reward function $\hat{r}^*_{\mathcal{A}}$. The fitness in the rightmost column of the table is cumulative over agent lifetimes of $10,000$ time steps and averaged over 200 randomly sampled environments. The table also shows the specific values of reward parameters. Noteworthy is the relatively large coefficient for the inverse-recency feature relative to the coefficient for the hunger-status feature in the best reward. Clearly, an intrinsic reward for executing state-action pairs not experienced recently emerges in the best reward function.

As can be seen in the table, the best reward function significantly outperforms the fitness-based reward function; indeed, with the latter the agent gets stuck and fails to accumulate fitness in most of the sampled environments. Agents using the best reward function, on the other hand, manage to achieve several orders of magnitude improvement in the amount of fitness obtained despite being coupled with a model that is wholly inadequate at predicting the food location (the partial observability causes the Markovian model to "hallucinate" about food at locations where the agent has experienced food before). Indeed, the advantage conferred by the best reward function is the (depth-first search like) systematic and persistent exploration that results from rewarding the experiencing of state-action pairs not experienced recently. Of course, the best reward function also has a positive reward value for the activity of eating (which leads to satiation), for otherwise the agent would not eat the worm even when co-located with it (except as an exploration effect).

To provide a reference point for the effect of exploration, we also implemented an agent that acts purely randomly and thus explores persistently though not systematically. As can been seen from the results in the table, the random agent does much better than the agent with the fitness-based reward (which gets stuck because the model hallucinates about food and thus the agent does not explore systematically or persistently). The agent with the best reward function, however, again outperforms the random agent (the former's model also hallucinates about food but the high positive coefficient associated with the inverse-recency feature overcomes this effect).

Summary. As in the results for Experiment 1, the best reward function positively rewards the activity of eating. What is most interesting about this experiment is that the agent's internal environment—which is of course invariant across the distribution over external environments—provides an inverse-recency feature. The best reward function exploits this feature to intrinsically reward activities that lead to the agent experiencing state-action pairs it has not visited recently, leading to systematic and persistent exploration. This exploration, in turn, distally produces much greater fitness than achieved by an agent using the fitness-based reward. Of course, the environment-specific movements to explore and find food are the result of the agent's planning processes executed throughout its lifetime.

VII. RELATION TO OTHER RESEARCH

The study most closely related to ours is that of Elfwing et al. [11] in which a genetic algorithm is used to search for "shaping rewards" and other learning algorithm parameters that improve an RL learning system's performance. Like ours, this work uses an evolutionary framework to demonstrate that performance can be improved by a suitable choice of reward function. However, its focus on shaping rewards reveals important differences. The key fact about what Ng et al. [23] called shaping rewards is that adding them to an RL agent's primary reward function does not change what policies are optimal.[12] In other words, shaping rewards do not alter the learning problem the agent is facing in the sense that the optimal solution remains the same, but they do offer the possibility—if suitably selected—of providing more informative performance feedback which can accelerate learning. Wiewiora [54] showed that adding shaping rewards

[12]This use of the term shaping differs from its original meaning due to Skinner [42].

is equivalent to initializing the agent's Q-function to non-zero values. Since these initial values are eventually "learned away," the problem reverts asymptotically to the problem initially set by the agent's primary reward function.

Although some shaping rewards might be considered to be intrinsic rewards, the fact that their influence disappears with continued learning is at odds with what psychologists call intrinsic rewards, which are as primary and as long-lived as an animal's more biologically-relevant primary rewards. From a theoretical perspective, since shaping rewards disappear with continued learning, they tend not to be useful in non-stationary environments. For example, the Boxes environment of our Experiment 1 with the step condition is non-stationary. Here, a shaping reward for manipulating boxes would only be useful if it lasted long enough to prevent the box-manipulating behavior from extinguishing before it became useful for incrementing fitness in the second half of the agent't life.

A more fundamental limitation of shaping rewards is that their property of leaving optimal policies unaltered is of limited use in situations where optimal policies cannot be attained due to limitations of the agent's learning algorithm or of the circumstances under which the agent must operate. For example, in our Experiment 1, agents' lives are generally not long enough to allow convergence to an optimal policy. If they could learn over a long enough period of time in a stationary environment, and with a learning algorithm and state representation that ensured convergence to an optimal policy, then a simple fitness-based reward function would allow convergence to a fitness-maximizing policy. Even if this were possible, though, the fitness of the entire lifetime is the most important factor, and this usually depends on learning efficiency more than the asymptotic result. Sutton et al. [48] make related observations about the limitations of asymptotic optimality.

The need for a departure from shaping rewards is even more clear in our Experiment 2 in which the agent cannot sense the location of food and the planning algorithm uses a learned model that makes the assumption that the environment is fully observable. With these limitations, the optimal policy with respect to the best fitness-based reward function gets stuck and is unable to systematically find food via planning. Thus, the best reward function should significantly alter behavior as achieved in our experiments by encouraging persistent and systematic exploration; such an alteration—or indeed any persistent alteration—can not be achieved via shaping rewards. In general, a major function of intrinsic rewards is to compensate for agent limitations, such as the short agent lifetimes in Experiment 1 or the non-Markovian nature of the environments in Experiment 2 (see [44] for further exploration of such compensation).

Although they did not directly touch on the issue of intrinsic versus extrinsic reward, Samuelson and Swinkels [29] put forward a related view regarding the nature of peoples' utility functions. They argue that their analysis shows

> ...that if the agent fully understands the causal and statistical structure of the world, the utility function "maximize the expected number of your descendants" does strictly better than one that puts weight on intermediate actions like eating and having sex. In the absence of such a perfect prior understanding of the world, however, there is evolutionary value in placing utility on intermediate actions. (p. 120, Samuelson and Swinkels [29])

Also related is research on transfer learning [49], which focuses on how learning to perform one task is useful in helping an agent learn to perform a different task. Multi-task learning, also reviewed in [49], explores transfer across multiple tasks dawn from a task distribution. Because our methodology assesses agent fitness over a task distribution, it has implications for transfer learning, especially multi-task learning, which remain to be explored. Good reward functions found by searching reward-function space tap into common aspects of these tasks to facilitate learning across the distribution. We are not aware of approaches to multi-task learning that rely on such searches. Although the variable-reward approach of Mehta et al. [21] involves multiple reward functions, it is quite different in that the tasks in the distribution differ in their reward functions rather than in other features, and no reward-function search is involved. However, the distinction between *agent-space* and *problem-space* in Konidaris and Barto's [17] approach to transfer learning is closely related to our observations because agent-space is determined by features associated with the agent that remain constant across multiple tasks. Thus, in Experiment 2, for example, we could say that the inverse-recency feature given significant weight in the best reward function is a feature of agent-space, suggesting that the agent-space/problem-space distinction may be a natural outcome of an evolutionary process.

The present paper used simple learning and planning agents and thus does not address hierarchical RL [2] and its implications for transfer learning, but our approach sets the stage for further examination of the claim made by Barto et al. [3] and Singh et al. [40] that intrinsic rewards facilitate the acquisition of skills that can form reusable building blocks for behavioral hierarchies. Evolutionary approaches to discovering useful hierarchical structure for RL, such as the work of Elfwing et al. [10], suggest that progress can be made in this direction.

VIII. Discussion and Conclusions

We believe that the new optimal reward framework presented by Singh et al. [41] and elaborated here clarifies the computational role and origin of intrinsic and extrinsic motivation. More specifically, the experimental results support two claims about the implications of the framework for intrinsic and extrinsic motivation.

First, both intrinsic and extrinsic motivation can be understood as emergent properties of reward functions selected because they increase the fitness of learning agents across some distribution of environments. When coupled with learning, a primary reward function that rewards behavior that is useful across many environments can produce greater evolutionary fitness than a function exclusively rewarding behavior directly related to fitness. For example, in both experiments above, eating is necessary for evolutionary success in all environments, so we see primary rewards generated by (satiated)

states resulting immediately from eating-related behavior. But optimal primary reward functions can also motivate richer kinds of behavior less directly related to basic needs, such as play and manipulation of the boxes in Experiment 1, that can confer significantly greater evolutionary fitness to an agent. This is because what is learned as a result of being intrinsically motivated to play with and manipulate objects contributes, within the lifetime of an agent, to that agent's ability to survive and reproduce.

Second, the difference between intrinsic and extrinsic motivation is one of degree—there are no hard and fast features that distinguish them. A stimulus or activity comes to elicit reward to the extent that it helps the agent attain evolutionary success based on whatever the agent does to translate primary reward to learned secondary reward, and through that to behavior during its lifetime. What we call intrinsically rewarding stimuli or activities are those that bear only a distal relationship to evolutionary success. Extrinsically rewarding stimuli or events, on the other hand, are those that have a more immediate and direct relationship to evolutionary success. In fact, in a strict sense, *all* stimuli or activities that elicit primary reward can be considered intrinsically motivated because they bear only a distal relationship to evolutionary success. Having sex is more directly related to evolutionary success (e.g., as measured by the longevity of one's genes in the population) than is childhood play, but both are merely predictors of evolutionary success, not that success itself. Crucially, however, all across this continuum the evolved (optimal) reward function has to be *ubiquitously useful across many different environments* in that the behavior learned from the reward function in each environment has to be good for that environment.

The experiments also clearly demonstrate that learning (specifically RL) exploits regularities within a single agent's lifetime, while the (evolutionary) reward function optimization exploits regularities across environments and agents. For example, in Experiment 1 the location of the boxes did not change within a single agent's lifetime (though they varied across environments) and so the value function learned via RL captured those within-environment regularities. Even more potentially significant and interesting is the role of the internal environment (cf. right panel in Figure 1) that remains relatively unchanged across individuals (whether within or across generations). This can lead the optimal primary reward function to encourage behaviors that involve features from this part of the agent's environment. In general, this might include behaviors that we think of as involving curiosity, novelty, surprise, and other internally-mediated features usually associated with intrinsic reward. Specifically, in Experiment 2 this led the primary reward to encourage the behavior of experiencing state-action pairs that had not been experienced recently. This in turn led to systematic and persistent exploration behavior by the agent which was beneficial across foraging environments. Although our observations do not support the view that dependence on internal environment states is a defining characteristic of intrinsic motivation, they nonetheless provide an explanation for why the archetypical examples of intrinsically rewarding behavior often exhibit this dependency. Prominent among the environmental features that are shared across populations of evolving agents are features of the agents' internal environments.

Our optimal reward framework and experimental results thus explain why evolution would give exploration, manipulation, play, etc. positive hedonic valence, i.e., make them rewarding, along with stimuli and activities that are more directly related to evolutionary success. The distinction between intrinsic and extrinsic motivation is therefore a matter of degree, but their source and role is computationally clear: both intrinsic and extrinsic motivation are emergent properties of a process that adjusts reward functions in pursuit of improved evolutionary success.

Finally, our optimal reward framework also has implications for a basic tenet of RL:

> . . . the reward signal is not the place to impart to the agent prior knowledge about how to achieve what we want it to do. ... The reward signal is your way of communicating to the robot what you want it to achieve, not how you want it achieved. (p. 56, Sutton and Barto [47])

This remains good cautionary advice for the agent designer attempting to impart prior knowledge through the reward function heuristically. The limitations of this approach is illustrated by many examples in which the agent learns to achieve rewarded subgoals without learning to achieve a problem's ultimate goal (e.g., [27]). However, our results demonstrate that reward functions do exist that incorporate prior knowledge in a way that produces significant gains in performance toward the ultimate goal of maximizing fitness. That these reward functions are the result of extensive search supports the essential role that evolution has in making biological reinforcement learning a useful component of adaptive natural intelligence.

Acknowledgements

Satinder Singh and Jonathan Sorg were supported by AFOSR grant FA9550-08-1-0418 and by NSF grant IIS 0905146. Richard Lewis was supported by ONR grant N000140310087 and by NSF grant IIS 0905146. Andrew Barto was supported by AFOSR grant FA9550-08-1-0418. Any opinions, findings, conclusions or recommendations expressed here are those of the authors and do not necessarily reflect the views of the sponsors.

REFERENCES

[1] D. H. Ackley and M. Littman. Interactions between learning and evolution. In C.G. Langton, C. Taylor, C.D. Farmer, and S. Rasmussen, editors, *Artificial Life II (Proceedings Volume X in the Santa Fe Institute Studies in the Sciences of Complexity)*, pages 487–509. Addison-Wesley, Reading, MA, 1991.

[2] A. G. Barto and S. Mahadevan. Recent advances in hierarchical reinforcement learning. *Discrete Event Dynamcal Systems: Theory and Applications*, 13:341–379, 2003.

[3] A. G. Barto, S. Singh, and N. Chentanez. Intrinsically motivated learning of hierarchical collections of skills. In *Proceedings of the International Conference on Developmental Learning (ICDL)*, 2004.

[4] A. G. Barto, R. S. Sutton, and C. W. Anderson. Neuronlike elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, 13:835–846, 1983. Reprinted in J. A. Anderson and E. Rosenfeld (eds.), *Neurocomputing: Foundations of Research*, pp. 535-549, MIT Press, Cambridge, MA, 1988.

[5] D. E. Berlyne. Curiosity and exploration. *Science*, 4153(3731):25–33, 1966.

[6] W. B. Cannon. *The Wisdom of the Body*. W. W. Norton, New York, 1932.

[7] C. N. Cofer and M. H. Appley. *Motivation: Theory and Research*. Wiley, New York, 1964.

[8] T. Damoulas, I. Cos-Aguilera, G. M. Hayes, and T. Taylor. Valency for adaptive homeostatic agents: Relating evolution and learning. In M. S. Capcarrere, A. A. Freitas, P. J. Bentley, C. G. Johnson, and J. Timmis, editors, *Advances in Artificial Life: 8th European Conference, ECAL 2005, LNAI vol. 3636*, pages 936–945. Springer-Verlag, Berlin, 2005.

[9] E. L. Deci and R. M. Ryan. *Intrinsic Motivation and Self-Determination in Human Behavior*. Plenum Press, N.Y., 1985.

[10] S. Elfwing, E. Uchibe, and K. Doya. An evolutionary approach to automatic construction of the structure in hierarchical reinforcement learning. In *Genetic and Evolutionary Computation - GECCO 2003, Part 1*, pages 507–509. Springer, 2003.

[11] S. Elfwing, E. Uchibe, K. Doya, and H. I. Christensen. Co-evolution of shaping rewards and meta-parameters in reinforcement learning. *Adaptive Behavior*, 16:400–412, 2008.

[12] H. F. Harlow. Learning and satiation of response in intrinsically motivated complex puzzle performance by monkeys. *Journal of Comparative and Physiological Psychology*, 43:289–294, 1950.

[13] C. L. Hull. *Principles of Behavior*. D. Appleton-Century, NY, 1943.

[14] C. L. Hull. *Essentials of Behavior*. Yale University Press, New Haven, 1951.

[15] C. L. Hull. *A Behavior System: An Introduction to Behavior Theory Concerning the Individual Organism*. Yale University Press, New Haven, 1952.

[16] L. P. Kaelbling, M. L. Littman, and A. W. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[17] G.D. Konidaris and A.G. Barto. Autonomous shaping: Knowledge transfer in reinforcement learning. In *Proceedings of the Twenty Third International Conference on Machine Learning (ICML 2006)*, pages 489–496, 2006.

[18] D. B. Lenat. *AM: An Artificial Intelligence Approach to Discovery in Mathematics*. PhD thesis, Stanford University, 1976.

[19] M. L. Littman and D. H. Ackley. Adaptation in constant utility nonstationary environments. In *Proceedings of the Fourth International Conference on Genetic Algorithms*, pages 136–142. 1991.

[20] S. M. McClure, N. D. Daw, and P. R. Montague. A computational substrate for incentive salience. *Trends in Neurosciences*, 26:423–428, 2003.

[21] N. Mehta, S. Natarajan, P. Tadepalli, and A. Fern. Transfer in variable-reward hierarchical reinforcement learning. *Machine Learning*, 73:289–312.

[22] K. E. Merrick and M. L. Maher. *Motivated Reinforcement Learing*. Springer-Verlag, Berlin, 2009.

[23] A. Ng, D. Harada, and S. Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Proceedings of the Sixteenth International Conference on Machine Learning*. Morgan Kaufmann, 1999.

[24] J. Olds and P. Milner. Positive reinforcement produced by electrical stimulation of septal areas and other regions of rat brain. *Journal of Comparative and Physiological Psychology*, 47:419–427, 1954.

[25] P.-Y. Oudeyer and F. Kaplan. What is intrinsic motivation? A typology of computational approaches. *Frontiers in Neurorobotics*, 2007.

[26] P.-Y. Oudeyer, F. Kaplan, and V. Hafner. Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, 11, 2007.

[27] J. Randlöv and P. Alström. Learning to drive a bicycle using reinforcement learning and shaping. In *Proceedings of the Fifteenth International Conference on Machine Learning (ICML) 1998*, pages 463–471, San Francisco, 1998. Morgan Kaufmann.

[28] R. M. Ryan and E. L. Deci. Intrinsic and extrinsic motivations: Classic definitions and new directions. *Contemporary Educational Psychology*, 25:54–67, 2000.

[29] L. Samuelson and J. Swinkels. Information, evolution, and utility. *Theoretical Economics*, 1:119–142, 2006.

[30] T. Savage. Artificial motives: A review of motivation in artificial creatures. *Connection Science*, 12:211–277, 2000.

[31] M. Schembri, M. Mirolli, and G. Baldassarre. Evolving internal reinforcers for an intrinsically motivated reinforcement-learning robot. In Y. Demiris, D. Mareschal, B. Scassellati, and J. Weng, editors, *Proceedings of the 6th International Conference on Development and Learning (ICDL)*, London, Imperial College: E1-6, 2007.

[32] J. Schmidhuber. Adaptive confidence and adaptive curiosity. Technical Report FKI-149-91, Technische Universitat Munchen, 1991.

[33] J. Schmidhuber. Adaptive confidence and adaptive curiosity. Technical Report FKI-149-91, Institut für Informatik, Technische Universität München, Arcisstr. 21, 800 München 2, Germany, 1991.

[34] J. Schmidhuber. A possibility for implementing curiosity and boredom in model-building neural controllers. In *From Animals to Animats: Proceedings of the First International Conference on Simulation of Adaptive Behavior*, pages 222–227, Cambridge, MA, 1991. MIT Press.

[35] J. Schmidhuber. What's interesting? Technical Report TR-35-97, IDSIA, Lugano, Switzerland, 1997.

[36] J. Schmidhuber. Artificial curiosity based on discovering novel algorithmic predictability through coevolution. In *Proceedings of the Congress on Evolutionary Computation*, volume 3, pages 1612–1618. IEEE Press, 1999.

[37] J. Schmidhuber and J. Storck. Reinforcement driven information acquisition in nondeterministic environments, 1993. Technical report, Fakultat fur Informatik, Technische Universit at Munchen.

[38] W. Schultz. Reward. *Scholarpedia*, 2(3):1652, 2007.

[39] W. Schultz. Reward signals. *Scholarpedia*, 2(6):2184, 2007.

[40] S. Singh, A. G. Barto, and N. Chentanez. Intrinsically motivated reinforcement learning. In *Advances in Neural Information Processing Systems 17: Proceedings of the 2004 Conference*, Cambridge MA, 2005. MIT Press.

[41] S. Singh, R. L. Lewis, and A. G. Barto. Where do rewards come from? In *Proceedings of the Annual Conference of the Cognitive Science Society*, pages 2601–2606, Amsterdam, 2009.

[42] B. F. Skinner. *The Behavior of Organisms*. Appleton-Century, NY, 1938.

[43] M. Snel and G. M. Hayes. Evolution of valence systems in an unstable environment. In *Proceedings of the 10th International Conference on Simulation of Adaptive Behavior: From Animals to Animats*, pages 12–21, Osaka, Japan, 2008.

[44] J. Sorg, S. Singh, and R.L. Lewis. Internal rewards mitigate agent boundedness. In *Proceedings of the Twenty Seventh International Conference on Machine Learning (ICML)*, Haifa, 2010.

[45] R. S. Sutton. Learning to predict by the method of temporal differences. *Machine Learning*, 3:9–44, 1988.

[46] R. S. Sutton. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Proceedings of the Seventh International Conference on Machine Learning*, pages 216–224, San Mateo, CA, 1990. Morgan Kaufmann.

[47] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.

[48] R. S. Sutton, A. Koop, and D. Silver. On the role of tracking in stationary environments. In *Proceedings of the 24th International Conference on Machine Learning*, 2007.

[49] M. E. Taylor and P. Stone. Transfer learning for reinforcement learning. *Journal of Machine Learning Research*, 10:1633–1685, 2009.

[50] E. Uchibe and K. Doya. Constrained reinforcement learning from intrinsic and extrinsic rewards. In *Proceedings of the IEEE International Conference on Developmental Learning*. London, UK, 2007.

[51] E. Uchibe and K. Doya. Finding intrinsic rewards by embodied evolution and constrained reinforcement learning. *Neural Networks*, 21(10):1447–1455, 2008.

[52] C. J. C. H. Watkins. *Learning from Delayed Rewards*. PhD thesis, Cambridge University, Cambridge, England, 1989.

[53] C. J. C. H. Watkins and P. Dayan. Q-learning. *Machine Learning*, 8:279–292, 1992.

[54] E. Wiewiora. Potential-based shaping and Q-value initialization are equivalent. *Journal of Artificial Intelligence Research*, 19:205–208, 2003.

[55] P. T. Young. Food-seeking, drive, affective process, and learning. *Psychological Review*, 56:98–121, 1949.

Satinder Singh is a Professor of Computer Science and Engineering at the University of Michigan, Ann Arbor. He received a B.Tech in electrical engineering from the Indian Institute of Technology, New Delhi in 1987, and a PhD in Computer Science from the University of Massachusetts, Amherst in 1993. He is director of the Artificial Intelligence Laboratory at the University of Michigan.

Richard Lewis is a Professor of Psychology and Linguistics at the University of Michigan, Ann Arbor. He received a B.S. with honors in computer science from the University of Central Florida in 1987, and a PhD in Computer Science from Carnegie Mellon University in 1993. He is director of the Language and Cognitive Architecture Laboratory, and a core member of the Cognition and Cognitive Neuroscience Program at the University of Michigan.

Andrew Barto is a Professor of Computer Science, University of Massachusetts, Amherst. He received a B.S. with distinction in mathematics from the University of Michigan in 1970, and a Ph.D. in Computer Science in 1975, also from the University of Michigan. He Co-Directs the Autonomous Learning Laboratory and is a core faculty member of the Neuroscience and Behavior Program of the University of Massachusetts.

Jonathan Sorg is a Ph.D. candidate in Computer Science at the University of Michigan. He received a B.S. in Computer Science from Northwestern University in 2005 and a M.S. in Computer Science from the University of Michigan in 2008.