Noname manuscript No.
(will be inserted by the editor)

An information-theoretic approach to curiosity-driven reinforcement learning

Susanne Still · **Doina Precup**

Received: date / Accepted: date

Abstract We provide a fresh look at the problem of exploration in reinforcement learning, drawing on ideas from information theory. First, we show that Boltzmann-style exploration, one of the main exploration methods used in reinforcement learning, is optimal from an information-theoretic point of view. Second, we address the problem of curiosity-driven learning. We propose that, in addition to maximizing the expected return, a learner should chose a policy that maximizes the predictive power of its own behavior, measured by the information that the most recent state-action pair carries about the future. This makes the world “interesting” and exploitable. The general result has the form of Boltzmann-style exploration with a bonus that contains a novel exploration-exploitation trade-off that emerges from the proposed optimization principle. Importantly, this exploration-exploitation trade-off is also present when the “ temperature”-like parameter in the Boltzmann distribution tends to zero, i.e. when there is no exploration due to randomness. As a result, exploration emerges as a directed behavior that optimizes information gain, rather than being modeled solely as behavior randomization.

1 Motivation

The problem of optimal decision making under uncertainty is crucial both to animals and to artificial intelligent agents. Reinforcement learning (RL) addresses this problem by proposing that agents should choose actions such as to maximizes an expected long-term return provided by the environment [23]. To achieve this goal, an agent has to *explore* its environment, while at the same time *exploiting* the knowledge it currently has in order to achieve good returns. In existing algorithms, this trade-off is achieved mainly through simple randomization of the action choices. Practical implementations rely heavily on heuristics, though theoretically principled approaches also exist (see Sec. 5 for a more detailed discussion). In this paper, we look at the exploration-exploitation trade-off from a fresh perspective: we use information-theoretic methods both to analyze an existing exploration method, and to propose a new one.

Recently, an information theoretic framework for behavioral learning has been presented by Still [19], with the goal of providing a good exploration strategy for an agent who wants to learn a predictive representation of its environment. We use this framework to tackle reward-driven behavioral learning. We propose an intuitive optimality criterion for exploration policies which includes both the reward received, as well as the complexity of the policy. Having a simple policy is not usually a stated goal in reinforcement learning, but it is desirable for bounded-rationality agents, and it is especially useful in the context of developmental agents, which should evolve increasingly complex strategies as they get more experience, and as their knowledge of the environment becomes more sophisticated. We show in Sec. 2 that the general solution of the proposed optimization problem is a Boltzmann-style exploration algorithm. The trade-off between the return, on the one hand, and the average bit cost of the policy, on the other hand, is controlled by a “temperature”-like parameter. At high temperatures, simplicity is more important than return. As the temperature

S. Still
Information and Computer Sciences
University of Hawaii, Manoa, USA
E-mail: sstill@hawaii.edu

D. Precup
School of Computer Science
McGill University, Montreal, Canada
E-mail: dprecup@cs.mcgill.ca

decreases, return becomes increasingly important; the policy converges to the optimal-return policy as the temperature goes to zero.[1]

Animals often explore their environment not only to gather rewards, but also just for the sake of learning about it. Such learning is useful because the environment may change over time, and the animal may need to adapt to this change. Hence, it is advantageous to know more about the environment than what is strictly necessary in order to maximize the long-term return under the current conditions. Similar arguments have been presented in [18] as well as in many papers on transfer of knowledge in reinforcement learning (see [24] for a survey). In Sec. 3, we formulate this goal in terms of maximizing future return, while at the same time maximizing the predictive power of the behavior, which we measure by the information carried about the future. Predictive information, defined as the mutual information between the past and the future within a time series, measures temporal correlations and is related to other measures of complexity [4, 7]. It provides a measure of how complex, or "interesting", a time series is. Our objective function also contains a term which ensures that the agent continues to prefer simple policies (as in the case of simple return maximization). This term penalizes behaviors for retaining more memory about the past than is necessary to predict the future. As a consequence it ensures that undesirable repetitive behaviors are avoided. We show that the resulting optimal policy contains a trade-off between exploration and exploitation which emerges naturally from the optimization principle.

Our approach is similar to rate distortion theory [17], which is based on the fact that approximating a true signal using a compressed representation will cause a loss, computed as the expected value of a distortion function. The choice of the distortion function implicitly provides the distinction of relevant and irrelevant features of the data. Information theoretic approaches inspired by some form of rate-distortion theory have been used widely in machine learning, for example for clustering and dimensionality reduction [26,21,6,20]. However, to our knowledge, this approach has not been used in reinforcement learning prior to our work.

The paper is structured as follows. In Sec. 2, we lay the information-theoretic foundation of exploration for a reinforcement learning agent, whose main goal is to optimize long-term returns. Next, we formulate the problem of curiosity-driven reinforcement learning and solve it using a similar principle that includes the maximization of predictive information (Sec. 3). Finally, we discuss algorithmic implementation issues in Sec. 4, and close with a discussion of the relationship of our approach to classical and current work in RL in Sec. 5.

2 Information-theoretic approach to exploration in reinforcement learning

We consider the standard RL scenario [23] in which an agent is interacting with an environment at a discrete time scale. At each time step t, the agent observes the state of the environment, $x_t \in \mathbf{X}$ and takes an action $a_t \in \mathbf{A}$. In response to its action, the agent receives an immediate (extrinsic) reward, r_{t+1} and the environment transitions to a next state x_{t+1}. We assume that the environment is Markovian. Hence, the reward is expressed as $r_{t+1} = R(x_t, a_t)$, where $R : \mathbf{X} \times \mathbf{A} \to \mathbb{R}$ is the reward function, and the next state x_{t+1} is drawn from the distribution $p(X_{t+1}|x_t, a_t)$[2]. The reward function and the next-state transition distributions constitute the *model of the environment*. A way of behaving, or *policy*, $\pi : \mathbf{X} \times \mathbf{A} \to [0, 1]$ is a probability distribution over actions conditioned on the state. Each policy has an *action-value function* associated with it:

$$Q^\pi(x, a) = E_\pi[r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots |X_t = x, A_t = a], \tag{1}$$

where $\gamma \in (0, 1)$ is a discount factor expressing the fact that later rewards should be emphasized less. The interpretation of this value function is that the agent starts in state x, chooses a as its first action and thereafter chooses actions according to π. The goal of a reinforcement learning agent is to find a policy that maximizes the value function for all state-action pairs. In a finite Markov Decision Process, there is always at least one deterministic policy that achieves this goal, and many methods can be used to find such a policy (see [23] for a comprehensive review). In some situations, e.g. when the state space $\mathbf{X}$ is too large and value functions cannot be represented exactly, policies are compared with respect to a starting state distribution, $p_0(X)$. Then, the goal is to maximize the expected return:

$$V^\pi = \sum_{x \in \mathbf{X}} \sum_{a \in \mathbf{A}} p_0(x) \pi(a|x) Q^\pi(x, a) \tag{2}$$

The advantage of using this criterion is that it allows a policy to be characterized by a single number, and offers a clear ordering of policies. Then, the optimal policy for the MDP maximizes V^π, for example, for the uniform starting distribution.

[1] We will refer to this parameter as the temperature in the rest of the paper. One has to keep in mind that this is a metaphor, not a physical temperature.

[2] Here and throughout, we use capital letters to denote random variables, and small letter to denote particular realizations of these variables.

Suppose that we had a set of policies that all produce the same expected return. Which policy should be preferred? If one is to implement the policy on a real system, e.g. a robot, then it is reasonable to prefer the simplest policy, i.e. the policy that can be specified with the smallest number of bits. To make this precise, let us re-interpret the meaning of a policy. The action can be viewed as a *summary* of the state of the system. Therefore, we view the act of mapping states onto actions as lossy compression. If a large group of states share the same optimal action, then that action can be viewed as a *compressed representation* for this state "cluster", a representation which is sufficient from the point of view of attaining a desired level of return.

In order to formalize this intuition, we revisit rate distortion theory, introduced by Shannon [17]. Rate distortion theory measures the cost of approximating a signal Z by a signal Y, using the expected value of some distortion function, $d(Z, Y)$. This distortion measure can, but need not, be a metric. Lossy compression is achieved by assigning Z to Y via the probabilistic map $P(Y|Z)$, such that the mutual information:

$$I(Z, Y) = \sum_{z \in \Omega_Z} \sum_{y \in \Omega_Y} P(z, y) \log_2 \left[\frac{P(z, y)}{P(z)P(y)} \right] = \sum_{z \in \Omega_Z} \sum_{y \in \Omega_Y} P(y|z)P(z) \log_2 \left[\frac{P(y|z)}{P(y)} \right]$$

is minimized. The minimization is constrained by fixing the expected distortion $\sum_{z \in \Omega_Z} \sum_{y \in \Omega_Y} P(z, y)d(z, y)$. In other words, recalling the meaning of information in terms of bit message length, among the representations with the same quality, the most compact one will be preferred.

We interpret return as a function that measures quality, rather than distortion. The action is interpreded as a lossy summary of the state; hence, among the policies with the same return, we will find the most compact one. Considering a set of policies that achieve a fixed average return V^π, we can express this principle through the following optimization problem:

$$\min_{\pi} \quad I^\pi(A, X) \tag{3}$$

$$\text{subject to:} \quad V^\pi = \text{const.} \quad \text{and:} \sum_{a \in \mathbf{A}} \pi(a|x) = 1, \forall x \in \mathbf{X} \quad \text{and:} \ \pi(a|x) \geq 0, \forall x \in \mathbf{X}, \forall a \in \mathbf{A}. \tag{4}$$

Here, π is the policy we seek, which can be viewed as a probabilistic assignment of states to actions. The second constraint ensures normalization. The average return of policy π, V^π, is defined in (2). The term $I^\pi(A, X)$ denotes the information that the action A carries about the state X under policy π, where the joint distribution is given by $p(X, A) = \pi(A|X)p^\pi(X)$:

$$I^\pi(A, X) = \sum_{x \in \mathbf{X}} \sum_{a \in \mathbf{A}} \pi(a|x)p^\pi(x) \log \left[\frac{\pi(a|x)}{p^\pi(a)} \right]. \tag{5}$$

Note that the information that the action carries about the state depends also on the stationary distribution of states under policy π, p^π (which we assume exists, as is standard in RL) and on the average action probability, defined for any action a as: $p^\pi(a) = \sum_{x \in \mathbf{X}} p^\pi(x)\pi(a|x)$.

This optimization problem is complex, because of the dependence on the stationary distribution, which in general is unknown (though computable) and which changes as the policy π evolves during learning. A standard approach for changing the policy in reinforcement learning is to assume that we fix the policy π, compute its return, but then we consider a small perturbation around it at a given time step t. Let $V_t^\pi(q)$ be the expected return if the agent acts according to policy π on all time steps, except on time step t, when it chooses its action according to a different action distribution q:

$$V_t^\pi(q) = \sum_{x_0 a_0 \ldots x_t, a_t} p_0(x_0) \left(\prod_{j=0}^{t-1} \pi(a_j|x_j)p(x_{j+1}|x_j, a_j) \right) q(a_t|x_t) \left[\sum_{i=0}^{t-1} \gamma^i R(x_i, a_i) + \gamma^t Q^\pi(x_t, a_t) \right]$$

where $q(a_t|x_t)$ is the new probability of choosing action a_t from the state x_t, which we seek. Let $I_q^\pi(A_t, X_t)$ denote the information that the action A_t carries about the state X_t:

$$I_q^\pi(A_t, X_t) = \sum_{x \in \mathbf{X}} \sum_{a \in \mathbf{A}} q(a|x)p_t^\pi(x) \log \left[\frac{q(a|x)}{p_t^\pi(a)} \right], \tag{6}$$

where x and a range over the possible values of random variables X_t and A_t, $p_t^\pi(x)$ is the probability of arriving at state x on time step t if the agent starts with a state drawn from p_0 and chooses actions according to π:

$$p_t^\pi(x) = p^\pi(X_t = x) = \sum_{x_0 a_0 \ldots x_{t-1} a_{t-1}} p_0(x_0)\pi(a_0|x_0)p(x_1|x_0, a_0) \ldots \pi(a_{t-1}|x_{t-1})p(X_t = x|x_{t-1}, a_{t-1}),$$

and $p_t^\pi(a) = p^\pi(A_t = a) = \sum_{x \in \mathbf{X}} p_t^\pi(x)q(A_t = a|x)$.

Now, the optimization problem can be written as:

$$\min_q I_q^\pi(A_t, X_t)$$
$$\text{subject to: } V_t^\pi(q) = \text{const.} \quad \text{and: } \sum_{a \in \mathbf{A}} q(a|x) = 1, \forall x \in \mathbf{X}, \text{ and: } q(a|x) \geq 0, \forall x \in \mathbf{X}, \forall a \in \mathbf{A}.$$

This optimization principle has a dual form, where we maximize the average return under the constraint that the "size" of the policy is kept constant. Note that this view is mathematically equivalent, but potentially very useful when we think of agents with limited computational capacity (e.g., robots with limited on-board computation). In this case, one may just want to find the best policy which still fits on the available physical system. Similar capacity constraints may apply to animals. The dual form is the following:

$$\max_q V_t^\pi(q) \tag{7}$$
$$\text{subject to: } I_q^\pi(A_t, X_t) = \text{const.} \quad \text{and: } \sum_{a \in \mathbf{A}} q(a|x) = 1; \ \forall x \in \mathbf{X}. \tag{8}$$

A similar cost function was given by Bagnell and Schneider, as well as Peters and Schaal [3,12]; however, they only used a linearization to compute a better type of policy gradient update.

We can now re-write the constrained optimization principle, using the Lagrange multipliers λ and $\mu(x)$:

$$\max_q F[q], \tag{9}$$

where the objective function is a functional of the policy $q(A_t|X_t)$, given by

$$F[q] = V_t^\pi(q) - \lambda I_q^\pi(A_t, X_t) + \sum_{x \in \mathbf{X}} \mu(x) \left(\sum_{a \in \mathbf{A}} q(a|x) - 1 \right). \tag{10}$$

The solution is obtained by setting the variation of F to zero which leads to the optimal policy

$$q_{\text{opt}}(A_t = a|X_t = x) = \frac{p_t^\pi(a)}{Z(x)} e^{\frac{1}{\lambda} Q^\pi(x,a)} = \frac{1}{Z(x)} e^{\frac{1}{\lambda} Q^\pi(x,a) + \log p_t^\pi(a)}, \ \forall x \in \mathbf{X}, a \in \mathbf{A} \tag{11}$$

which has to be solved self consistently, together with:

$$p_t^\pi(a) = \sum_{x \in \mathbf{X}} q_{\text{opt}}(a|x) p_t^\pi(x), \ \forall a \in \mathbf{A} \tag{12}$$

The partition function $Z(x) = \sum_{a \in \mathbf{A}} p_t^\pi(a) e^{\frac{1}{\lambda} Q^\pi(x,a)}$ ensures normalization. This solution is similar to Boltzmann exploration, also known as softmax action selection [23]. The only difference is that here, we have an additional "complexity penalty", $\log p_t^\pi(a)$. We note that by a similar calculation, if one tries to optimize the return at a fixed level of randomness (using the Shannon entropy as a measure for randomness), one recovers exactly Boltzmann exploration. This follows immediately from the results in [14], and the arguments presented in [8]. In contrast, here we penalize explicitly for the complexity of the policy, measured by the coding cost. The result is that there is a penalty for using more actions than necessary (comp. Eq. (11)). This is useful not only when the agent has limited computational capacity, but also when the action space is very large (for example, in combinatorial optimization or inventory control problems). In this case, Eq. (11) may force the agent to use only a subset of the entire action space, which makes the learning task easier. The policy update in Eq. (11) appears also related to the ones suggested in references [2,11] despite of their different roots.

This Boltzmann-style softening of the policy optimally trades the complexity of the policy for average return. The trade-off is governed by the "temperature"-like parameter λ, and exploration takes place due to fluctuations only at non-zero temperature, when emphasis is put on the compactness of the policy. As λ tends to zero, the information minimization constraint in Eq. (10) becomes less relevant, and one can easily show that in the limit, the optimal policy becomes deterministic if there are no degeneracies.[3] The optimal action becomes a function of the past, and is chosen to maximize the return:

$$a^{\text{opt}}(x) = \arg\max_a Q^\pi(x, a) \tag{13}$$

[3] If the equivalent of the ground state is degenerate, then all N actions that maximize $Q^\pi(x, a)$ occur with probability $1/N$, while all other actions occur with probability 0.

3 Curiosity-driven reinforcement learning

Intuitively, exploration is driven by the curiosity to visit unknown areas of the state space. The theory we have developed so far is lacking any notion of curiosity. Apart from the rate constraint, the agent is just maximizing the return, as defined based on external rewards received from the environment. In this section, we present a formalization of curiosity based on information-theoretic principles. Drawing on ideas from our previous work [19], we postulate that the main goal of a curious agent is to create a time series of states that is interesting. We measure "interestingness" by means of the predictive power that the agent's behavior carries, as defined in [19]. Intuitively, if a time series has high predictive information, there will be data available for learning about a variety of situations, and also about different ways of behaving. In the context of a fully observable Markovian environment, this is the mutual information carried by the state of the environment (i.e., the sensation of the agent) at time t, together with the action, about the state of the environment at time $t+1$. Then, our goal becomes to find the policy that maximizes predictive power. We note that this is important not only in Markovian environments, but also in extensions to Partially Observable Markov Decision Processes (POMDPs), where the exact state of the environment is unknown. Maximizing predictive information is highly desirable in this setting, because it means that the agent is able to predict well its future sensation given the past data. The extension to the POMDP setting is straightforward given the theory outlined in [19].

Formally, by taking predictive power into account, we now have another constraint in the optimization principle. Using Lagrange multipliers, as before, we can write

$$\max_q F[q], \tag{14}$$

where the objective function is now given by

$$F[q] = I_q^\pi(\{X_t, A_t\}, X_{t+1}) + \alpha V_t^\pi(q) - \lambda I_q^\pi(A_t, X_t) + \sum_{x \in \mathbf{X}} \mu(x) \left(\sum_{a \in \mathbf{A}} q(a|x) - 1 \right). \tag{15}$$

We obtain the solution as before. However, now we have an additional contribution from $\delta I_q^\pi(\{X_t, A_t\}, X_{t+1})/\delta q$, as the following term:

$$D_{\mathrm{KL}}[p(X_{t+1}|X_t, A_t) \| p^\pi(X_{t+1})] \tag{16}$$

where the Kullback-Leibler divergence is defined as

$$D_{\mathrm{KL}}[p_1(X) \| p_2(X))] = \sum_x p_1(x) \log \left[\frac{p_1(x)}{p_2(x)} \right], \tag{17}$$

and

$$p^\pi(X_{t+1} = x') = \sum_{a \in \mathbf{A}} \sum_{x \in \mathbf{X}} p(X_{t+1} = x'|x, a) q(a|x) p_t^\pi(x), \forall x' \in \mathbf{X} \tag{18}$$

With the extra contribution (16), the optimal solution now becomes

$$q_{\mathrm{opt}}(A_t = a|X_t = x) = \frac{p_t^\pi(a)}{Z(x)} e^{\frac{1}{\lambda}\left(D_{\mathrm{KL}}[p(X_{t+1}|X_t=x, A_t=a) \| p^\pi(X_{t+1})] + \alpha Q^\pi(x,a),\right)}, \forall x \in \mathbf{X}, \forall a \in A. \tag{19}$$

The first term in the exponent drives the agent towards exploration. The optimal action will maximize the divergence between the distribution over the next state, given the curent state x and the action a, to the average distribution over the next state. This means that the optimal action will produce a next state with a conditional probability distribution far from the average distribution. The second term is the value maximization, as before. The exponent in Eq. (19) thus represents a trade-off between exploration and exploitation. This *emerges* from the optimization principle.

As $\lambda \to 0$, the policy will become deterministic, and with probability one, the chosen action will be the one that maximizes the functional in the exponent of equation (19):[4]

$$a^{\mathrm{opt}}(x) = \arg\max_a \big[D_{\mathrm{KL}}[p(X_{t+1}|X_t = x, A_t = a)||p^\pi(X_{t+1})] + \alpha Q^\pi(x, a) \big] \tag{20}$$

Note that the optimal action includes a natural trade–off between exploration and exploitation, even when the policy is deterministic! This is not the case for pure Boltzmann-style exploration, where the optimal action under a deterministic policy simply maximizes the return, subject possibly to size constraints (see Eq. (13)).

The parameter α can be viewed as a measure of how interested the agent is in obtaining a reward. For example, if the reward is energy intake, then α could be set by measuring the charge of the batteries of a robot, and would represent how "hungry" the agent is.

[4] This is true if there are no degeneracies, otherwise all those actions occur with equal probability, as in Sec. 2.

3.1 Illustrations

To build some intuition about what this approach does, we consider a couple of simple examples. First, imagine a world in which there are two states, $x \in \{0, 1\}$, and assume that $\alpha = 0$, so the optimal action becomes the one that maximizes only the predictive power. Consider a continuous range of actions $a \in [0, 1]$. The value of the action expresses how strongly the agent tries to stay in the same state or leave it, such that $a = 0$ means that the agent wants to remain in the same state, $a = 1/2$ means that the agent is ambivalent about staying or leaving, and $a = 1$ means that the agent tries to switch state. Let the Markovian transitions of the environment be given by $p(x'|x, a) = a, \forall x, x', \forall a$. Then, the optimal policy, Eq. (19), chooses only those two actions which result in the largest predictability, namely $a = 0$ and $a = 1$, and it chooses between these two actions with equal probability. This "clever random" policy is an example of balance between control and exploration, as mentioned in [19].

As a second case, consider a two-state world in which there are only two actions, STAY or FLIP, $a \in \{s, f\}$, and the transition probabilities are such that one state is completely reliable: $p(0|0, s) = p(1|0, f) = 1$, while the other state is completely unreliable $p(0|1, s) = p(0|1, f) = 1/2$. This is a test for our information theoretic objective: if we are doing the right thing, then we should find that in the absence of a reward (or an interest in a reward, $\alpha = 0$), the optimal curious policy should enable exploration of the combined state-action space, which means that we should *not* stay in the more reliable state with probability 1. Thus, if we find that the optimal policy is $\pi(s|0) = 1$, then we know that our objective is wrong. Maximizing $I[\{X_t, A_t\}; X_{t+1}]$ results asymptotically in the policy $\pi(s|0) = 3/4$, which balances between exploration and choosing a reliable state, i.e. control. Note that it is obvious that the policy is random in state 1, since $D_{\mathrm{KL}}[p(X_{t+1}|X_t = 1, A_t = s)||p(X_{t+1})] = D_{\mathrm{KL}}[p(X_{t+1}|X_t = 1, A_t = f)||p(X_{t+1})]$. All calculations for this section are in the Appendix.

If our criterion was incorrect, the policy for state 0 would be $\pi(s|0) = 1$, making the agent stick to the more predictable state. This mean that some of the state-action space is never explored, which is undesirable for learning. It is instructive to see that maximizing $I[X_t; X_{t+1}]$, on the other hand, results in $\pi(s|0) = 1$. This simple case should be compared to results reported in [1].

4 Algorithmic issues

The optimal solution consists of Eq. (19), which has to be solved self consistently, together with Eq. (18). Furthermore, the action-value function Q has to be estimated. In this section, we discuss how this can be implemented in practice.

We propose an implementation that is inspired by the usual Boltzmann exploration algorithm. The algorithm proceeds as follows.

1. Initialize $t \leftarrow 0$ and get initial state x_0. Initialize $\pi(a|x), \forall x \in \mathbf{X}, \forall a \in \mathbf{A}$ (e.g., uniformly randomly) and initialize the action-value function Q.
2. Repeat at every time step t
   (a) Update $p_t(x), \forall x \in \mathbf{X}$ (the current estimate of the state visitation distribution)
   (b) Initialize $q^{(0)}(a|x), \forall a \in \mathbf{A}, \forall x \in \mathbf{X}$
   (c) Repeat the following updates, until the difference between $q^{(j)}$ and $q^{(j+1)}$ is small:

$$p^{(j)}(a) \leftarrow \sum_x q^{(j)}(a|x)p_t(x), \forall a \in \mathbf{A} \tag{21}$$

$$p^{(j)}(x') \leftarrow \sum_x \sum_a p(x'|a, x)q^{(j)}(a|x)p_t(x), \forall x' \in \mathbf{X} \tag{22}$$

$$q^{(j+1)}(a|x) \leftarrow \frac{p^{(j)}(a)}{Z^{(j)}(x)} \exp\left[\frac{1}{\lambda}\left(D_{\mathrm{KL}}[p(X_{t+1}|a, x)||p^{(j)}(X_{t+1})] + \alpha Q(x, a)\right)\right], \forall x \in \mathbf{X}, \forall a \in \mathbf{A} \tag{23}$$

   Update $\pi \leftarrow q^{(j+1)}$
   (d) Choose action $a_t \sim \pi(\cdot|x_t)$ and obtain reward r_{t+1} and next state x_{t+1}
   (e) Update the action-value function estimates Q.
   (f) $t \leftarrow t + 1$

In this algorithm, step 2a can be performed exactly by using the true model and all the previous policies; the update of the model is similar the one in Eq. (22); we note that this is exactly the same type of update used in the forward algorithm in a Hidden Markov Model (HMM). However, this computation can be expensive if the number of states is large. As a result, in this case we would use the state samples $x_k, k \leq t$, to estimate $p_t(x)$ approximately.

The initial value $q^{(0)}(a|x)$ is important, as it will influence the point to which iteration 2c converges (convergence is guaranteed to a locally optimal solution). A good solution would be to to start with the result of the previous iteration, under the assumption that the policy will change fairly smoothly from one time step to the next.

In principle, the action-value function Q should be re-computed exactly at every step, using the known model and the computed policy. This involves solving a system of linear equations with $|\mathbf{X}| \times |\mathbf{A}|$ unknowns. While this may be feasible for small environments, it is computationally expensive for larger problems. In this case, the value $Q(x_t, a_t)$ can instead be updated incrementally, using the standard temporal-difference learning approach (i.e., a learning rule like Sarsa or Q-learning; see [23] for details). Intuitively, this approach should work well if the policy changes slowly, because the action-value function will only change around the current state x_t. Similarly, in order to save computation, the policy may be re-computed only at x_t, rather than at all states $x \in \mathbf{X}$, as indicated in Eq. (23).

If the agent has no knowledge of the environment, then it can use the samples received to fit an approximate model, $\hat{p}(X_{t+1}|X_t, A_t)$, and then use this model in the computation above. The model, action values, and distributions of interest can all be updated incrementally from samples. If a batch of samples is gathered first and then we run the algorithm above, we obtain an approach fairly close to batch model-based reinforcement learning. If on every time step t we update the model estimate $\hat{p}(X_{t+1}|X_t, A_t)$ and immediately use it in the policy computation, we obtain an algorithm very close to incremental, model-free reinforcement learning.

The "temperature"-like parameter λ determines how deterministic the resulting policy is. There are different possibilities for choosing this parameter. In the simplest case, the parameter is fixed to a pre-specified value, for example dictated by the capacity/memory constraints of a robot. This selects a fixed trade-off between complexity and utility. More generally, a process known as deterministic annealing [14] can be employed at every time step. It consists of starting with a large temperature, running the iterative algorithm until convergence, then lowering the temperature by a factor (the annealing rate) and continuing this process, until the policy is deterministic, always using the current result as initial conditions for the iterations at the next (lower) temperature. This method obtains, at each time step, the deterministic, optimal policy, according to the criterion. The procedure is computationally intensive, but guarantees that actions are always chosen in a way that maximizes the optimization criterion, given that the annealing rate is sufficiently slow. Finally, the temperature can be fixed during each time step, but lowered as a function of time, $\lambda(t)$, until it approaches zero. This approach is preferable when the agent's knowledge about the world increases with time, and when there is no other fixed constraint of the complexity of the desired policy. Methods such as the ones outlined in [20] can be used to find (a bound on) $\lambda(t)$. Finally, if a complexity constraint is given by the design of the agent, this scheme can be modified to include a $\lambda_{\min} = \lim_{t\to\infty} \lambda(t)$.

If the algorithm is implemented using only exact computations (i.e., the most computationally expensive version, outlined above), it is guaranteed to converge to a locally optimal solution for the proposed optimization criterion. Convergence analysis for the case in which samples are used incrementally is quite tricky and we leave it for future work.

5 Related work

The textbook by Sutton & Barto [23] summarizes several randomization-based exploration approaches used in reinforcement learning, such as Boltzmann exploration and ϵ-greedy (in which there is simply a fixed, small probability of trying out actions which appear sub-optimal). Many heuristic variations have been proposed, in which bonuses are added to the value function to encourage more efficient exploration (e.g. [25, 13])

A different strategy, which yields interesting theoretical results, is that of optimism in the face of uncertainty: if a state has not been visited sufficiently for the agent to be familiar with it, it is automatically considered good, so the agent will be driven towards it. This ensures that an agent will explore new areas of the state space. The first sample-complexity results for reinforcement learning using this idea were provided by Kearns and Singh in [9]. The authors assumed that a state is "known" if it has been visited a sufficiently large number of times. The RMAX algorithm proposed by Brafman and Tennenholtz in [5] is a practical implementation of this idea. An extensive theoretical analysis of this approach was given by Strehl, Li and Litman in [22], showing sample-complexity results both for reinforcement learning methods that learn a model and ones that learn directly a value function. Those PAC-style bounds are not directly related to our work.

Previous work on curiosity-driven reinforcement learning is centered around the idea that agents are motivated by an internal reward signal, and in the process of maximizing this reward, they learn a collection of skills. In early work [15], Schmidhuber proposed different kinds of internal reward signals. More recently, a hierarchical learning approach was put forth by Singh, Barto & Chentanez in [18]. In this case, both an external and an internal reward signal are used to learn a behavior policy. At the same time, the extrinsic reward is used to learn multiple temporally extended behaviors. The particular setting proposed for the intrinsic reward is attempting to provide a novelty bonus. We note that the intrinsic reward is only used to generate behavior. The paper assumes that there are certain events in the world that are "salient" and which the agent will be motivated to seek. Oudeyer and colleagues implemented these ideas in robotics tasks (see e.g. [10]). More recently, Schmidhuber (in [16] and related works) proposed a novel approach to creativity and exploration, which is related to information theory, yet different. The relationship to our work deserves further investigation.

The recent work on differential dynamic programming (e.g. [28, 2]) addresses the problem of finding closed-form solutions to reinforcement learning problems, by reformulating the optimization objective. More specifically, the system is considered to have a "passive" dynamics (induced by a default policy). The optimization criterion then includes both the value function and a term that penalizes deviations form this "passive" dynamics (using the KL-divergence between the state distributions induced by the sought policy and the default policy). This line of work comes from the perspective of continuous control. The results obtained for the optimal policy look similar to the updates we obtain, but the motivation behind the approach is very different.

A similarly defined policy update is also obtained by Peters et al. [11], coming from yet another different angle. They formulate an optimization problem in which the goal is to utilize the existing samples as well as possible. They formulate a policy search algorithm in which new policies are penalized if they induce a state distribution that is different from the empirical distribution observed in the past data. The fact that very different points of view lead to syntactically similar policy updates is intriguing and we plan to study it further in future work.

The idea of using information theory in reinforcement learning has been in some prior work. Ay et al. [1] explore the maximization of $I[X_t, X_{t+1}]$ for *linear* models. The main conceptual difference in our work is that we penalize policies with more memory than is needed for prediction, a notion that is not present in [1]. In [27], Tishby and Polanyi propose an MDP formulation in which rewards are traded off against information. The authors observe that information also obeys Bellman-like equations, and use this observation to set up dynamic programming algorithms for solving such MDPs. While we share the idea of using information, our work is different in a few important aspects. First, the development in [27] is with respect to a single state distribution, while we account for the state distributions induced by different policies. Second, in their formulation the information value ends up mixed with the value function. In our case, information influences the exploration policy, but ultimately, one can still obtain a value function, and a policy, that reflect only reward optimization. Another important distinction is that in their formulation, deterministic policies are more "complex" than randomized ones, whereas in our case, a deterministic policy that is constant everywhere would still be considered "simple". We anticipate that such a treatment will be important in the generalization of these ideas to continuous states and actions (where simple policies will share the same choices across large subsets of states). Little and Sommer (Technical report 2010, unpublished) considered several measures for estimating (or approximating) the information gain of an action in the context of past data. They found that learning efficiency is strongly dependent on temporal integration of information gain but less dependent on the particular measure used to quantify information gain.

Our work can be viewed as defining implicitly an intrinsic reward, based on the idea of maximizing how interesting is the time series experienced by the agent. We note that the intrinsic rewards proposed by Singh, Barto and Chentanez in [18] also involve the probability of the next state given the current state, under different extended behaviors. However, this is proposed as a heuristic. The relationship to our results remains to be explored.

6 Conclusion and future work

In this paper, we introduced a new information-theoretic perspective on the problem of optimal exploration in reinforcement learning. We focused, for simplicity, on Markovian environments, in which the state of the environment is observable and does not have to be learned.

We showed that a soft policy similar to Boltzmann exploration optimally trades return and the coding cost (or complexity) of the policy. By postulating that an agent should, in addition to maximizing the expected return, also maximize its predictive power, at a fixed policy complexity, we derived a trade-off between exploration and exploitation that does not rely on randomness in the action policy, and thereby may be more adequate to model exploration than previous schemes.

This work can be extended easily to Partially Observable Markov Decision Processes, using the framework in [19]. In this case, the additional goal is to build a good, predictive internal representation of the environment. Our theoretical framework can also be extended to continuous states and actions; very little work has been done so far in this direction [29] and none using rewards.

A third important direction for future work is empirical: we are currently evaluating the proposed method in comparison to existing exploration techniques, and experience in large domains will be especially necessary in the future.

7 Acknowledgements and historical note

This research was funded in part by NSERC and ONR. The paper was put into circulation as a pre-print starting in the summer of 2006.

References

1. N. Ay, N. Bertschinger, R. Der, F. Guttler, and E. Olbrich. Predictive information and explorative behavior of autonomous robots. *European Physical Journal B*, 63:329–339, 2008.
2. M. G. Azar and H.J. Kappen. Dynamic policy programming. *Journal for Machine Learning Research*, arXiv:1004.2027:1–26, 2010.
3. J. A. Bagnell and J. Schneider. Covariant policy search. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2003.
4. W. Bialek, I. Nemenman, and N. Tishby. Predictability, complexity and learning. *Neural Computation*, 13:2409–2463, 2001.
5. R. I. Brafman and M. Tennenholtz. R-max - a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, pages 213–231, 2002.
6. D. Chigirev and W. Bialek. Optimal manifold representation of data: An information theoretic perspective. In *Proceedings of NIPS*, 2004.
7. J. P. Crutchfield and D. P. Feldman. Synchronizing to the environment: Information theoretic limits on agent learning. *Adv. in Complex Systems*, 4(2):251–264, 2001.
8. E. T. Jaynes. Information Theory and Statistical Mechanics. *Physical Review*, 106(4):620–630, 1957.
9. M. Keanrs and S. Singh. Near-optimal reinforcement learning in polynomial time. In *Proceedings of ICML*, pages 260–268, 1998.
10. P-Y. Oudeyer P-Y, F. Kaplan, and V. Hafner. Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, 11(2):265–286, 2007.
11. J. Peters, K. Muelling, and Y. Altun. Relative entropy policy search. In *Proceedings of the Twenty-Fourth National Conference on Artificial Intelligence (AAAI)*, 2010.
12. J. Peters and S. Schaal. Reinforcement learning of motor skills with policy gradients. *Neural Networks*, 21(4):682–697, 2008.
13. B. Ratitch and D. Precup. Using MDP characteristics to guide exploration in reinforcement learning. In *Proceedings of ECML*, pages 313–324, 2003.
14. K. Rose. Deterministic annealing for clustering, compression, classication, regression, and related optimization problems. *Proc. IEEE*, 86(11):22102239, 1998.
15. J. Schmidhuber. Curious model-building control systems. In *Proceedings of IJCNN*, pages 1458–1463, 1991.
16. J. Schmidhuber. Art & science as by-products of the search for novel patterns, or data compressible in unknown yet learnable ways. In *Multiple ways to design research. Research cases that reshape the design discipline*, pages 98–112. Swiss Design Network - Et al. Edizioni, 2009.
17. C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, 1948.
18. Satinder Singh, Andrew G. Barto, and Nuttapong Chentanez. Intrinsically motivated reinforcement learning. In *Proceedings of NIPS*, pages 1281–1288, 2005.
19. S. Still. Statistical mechanics approach to interactive learning. *European Physics Letters*, 85, 2009.
20. S. Still and W. Bialek. How many clusters? an information theoretic perspective. *Neural computation*, 16:2483–2506, 2004.
21. S. Still, W. Bialek, and L. Bottou. Geometric clustering using the information bottleneck method. In *Proceedings of NIPS*, 2004.
22. Alexander L. Strehl, Lihong Li, and Michael L. Littman. Incremental model-based learners with formal learning-time guarantees. In *Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence*, 2006.
23. R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT Press, 1998.
24. M. E. Taylor and P. Stone. Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10(1):1633–1685, 2009.
25. S. Thrun and K. Moeller. Active exploration in dynamic environments. In *Advances in Neural Information Processing Systems (NIPS) 4*, pages 531–538, 1992.
26. N. Tishby, F. Pereira, and W. Bialek. The information bottleneck method. In *Proceedings of the 37th Annual Allerton Conference*, pages 363–377, 1999.
27. N. Tishby and D. Polani. Information theory of decisions and actions. In *Perception-reason-action cycle: Models, algorithms and systems*. Springer, 2010.
28. E. Todorov. Efficient computation of optimal actions. *PNAS*, 106(28):11478–11483, 2009.
29. D. Wingate and S. Singh. On discovery and learning of models with predictive representations of state for agents with continuous actions and observations. In *Proceedings of International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1128–1135, 2007.

Appendix

Clever random policy.

There are two world states, $x \in \{0, 1\}$ and a continuous action set, $a \in [0, 1]$. The value of the action sets how strongly the agent tries to stay in or leave a state, and $p(x'|x, a) = a$. The interest in reward is switched off ($\alpha = 0$), so that the optimal action becomes the one that maximizes only the predictive power.

–Policies that maximize $I[X_{t+1}, \{X_t, A_t\}] = H[X_{t+1}] - H[X_{t+1}|X_t, A_t]$.

For brevity of notation, we drop the index t for the current state and action.

$$I[X_{t+1}, \{X, A\}] = H[X_{t+1}] - H[X_{t+1}|X, A] \tag{24}$$

The second term in (24) is minimized and equal to zero for all policies that result in deterministic world transitions. Those are all policies for which $\pi(\tilde{a}|x) = 0$ for all $\tilde{a} \notin \{0, 1\}$. This limits the agent to using only two (the most extreme) actions: $a \in \{0, 1\}$. Since we have only two states, policies in this class are determined by two probabilities, for example the flip probabilities $\pi(A = 0|X = 1)$ and $\pi(A = 1|X = 0)$.

The first term in (24) is maximized for $p(X_{t+1} = 1) = p(X_{t+1} = 0) = 1/2$. Setting $p(X_{t+1} = 1)$ to $1/2$ yields

$$\pi(A = 0|X = 1)p(X = 1) + \pi(A = 1|X = 0)p(X = 0) = \frac{1}{2}. \tag{25}$$

We assume that $p(X = 0)$ is estimated by the learner. Eqn (25) is true *for all* values of $p(X = 0)$, if $\pi(A = 0|X = 1) = \pi(A = 1|X = 0) = 1/2$. We call this the "clever random" policy (π_R). The agent uses only those actions that make the world transitions deterministic, and uses them at random, i.e. it explores within the subspace of actions that make the world deterministic. This policy maximizes $I[X_{t+1}, \{X, A\}]$, independent of the estimated value of $p(X = 0)$.

However, when stationarity holds, $p(X = 0) = p(X = 1) = 1/2$, then all policies for which

$$\pi(A = 0|X = 1) = \pi(A = 0|X = 0) \tag{26}$$

maximize $I[X_{t+1}, \{X, A\}]$. Those include "STAY-STAY", and "FLIP-FLIP".

–Self consistent policies.

Since $\alpha = 0$, the term in the exponent of Eqn. (19), for a given state x and action a, is:

$$D^\pi(x, a) := \mathcal{D}[P(X_{t+1}|X_t = x, A_t = a)\|P(X_{t+1})] = H[a] + a \log \left[\frac{p(X_{t+1} = \bar{x})}{p(X_{t+1} = x)}\right] + \log[p(X_{t+1} = x)] \tag{27}$$

with $\bar{x}$ being the opposite state, and $H[a] = -(a \log(a) + (1 - a) \log(1 - a)$. Note that $H[0] = H[1] = 0$.

The clever random policy π_R is self-consistant, because under this policy, for all x, both actions, STAY ($a = 0$) and FLIP ($a = 1$) are equally likely. This is due to the fact that $p(X_{t+1} = x) = p(X_{t+1} = \bar{x}) = 1/2$, hence $D^{\pi_R}(x, 0) = D^{\pi_R}(x, 1), \forall x$.

If stationarity holds, $p(X = 0) = 1/2$, and no policy other than π_R, which uses only actions $a \in \{0, 1\}$ is self-consistent. This is because under other policies we also have that $p(X_{t+1} = x) = p(X_{t+1} = \bar{x}) = 1/2$, and we have $H[0] = H[1] = 0$; therefore,

$$D^\pi(x, 0) - D^\pi(x, 1) = H[A = 0] - H[A = 1] + \log[p^\pi(X_{t+1} = x)] - \log[p^\pi(X_{t+1} = \bar{x})] = 0. \tag{28}$$

This means that the algorithm gets to π_R after one iteration. We can conclude that π_R is the unique optimal self-consistant solution.

A reliable and an unreliable state.

There are two possible actions, STAY (s) or FLIP (f), and two world states, $x \in \{0, 1\}$, distinguished by the transitions: $p(X_{t+1} = 0|X_t = 0, A_t = s) = p(X_{t+1} = 1|X_t = 0, A_t = f) = 1$, while $p(X_{t+1} = x|X_t = 1, a) = 1/2, \forall x, \forall a$. In other words, state 0 is fully reliable, and state 1 is fully unreliable, in terms of the action effects.

The information is given by

$$I[X_{t+1}, \{X, A\}] = H[X_{t+1}] - H[X_{t+1}|X, A] = -p(X_{t+1}) \log_2[p(X_{t+1})] - p(X_t = 1) \tag{29}$$

Starting with a fixed value for $p(X_t = 1)$ which is estimated from past experiences, this is maximized by $p(X_{t+1} = 1) = 1/2$. We have $p(X_{t+1} = 0) = \pi(A = s|X = 0)p(X = 0) + \frac{1}{2}p(X = 1)$. Therefore, $p(X_{t+1} = 1) = 1/2 \Leftrightarrow \pi(A = s|X = 0) = 1/2$, which implies that asymptotically $p(X_t = 1) = 1/2$, and thus $I[X_{t+1}, \{X, A\}] = 1/2$. Asymptotically, $p(X_t = 0) = p(X_{t+1} = 0)$, and the information is given by $-(p(X = 0) \log_2[p(X = 0)/(1 - p(X = 0))] + \log_2[1 - p(X = 0)]) + p(X = 0) - 1$. Setting the first derivative, $1 - \log_2[p(X = 0)/(1 - p(X = 0))]$, to zero implies that the extremum lies at $p(X = 0) = 2/3$, where the information reaches $\log_2(3) - 1/3 \simeq 5/4$ bits. Now, $p(X_{t+1} = 0) = 2/3$ implies that $\pi(A = s|X = 0) = 3/4$. Asymptotically, the optimal strategy is to stay in the reliable state with probability $3/4$. We conclude that the agent starts with the random strategy in state 0, i.e. $\pi(A = s|X = 0) = 1/2$, and asymptotically finds the strategy $\pi(A = s|X = 0) = 3/4$. This asymptotic strategy still allows for exploration, but it results in a more controlled environment than the purely random strategy.