A Survey of Exploration Strategies in Reinforcement Learning

Roger McFarlane
McGill University
School of Computer Science
roger.mcfarlane@mail.mcgill.ca

Abstract

A fundamental issue in reinforcement learning algorithms is the balance between exploration of the environment and exploitation of information already obtained by the agent. This paper surveys exploration strategies used in reinforcement learning and summarizes the existing research with respect to their applicability and effectiveness.

1. Introduction

Reinforcement learning is, loosely defined, any problem in which an agent learns to control (or behave in) an unknown environment by interacting with that environment. Typically, the agent's performance is evaluated as it goes about acquiring "understanding" of its environment. Clearly, an important facet of this problem has to do with how the agent goes about the process of discovering its environment. This report surveys the common exploration strategies employed in reinforcement learning and summarizes their effectiveness and applicability.

There exists a large body of research pertaining to exploration in reinforcement learning, and more generally, to the problem of automated decision making in the presence of uncertainty. This report does not attempt to summarize or distil that wealth of information into a unified document. Rather, this report serves as an introductory overview of the broad classes and categories of approaches that are documented in the available literature.

The report begins with a brief introduction to the role of exploration in reinforcement learning. In particular, the trade-off between exploring the environment and exploiting that which the agent has already learned is discussed. This is followed by a catalogue overview of the commonly employed strategies and techniques. Finally, a summary of the merits and comparative effectiveness of each strategy is provided.

2. The Role of Exploration

This section provides an overview of the role exploration plays in reinforcement learning. Of particular interest is the manner in which exploration strategies balance the need to learn more about the environment with the agent's "desire" to perform well based on its current knowledge of the environment. (Thrun 1992a) poses several questions that are useful for framing this discussion.

- ❑ How can learning time be minimized?
- ❑ How can costs be minimized?
- ❑ What impact does the exploration strategy have on the speed and cost of learning?
- ❑ How does on trade off exploration and exploitation?

The first question is the central motivation for exploration. Clearly, the more effectively and efficiently an agent explores and learns from its environment, the more effectively and efficiently it uses its time, and hence, the less time is required for learning. Intuitively, one might infer that pure exploration during learning would be the most effective learning approach. However, this is not the case. Pure exploration may waste much time and computing resources exploring task-irrelevant parts of the environment. This also means that the agent's performance during learning will be poor, relatively speaking, since it spends some unnecessary portion of its time performing actions that do not help it achieve its goals.

Note that our intuition for exploration included notions of efficacy and efficiency. The second question embodies the issue of effective and efficient exploration. Exploration is an attempt to maximize knowledge gain. This question constrains the cost at which that knowledge is acquired. Typically, questions one and two have opposing answers: the smaller the learning time, the larger the cost (i.e., the poorer the performance of the agent during its learning phase), and vice versa.

As it turns out, it is often advantageous to find some suitable balance of both exploration and exploitation. If the agent can exploit its current knowledge of the environment, it may be able to identify the most worthwhile parts of the environment to explore. Further, minimizing the learning costs (i.e., maximizing the agent's performance during learning) cannot be done without some degree of exploration of the environment such that effective behaviours can be discovered.

The third question focuses on the applicability of the exploration strategy to the task at hand. An exploration strategy that performs well in one environment may be ill-suited to another environment. (Thrun, 1992a) observed that the impact of the exploration technique on both learning time and learning costs can be "enormous."

The fourth question considers the precise bias or tendency to exploration or exploitation that is suitable for the agent. Most of the strategies discussed below can be tuned to prefer exploration or exploitation to varying degree. In particular, it leads to the question of how best to combine these two concepts.

Numerous articles, papers, and theses have been published which focus on potential solutions to the effectively answering the preceding questions. The remainder of this report provides a representative sampling of these strategies.

3. Exploration Strategies

This section provides a catalogue of the commonly employed exploration strategies. The techniques outlined below can be divided into two categories: undirected and directed. Directed exploration strategies consider the agents history in the environment in order to influence the parts of the environment that the agent will further explore. Undirected exploration techniques randomly explore the environment without consideration of the previous history of the learning process.

The following sections describe:

1. Undirected exploration
2. Directed exploration based on the observing the learning process
3. Other directed techniques based on alternative learning models

In each case, the general idea behind each approach is described and a brief commentary on its efficacy is provided.

3.1 Undirected Exploration

The primary characteristic of undirected exploration is that actions are generated based on some random distribution and does not take into consideration the learning process itself. As a means of dealing with the trade-off between exploration and exploitation, undirected methods often modify the probability distribution from which actions are randomly selected.

In this section, we describe three undirected exploration strategies: Random Exploration, Semi-Uniform Distributed Exploration and Boltzmann Distributed Exploration. Although these strategies are often used, it has been demonstrated (Thrun, 1992b) that, as a strategy class, undirected exploration techniques are inferior to directed exploration techniques, discussed in Sections 3.2 and 3.4.

Throughout this discussion of undirected exploration strategies, let the exploitation measure *f(a)* of an action be defined by the following formula, where *s* is the current state and $\widehat{V}(x)$ is the current estimate for the value of state *x*.

$$f(a) = \sum_{s' \in States} P_{s \to s'}(a) \cdot \hat{V}(s')$$

3.1.1 Random Exploration

One approach for exploring the state space is to generate actions randomly with uniform probability. This method might be used of the cost of exploration during learning is not under consideration. For example, if the task to be learned is divided into a learning phase and a performance phase and the cost during learning is being ignored, then this method may be applicable. However, in many situations, the agent's performance during learning is an important facet of the problem formulation. This method is not well suited to such problems; purely random exploration is the least efficient exploration method in terms of costs (Thrun, 1992b).

3.1.2 Semi-Uniform Distributed Exploration

Rather than generating actions randomly with uniform probability, an alternative would be to generate actions with a probability distribution that is based on the utility estimates that are currently available to the agent. One such approach selects the action having the highest current utility estimate with some predefined probability P_{best}. Each of the other actions is selected with probability $1 - P_{best}$ regardless of its currently utility estimate.

$$P(a) = \begin{cases} P_{best} + \dfrac{1 - P_{best}}{\#\text{ of actions}} & \text{, if } a \text{ maximizes } f(a) \\ \dfrac{1 - P_{best}}{\#\text{ of actions}} & \text{, otherwise} \end{cases}$$

The P_{best} parameter facilitates a continuous blend of exploration and exploitation that ranges from purely random exploration ($P_{best} = 0$) to pure exploitation ($P_{best} = 1$).

This method has been found to outperform purely random exploration. It has been also found to outperform the more sophisticated Boltzmann-distributed exploration explained next (Thrun, 1992a).

3.1.3 Boltzmann-Distributed Exploration

Whereas semi-uniform exploration considers the utility of the actions only to select the action to which the fixed probability P_{best} should be assigned, Boltzmann-distributed exploration considers the estimated utility of all actions using the following equation.

$$P(a) = \frac{e^{f(a)\cdot\theta^{-1}}}{\sum_{a' \in \text{Actions}} e^{f(a')\cdot\theta^{-1}}}$$

The gain factor $\theta > 0$, also called the temperature, determines the amount of randomness used in the action selection procedure. As $\theta \to 0$ the system tends to pure exploitation. As $\theta \to \infty$ the resulting distribution approaches the uniform distribution (i.e., random exploration). θ can be gradually decreased over time in order to decrease exploration.

This method works well if the best action is well separated from the others, but suffers somewhat when the values of the actions are close; it may also converge unnecessarily slowly unless the temperature schedule is manually tuned with great care (Kaelbling et al, 1996).

3.2. Basic Directed Exploration

In contrast to the undirected exploration techniques discussed above, directed exploration strategies retain knowledge of the learning process itself. This retained knowledge is used to guide the exploration process.

In this section, we discuss four basic directed exploration strategies: Counter-based exploration, Counter-based exploration with decay, Counter/Error-based exploration and Recency-based exploration. It has been shown (Thrun, 1992b) that directed exploration strategies are inherently superior to undirected strategies.

As for the discussion for undirected exploration strategies, let the exploitation measure *f(a)* of an action be defined by the following formula, where *s* is the current state and $\widehat{V}(x)$ is the current estimate for the value of state *x*.

$$f(a) = \sum_{s' \in States} P_{s \to s'}(a) \cdot \hat{V}(s')$$

3.2.1 Counter-Based Exploration

In Counter-based exploration, the number of visits to each state *s* is maintained. Actions are evaluated using a combination of the exploitation value and an exploration term. One such combination is given by the following formula (Thrun, 1992b) in which the exploration term is the quotient of the counter value for the current state and the expected counter value for the state that results from taking an action. Other possible formulations use the difference between the counter value for the current state and the expected counter value for the state that results from taking an action.

$$eval_c(a) = \alpha \cdot f(a) + \frac{c(s)}{E[c \mid s, a]}$$

$$E[c \mid s, a] = \sum_{s} P_{s \to s'}(a) \cdot c(s')$$

In the given equation, $\alpha \geq 0$ is constant factor weighting exploration versus exploitation. The action having the maximum evaluation is chosen. Refer to (Sato et al, 1990) and (Barto and Singh, 1990) for related counter-based exploration strategies.

In general, counter-based exploration evaluates states on the basis of how often then occur (Thrun, 1992b). Ideally, however, one would prefer to visit states that yield the best performance improvement. The decay and error based strategies are modifications to this end.

3.2.2 Counter-Based Exploration with Decay

The basic counter-based strategy can be augmented to more accurately reflect the state of exploration by also storing information regarding when a state occurred. For example, if two states have been visited equally often, but one of the two was late visited very early in the exploration process, it makes intuitive sense to prefer to revisit that earlier state to see of what has since been learned improves the agent's performance from that state. This intuitive heuristic, which is more directly applied in recency-based exploration described below, leads to an extension to counter-based exploration with decay. At each timestep, every counter is multiplied by a fixed decay $\lambda \leq 1$.

$$c(s) \leftarrow \lambda \cdot c(s) \quad \forall s$$

Given an appropriately chosen λ the resulting decayed counter values more effectively reflect the utility values than un-decayed counter values (Thrun, 1992b).

3.2.3 Counter/Error-Based Exploration

An extension to the counter-based exploration strategy is to guide the exploration based on the changes in the state utility estimates (Schmidhuber, 1991), (Thrun and Möller, 1991), (Thrun and Möller, 1992). Intuitively, the action having the greatest impact on the current estimates (i.e., the action from which the agent will learn the most) is the most attractive choice to make. The resulting idea of the heuristic is to prefer actions leading to states whose estimate has recently changed the most.

$$eval_c(a) = \alpha \cdot f(a) + \frac{c(s)}{E[c \mid s,a]} + \beta \cdot E\left[\Delta\hat{V}_{last} \mid s,a\right]$$

In the above formula, $\beta > 0$ is a constant factor that determines the weight with which to consider the error-heuristic and $E[\Delta\hat{V}_{last} \mid s,a]$ is the expected change in value which will result from taking action a from state s.

3.2.4 Recency-Based Exploration

(Sutton, 1990) describes an alternative to basic counter-based exploration. As mentioned above, the basic idea is to prefer visiting states that have not been observed recently. To facilitate this, an additional value $\rho(s)$ denoting the last visit time is associated with each state. The difference between the current time and $\rho(s)$ measures how recently the state was last observed. Based on this value, actions are evaluated using the following formula.

$$eval_r(a) = \alpha \cdot f(a) + \sqrt{E[\rho \mid s,a]}$$

As before, $\alpha \geq 0$ is a constant weight factor and the action having the highest evaluation is selected. $E[\rho|s,a]$ is the expected recency or the resulting state. Note that this strategy behaves very differently than the counter-based techniques. For example, if a state has occurred very few times, but has occurred very recently, actions yielding this state are not likely to be selected.

3.2.5 Local vs. Non-Local

A differentiating distinction of directed exploration strategies is the amount of global knowledge used for exploration. Local exploration techniques (a family including all of those mentioned above) select actions based on the immediate usefulness expected of the next observed state. On the other hand, non-local exploration considers the future usefulness of exploration within some time horizon. (Sutton, 1990) describes a technique, based on dynamic programming, for using knowledge from the entire state-action space to guide exploration in finite domains. This technique may be applied to the previously described strategies to extend them to non-local exploration (Thrun, 1992a).

Non-local exploration techniques are typically more complex in time due to their need to perform additional planning. Most of the surveyed literature, and in turn, this paper, focuses on local exploration.

3.3 Other Directed Strategies

There have recently been a number of other exploration strategies that consider exploration as an inherent part of the optimal behaviour problem. In these strategies, the underlying learning model considers exploring the environment. This section surveys a number of different approaches that are described in recent literature. In particular, many of these techniques consider, either directly or indirectly, the global state-action space.

3.3.2 Competence Maps

(Thrun and Moller, 1992) use an auxiliary data structure which estimates the degree to which the agent believes it has sufficient knowledge to make good decisions in that regions of the environment. This estimate is used for exploring the world by selecting actions that lead the agent into areas of the environment in which it believes it has the least competence. Exploration and exploitation are balanced by alternating between pure exploitation and pure exploration.

Experimental results on the part of (Thrun and Moller, 1992) demonstrate that this method is as effective in discovering near optimal policies as the comparison algorithms. It has the additional advantage of finding multiple near-optimal policies in those cases where more than one such policy exists.

3.3.3 Interval Estimation

(Kaelbling, 1993) describes an application of second order statistics to infer the potential with which a given action belongs to the optimal policy. Rather than simply learning an approximation of an actions value, interval estimation methods learn that an actions value falls within some upper and lower bound with a degree of confidence. The action selected by this method is the one having the confidence interval has the highest upper limit.

Unfortunately, interval estimation methods are problematic in practice because of the complexity of the statistical methods used to estimate the confidence intervals; moreover, the underlying statistical assumptions required by these methods are often not satisfied. (Sutton and Barto, 1998).

3.3.1 Multiple-Hypothesis Heuristics

(Santamaría and Ram, 1997) describe a modification to the Q-Learning algorithm in which an additional level of indirection is introduced between the update function and the state representation. Their heuristic approach is to modify an approximation of the agent's belief about the environment, which itself sits on top of the existing Q-Learning approximation of the state-action value function/table. During learning, the state-action values are modified, as well as the belief layer through which the learning algorithm interacts with the state-action values.

Experimental results provided by (Santamaría and Ram, 1997) indicate that the approach is computationally feasible and that it works effectively in the specific test environment.

There is little additional experimental data regarding this approach. As such, this strategy can, at best, be considered promising.

3.3.4 Model Based Exploration

(Weiring and Schmidhuber, 1998) extent interval estimation methods to apply to model based reinforcement problems. In these problems, the agent attempts to construct a model of how to effectively explore its environment. Essentially, the agent constructs an estimate of how useful improved knowledge of some part of the environment will be to its performance.

In their experiments, the researches report significant success using a hybrid approach. The agent begins to explore using a basic interval estimation method and, after learning enough to establish a model, continues with their model-based method.

3.3.5 Dynamic Programming

An additional family of approaches are closely related to dynamic programming. (Sutton, 1990) describes the DYNA-Q algorithm which implements model based Q-Learning. (Meuleau and Bourgine, 1998) extend these ideas in order to achieve global-scale reasoning about the agent's uncertainty about the environment using the following key concepts.

- ❑ Exploration Bonuses
  These allow for the quantification of uncertainty in the same units as rewards and make explicit the rationale behind the choice of a non-optimal action.
- ❑ Back-propagation of Exploration Bonuses
  This allows for "intelligent" and complete exploration of the environment using only local measures of uncertainty.

(Dayan and Sejnowski, 1996) also builds on these ideas in order to translate the uncertainty model into a system of exploratory behaviour using exploration bonuses.

These ideas, which combine model-based exploration with the more established strategies, demonstrate very good performance in the literature (Meuleau and Bourgine, 1998). However, the documented trials are for fairly simple, finite environments; the practical scalability is not yet proven (Kaelbling et al, 1996), (Meuleau and Bourgine, 1998), (Dayan and Sejnowski, 1996).

4. Discussion

There exists a wide variety of reinforcement learning techniques. Subservient to these are myriad exploration strategies, as described in previous sections. One of the key issues is the scalability of these solutions to larger, more complex, problems (Kaebling et al., 1996). In particular, many recent approaches are moving away from the tabular approach, on which much of the theory is based, to a function approximation method (e.g., neural networks, decision trees, tile coding, etc).

(Thrun, 1992a) demonstrates that directed techniques are inherently superior to undirected techniques. However, almost of the directed techniques seek to explore the entire state-action space. For even moderately complex problems the state-action space is too large to conduct an exhaustive search. The more recent approaches that attempt to locate "interesting" regions of the state-action space for exploration are extremely promising in this regard.

The experimental results of (Meuleau and Bourgine, 1998) demonstrate that the following classification of exploration strategies in order of increasing efficiency and efficacy:

1. Undirected exploration
2. Directed, local exploration
3. Directed, global exploration

These observations are consistent with the general trends observable throughout the literature.

5. Conclusion

A fundamental issue in reinforcement learning algorithms is the balance between exploration of the environment and exploitation of information already obtained by the agent. This report has surveyed exploration strategies used in reinforcement learning and summarized the existing techniques with respect to their applicability and effectiveness.

As discussed, directed techniques utilizing global exploration strategies have been shown to be the most effective and efficient class of strategies currently available.

References

(Dayan and Sejnowski, 1996) P Dayan and T. J. Sejnowski. "Exploration bonuses and dual control". *Machine Learning*, 25:5--22. 1996.

(Dearden et al., 1998) R. Dearden, N. Friedman, and S. Russell. "Bayesian Q-Learning." In *AAAI-98*, pages 761–68 . AAAI Press, 1998.

(Kaelbling, 1993) L. Kaelbling. *Learning in Embedded Systems*. MIT Press. 1993.

(Kaelbling et al, 1996) L. P. Kaelbling, M. L. Littman and A.W. Moore, (1996). "Reinforcement learning: a survey", *Journal of Artificial Intelligence Research, Vol. 4*, pp. 237—285.

(Kearns and Singh, 1998) M Kearns and S. Singh. "Near-optimal reinforcement learning in polynomial time." In *ICML-98*, 260–268. 1998.

(Meuleau and Bourgine, 1998) N. Meuleau and P. Bourgine. "Exploration of multi-state environments: Local measures and back-propagation of uncertainty". *Machine Learning*. 1998.

(Sathiya Keerthi and Ravindran 1995). S. Sathiya Keerthi and B. Ravindran. “A tutorial survey of reinforcement learning.” Published by the Indian Academy of Sciences. 1995.

(Santamaría and Ram, 1997) J. C. Santamaría and A Ram. “A New Heuristic Approach for Dual Control.” In *Proceeding of the AAAI-97 Workshop on On-Line Search*, Providence, Rhode Island, July 1997.

(Sato et al, 1990) M. Sato, K. Abe and H Takeda. “Learning Control of Finite Markov Chains with Explicit Trade-off between Estimation and Control.” In D.S et al. Touretzky, editor, *Connectionist Models, Proceedings of the 1990 Summer School*, pages 287-300, San Mateo, CA. 1990. Morgan Kaufmann.

(Schmidhuber, 1991) J.H. Schmidhuber. “Adaptive confidence and adaptive curiosity.” *Technical Report FKI-149-91.* Technische Universität München. 1991.

(Sutton, 1990) 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, June 1990*, pages 216-224, 1990.

(Sutton and Barto, 1998) R. S. Sutton and A. G. Barto. *Reinforcement Learning*. MIT Press. 1998.

(Thrun and Moller, 1991) S. B. Thrun and K. Moller. “On planning and exploration in non-discrete environments.” *Technical Report 528*, GMD, Sankt Augustin, FRG. 1991.

(Thrun and Moller, 1992) S. B. Thrun and K. Moller. “Active exploration in dynamic environments.” In, JE Moody, SJ Hanson & RP Lippmann editors, *Advances in Neural Information Processing Systems, 4*, 531–538. San Mateo, CA: Morgan Kaufmann. 1992.

(Thrun, 1992a) S. B. Thrun. “The role of exploration in learning control.” In DA White & DA Sofge, editors, *Handbook of Intelligent Control: Neural, Fuzzy and Adaptive Approaches*. New York, NY: Van Nostrand Reinhold. 1992.

(Thrun, 1992b) S. B. Thrun. “Efficient exploration in reinforcement learning.” *Technical Report CMU-CS-92-102*, School of Computer Science, Carnegie Mellon University. 1992.

(Thrun 1993) S. B. Thrun. “Exploration and model building in mobile robot domains.” *Proceedings of the International Conference on Neural Networks* (pp. 175–180). San Francisco, CA: IEEE Neural Network Council. 1993.

(Wiering and Schmidhuber, 1998) M. A. Wiering and J. Schmidhuber. “Efficient model based exploration.” In J. A. Meyer and S. W. Wilson, editors, *Proceedings of the Sixth International Conference on Simulation of Adaptive Behaviour: From Animals to Animats 6*, pages 223–228. MIT Press/Bradford Books. 1998.

(Wyatt, 1997) J. Wyatt. “Exploration and Inference in Learning from Reinforcement.” PhD thesis, Department of Artificial Intelligence, University of Edinburgh. 1997