# A way around the exploration-exploitation dilemma

**Erik J Peterson (erik.exists@gmail.com)**
Carnegie Mellon, Dept. of Psychology
5000 Forbes Ave.
Pittsburgh, PA 15213

**Timothy Verstynen (timothyv@andrew.cmu.edu)**
Carnegie Mellon, Dept. of Psychology
5000 Forbes Ave.
Pittsburgh, PA 15213

## Abstract

**The exploration-exploitation dilemma is considered a fundamental but intractable problem in the learning and decision sciences. Here we challenge the basic formulation of the dilemma by defining independent mathematical objectives for exploration and exploitation, rather than having them share a single objective. This dual formulation greedy algorithm that provably and optimally maximizes both information and reward value.**

**Keywords:** reinforcement learning; exploration; theory

## Introduction

The exploration-exploitation dilemma is a fundamental but intractable problem in the learning and decision sciences (Gupta, Smith, & Shalley, 2006; Berger-Tal, Nathan, Meron, & Saltz, 2014; Sutton & Barto, 2018; Thrun, 1992; Dayan & Sejnowski, 1996; Findling, Skvortsova, Dromnelle, Palminteri, & Wyart, 2018; Gershman, 2018). In the problem, the actions taken by an agent (e.g., rodent, human, computer algorithm) are based on a set of learned values (Sutton & Barto, 2018; Roughgarden, 2019). Exploitation is defined as choosing the most valuable action. Exploration is defined as simply making any other choice. When an only one action can be taken, there must be a trade-off between exploitation and exploration. This trade-off is not a dilemma.

To become a dilemma–and a fundamental problem–the trade-off must satisfy two more conditions. The first is a shared objective. For example, during reinforcement learning exploration and exploitation both attempt to maximize reward (e.g., food, water) (Sutton & Barto, 2018). The second is partial observability. For example, if a mouse is placed in maze with four arms it might know that down one arm is a small pellet of food, but have no knowledge of the other arms. In this condition choosing to explore might lead to more chow, or to less. Limited knowledge on a common objective defines the dilemma. It also makes a direct mathematical solution to the problem intractable (Thrun, 1992; Dayan & Sejnowski, 1996; Findling et al., 2018; Gershman, 2018).

But is the dilemma really fundamental?

## Results

In the natural world exploration can have two different explanations. If there is no reason to expect a reward, exploration is described theoretically as a search for novel or maximum information (Mehlhorn et al., 2015; Pathak, Agrawal, Efros, & Darrell, 2017; Wang & Hayden, 2019; Calhoun et al., 2015; J. Schmidhuber, 2008; Schwartenbeck et al., 2019). For example, when a mouse is placed in a maze it has never visited before it will explore extensively even if no tangible rewards are present (Mehlhorn et al., 2015). On the other hand if reward is expected exploration gets re-interpreted as a reinforcement learning problem (Sutton & Barto, 2018).

Here we conjecture *all* exploration needs only a single explanation based on maximum information. This simple conjecture let us define independent mathematical objectives for exploration and exploitation. Unlike dilemma, solving these independent objectives is quite tractable.

Our contribution is threefold. We first offer five axioms that serve as a basis to estimate the value of any information. Next we prove the computer science method of dynamic programming (Bellmann, 1954; Sutton & Barto, 2018) provides an optimal way to maximize this information value. Finally, we describe a simple greedy scheduling algorithm (Roughgarden, 2019) that can maximize both information value and reward value.

### Information is not a reward

In principle there is no need for exploration and exploitation to share a common objective; we know exploration happens without reward to motivate it (Mehlhorn et al., 2015; Gupta et al., 2006; Berger-Tal et al., 2014; Schwartenbeck et al., 2019; Pathak et al., 2017).

In practice reward and information contain at least two fundamentally distinct ideas. First, rewards are a conserved resource. Information is not. For example, if a mouse shares a potato chip with a cage-mate, she must break the chip up leaving less food for herself. Second, reward value is constant during learning while the value of information must decline[1]. For example, if a mouse learns where a bag of potato chips is it will generally keep going back and eating more. Whereas if a student learns the capital of the United States, there is no value to the student in being told again that the capital of the United States is Washington DC.

[1] Assuming a stable environment

## A definition of information value

To make our eventual solution to the exploration-exploitation trade-off general, we separate reward value from information axiomatically.

To form a basis for our axioms, we reasoned that the value of any observation $s$ made by an agent (e.g., rodent, human, computer algorithm) depends entirely on what the agent learns by making that observation. In other words, how it changes the agent's memory. A memory in our definition consists of only an (invertible) encoder, a decoder, and the mathematical idea of a set.

We let $M$ be finite set, whose maximum size is $N$ (denoted $M^N$). This memory $M$ learns by making observations $s$ from a finite state space $s \in S^N$. Learning in $M$ happens by an invertible encoder $f$, such that $M_{t+1} = f(M_t, s)$ and $M_t = f^{-1}(M_{t+1}, s)$. Memories $z$ are recovered from $M$ by a decoder $g$, such that $z = g(M, s)$. For simplicity the initial memory $M_0$ is an empty set, $M_0 = \emptyset$.

This definition of memory is extremely general–covering all modes of working or episodic memory (Miller, 1956; Tulving, 2002), probabilistic memory (Park & Pillow, 2017; Itti & Baldi, 2009; Friston et al., 2016), and even memories based on compression or latent-states (Kingma & Welling, 2013; Ganguli & Sompolinsky, 2010).

We could have used state prediction (Schmidhuber, 1991; Schwartenbeck et al., 2019; Pathak et al., 2017), information theory (Polani, Martinetz, & Kim, 2001; Kolchinsky & Wolpert, 2018), or Bayesian learning (Itti & Baldi, 2009; Park & Pillow, 2017; Friston et al., 2016) to estimate the value of information. Each of these though require strong assumptions which are formidable if it turns out–for example–animals are not in fact general Bayesian reasoning systems.

Our goal was to describing behavior across the natural world, as well as to model behavior in artificial environments. This lead us to work with high level of mathematical abstraction, and to develop our general notion of memory and to working from simple axioms. As a result to these choices, the theory we develop applies to *all* of these more standard approaches.

### Axiomatic information value ($E$)

**Axiom 1** (Axiom of Now). *$E$ is a distance $\Delta M$ between $M_{t+1}$ and $M_t$.*

That is, the value of an observation $s$ depends only on much how current memory changes.

**Axiom 2** (Axiom of Novelty). *$E = 0$ if and only if $\Delta M = 0$.*

An observation that doesn't change the memory has no value.

**Axiom 3** (Axiom of Scholarship). *$E \geq 0$.*

In principle all new information is valuable, even if the consequences later on are negative.

**Axiom 4** (Axiom of Specificity). *$E$ is monotonic with the compactness $C$ of $\Delta M$ (Eq. 8).*

More specific information is more valuable than less specific information. (We use the mathematical idea of compactness to formalize specificity.)

**Axiom 5** (Axiom of Equilibrium). $\frac{\Delta^2 M}{\Delta\theta^2} < 0$

The environment must learnable by $M$. (Learnable in the sense of in Valiant's definition of probably approximately correct algorithms (Valiant, 1984)).

## Exploration as a dynamic programming problem

An useful solution for learning to maximize information value, $E$, would be a dynamic programming solution. Dynamic programming guarantees total cumulative value is maximized by a simple, deterministic, and greedy algorithm.

In Theorem 1 we prove our definition of memory (above) has one critical property, optimal substructure, that is needed for a dynamic programming solution (Bellmann, 1954; Roughgarden, 2019). The other two properties, $E \geq 0$ and the Markov property (Bellmann, 1954; Roughgarden, 2019), are fulfilled by the Axioms 3 and 1 respectively.

To write down the Bellman solution for $E$ as a dynamic programming problem we need some new notation. Let $a$ be an action drawn from a finite action space $A^K$. Let time $t$ denote an index $t = (1, 2, 3, \dots, \infty)$. Let $\pi$ denote any policy function that maps a state $s$ to an action $a$, $\pi : s \to a$, such that $\forall s_t \in S, \forall a_t \in A$. Let $\delta$ be a transition function which maps $(s_t, a_t)$ to a new state $s_{t+1}$, $\delta : (s_t, a_t) \to s_{t+1}$ where, once again, $\forall s_t \in S, \forall a_t \in A$.

For simplicity we redefine $E$ as $F(M_t, a_t)$–a "payoff function" in dynamic programming (Eq 1).

$$\begin{aligned} F(M_t, a_t) &= E(M_{t+1}, M_t) \\ &\text{subject to the constraints} \\ a_t &= \pi(s_t) \\ s_{t+1} &= \delta(s_t, a_t), \\ M_{t+1} &= f(M_t, s_t) \end{aligned} \tag{1}$$

The value function for $E$ is Eq 2.

$$V_{\pi_E}(M_0) = \left[ \max_{a \in A} \sum_{t=0}^{\infty} F(M_t, a_t) \,\middle|\, \pi_E,\, M \right] \tag{2}$$

To find the recursive Bellman solution we decompose Eq. 2 into an initial value $F_0$ and the remaining series in the summation. When this decomposition is applied recursively (Eq 4) we arrive at an iterative optimal and greedy solution (Eq. 3).

$$\begin{aligned} V^*_{\pi_E}(M_0) &= \max_{a \in A} \left[ \sum_{t=0}^{\infty} F(M_t, a_t) \right] \\ &= \max_{a \in A} \left[ F(M_0, a_0) + \sum_{t=1}^{\infty} F(M_{t+1}, a_{t+1}) \right] \\ &= F(M_0, a_0) + \max_{a \in A} \left[ \sum_{t=1}^{\infty} F(M_{t+1}, a_{t+1}) \right] \\ &= F(M_0, a_0) + V^*_{\pi_E}(M_{t+1}) + V^*_{\pi_E}(M_{t+2}),\ \dots \end{aligned} \tag{3}$$

$$V^*_{\pi_E}(M_t) = F(M_t, a_t) + \max_{a \in A} \Big[ F(M_{t+1}, a_t) \Big] \tag{4}$$

## A note on exploration quality

Having a policy $\pi^*_E$ that maximizes $E$ does not necessarily ensure exploration is of good quality. Fortunately $\pi^*_E$ also ensures exhaustive exploration of any *finite* space $S$ (Theorems 2 and 3).

## Scheduling a way around the dilemma

The objective of reinforcement learning is maximize reward value $V_R$ using a action policy, which we'll denote $pi_R$ (Eq. 5).

$$V_R(a_t) = \sum_{t=0}^{\infty} R_t \Big| \, a_t \sim \pi_R \tag{5}$$

Any reinforcement learning algorithm (Sutton & Barto, 2018) that operates on a discrete state-space is compatible with our approach. Here we use $\pi_R$ to denote any such suitable algorithm.

We imagine the policies for exploration and exploitation acting as two possible "jobs" competing for control of this fixed behavioral resource. By definition each of these "jobs" naturally produce non-negative values that an optimal job scheduler could use: $E$ for information or $R$ for reward/reinforcement learning. Each of the "jobs" takes a constant amount of time and each policy can only take one action a time (by definition).

The solution to scheduling problems that produce non-negative values and have fixed run times is known to be a simple greedy algorithm (Roughgarden, 2019). We restate this solution as a set of inequalities controlling the action policy, $\pi_\pi$ (Eq. 6).

$$\pi_\pi = \begin{cases} \pi^*_E & : E_t - \varepsilon > R_t \\ \pi_R & : E_t - \varepsilon \leq R_t \end{cases}$$

$$\text{subject to the constraints} \tag{6}$$

$$R \in \{0, 1\}$$

$$p(R) < 1$$

**The fine print** The inequalities in Eq 6 ensure total value summed over $E_t$ and $R_t$ is maximized (Theorem 4). They do not ensure each policy maintains its individual optimality. To do this we introduce two mild but sufficient constraints the reward distribution (Theorem 4, Eq 6).

In stochastic environments learning in $M$ may show small continual fluctuations. These will often get reflected in $E$. To allow Eq. 6 to achieve a stable solution we introduce $\varepsilon$, a boredom threshold for exploration. When $E_t < \varepsilon$ we say the policy is "bored" with exploration. To be consistent with the value axioms, $\varepsilon > 0$ and $E + \varepsilon \geq 0$.

The initial value $E_0$ for $\pi^*_E$ can be arbitrary with the limit $E_0 > 0$. In theory $E_0$ does not change $\pi^*_E$'s long term behavior, but different values will change the algorithm's short-term dynamics and so might be quite important in practice. By definition a pure greedy policy, like $\pi^*_E$, can't handle ties. There is simply no mathematical way to rank equal values. Theorems 3 and 2 ensure that any tie breaking strategy is valid. However like the choice of $E_0$, tie breaking can strongly effect the transient dynamics.

To ensure the default policy is reward maximization, Eq. 6 breaks ties between $R_t$ and $E_t$ in favor of $\pi_R$.

## Exploration without regret

In reinforcement learning it is common to use regret $G$ to quantify the cost or benefit of exploration (Eq. 7).

$$G_t = V^* - V_{Et} \tag{7}$$

Where $V^*$ represents the value of a pure exploitation, and $V_{Et}$ represents the value of some exploratory action at time $t$.

An ideal solution to the dilemma would ensure that for all observation times $G_t > 0$. While such an ideal solution is in practice intractable, we can partially state the properties of any ideal solution.

As long as some actions are more valuable that others, any stochastic exploration policy will cause to some $G_t < 0$. So, an ideal policy must be deterministic.

By definition if $G_t > 0$ for all $t$ then each action $a_t$ must be greedy. That is $a_t$ is not greedy, it possible that $G_t$ becomes $< 0$. So an ideal policy must be greedy.

Given the dilemma requires partial observability (by definition) coupled with the fact that you can only overcome partial observability with randomness, but that any random policy ensures $G_t < 0$ for some $t$, the only candidate for an ideal solution can come from modifying the dilemma problem itself. In terms of regret then, our approach ensures $G_t > 0$ for all $t$ and so is an ideal solution for exploration and is as result–as close as possible–to an ideal to the dilemma.

## Algorithmic complexity

Computational complexity is an important concern for any learning algorithm (Valiant, 1984). The worst case run time for $\pi_\pi$ is linear and additive in its policies. That is, if in isolation it takes $T_E$ steps to earn $E_T = \sum_{T_E} E$, and $T_R$ steps to earn $r_T = \sum_{T_R} R$, the worst case training time for $\pi_\pi$ is $T_E + T_R$. This in only true though if neither policy can learn from the other's actions. There is however no reason each policy can't observe the transitions $(s_t, a_t, R, s_{t+1})$ caused by the other. If this is allowed, worst case training time improves to $\max(T_E, T_R)$.

# Summary

The exploration-exploitation dilemma is only a dilemma if we assume both exploration and exploitation must have the same objective. By recognizing that exploration can be formalized, quite generally, on its own we've found a way around the dilemma.

Exploration in this solution aims to maximize information value. This policy allows an animal to learn a general and reward-independent memory of the world. (General learning like this is critical for long-term planning (Sutton & Barto, 2018).) An important "side effect" of learning a general memory is that reward or reinforcement learning performances must also improve, to optimality.

## References

Bellmann, R. (1954). The theory of dynamic programming. *Bull. Amer. Math. Soc*, *60*(6), 503-515.

Berger-Tal, O., Nathan, J., Meron, E., & Saltz, D. (2014, April). The Exploration-Exploitation Dilemma: A Multidisciplinary Framework. *PLoS ONE*, *9*(4), e95693. doi: 10.1371/journal.pone.0095693

Calhoun, A. J., Tong, A., Pokala, N., Fitzpatrick, J. A., Sharpee, T. O., & Chalasani, S. H. (2015, April). Neural Mechanisms for Evaluating Environmental Variability in Caenorhabditis elegans. *Neuron*, *86*(2), 428-441. doi: 10.1016/j.neuron.2015.03.026

Dayan, P., & Sejnowski, T. J. (1996, October). Exploration bonuses and dual control. *Machine Learning*, *25*(1), 5-22. doi: 10.1007/BF00115298

Findling, C., Skvortsova, V., Dromnelle, R., Palminteri, S., & Wyart, V. (2018, October). *Computational noise in reward-guided learning drives behavioral variability in volatile environments* (Preprint). Neuroscience. doi: 10.1101/439885

Friston, K., FitzGerald, T., Rigoli, F., Schwartenbeck, P., ODoherty, J., & Pezzulo, G. (2016, September). Active inference and learning. *Neuroscience & Biobehavioral Reviews*, *68*, 862-879. doi: 10.1016/j.neubiorev.2016.06.022

Ganguli, S., & Sompolinsky, H. (2010). Short-term memory in neuronal networks through dynamical compressed sensing. , 9.

Gershman, S. J. (2018, April). Deconstructing the human algorithms for exploration. *Cognition*, *173*, 34-42. doi: 10.1016/j.cognition.2017.12.014

Gupta, A. A., Smith, K., & Shalley, C. (2006). The Interplay between Exploration and Exploitation. *The Academy of Management Journal*, *49*(4), 693-706.

Itti, L., & Baldi, P. (2009, June). Bayesian surprise attracts human attention. *Vision Research*, *49*(10), 1295-1306. doi: 10.1016/j.visres.2008.09.007

Kingma, D. P., & Welling, M. (2013, December). Auto-Encoding Variational Bayes. *arXiv:1312.6114 [cs, stat]*.

Kolchinsky, A., & Wolpert, D. H. (2018, December). Semantic information, autonomous agency and non-equilibrium statistical physics. *Interface Focus*, *8*(6), 20180041. doi: 10.1098/rsfs.2018.0041

Mehlhorn, K., Newell, B. R., Todd, P. M., Lee, M. D., Morgan, K., Braithwaite, V. A., ... Gonzalez, C. (2015, July). Unpacking the exploration–exploitation tradeoff: A synthesis of human and animal literatures. *Decision*, *2*(3), 191-215. doi: 10.1037/dec0000033

Miller, G. (1956). The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *The Psychological Review*, *63*, 81-97.

Park, I. M., & Pillow, J. W. (2017, August). *Bayesian Efficient Coding* (Preprint). Neuroscience. doi: 10.1101/178418

Pathak, D., Agrawal, P., Efros, A. A., & Darrell, T. (2017, July). Curiosity-Driven Exploration by Self-Supervised Prediction. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (p. 488-489). Honolulu, HI, USA: IEEE. doi: 10.1109/CVPRW.2017.70

Polani, D., Martinetz, T., & Kim, J. (2001). An Information-Theoretic Approach for the Quantification of Relevance. In G. Goos, J. Hartmanis, J. van Leeuwen, J. Kelemen, & P. Sosík (Eds.), *Advances in Artificial Life* (Vol. 2159, p. 704-713). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/3-540-44811-X$_82$

Roughgarden, T. (2019). *Algorithms Illuminated (Part 3): Greedy Algorithms and Dynamic Programming* (Vol. 1) (No. 3).

Schmidhuber. (1991). A possibility for implementing curiosity and boredom in model-building neural controllers. *Proc. of the international conference on simulation of adaptive behavior: From animals to animats*, 222-227.

Schmidhuber, J. (2008, December). Driven by Compression Progress: A Simple Principle Explains Essential Aspects of Subjective Beauty, Novelty, Surprise, Interestingness, Attention, Curiosity, Creativity, Art, Science, Music, Jokes. *arXiv:0812.4360 [cs]*.

Schwartenbeck, P., Passecker, J., Hauser, T. U., FitzGerald, T. H., Kronbichler, M., & Friston, K. J. (2019). Computational mechanisms of curiosity and goal-directed exploration. *eLife*(e41703), 45.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction* (Second edition ed.). Cambridge, Massachusetts: The MIT Press.

Thrun, S. B. (1992). Eficient Exploration In Reinforcement Learning. *NIPS*, 44.

Tulving, E. (2002, February). Episodic Memory: From Mind to Brain. *Annual Review of Psychology*, *53*(1), 1-25. doi: 10.1146/annurev.psych.53.100901.135114

Valiant, L. (1984). A theory of the learnable. *Communications of the ACM*, *27*(11), 1134-1142.

Wang, M. Z., & Hayden, B. Y. (2019, August). Monkeys are curious about counterfactual outcomes. *Cognition*, *189*, 1-10. doi: 10.1016/j.cognition.2019.03.009

## Mathematical Appendix.

### Compactness

The compactness $C$ of a hyper-cube has a simple formula, $C = \frac{P^2}{A}$ where $P$ is the cube's perimeter and $A$ is its area. Therefore as $M_t$ moves to $M_{t+1}$, the we can measure the distances $\{d_i, d_{i+1}, d_{i+2}, \dots d_N\}$ for all $O \leq N$ observed states $Z$, such that $Z \subseteq S$ and treat them as if they formed a $O$-dimensional hyper-cube. In measuring this imagined cube we arrive at a geometric estimate for the compactness of changes to memory (Eq. 8).

$$C = \frac{\left(2\sum_i^O d_i\right)^2}{\prod_i^O d_i} \tag{8}$$

### Information value as a dynamic programming problem

To use theorems from dynamic programming (Roughgarden, 2019; Sutton & Barto, 2018) we must prove our memory $M$ has optimal substructure. By optimal substructure we mean that $M$ can be partitioned into a small number collection or series, each of which is an optimal dynamic programming solution. In general by proving we can decompose some optimization problem into a set of smaller problems whose optimal solution is easy to find or prove, it is trivial to prove that we can also grow the series optimally, which, in turn, allows for direct proofs by induction.

**Theorem 1** (Optimal substructure)**.** *Assuming transition function $\delta$ is deterministic, if $V^*_{\pi_E}$ is the optimal information value given by $\pi_E$, a memory $M_{t+1}$ has optimal substructure if the the last observation $s_t$ can be removed from $M_t$, by $M_{t+1} = f^{-1}(M_{t+1}, s_t)$ where the resulting value $V^*_{t-1} = V^*_t - F(M_t, a_t)$ is also optimal.*

*Proof.* Given a known optimal value $V^*$ given by $\pi_E$ we assume for the sake of contradiction there also exists an alternative policy $\hat{\pi}_E \neq \pi_E$ that gives a memory $\hat{M}_{t-1} \neq M_{t-1}$ and for which $\hat{V}^*_{t-1} > V^*_{t-1}$.

To recover the known optimal memory $M_t$ we lift $\hat{M}_{t-1}$ to $M_t = f(\hat{M}_{t-1}, s_t)$. This implies $\hat{V}^* > V^*$ which in turn contradicts the purported original optimality of $V^*$ and therefore $\hat{\pi}_E$. □

### A greedy policy explores exhaustively

Our proofs for exploration breadth are really sorting problems. If every state must be visited (or revisited) until learned, then under a greedy policy every state's value must, at one time or another, be the maximum value.

Let $Z$ be the set of all visited states, where $Z_0$ is the empty set $\{\}$ and $Z$ is built iteratively over a path $P$, such that $Z = \{s | s \in P \text{ and } s \notin Z\}$.

Sorting requires ranking, so we define an algebraic notion of inequality for any three numbers $a, b, c \in \mathbb{R}$ are defined in Eq. 9.

$$a \leq b \Leftrightarrow \exists\, c;\ b = a + c \tag{9}$$

$$a > b \Leftrightarrow (a \neq b) \wedge (b \leq a) \tag{10}$$

**Theorem 2** (State search: completeness and uniqueness)**.** *A greedy policy $\pi$ is the only deterministic policy which ensures all states in $S$ are visited, such that $Z = S$.*

*Proof.* Let $\mathbf{E} = (E_1, E_2, ...)$ be ranked series of $E$ values for all states $S$, such that $(E_1 \geq E_2, \geq ...)$. To swap any pair of values $(E_i \geq E_j)$ so $(E_i \leq E_j)$ by Eq. 9 $E_i - c = E_j$.

Therefore, again by Eq. 9, $\exists \int \delta E(s) \rightarrow -c$.

*Recall*: $\nabla \mathcal{L}_M < 0$

However if we wished to instead swap $(E_i \leq E_j)$ so $(E_i \geq E_j)$ by definition $\nexists c; E_i + c = E_j$, as $\nexists \int \delta \rightarrow c$.

To complete the proof, assume that some policy $\hat{\pi}_E \neq \pi^*_E$. By definition policy $\hat{\pi}_E$ can be any action but the maximum, leaving $k-1$ options. Eventually as $t \rightarrow T$ the only possible swap is between the max option and the $kth$, but as we have already proven this is impossible as long as $\nabla \mathcal{L}_M < 0$. Therefore, the policy $\hat{\pi}_E$ will leave at least 1 option unexplored and $S \neq Z$. □

**Theorem 3** (State search: convergence)**.** *Assuming a deterministic transition function $\Lambda$, a greedy policy $\pi_E$ will resample $S$ to convergence as $t \rightarrow T$, $E_t \rightarrow 0$.*

*Proof. Recall*: $\nabla \mathcal{L}_M < 0$.

Each time $\pi^*_E$ visits a state $s$, so $M \rightarrow M'$, $F(M', a_{t+1}) < F(M, a_t)$

In Theorem 2 we proved only a deterministic greedy policy will visit each state in $S$ over $T$ trials.

By induction, if $\pi^* E$ will visit all $s \in S$ in $T$ trials, it will revisit them in $2T$, therefore as $T \rightarrow \infty$, $E \rightarrow 0$. □

### Optimality of $\pi_\pi$

**Theorem 4** (Optimality of $\pi_\pi$)**.** *Assuming an infinite time horizon, if $\pi_E$ is optimal and $\pi_R$ is optimal, then $\pi_\pi$ is also optimal in the same sense as $\pi_E$ and $\pi_R$.*

*Proof.* The optimality of $\pi_\pi$ can be seen by direct inspection. If $p(R = 1) < 1$ and we have an infinite horizon, then $\pi_E$ will have a unbounded number of trials meaning the optimally of $P^*$ holds. Likewise, $\sum E < \varepsilon$ as $T \rightarrow \infty$, ensuring $pi_R$ will dominate $\pi_\pi$ therefore $\pi_R$ will asymptotically converge to optimal behavior. □

In proving the total optimality of $\pi_\pi$ we limit the probability of a positive reward to less than one, denoted by $p(R_t = 1) < 1$. Without this constraint the reward policy $\pi_R$ would always dominate $\pi_\pi$ when rewards are certain. While this might be useful in some circumstances, from the point of view $\pi_E$ it is extremely suboptimal as the model would never explore. Limiting $p(R_t = 1) < 1$ is reasonable constraint, as rewards in the real world are rarely certain. A more naturalistic but complex way to handle this edge case might be to introduce reward satiety, and have reward value decay with repeated exposure.