ε-BMC: A Bayesian Ensemble Approach to Epsilon-Greedy Exploration in Model-Free Reinforcement Learning

Michael Gimelfarb **Scott Sanner** **Chi-Guhn Lee**

Mechanical and Industrial Engineering
University of Toronto
ON M5S 3G8, Canada

Abstract

Resolving the exploration-exploitation trade-off remains a fundamental problem in the design and implementation of reinforcement learning (RL) algorithms. In this paper, we focus on model-free RL using the epsilon-greedy exploration policy, which despite its simplicity, remains one of the most frequently used forms of exploration. However, a key limitation of this policy is the specification of ε. In this paper, we provide a novel Bayesian perspective of ε as a measure of the uncertainty (and hence convergence) in the Q-value function. We introduce a closed-form Bayesian model update based on Bayesian model combination (BMC), based on this new perspective, which allows us to adapt ε using experiences from the environment in constant time with monotone convergence guarantees. We demonstrate that our proposed algorithm, ε-`BMC`, efficiently balances exploration and exploitation on different problems, performing comparably or outperforming the best tuned fixed annealing schedules and an alternative data-dependent ε adaptation scheme proposed in the literature.

1 INTRODUCTION

Balancing exploration with exploitation is a well-known and important problem in reinforcement learning [Sutton and Barto, 2018]. If the behaviour policy focuses too much on exploration rather than exploitation, then this could hurt the performance in an on-line setting. Furthermore, on-policy algorithms such as SARSA or TD(λ) might not converge to a good policy. On the other hand, if the exploration policy focuses too much on exploitation rather than exploration, then the state space might not be explored sufficiently and an optimal policy would not be found.

Historically, numerous exploration policies have been proposed for addressing the exploration-exploitation trade-off in model-free reinforcement learning, including Boltzmann exploration and epsilon-greedy [McFarlane, 2018]. There, this trade-off is often controlled by one or more tuning parameters, such as ε in epsilon-greedy or the temperature parameter in Boltzmann exploration. However, these parameters typically have to be handcrafted or tuned for each task in order to obtain good performance. This motivates the design of exploration algorithms that adapt their behaviour according to some measure of the learning progress. The simplest approaches adapt the tuning parameters of a fixed class of exploration policies such as epsilon-greedy [Tokic, 2010]. Other methods, such as count-based exploration [Thrun, 1992, Bellemare et al., 2016, Ostrovski et al., 2017] and Bayesian Q-learning [Dearden et al., 1998], use specialized techniques to develop new classes of exploration policies.

However, despite the recent developments in exploration strategies, epsilon-greedy is still often the exploration approach of choice [Vermorel and Mohri, 2005, Heidrich-Meisner, 2009, Mnih et al., 2015, Van Hasselt et al., 2016]. Epsilon-greedy is both intuitive and simpler to tune than other approaches, since it is completely parameterized by one parameter, ε. Another benefit of this policy is that it can be easily combined with more sophisticated frameworks, such as options [Bacon et al., 2017]. Unfortunately, the performance of epsilon-greedy in practice is highly sensitive to the choice of ε, and existing methods for adapting ε from data are ad-hoc and offer little theoretical justification.

In this paper, we take a fully *Bayesian* perspective on adapting ε based on return data. Recent work has demonstrated the strong potential of a Bayesian approach for parameter tuning in model-free reinforcement learning

[Downey and Sanner, 2010]. Another key advantage of a fully Bayesian approach over heuristics is the ability to specify priors on parameters, such as the predictive inverse variance of returns, τ in this work, which are more robust to noise or temporary digressions in the learning process. In addition, our approach can be combined with other exploration policies such as Boltzmann exploration [Tokic and Palm, 2011]. Specifically, we contribute:

1. A new Bayesian perspective of expected SARSA as an ε-weighted mixture of two models, the greedy (Q-learning) bootstrap and one which averages uniformly over all Q-values (Section 4.1);
2. A Bayesian algorithm ε-BMC (Algorithm 1) that is robust (Section 4.2), general, and adapts ε efficiently (Section 4.3);
3. A theoretical convergence guarantee of our proposed algorithm (Theorem 1).

Empirically, we evaluate the performance of ε-BMC on domains with discrete and continuous state spaces, using tabular and approximate RL methods. We empirically show that our algorithm can outperform exploration strategies that fix or anneal ε based on time, and even existing adaptive algorithms. In the end, ε-BMC is a novel, efficient and general approach to adapting the exploration parameter in epsilon-greedy policies that empirically outperforms a variety of fixed annealing schedules and other ad-hoc approaches.

2 RELATED WORK

Our paper falls within the scope of adaptive epsilon greedy algorithms. Perhaps the most similar approach to our work is the Value Differences Based Exploration (VDBE) algorithm of Tokic [2010], in which ε was modelled using a moving average and updated according to the Bellman (TD) error. However, that algorithm was presented for stationary-reward multi-armed bandits. Tokic and Palm [2011] combined the ideas of VDBE with Boltzmann exploration to create the VDBE-Softmax algorithm. dos Santos Mignon and da Rocha [2017] later developed a similar heuristic algorithm that worked on non-stationary multi-armed bandit problems. However, all these approaches are heuristic in nature; our paper approaches the problem from a Bayesian perspective.

Modelling Q-values using normal-gamma priors, as done in our paper, is a cornerstone of Bayesian Q-learning [Dearden et al., 1998]. However, that paper is fundamentally different from ours, in that it addresses the problem of exploration by adding a bonus to the Q-values that estimates the myopic *value of perfect information*. Our paper, on the other hand, applies the normal-gamma prior only to model the variance of the returns, while the exploration is handled using the epsilon-greedy policy with ε modelled using a Beta distribution.

3 PRELIMINARIES

3.1 MARKOV DECISION PROCESSES

In this paper, we denote a *Markov decision process* (MDP) as a tuple $\langle \mathcal{S}, \mathcal{A}, T, R, \gamma \rangle$, where $\mathcal{S}$ is a set of states, $\mathcal{A}$ is a finite set of actions, $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, \infty)$ is a transition function for the system state, $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is a bounded reward function, and $\gamma \in (0, 1)$ is a discount factor.

Randomized exploration policies are sequences of mappings from states to probability distributions over actions. Given an MDP $\langle \mathcal{S}, \mathcal{A}, T, R, \gamma \rangle$, for each state-action pair (s, a) and policy π, we define the expected return

$$Q^\pi(s,a) = \mathbb{E}_{\pi,T}\left[\sum_{t=0}^{\infty} \gamma^t r_{t+1} \middle| s_0 = s, a_0 = a\right],$$

where s_t is sampled from T, a_t are sampled from $\pi(\cdot|s_t)$ and $r_{t+1} = R(s_t, a_t, s_{t+1})$. The associated value function is $V^\pi(s) = \max_{a \in \mathcal{A}} Q^\pi(s, a)$, and the objective is to learn an optimal policy π^* that attains the supremum of $V^\pi(s)$ over all policies. Puterman [2014] contains a more detailed treatment of this subject.

3.2 REINFORCEMENT LEARNING

In the reinforcement learning setting, neither T nor R are known, so optimal policies are learned from experience, defined as sequences of transitions $(s_t, a_t, r_{t+1}, s_{t+1}, a_{t+1})$, $t = 0, 1, \ldots$ broken up into *episodes*. Here, states and rewards are sampled from the environment, and actions follow some exploration policy π. Given an estimate $\tilde{G}_t$ of the expected return at time t starting from state $s = s_t$ and taking action $a = a_t$, *temporal difference (TD) learning* updates the Q-values as follows:

$$Q_{t+1}(s,a) = Q_t(s,a) + \eta_t \left(\tilde{G}_t - Q_t(s,a)\right),$$

where $\eta_t \in (0, 1]$ is a problem-dependent learning rate parameter.

Typically, $\tilde{G}_t$ is computed by bootstrapping from the current Q-values, in order to reduce variance. Two of the most popular bootstrapping algorithms are *Q-learning* and *SARSA*, given respectively as:

$$\tilde{G}_t^Q = r_{t+1} + \gamma \max_{a' \in \mathcal{A}} Q_t(s_{t+1}, a'), \qquad (1)$$

$$\tilde{G}_t^{SARSA} = r_{t+1} + \gamma Q_t(s_{t+1}, a_{t+1}). \tag{2}$$

Q-learning is an example of an off-policy algorithm, whereas SARSA is on-policy. Under relatively mild conditions and in tabular settings, Q-learning has been shown to converge to the optimal policy with probability one [Watkins and Dayan, 1992].

One additional algorithm that is important in this work is *expected SARSA*

$$\tilde{G}_t^{ExpS} = r_{t+1} + \gamma \mathbb{E}_{a' \sim \pi}[Q_t(s_{t+1}, a')], \tag{3}$$

which is similar to SARSA, but in which the uncertainty of the next action a_{t+1} with respect to π is averaged out. This results in considerable variance reduction as compared to SARSA, and theoretical properties of this algorithm are detailed in Van Seijen et al. [2009]. Sutton and Barto [2018] provides a comprehensive treatment of reinforcement learning methods.

3.3 EPSILON-GREEDY POLICY

In this paper, exploration is carried out using ε-*greedy policies*, defined formally as

$$\pi^{\varepsilon}(a|s) = \begin{cases} 1 - \varepsilon_t + \frac{\varepsilon_t}{|\mathcal{A}|} & \text{if } a = \arg\max_{a' \in \mathcal{A}} Q_t(s, a') \\ \frac{\varepsilon_t}{|\mathcal{A}|} & \text{otherwise} \end{cases} . \tag{4}$$

In other words, π^{ε} samples a random action from $\mathcal{A}$ with probability $\varepsilon_t \in [0, 1]$, and otherwise selects the greedy action according to Q_t. As a result, ε_t can be interpreted as the relative importance placed on exploration.

The optimal value of the parameter ε_t is typically problem-dependent, and found through experimentation. Often, ε_t is annealed over time in order to favor exploration at the beginning, and exploitation closer to convergence [Sutton and Barto, 2018]. However, such approaches are not adaptive since they do not take into account the learning process of the agent. In this paper, our main objective is to derive a data-driven tuning strategy for ε_t, that depends on current learning progress rather than trial and error.

4 ADAPTIVE EPSILON-GREEDY

In this section, we show how the expected return under epsilon-greedy policies can be written as an average of two return models weighted by ε. There are two relevant Bayesian methods for combining multiple models based on evidence: *Bayesian model averaging* (BMA), and *Bayesian model combination* (BMC). Generally, the Bayesian model combination approach is preferred to model averaging, since it provides a richer space of hypotheses and reduced variance [Minka, 2002]. By interpreting ε as a random variable whose posterior distribution can be updated on the basis of observed data, BMC naturally leads to a method for ε adaptation.

4.1 A BAYESIAN INTERPRETATION OF EXPECTED SARSA WITH THE EPSILON-GREEDY POLICY

We begin by combining the definition of expected SARSA (3) with the ε-greedy policy (4). For $s' = s_{t+1}$, $a^* = \arg\max_{a'} Q_t(s', a')$, and $r' = r_{t+1}$ we have

$$\begin{aligned}
&\tilde{G}_t^{ExpS} \\
&= r' + \gamma \sum_{a \in \mathcal{A}} \pi^{\varepsilon}(a|s') Q_t(s', a) \\
&= r' + \gamma \left(1 - \varepsilon_t + \frac{\varepsilon_t}{|\mathcal{A}|}\right) Q_t(s', a^*) + \gamma \frac{\varepsilon_t}{|\mathcal{A}|} \sum_{a \neq a^*} Q_t(s', a) \\
&= r' + (1 - \varepsilon_t)\, \gamma Q_t(s', a^*) + \varepsilon_t \gamma \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} Q_t(s', a) \\
&= (1 - \varepsilon_t)\, \tilde{G}_t^Q + \varepsilon_t \tilde{G}_t^U,
\end{aligned} \tag{5}$$

where $\tilde{G}_t^Q$ is the Q-learning bootstrap (1) and

$$\tilde{G}_t^U = r_{t+1} + \gamma \frac{1}{|\mathcal{A}|} \sum_{a' \in \mathcal{A}} Q_t(s_{t+1}, a') \tag{6}$$

is an estimate that uniformly averages over all the action-values, which we call the *uniform* model. This leads to the following important observation: *expected SARSA can be viewed as a probability-weighted average of two models, the greedy model $\tilde{G}_t^Q$ that trusts the current Q-value estimates and acts optimally with respect to them, and the uniform model $\tilde{G}_t^U$ that completely distrusts the current Q-value estimates and consequently places a uniform belief over them.* Under this interpretation, ε_t and $1 - \varepsilon_t$ are the posterior beliefs assigned to the two aforementioned models, respectively. In the following subsections, we verify this simple fact algebraically in the context of Bayesian model combination. We also develop a method for maintaining the (approximate) posterior belief state efficiently, with a computational cost that is constant in both space and time, and with provable convergence guarantees.

4.2 BAYESIAN Q-LEARNING

In order to facilitate tractable learning and inference, we assume that the return observation $q_{s,a}$ at time t, given the model $m \in \{Q, U\}$, is normally distributed:

$$q_{s,a} | m, \tau \sim \mathcal{N}\left(\tilde{G}_t^m, \tau^{-1}\right), \tag{7}$$

where the means $\tilde{G}_t^Q$ and $\tilde{G}_t^U$ are given in (1) and (6), respectively, and $\tau > 0$ is the inverse of the variance, or *precision*. This assumption can be justified, by viewing the return as a discounted sum of future (random) reward observations, and appealing to the central limit theorem when γ is close to 1 and the MDP is ergodic [Dearden et al., 1998].

There are two special cases of interest in this work. In the first case, τ is allowed to be constant across all state-action pairs and models, and naturally leads to a state-independent ε adaptation. This approach is particularly advantageous when it is costly or impossible to maintain independent statistics per state, such as when the state space is very large or continuous in nature. In the second case, independent statistics are maintained per state and lead to state-dependent exploration.

In order to update τ, we consider the standard normal-gamma model:

$$\begin{aligned} \mu, \tau &\sim \text{NormalGamma}\left(\mu_0, \tau_0, a_0, b_0\right), \\ q_{s,a} | \mu, \tau &\sim \mathcal{N}\left(\mu, \tau^{-1}\right), \end{aligned} \tag{8}$$

where $q_{s,a}$ are i.i.d. given μ and τ. Since the returns in different state-action pairs are dependent, this assumption is likely to be violated in practice. However, it leads to a compact learning representation necessary for tractable Bayesian inference, and has been used effectively in the existing literature in similar forms [Dearden et al., 1998]. Furthermore, (8) is not used to model the Q-values directly in our paper, but rather, to facilitate robust estimation of τ, as we now show.

Given data $\mathcal{D} = \{q_{s_i,a_i,i} \,|\, i = 0, 1 \ldots t{-}1\}$ of previously observed returns, the joint posterior distribution of μ and τ with likelihood (7) and prior (8) is also normal-gamma distributed, and so the marginal posterior distribution of τ, $\mathbb{P}(\tau|\mathcal{D})$, is gamma distributed with parameters:

$$\begin{aligned} a_t &= a_0 + \frac{t}{2}, \\ b_t &= b_0 + \frac{t}{2}\left(\hat{\sigma}_t^2 + \frac{\tau_0}{\tau_0 + t}(\hat{\mu}_t - \mu_0)^2\right), \end{aligned} \tag{9}$$

where $\hat{\mu}_t$ and $\hat{\sigma}_t^2$ are the sample mean and variance of the returns in $\mathcal{D}$, respectively [Bishop, 2006]. These quantities can be updated online after each new observation d' in constant time [Welford, 1962].

Finally, for each model $m \in \{Q, U\}$, we marginalize over the uncertainty in τ, using (7) and (9) as follows:

$$\begin{aligned} \mathbb{P}(q_{s,a}|m,\mathcal{D}) &= \int_0^\infty \mathbb{P}(q_{s,a}|m,\tau)\mathbb{P}(\tau|\mathcal{D})\,\mathrm{d}\tau \\ &\propto \int_0^\infty \tau^{1/2} e^{-\frac{\tau}{2}(q_{s,a} - \tilde{G}_t^m)^2} \tau^{a_t - 1} e^{-b_t \tau}\,\mathrm{d}\tau \\ &= \int_0^\infty \tau^{a_t + \frac{1}{2} - 1} e^{-\left(b_t + \frac{1}{2}(q_{s,a} - \tilde{G}_t^m)^2\right)\tau}\,\mathrm{d}\tau \\ &\propto \left(b_t + \frac{1}{2}(q_{s,a} - \tilde{G}_t^m)^2\right)^{-\frac{2a_t+1}{2}}. \end{aligned}$$

Finally, we have:

$$q_{s,a}|m, \mathcal{D} \sim \text{St}\left(\tilde{G}_t^m, \frac{a_t}{b_t}, 2a_t\right), \tag{10}$$

where $\text{St}(\mu, \lambda, \nu)$ is the three-parameter Student t-distribution [Bishop, 2006]. Therefore, marginalizing over the unknown precision τ leads to a t-distributed likelihood function. Alternatively, one could simply use the Gaussian likelihood in equation (7) and treat τ as a problem-dependent tuning parameter. However, the heavy tail property of the t-distribution is advantageous in the non-stationary setting typically encountered in reinforcement learning applications, where Q-values change during the learning phase. We now show how to link this update with the expected SARSA decomposition (5) to derive an adaptive epsilon-greedy policy.

4.3 EPSILON-BMC: ADAPTING EPSILON USING BAYESIAN MODEL COMBINATION

In the general setting of *Bayesian model combination*, we model the uncertainty in Q-values for each state-action pair (s, a) as random variables with posterior distribution $\mathbb{P}(q_{s,a}|\mathcal{D})$. The expected posterior return can be written as an average over all possible *combinations* of greedy and uniform model,

$$\mathbb{E}[q_{s,a}|\mathcal{D}] = \int_0^1 \mathbb{E}[q_{s,a}|w]\,\mathbb{P}(w|\mathcal{D})\,\mathrm{d}w, \tag{11}$$

where w is the weight assigned to the uniform model and $1 - w$ is the weight assigned to the greedy model [Monteith et al., 2011]. As will be verified shortly, the expectation of this weight w given the past return data $\mathcal{D}$ will turn out to be a Bayesian interpretation of ε_t.

The belief over w is maintained as a posterior distribution $p_t(w) = \mathbb{P}(w|\mathcal{D})$. Continuing from (11) and using (10):

$$\begin{aligned} &\mathbb{E}[q_{s,a}|\mathcal{D}] \\ &= \int_0^1 \mathbb{E}[q_{s,a}|w,\mathcal{D}]\,\mathbb{P}(w|\mathcal{D})\,\mathrm{d}w \\ &= \int_0^1 \sum_{m \in \{Q,U\}} \mathbb{E}[q_{s,a}|m,\mathcal{D}]\,\mathbb{P}(m|w)\,\mathbb{P}(w|\mathcal{D})\,\mathrm{d}w \\ &= \int_0^1 \left(\mathbb{E}[q_{s,a}|Q,\mathcal{D}](1-w) + \mathbb{E}[q_{s,a}|U,\mathcal{D}]w\right)\mathbb{P}(w|\mathcal{D})\,\mathrm{d}w \\ &= \left(1 - \mathbb{E}[w|\mathcal{D}]\right)\mathbb{E}[q_{s,a}|Q,\mathcal{D}] + \mathbb{E}[w|\mathcal{D}]\,\mathbb{E}[q_{s,a}|U,\mathcal{D}] \\ &= \left(1 - \mathbb{E}[w|\mathcal{D}]\right)\tilde{G}_t^Q + \mathbb{E}[w|\mathcal{D}]\tilde{G}_t^U, \end{aligned}$$

which is exactly (5) except that now $\varepsilon_t = \mathbb{E}[w|\mathcal{D}]$. We have thus shown that *the expected SARSA bootstraps with data-driven ε_t can be viewed in terms of Bayesian model combination.* We denote this new estimate ε_t^{BMC}.

The posterior distribution $p_t(w) = \mathbb{P}(w|\mathcal{D})$ is updated recursively by Bayes' rule:

$$\begin{aligned} p_t(w) &\propto \mathbb{P}(d'|w,\mathcal{D})p_{t-1}(w) \\ &\propto \sum_{m\in\{Q,U\}} \mathbb{P}(d'|m,\mathcal{D})\mathbb{P}(m|w)p_{t-1}(w) \\ &\propto \left(\mathbb{P}(d'|U,\mathcal{D})w + \mathbb{P}(d'|Q,\mathcal{D})(1-w)\right)p_{t-1}(w), \end{aligned}$$

for every new observation d'. Since the number of terms in p_t grows exponentially in $|\mathcal{D}|$, it is necessary to use posterior approximation techniques to compute $\mathbb{E}[w|\mathcal{D}]$.

One approach to address the intractability of computing an exact posterior $p_t(w)$ is to sample directly from the distribution. However, such an approach is inherently noisy and inefficient in practice. Instead, we apply the Dirichlet moment-matching technique [Hsu and Poupart, 2016, Gimelfarb et al., 2018] that was shown to be effective at differentiating between good and bad models from data, easy to implement, and efficient. In particular, we apply the approach in Gimelfarb et al. [2018] with the models $\tilde{G}_t^Q$ and $\tilde{G}_t^U$, by matching the first and second moments of p_t to those of the beta distribution $\mathrm{Beta}(\alpha_t, \beta_t)$ and solving the resulting system of equations for α_t and β_t. The closed-form solution is:

$$m_t = \frac{\alpha_t}{\alpha_t+\beta_t+1}\frac{e_t^U(\alpha_t+1)+e_t^Q\beta_t}{e_t^U\alpha_t+e_t^Q\beta_t}, \tag{12}$$

$$v_t = \frac{\alpha_t}{\alpha_t+\beta_t+1}\frac{\alpha_t+1}{\alpha_t+\beta_t+2}\frac{e_t^U(\alpha_t+2)+e_t^Q\beta_t}{e_t^U\alpha_t+e_t^Q\beta_t}, \tag{13}$$

$$r_t = \frac{m_t - v_t}{v_t - m_t^2}, \tag{14}$$

$$\alpha_{t+1} = m_t r_t, \tag{15}$$

$$\beta_{t+1} = (1-m_t)r_t, \tag{16}$$

where e_t^U and e_t^Q are the respective probabilities of observing a return $d' = \tilde{G}_t^{ExpS}$ under the distributions (10). It follows that

$$\varepsilon_t^{BMC} \approx \mathbb{E}_{\mathrm{Beta}(\alpha_t,\beta_t)}[w|\mathcal{D}] = \frac{\alpha_t}{\alpha_t+\beta_t}. \tag{17}$$

All quantities, including a_t, b_t, α_t and β_t, can be computed online in constant time and with minimal storage overhead without caching $\mathcal{D}$. We call this approach ε−BMC and present the corresponding pseudo-code in Algorithm 1. Therein, lines with a * indicate additions to the ordinary expected SARSA algorithm.

Algorithm 1 ε−BMC with Expected SARSA

```
1: * initialize $\mu_0, \tau_0, a_0, b_0, \hat{\mu} = 0, \hat{\sigma}^2 = \infty, \alpha, \beta$
2: for each episode do
3:     initialize $s$
4:     for each step in the episode do
5:         * $\varepsilon \leftarrow \frac{\alpha}{\alpha+\beta}$
6:         choose action $a$ using $\varepsilon$-greedy policy $\pi^\varepsilon$ (4)
7:         take action $a$, observe $r$ and $s'$
8:         $\tilde{G}^Q \leftarrow r + \gamma \max_{a'} Q(s', a')$
9:         $\tilde{G}^U \leftarrow r + \gamma \frac{1}{|\mathcal{A}|}\sum_{a'} Q(s', a')$
10:        $\tilde{G}^{ExpS} \leftarrow r + \gamma \sum_{a'} \pi^\varepsilon(a'|s')Q(s', a')$ {note
           $\tilde{G}^{ExpS} = (1-\varepsilon)\tilde{G}^Q + \varepsilon\tilde{G}^U$}
11:        $Q(s,a) \leftarrow Q(s,a) + \eta[\tilde{G}^{ExpS} - Q(s,a)]$
12:        * update $\hat{\mu}$ and $\hat{\sigma}^2$ using observation $\tilde{G}^{ExpS}$
13:        * compute $a$ and $b$ using (9)
14:        * compute $e^Q$ and $e^U$ using (10)
15:        * update $\alpha$ and $\beta$ using (12)-(16)
16:        $s \leftarrow s'$
17:    end for
18: end for
```

In Algorithm 1, the expected SARSA return $\tilde{G}^{ExpS}$ was used to update both the posterior on ε and the Q-values. However, it is possible to update the Q-values in line 11 using a different estimator of the future return, including Q-learning (1), SARSA (2), or other approaches. The Q-values could also be approximated using a deep neural network or other function approximator. The algorithm can also be run off-line by caching $\mathcal{D}$ for an entire episode and then updating ε. State-dependent exploration can be easily implemented by maintaining the posterior statistics independently for each state, or approximating them using a neural network.

A final advantage of our algorithm is a provable convergence guarantee under fairly general assumptions.

Theorem 1 (Monotone Convergence of ε-BMC)**.** *Suppose $0 < \alpha_0 \leq \beta_0 < \infty$. Then, $\varepsilon_{t+1}^{BMC} \leq \varepsilon_t^{BMC}$ for all $t = 0, 1 \dots$, therefore ε_t^{BMC} converges as $t \to \infty$.*

Proof. Let $\varepsilon_t = \varepsilon_t^{BMC}$ and observe that:

$$\begin{aligned} &\varepsilon_{t+1} \\ &= \frac{\alpha_{t+1}}{\alpha_{t+1}+\beta_{t+1}} = \frac{m_t r_t}{r_t} = m_t \\ &= \frac{\alpha_t}{\alpha_t+\beta_t+1}\frac{e_t^U(\alpha_t+1)+e_t^Q\beta_t}{e_t^U\alpha_t+e_t^Q\beta_t} \\ &= \frac{m_{t-1}r_{t-1}}{r_{t-1}+1}\frac{e_t^U(m_{t-1}r_{t-1}+1)+e_t^Q(1-m_{t-1})r_{t-1}}{e_t^U m_{t-1}r_{t-1}+e_t^Q(1-m_{t-1})r_{t-1}} \end{aligned}$$

$$= \varepsilon_t \frac{\left(e_t^U m_{t-1} + e_t^Q(1-m_{t-1})\right) r_{t-1} + e_t^U}{\left(e_t^U m_{t-1} + e_t^Q(1-m_{t-1})\right)(r_{t-1}+1)}, \tag{18}$$

where the first line uses (15)-(17), the second uses (12) and the third uses (15) and (16). Then, since $d_t = \tilde{G}_t^{ExpS} = (1-\varepsilon_t)\tilde{G}_t^Q + \varepsilon_t \tilde{G}_t^U$:

$$(d_t - \tilde{G}_t^U)^2 = (1-\varepsilon_t)^2(\tilde{G}_t^Q - \tilde{G}_t^U)^2,$$
$$(d_t - \tilde{G}_t^Q)^2 = \varepsilon_t^2(\tilde{G}_t^Q - \tilde{G}_t^U)^2.$$

Now, if $\varepsilon_t \leq \frac{1}{2}$, then this implies that

$$(d_t - \tilde{G}_t^U)^2 \geq (d_t - \tilde{G}_t^Q)^2 \implies e_t^U \leq e_t^Q$$
$$\implies e_t^U \leq e_t^U m_{t-1} + e_t^Q(1-m_{t-1}),$$

and from (18) we conclude that $\varepsilon_{t+1} \leq \varepsilon_t$. The first statement of the theorem follows from the assumption $\varepsilon_0 \leq \frac{1}{2}$ and a standard induction argument. The second statement follows from the monotone convergence theorem (see, e.g. Rudin [1976], pg. 56). □

The convergence of ε-BMC holds using any value function representation, including neural networks. It is important to note that convergence of ε-BMC can only be guaranteed when ε is initialized in $[0, 0.5]$. However, this is not a concern in practice, since it has been found that there is no significant gain in using values of ε larger than 0.5 [dos Santos Mignon and da Rocha, 2017].

5 EMPIRICAL EVALUATION

To demonstrate the ability of Algorithm 1 to adapt in a variety of environments, we consider a deterministic, finite state grid-world domain, the continuous state cart-pole control problem, and a stochastic, discrete state supply-chain problem. The third domain was chosen to show how our algorithm performs when the action space is large. We considered two different reinforcement learning algorithms: on-policy tabular expected SARSA [Sutton and Barto, 2018], and off-policy DQN with experience replay [Mnih et al., 2015] [1]. The parameter settings are listed in Tables 1 and 2 in the supplementary materials [2]. All experiments were run independently 100 times, and mean curves with shaded standard error are reported.

In the empirical evaluation of Algorithm 1, our goal is to quantify the added value of adapting the ε parameter in epsilon-greedy policies using a Bayesian approach, rather than compare the performance of epsilon-greedy policies against other approaches, which has been investigated in the literature in various settings [Vermorel and Mohri, 2005, Tijsma et al., 2016]. Therefore, Algorithm 1 is compared against different annealing schedules for ε_t, broken down into the following categories:

- **constant:** $\varepsilon_t = c$, where $c \in \{0.01, 0.05, 0.1, 0.25, 0.5\}$;
- **geometric:** $\varepsilon_t = \frac{1}{2}\rho^t$, where $\rho \in \{0.85, 0.9, 0.95, 0.975, 0.99\}$ and t is the episode number;
- **power:** $\varepsilon_t = \frac{1}{2}(t+1)^{-\beta}$, where $\beta \in \{0.25, 0.5, 1.0, 1.5\}$ and t is the episode number;
- **adaptive:** VDBE [Tokic, 2010] with $\varepsilon_0 = 0.5$, $\delta = 1/|\mathcal{A}|$, and $\sigma \in \{0.01, 0.05, 0.1, 0.5, 1.0, 10.0, 100.0\}$.

We do not compare to Tokic and Palm [2011], since that paper falls outside the scope of epsilon-greedy policies. However, we reiterate that it is a trivial matter to interchange VDBE and ε-BMC in that framework.

5.1 GRID-WORLD

The first benchmark problem is the discrete deterministic 5-by-5 grid-world navigation problem with sub-goals presented in Ng et al. [1999]. Valid moves incur a cost of 0.1, and invalid moves incur a cost of 0.2, in order to encourage the agent to solve the task in the least amount of time. We set $\gamma = 0.99$. Testing consists of running a single episode, starting from the same initial state, using the greedy policy at the end of each episode. The results are shown in Figures 1 and 2.

5.2 CART-POLE

The second problem is the continuous deterministic cart-pole control problem. A reward of 1.0 is provided until the pole falls, to encourage the agent to keep the pole upright. We also set $\gamma = 0.95$. To run the tabular expected SARSA algorithm and VDBE, we discretize the four-dimensional state space into $3 \times 3 \times 4 \times 3 = 108$ equal regions. Since the initial position is randomized, testing consists of evaluating the greedy policy on 10 independent episodes and averaging the returns. Since over-fitting was a significant concern for DQN, we stop training as soon as perfect test performance (the pole has not been dropped) was observed over four consecutive episodes. The results are shown in Figures 3 and 4.

[1] As noted in the previous section, we only need to replace line 11 of Algorithm 1 with the DQN update.

[2] The code and supplementary materials can be found at `https://github.com/mike-gimelfarb/bayesian-epsilon-greedy`.

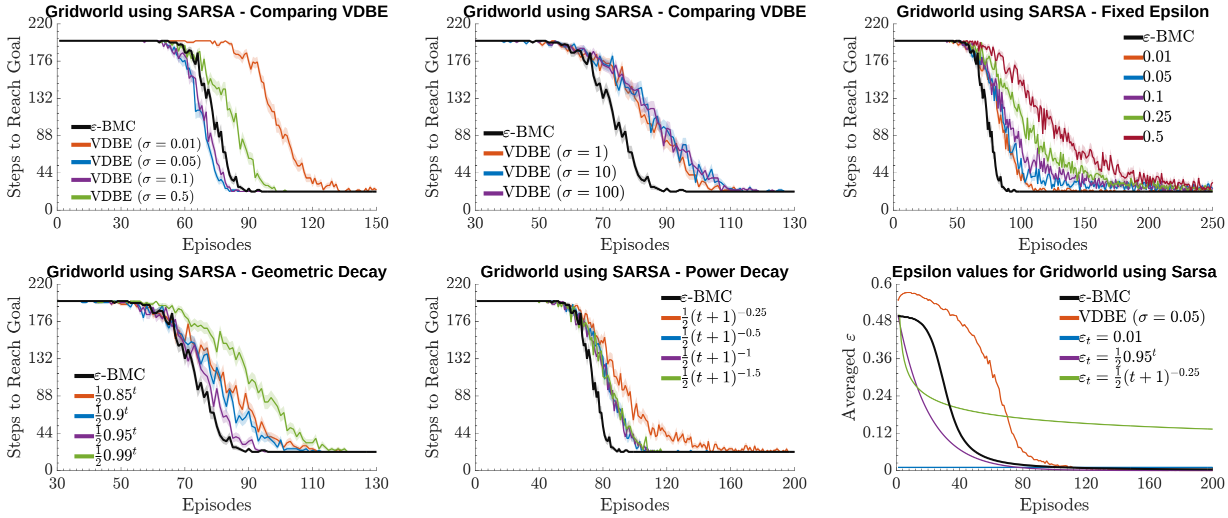

Figure 1: Average performance (steps to reach the final goal) on the grid-world domain using expected SARSA.

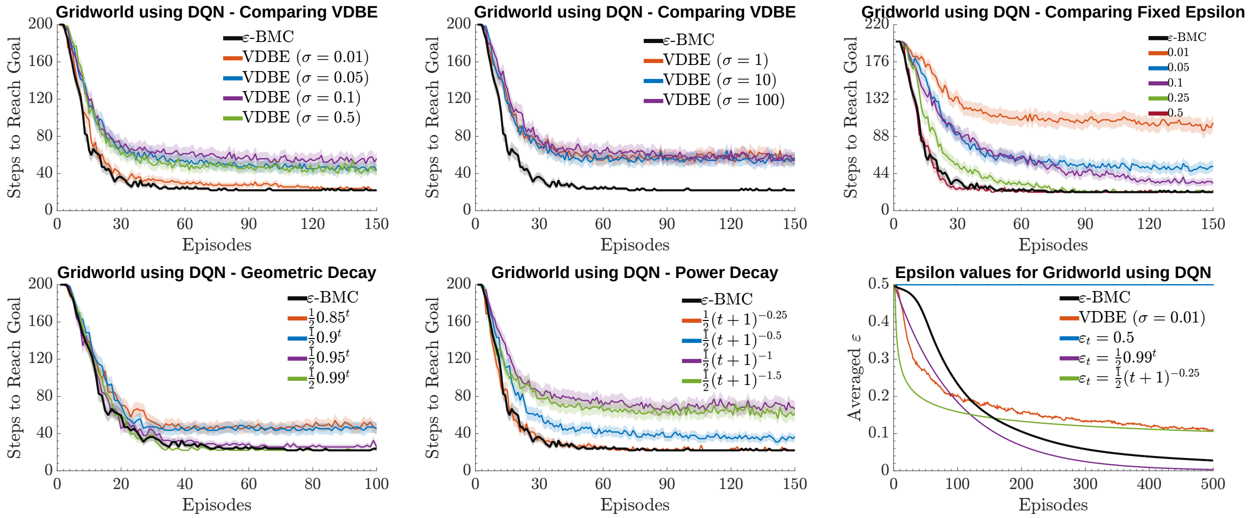

Figure 2: Average performance (steps to reach the final goal) on the grid-world domain using deep Q learning.

5.3 SUPPLY-CHAIN MANAGEMENT

This supply-chain management problem was described in Kemmer et al. [2018], and consists of a factory and warehouses. The agent must decide how much inventory to produce at the factory and how much inventory to ship to the warehouse(s) to meet the demand. The parameters used in our experiment are: $K = 1$, $p = 0.5$, $\kappa_{pr} = 0.1$, $\kappa_{st,j} = 0.02$, $\kappa_{tr,j} = 0.1$, $\zeta_j = 5, c_j = 50, \rho_{max} = 10$, and demand follows a Poisson distribution with rate $\lambda = 2.5$. We also set a transportation limit of 10 items per period, and assume that unfulfilled demand is not backlogged, but lost forever. The initial state is always $(10, 0)$ for training and testing (e.g. 10 items initially at the factory and 0 at the warehouse). We set $\gamma = 0.95$. Like cart-pole, testing is done by averaging the returns of 10 independent trials using the greedy policy. The results are illustrated in Figures 5 and 6.

5.4 DISCUSSION

Overall, we see that ε-BMC consistently outperformed all other types of ε annealing strategies, including VDBE, or performed similarly. However, ε-BMC converged slightly later than VDBE on the grid-world domain and the fixed annealing strategy $\varepsilon_t = \frac{1}{2}(t+1)^{-0.25}$ on the supply-chain problem, using tabular expected SARSA. However, in the former case, ε-BMC outperformed all fixed tuning strategies, and in the latter case, it outperformed VDBE by a large margin. These observations are related

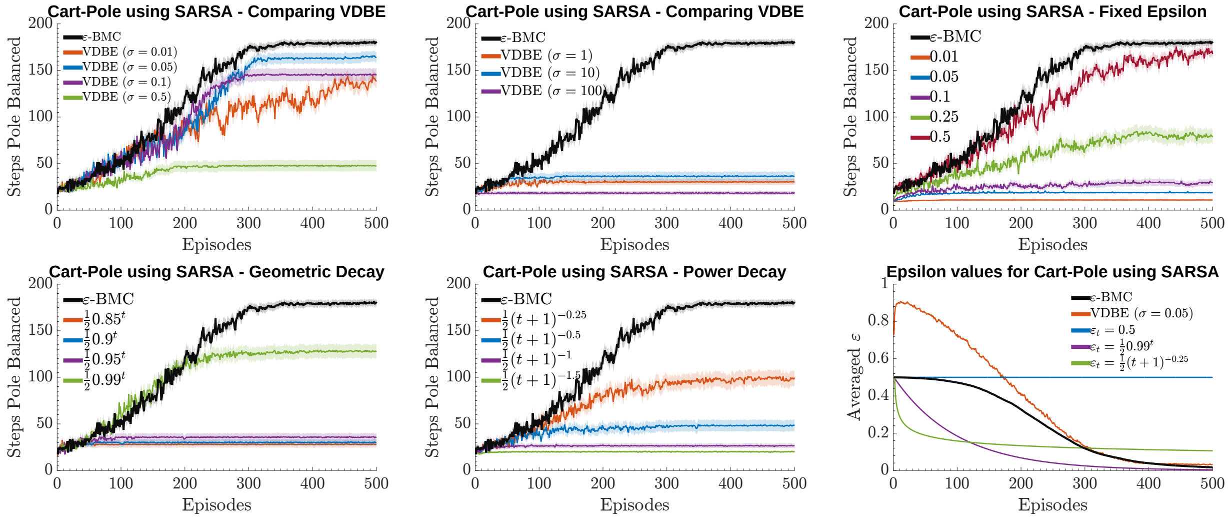

Figure 3: Average performance (time steps the pole is balanced) on the cart-pole domain using expected SARSA.

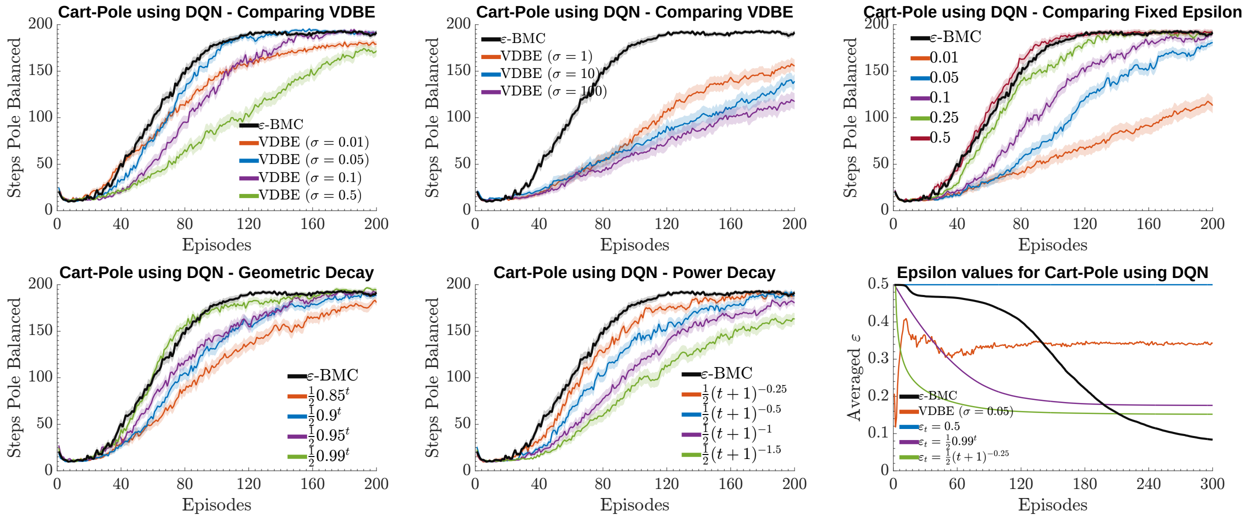

Figure 4: Average performance (time steps the pole is balanced) on the cart-pole domain using deep Q-learning.

to the speed of convergence; asymptotically, ε-BMC approached the performance of the best policy that was attained (for grid-world this is indeed the optimal policy).

While it performed well on the simple grid-world domain, VDBE performed considerably worse than ε-BMC on the more complex supply-chain problem. We believe that the Bayesian approach of ε-BMC smooths out the noise in the return signals better than VDBE and other ad-hoc approaches for adapting ε. This also suggests why our algorithm performed better on DQN.

Furthermore, we see that no single family of annealing strategies worked consistently well across all domains and algorithms. For instance, geometric decay strategies worked well on the grid-world domain, while performing poorly on the supply-chain problem using tabular SARSA. The power decay strategies worked well on the supply-chain problem using tabular SARSA, but failed to match the performance of other strategies when switching to DQN. Also, the performance of VDBE was highly sensitive to the choice of the σ parameter. A lower value of σ worked well for grid-world and cart-pole, but higher values of σ worked better for supply-chain. The performance of ε-BMC was relatively insensitive to the choice of prior parameters for μ and τ (a_0, b_0, μ_0, τ_0), so we were able to use the same values in all our experiments. However, unsurprisingly, it was more sensitive to the strength of the prior on ε (α_0, β_0). Since we can always set $\alpha_0 \approx \beta_0$, this effectively reduces to the problem of selecting a single parameter that controls the strength

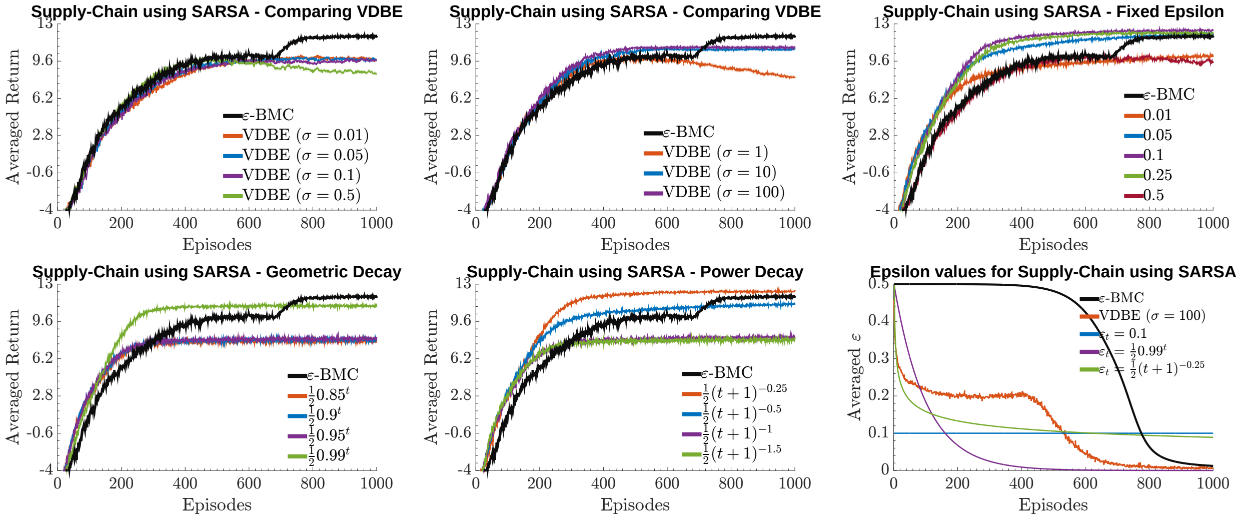

Figure 5: Average performance (return) on the supply-chain domain using expected SARSA.

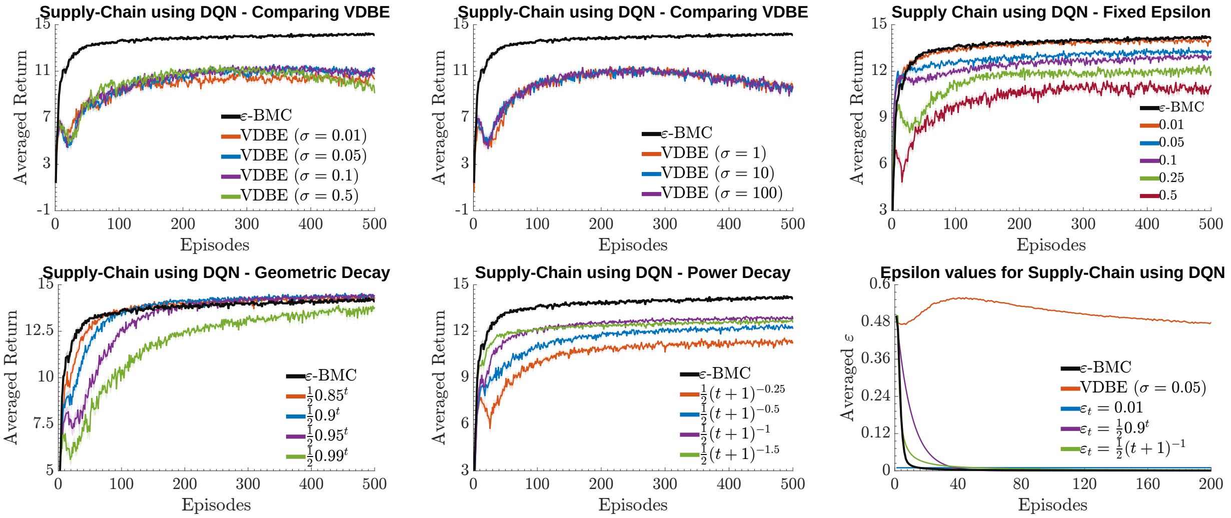

Figure 6: Average performance (return) on the supply-chain domain using deep Q-learning.

of the prior on ε. This is considerably easier to do than to select both a good annealing method *and* the tuning parameter(s).

6 CONCLUSION

In this paper, we proposed a novel Bayesian approach to solve the exploration-exploitation problem in general model-free reinforcement learning, in the form of an adaptive epsilon-greedy policy. Our novel algorithm, ε-BMC, is a novel approach for tuning the ε parameter automatically from return observations based on Bayesian model combination and approximate moment-matching based inference. It was argued to be general, efficient, robust, and theoretically grounded, and was shown empirically to outperform fixed annealing schedules for ε and even a state-of-the-art ε adaptation scheme.

In future work, it would be interesting to evaluate the performance of ε-BMC combined with Boltzmann exploration [Tokic and Palm, 2011], as well as the state-dependent version. We believe that it is possible to obtain a Bayesian interpretation of VDBE by placing priors over the Bellman errors and updating them using data, but we have not investigated this approach. It would also be interesting to extend our approach to handle options.

Acknowledgements

We thank the reviewers for their insightful comments and suggestions.

References

P.-L. Bacon, J. Harb, and D. Precup. The option-critic architecture. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

M. Bellemare, S. Srinivasan, G. Ostrovski, T. Schaul, D. Saxton, and R. Munos. Unifying count-based exploration and intrinsic motivation. In *Advances in Neural Information Processing Systems*, pages 1471–1479, 2016.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

R. Dearden, N. Friedman, and S. Russell. Bayesian q-learning. In *AAAI/IAAI*, pages 761–768, 1998.

A. dos Santos Mignon and R. L. d. A. da Rocha. An adaptive implementation of ε-greedy in reinforcement learning. *Procedia Computer Science*, 109:1146–1151, 2017.

C. Downey and S. Sanner. Temporal difference bayesian model averaging: A bayesian perspective on adapting lambda. In *International Conference on Machine Learning*, pages 311–318, 2010.

M. Gimelfarb, S. Sanner, and C.-G. Lee. Reinforcement learning with multiple experts: A bayesian model combination approach. In *Advances in Neural Information Processing Systems*, pages 9549–9559. Curran Associates, 2018.

V. Heidrich-Meisner. Interview with richard s. sutton. *KI*, 23(3):41–43, 2009.

W.-S. Hsu and P. Poupart. Online bayesian moment matching for topic modeling with unknown number of topics. In *Advances In Neural Information Processing Systems*, pages 4536–4544, 2016.

L. Kemmer, H. von Kleist, D. de Rochebouet, N. Tziortziotis, and J. Read. Reinforcement learning for supply chain optimization. In *European Workshop on Reinforcement Learning 14*, 10 2018.

R. McFarlane. A survey of exploration strategies in reinforcement learning. Unpublished, 2018.

T. P. Minka. Bayesian model averaging is not model combination. Unpublished, 2002.

V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518 (7540):529, 2015.

K. Monteith, J. L. Carroll, K. Seppi, and T. Martinez. Turning bayesian model averaging into bayesian model combination. In *Neural Networks (IJCNN)*, pages 2657–2663, 2011.

A. Y. Ng, D. Harada, and S. Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *International Conference on Machine Learning*, volume 99, pages 278–287, 1999.

G. Ostrovski, M. G. Bellemare, A. van den Oord, and R. Munos. Count-based exploration with neural density models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2721–2730, 2017.

M. L. Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

W. Rudin. *Principles of mathematical analysis*. McGraw-Hill Book Co., New York, third edition, 1976. ISBN 0-07-085613-3.

R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

S. B. Thrun. Efficient exploration in reinforcement learning. Technical report, CMU-CS-92-102, School of Computer Science, Carnegie Mellon, 1992.

A. D. Tijsma, M. M. Drugan, and M. A. Wiering. Comparing exploration strategies for q-learning in random stochastic mazes. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–8, 2016.

M. Tokic. Adaptive ε-greedy exploration in reinforcement learning based on value differences. In *Annual Conference on Artificial Intelligence*, pages 203–210, 2010.

M. Tokic and G. Palm. Value-difference based exploration: adaptive control between epsilon-greedy and softmax. In *Annual Conference on Artificial Intelligence*, pages 335–346, 2011.

H. Van Hasselt, A. Guez, and D. Silver. Deep reinforcement learning with double q-learning. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

H. Van Seijen, H. Van Hasselt, S. Whiteson, and M. Wiering. A theoretical and empirical analysis of expected sarsa. In *Adaptive Dynamic Programming and Reinforcement Learning, 2009. ADPRL'09. IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning*, pages 177–184, 2009.

J. Vermorel and M. Mohri. Multi-armed bandit algorithms and empirical evaluation. In *European conference on machine learning*, pages 437–448. Springer, 2005.

C. J. Watkins and P. Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

B. Welford. Note on a method for calculating corrected sums of squares and products. *Technometrics*, 4(3): 419–420, 1962.