Towards Tractable Optimism in Model-Based Reinforcement Learning

Aldo Pacchiano
Philip Ball
Jack Parker-Holder
Krzysztof Choromanski
Stephen Roberts

1 UC Berkeley
2 University of Oxford
3 Google Brain Robotics

Abstract

The principle of optimism in the face of uncertainty is prevalent throughout sequential decision making problems such as multi-armed bandits and reinforcement learning (RL). To be successful, an optimistic RL algorithm must over-estimate the true value function (optimism) but not by so much that it is inaccurate (estimation error). In the tabular setting, many state-of-the-art methods produce the required optimism through approaches which are intractable when scaling to deep RL. We re-interpret these scalable optimistic model-based algorithms as solving a tractable noise augmented MDP. This formulation achieves a competitive regret bound: $\tilde{O}(\sqrt{|S|H|A|T})$ when augmenting using Gaussian noise, where $T$ is the total number of environment steps. We also explore how this trade-off changes in the deep RL setting, where we show empirically that estimation error is significantly more troublesome. However, we also show that if this error is reduced, optimistic model-based RL algorithms can match state-of-the-art performance in continuous control problems.

1 INTRODUCTION

Reinforcement Learning (RL, Sutton and Barto [1998]) considers the problem of an agent taking sequential actions in an uncertain environment to maximize some notion of reward. Model-based reinforcement learning (MBRL) algorithms typically approach this problem by building a “world model” [Sutton, 1991], which can be used to simulate the true environment. This facilitates efficient learning, since the agent no longer needs to query to true environment for experience, and instead plans in the world model. In order to learn a world model that accurately represents the dynamics of the environment, the agent must collect data that is rich in experiences [Sekar et al., 2020]. However, for faster convergence, data collection must also be performed efficiently, wasting as few samples as possible [Ball et al., 2020]. Thus, the effectiveness of MBRL algorithms hinges on the exploration-exploitation dilemma.

This dilemma has been studied extensively in the tabular RL setting, which considers Markov Decision Processes (MDPs) with finite states and actions. Optimism in the face of uncertainty (OFU) [Audibert et al., 2007, Kocsis and Szepesvári, 2006] is a principle that emerged first from the Multi-Arm Bandit literature, where actions having both large expected rewards (exploitation) and high uncertainty (exploration) are prioritized. OFU is a crucial component of several state-of-the-art algorithms in this setting [Silver et al., 2016], although its success has thus far failed to scale to larger settings.

However, in the field of deep RL, many of these theoretical advances have been overlooked in favor of heuristics [Burda et al., 2019], or simple dithering based approaches for exploration [Mnih et al., 2013]. There are two potential reasons for this. First, many of the theoretically motivated OFU algorithms are intractable in larger settings. For example, UCRL2 [Jaksch et al., 2010] a canonical optimistic RL algorithm, requires the computation of an analytic uncertainty envelope around the MDP, which is infeasible for continuous MDPs. Despite its many extensions [Filippi et al., 2010, Jaksch et al., 2010, Fruit et al., 2018, Azar et al., 2017b, Bartlett and Tewari, 2012, Tossou et al., 2019], none address generalizing the techniques to continuous (or even large discrete) MDPs.

Second, OFU algorithms must strike a fine balance in what we call the Optimism Decomposition. That is, they need to be optimistic enough to upper bound the true value function, while maintaining low estimation error. Theoretically motivated OFU algorithms predominantly focus on the prior. However, when moving to the deep RL setting, several sources of noise make estimation error a thorn in the side of optimistic approaches. We show that an optimistic algorithm can fixate on exploiting the least accurate models, which causes the majority of experience the agent learns from to be worthless, or even harmful for performance.

In this paper we seek to address both of these issues, paving the way for OFU-inspired algorithms to gain prominence in...
the deep RL setting. We make two contributions:

**Making provably efficient algorithms tractable** Our first contribution is to introduce a new perspective on existing tabular RL algorithms such as UCRL2. We show that a comparable regret bound can be achieved by being optimistic with respect to a noise augmented MDP, where the noise is proportional to the amount of data collected during learning. We propose several mechanisms to inject noise, including count-scaled Gaussian noise and the variance from a bootstrap mechanism. Since the latter technique is used in many prominent state-of-the-art deep MBRL algorithms [Kurttach et al., 2018; Janner et al., 2019; Chua et al., 2018; Ball et al., 2020], we have all the ingredients we need to scale to that paradigm.

**Addressing model estimation error in the deep RL paradigm** We empirically explore the Optimism Decomposition in the deep RL setting, and introduce a new approach to reduce the likelihood that the weakest models will be exploited. We show that we can indeed produce optimism with low model error, and thus match state of the art MBRL performance.

The rest of the paper is structured as follows: 1) We begin with background and related work, where we formally introduce the Optimism Decomposition; 2) In Section 3 we introduce noise augmented MDPs, and draw connections with existing algorithms; 3) We next provide our main theoretical results, followed by empirical verification in the tabular setting; 4) We rigorously evaluate the Optimism Decomposition in the deep RL setting, demonstrating the scalability of our approach; 5) We conclude and discuss some of the exciting future directions we hope to explore.

## 2 BACKGROUND AND RELATED WORK

In this paper we study a sequential interaction between a learner and a finite horizon MDP $M = (S, A, P, H, r, P_0)$, where $S$ denotes the state space, $A$ the actions, $P$ its dynamics, $H$ its episode horizon, $r \in \mathbb{R}^{[S \times |A|]}$ the rewards and $P_0$ the initial state distribution. For any state action pair $(s, a)$, we call $r(s, a)$ their true reward, which we assume to be a random variable in $[0, 1]$. $P$ represents the dynamics and defines the distribution over the next states, i.e., $s' \sim P(s, a)$ with probability $P(s, a, s')$. At the beginning of each round $k$, the learner computes a policy $\pi_k$ which it uses to collect rewards and transition tuples in $M$, for a total of $H$ steps. We use $k$ to denote the episode number and $h$ to index a timestep within an episode.

Since we do not know the true reward nor dynamics, we must instead approximate these through estimators. For state action pair $(s, a)$, we denote the average reward estimator as $\hat{r}_k(s, a) \in \mathbb{R}$ and the average dynamics estimator $\hat{P}_k(s, a) \in \Delta_{|S|}$, where index $k$ refers to the episode.

When training, the learner collects dynamics tuples during its interactions with $M_k$, which in turn it uses during each round $t$ to produce a policy $\pi_k$ and an approximate MDP $M_k = (S, A, \hat{P}, H, \hat{r}, P_0)$. In our theoretical results we will allow $\hat{P}(s, a)$ to be a signed measure whose entries do not sum to one. This is purely a semantic devise, rendering the exposition of our work easier and more general, and in no way affects the feasibility of our algorithms and arguments.

For any policy $\pi$, let $V(\pi)$ be the (scalar) value of $\pi$ and let $V_k(\pi)$ be the value of $\pi$ operating in the approximate MDP $M_k$. We define $E_{\pi}$ as the expectation under the dynamics of the true MDP $M$ and using policy $\pi$ (analogously $E_{\pi}$ as the expectation under $M_k$). The true and approximate value function for a policy $\pi$ are defined as follows:

$$V(\pi) = E_{\pi} \left[ \sum_{h=0}^{H-1} r(s_h, a_h) \right], V_k(\pi) = E_{\pi_k} \left[ \sum_{h=0}^{H-1} \hat{r}_k(s_h, a_h) \right].$$

We will evaluate our method using *regret*, the difference between the value of the optimal policy and the value from the policies it executed. Formally, in the episodic RL setting the regret of an agent using policies $\{\pi_k\}_{k=1}^K$ is (where $K$ is number of episodes and $T = KH$):

$$R(T) = \sum_{k=1}^K V(\pi^*) - V(\pi_k),$$

where $\pi^*$ denotes the optimal policy for $M$, and $V(\pi_k)$ is $\pi_k$ true value function. Furthermore, for each $h \in \{1, \ldots, H\}$ we call $V^h(\pi) \in \mathbb{R}^{[S]}$ the value vector satisfying $V^h(\pi)[s] = E_{\pi} \left[ \sum_{h'=h}^{H-1} r(s_{h'}, a_{h'}) | s_h = s \right]$, similarly we define $V_k^h(\pi) \in \mathbb{R}^{[S]}$ as $V_k^h(\pi)[s] = E_{\pi_k} \left[ \sum_{h'=h}^{H-1} \hat{r}(s_{h'}, a_{h'}) | s_h = s \right]$ where $V^h(\pi)[s] = 0$. Bold represents a vector-valued quantity.

The principle of optimism in the face of uncertainty (OFU) is used to address the exploration-exploitation dilemma in sequential decision making processes by performing both simultaneously. In RL, “model based” OFU algorithms [Jaksch et al., 2010; Fruit et al., 2018; Tossou et al., 2019] proceed as follows: at the beginning of each episode $k$ a learner selects an approximate MDP $M_k$ from a model cloud $M_k$ and a policy $\pi_k$ whose approximate value function $V_k(\pi_k)$ is optimistic, that is, it overestimates the optimal policy’s true value function $V(\pi^*)$. Our approach follows the same paradigm, but instead of using a continuum of models as in [Jaksch et al., 2010; Azar et al., 2017a] we allow $M_k$ to be a discrete set (i.e. an ensemble). For OFU inspired algorithms we re-write $R(T)$ as:

$$R(T) = \sum_{k=1}^K V(\pi^*) - V_k(\pi_k) + \sum_{k=1}^K \hat{V}_k(\pi_k) - V(\pi_k).$$

We refer to this as the *Optimism Decomposition*, since it breaks down the regret into its’ two major components. OFU
we are the first to propose algorithms for the deep RL setting. Two of the most prominent model-based OFU algorithms setting, opening the door to new scalable algorithms. However, our method is simple to implement in the deep RL setting, inspired by UCRL, but shift towards noise augmentation as in [Jaksch et al., 2010, Agrawal and Jia, 2017]. We are inspired by UCRL, but shift towards noise augmentation rather than an intractable model cloud. We thus call our algorithms Noise Augmented UCBVI Yes Model Based

Table 1: Prominent tabular RL algorithms and their noise augmented equivalents.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Scalable?</th>
<th>Model/Value Based</th>
<th>Regret Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>UCRL</td>
<td>No</td>
<td>Model Based</td>
<td>(\tilde{O}(</td>
</tr>
<tr>
<td>UCBVI</td>
<td>No</td>
<td>Model Based</td>
<td>(\tilde{O}(\sqrt{H</td>
</tr>
<tr>
<td>RLSVI</td>
<td>Yes</td>
<td>Value Based</td>
<td>(\tilde{O}(</td>
</tr>
<tr>
<td>Posterior Sampling</td>
<td>Yes</td>
<td>Model Based</td>
<td>(\tilde{O}(</td>
</tr>
<tr>
<td>Noise Augmented UCBVI</td>
<td>Yes</td>
<td>Model Based</td>
<td>(\tilde{O}(</td>
</tr>
</tbody>
</table>

In this paper we aim to shed light on how to transfer the principle of optimism into the realm of model based deep RL with deep function approximation. To our knowledge we are the first to propose algorithms for the deep RL setting inspired by the optimism principle prevalent throughout the theoretical RL literature.

Two of the most prominent model-based OFU algorithms are UCRL2 and UCBVI. In the case of UCRL2, optimism is produced by analytically optimizing over the entire dynamics uncertainty set. It is easy to see that this is intractable beyond the tabular setting. In the case of UCBVI, optimism is produced by adding a bonus directly at the value function level. This is also intractable in the deep RL setting as it requires a count model over the visited states and actions.

Optimism beyond tabular models has been theoretically studied in [Du et al., 2020], [Jin et al., 2019] that showed the value of OFU where the MDP satisfies certain linearity properties but their practical impact has been limited.

Other methods which have successfully scaled from the tabular setting to deep RL are Posterior Sampling and RLSVI [Osband et al., 2016b, Russo, 2019]. First we note that RLSVI is not model based in spirit and it certainly does not use a model ensemble. While RLSVI is a philosophically different algorithm to ours, they also propose the use of Gaussian noise perturbations and so it is closely related to our work.

Our approach is also inspired by [Agrawal and Jia, 2017] and [Xu and Tewari]. The parametric approach to posterior sampling studied in [Agrawal and Jia, 2017] can be easily analyzed under our framework, which can be seen as a generalization of the their posterior sampling algorithm. Crucially however, our method is simple to implement in the deep RL setting, opening the door to new scalable algorithms.

Next we show that optimism can be achieved by a simple noise augmentation procedure. This gives rise to provably efficient algorithms for tabular RL problems. We discuss variants of both UCRL and UCBVI which make use of this to simultaneously scale to deep RL while maintaining their theoretical guarantees.

### 3 ALGORITHMS

The aim of this section is to show that we can produce new versions of two popular OFU algorithms solely making use of noise augmentation. First, we focus on showing these noise augmented algorithms are theoretically competitive, before discussing practical implementations of our approach in the deep RL context.

**Algorithm 1 Noise Augmented RL (NARL)**

**Input:** Finite horizon MDP \(M = (S, A, P, H, r, P_0)\), Episodes \(K\), Initial reward and dynamics augmentation noise distributions \(\{P_r(s, a)\}_{s,a \in S \times A}\) and \(\{P_p(s, a)\}_{s,a \in S \times A}\), sampling frequencies \(M_r, M_p\).

**Initialize:** the transition and rewards data buffer \(D(s, a) = \emptyset\) for each \(s, a \in S \times A\). for \(k = 1, \ldots, K - 1\) do

1. (*\) Execute policy \(\pi_k\) for a length \(H\) episode and update \(D\).
2. Produce \(\{P_{r+1}(s, a)\}_{s,a \in S \times A}\) and (if UCRL) \(\{P_{p+1}(s, a)\}_{s,a \in S \times A}\).

Although we present our results for the case of undiscounted episodic reinforcement learning problems, our results extend to the average reward setting with bounded diameter MDPs as in [Jaksch et al., 2010, Agrawal and Jia, 2017]. We are inspired by UCRL, but shift towards noise augmentation rather than an intractable model cloud. We thus call our
approach Noise Augmented Reinforcement Learning or NARL.

NARL initializes an empty data buffer of rewards and transitions \( D \). We denote by \( N_k(s) \) the number of times state \( s \) has been encountered in the algorithm’s run up to the beginning of episode \( k \) (before \( \pi_k \) is executed). Similarly we call \( N_k(s, a) \) the number of times the pair \( (s, a) \) has been encountered up to the beginning of episode \( k \), and let \( N_k(s, a) = \sum_{a \in A} N_k(s, a) \). We make the following assumption:

**Assumption 1 (Rewards).** We assume the rewards are 1−sub Gaussian with mean values in \([0, 1]\).

### 3.1 CONCENTRATION

We start by recalling the mean estimators \( \{\hat{\pi}_k(s, a)\}_{(s, a) \in S \times A} \) and \( \{\hat{P}_k(s, a)\}_{(s, a) \in S \times A} \) concentrate around their true values. We make use of a time uniform concentration bound that leverages the theory of self normalization [Peña et al. 2008|Abbasi-Yadkori et al. 2011] to obtain the following:

**Lemma 1 (Lemma 1 of Maillard and Asadi 2018).** For all \((s, a) \in S \times A:\)

\[
\begin{align*}
\mathbb{P}( \forall t \in N \mid \left| r(s, a) - \hat{r}_k(s, a) \right| \geq \beta_k(N_k(s, a), \delta') \leq \delta, \\
\mathbb{P}( \forall t \in N \mid \left| P(s, a) - \hat{P}_k(s, a) \right|_1 \leq \beta_P(N_k(s, a), \delta') \leq \delta, \\
s.t. \beta_k(n, \delta') := \sqrt{\frac{\log(n/\delta')}{n}}, \quad \beta_P(n, \delta') := \sqrt{\frac{\log(n/\delta')}{n}}.
\end{align*}
\]

A more precise version of these bounds is stated in the Appendix. Equipped with these bounds, for the rest of the paper we condition on the event:

\[ \mathcal{E} := \{ k \in N \mid \forall(s, a) \in S \times A, \]
\[ \mid r(s, a) - \hat{r}_k(s, a) \mid \leq \beta_k(N_k(s, a), \delta'), \]
\[ \mid P(s, a) - \hat{P}_k(s, a) \mid_1 \leq \beta_P(N_k(s, a), \delta') \}. \]

If \( \delta' = \frac{\delta}{2|S||A|} \), Lemma 1 implies \( \mathbb{P}(\mathcal{E}) \geq 1 - \delta. \)

At the beginning of the \( k \)-th episode the learner produces \( M_r \) reward augmentation noise scalars \( \xi^m_k(s, a) \sim \mathcal{N}(0, \sigma^2_{r, \xi}(s, a)) \) and possibly \( M_P \) dynamics augmentation \(|S|\)-dimensional noise vectors \( \xi^m(s, a) \sim \mathcal{N}(0, \sigma^2_{r, \xi}(s, a)) \), for each state action pair \((s, a) \in S \times A\). This notation will become clearer in the subsequent discussion.

### 3.2 NOISE AUGMENTED UCRL

We start by showing that an appropriate choice for the noise variables \( \{\mathcal{N}(0, \sigma^2_{r, \xi}(s, a))\}_{s, a \in S \times A} \) and \( \{\mathcal{N}(0, \sigma^2_{r, \xi}(s, a))\}_{s, a \in S \times A} \) yields an algorithm akin to UCRL2 and with provable regret guarantees.

Our main result of this section (Theorem 1) states that in the tabular setting, if we set \( \mathcal{N}(0, \sigma^2_{r, \xi}(s, a)) \) and \( \mathcal{N}(0, \sigma^2_{r, \xi}(s, a)) \) for appropriate values of \( \sigma^2_{r, \xi}(s, a) \) and \( \sigma^2_{r, \xi}(s, a) \) we can obtain a regret guarantee of order \( \tilde{O}(|S|H\sqrt{|A|T}) \), which is competitive w.r.t. UCRL2 that achieves \( \bar{O}(|S|H\sqrt{|A|T}). \) These results can be extended beyond Gaussian noise augmentation provided the noise distributions satisfy quantifiable anticoncentration properties. For example, when using the dynamics noise given by posterior sampling of dynamics vectors, we recover the results of Agrawal and Jia [2017] in the episodic setting. Our results can be easily extended to the bounded diameter, average reward setting.

Noise Augmented Extended Value Iteration (NAEVI) proceeds as follows: at the beginning of episode \( k \) we compute a value function \( V_k \) as:

\[
\bar{V}_k(s) = \max_{a \in A} \left( \hat{r}_k(s, a) + \mathbb{E}_{\xi^m_k(s, a)} \left[ \mathcal{N}(0, \sigma^2_{r, \xi}(s, a)) \right] \right) + \max_{m} \xi^m_k(s, a) \cdot \bar{V}_k(s) \]

(2)

where \( A := \max_m \xi^m_k(s, a) \) and \( B := \max_m \xi^m_k(s, a) \). These results can be extended beyond Gaussian noise augmentation provided the noise distributions satisfy quantifiable anticoncentration properties. For example, when using the dynamics noise given by posterior sampling of dynamics vectors, we recover the results of Agrawal and Jia [2017]. Our results can be easily extended to the bounded diameter, average reward setting.

Although \( \bar{V}_k(s) \) may not be a probability measure, for convenience we still treat it as a signed measure and write \( \mathbb{E}_{\xi^m_k(s, a)} \left[ \cdot \right] := \langle \hat{P}_k(s, a) + \xi^m_k(s, a), \cdot \rangle. \) Let \( M_k = (S, A, \mathcal{P}, H, \bar{r}, P_0) \) be the approximate MDP resulting from collecting the maximizing rewards \( \bar{P}_k \) and dynamics vectors \( \bar{P}_k(s, a) \) while executing NAEVI. In other words, for any state action pair \((s, a) \in S \times A\):

\[
\bar{r}_k(s, a) = \max_{m = 1, \ldots, M_r} \bar{r}^m_k(s, a), \quad \bar{P}_k(s, a) = \mathbb{P}_{\{\bar{P}^m_k(s, a)\}_{m=1}^{M_r}} \mathcal{N}(0, \sigma^2_{r, \xi}(s, a)) \cdot \bar{V}_k(s) \]

(3)

Our main result in this section is the following theorem:
Theorem 1. Let $\epsilon \in (0, 1)$, $\delta = \frac{\epsilon}{4T}$, $M_r \geq \log \left( \frac{2|H|A|A|}{\epsilon^2} \right)$, and $M_P \geq 3 + \log \left( \frac{2|S|A|A|}{\epsilon^2} \right)$. The regret $R(T)$ of UCRL2 with Gaussian noise augmentation satisfies the following bound with probability at least $1 - \epsilon$:

$$R(T) \leq \tilde{O}(\sqrt{|S|H\sqrt{|A|T}})$$

where

$$\tilde{O}(\cdot)$$

hides logarithmic factors in $|A|$, $|S|$, $\epsilon$ and $T$ and:

$$\tilde{O}(\cdot)$$

hides logarithmic factors in $|A|$, $|S|$, $\epsilon$ and $T$ and:

$$\xi_{k,r}^{(m)} \sim \mathcal{N}(0, \sigma_r^2), \quad \text{s.t. } \sigma_r = 2\beta_r N_k(s, a), \quad \frac{\delta}{2|S||A|}$$

$$\xi_{k,r}^{(m)} \sim \mathcal{N}(0, \sigma_p^2), \quad \text{s.t. } \sigma_p = 2\beta_p \left( N_k(s, a) + \frac{\delta}{|S||A|} \right)$$

We remark these bounds are not optimal in $H$ and $S$, nevertheless, Theorem 1 shows this simple (and computationally scalable) noise augmented algorithm satisfies a regret guarantee. Our proof techniques are inspired but not the same as those of Agrawal and Jia [2017]. Our proofs proceed in two parts: returning to the Optimism Decomposition (Equation 1), we deal with the Optimism and Estimation Error separately. The details of all proofs are in the Appendix.

3.3 NOISE AUGMENTED UCBVI

In this section we show that a simple modification of the previous algorithm can yield an even stronger regret guarantee. The chief insight is to note that under Assumption 1 the scale of the value function is at most $H$ and therefore instead of adding dynamics noise vectors $\xi_k^{(m)}$ it is enough to simply scale up the variance of the reward noise components to ensure optimism at the value function level.

Noise Augmented Value Iteration (NAVI) proceeds as follows: at the beginning of episode $k$ we compute a $Q$-function $\tilde{Q}_k$ as:

$$\tilde{Q}_{k,h}(s, a) = \min \{ \tilde{Q}_{k-1,h}(s, a), H, \tilde{r}_k(s, a) \} +$$

$$\mathbb{E}_{a' \sim \pi_{k}(s, a)} \left[ \tilde{V}_{k,h+1}(s, a') \right]$$

$$\tilde{V}_{k,h}(s, a) = \max_{a \in \mathcal{A}} \tilde{Q}_{k,h}(s, a).$$

(4)

Where $\tilde{r}_k(s, a)$ is defined as in Equation 3. The policy executed at time $k$ by Noise Augmented UCBVI is the greedy policy w.r.t. $\tilde{Q}_{k,h}(s, a)$.

Our main result in this section is the following theorem:

Theorem 2. Let $\epsilon \in (0, 1)$, $\delta = \frac{\epsilon}{4T}$ and $M_r \geq \log \left( \frac{2|S|A|A|}{\epsilon^2} \right)$. The regret $R(T)$ of UCBVI with Gaussian noise augmentation satisfies the following bound with probability at least $1 - \epsilon$:

$$R(T) \leq \tilde{O}(\sqrt{|S|H\sqrt{|A|T}})$$

4 ANTI-CONCENTRATION AND OPTIMISM

The fundamental principle behind our bounds is that noise injection gives rise to optimism. In order to show this we rely on anti-concentration properties of the noise augmentation distributions. For the sake of simplicity we present simple results regarding Gaussian noise variables, their anti-concentration properties and one-step optimism. More nuanced results extending the discussion to Bootstrap sampling are in the Appendix.
Benign variance. We start by showing that whenever the noise is Gaussian and has an appropriate variance, with a constant probability each of the noise perturbed reward estimators \( r_k^{(m)} \) is at least as large as the empirical mean, plus the confidence radius \( \beta_r(N_k(s,a),\frac{\delta}{2|S||A|}) \). We can boost this probability by setting \( M_r \) to be sufficiently large. The main ingredient behind this proof is the following Gaussian anti-concentration result:

**Lemma 2. Lower bound on Gaussian density \( N(\mu,\sigma^2) \):**

\[
P(X - \mu > t) \geq \frac{1}{\sqrt{2\pi}} \frac{\sigma}{t + \sigma^2} e^{-\frac{t^2}{2\sigma^2}}.
\]

Using Lemma 2 we can show that as long as the standard deviation of \( r_k^{(m)}(s,a) \) is set to the right value, \( r_k^{(m)}(s,a) \) overestimates the true reward \( r(s,a) \) with constant probability.

**Lemma 3. Let \( (s,a) \in S \times A \). If \( r_k^{(m)}(s,a) \sim \tilde{r}_k(s,a) + N(0,\sigma^2) \) for \( \sigma = 2\beta_r(N_k(s,a),\frac{\delta}{2|S||A|}) \) then:**

\[
P(\tilde{r}_k^{(m)}(s,a) \geq r(s,a) | E) \geq \frac{1}{10}.
\]

Lemma 3 implies that with constant probability the values \( \tilde{r}_k^{(m)}(s,a) \) are an overestimate of the true rewards. It is also possible to show that despite this property, \( \tilde{r}_k(s,a) \) remain very close to \( \tilde{r}_k(s,a) \) and therefore to \( r(s,a) \). In summary:

**Corollary 1. The sampled rewards \( \tilde{r}_k(s,a) \) are optimistic:**

\[
P(\tilde{r}_k(s,a) = \max_{m=1,...,M_r} \tilde{r}_k^{(m)}(s,a) \geq r(s,a) | E) \geq 1 - \left(\frac{1}{10}\right)^{M_r}.
\]

while at the same time not being too far from the true rewards:

\[
P\left( |\tilde{r}_k(s,a) - r(s,a) | \geq L\beta_r(N_k(s,a),\frac{\delta}{2|S||A|}) \right) \leq \frac{\delta}{|S||A|}.
\]

Where \( L = \left( 2\sqrt{\log\left( \frac{4|S||A|\delta}{\sigma} \right)} + 1 \right) \).

Corollary 1 shows the trade-offs when increasing the number of models in an ensemble: it increases the amount of optimism, at the expense of greater estimation error of the sample rewards. A similar statement can be made of the dynamics in UCBRL, but we defer the details to the appendix.

**Bootstrap optimism** The necessary anti-concentration properties of the sampling distribution corresponding to Theorem 2 are explored in the Appendix.

**RLSVI Comparison** RLSVI’s regret bound is of the order of \( \mathcal{O}(H^3S\sqrt{AK}) \) while NARL-UCRL2 is of the order of \( \mathcal{O}(HS\sqrt{AK}) \) and NARL-UCBVI is of the order of \( \mathcal{O}(HS\sqrt{AK}) \). Removing an additional \( \sqrt{S} \) factor may prove more challenging since it is akin to the extra \( \sqrt{d} \) factor present in the worst case regret bounds for Thompson sampling in Linear bandits. Our rates for noise augmented NARL-UCBVI are superior to RLSVI, and even better in its \( H \) dependence than the latest regret bounds for the RLSVI setting (see Agrawal et al. [2020]). NARL and RLSVI are incomparable when moving into the function approximation regime since in this setting RLSVI will not be a model based algorithm. We also want to remark that the existing works on RLSVI are purely theoretical works and therefore there is no empirical evidence in neither Russo [2019] nor Agrawal et al. [2020] regarding its usefulness in a deep RL setting.

**Comparison to Thompson Sampling using Dirichlet prior:** As explained above, the objective of this work is not to be state of the art and get the optimal regret guarantees. Our dependence on \( S \) should be compared not with this approach but with RLSVI. Although the use of a Dirichlet prior allows the authors to get a better dependence on \( S \), the resulting algorithm is infeasible in the Deep RL paradigm. It is unclear what would the equivalent of maintaining such a prior be when making use of function approximation.

## 5 TABULAR EXPLORATION EXPERIMENTS

In this section we evaluate Noise Augmented UCRL (referred to as NARL) in the tabular setting, as is common for work in theoretical RL. We consider two implementations of NARL: (1) Gaussian, where we use use one model, but sample \( M = 10 \) noise vectors from a Gaussian distribution, with variance \( \frac{1}{N_k(s,a)} \) for a constant \( c \), which we set to 1, and (2) Bootstrap, where we maintain \( M = 10 \) models, each having access to 50% of the data. We compare these against UCRL2 [Jaksch et al. 2010] and Optimistic Posterior Sampling (OPSRL), using an open source implementation [1].

We begin with the RiverSwim environment [Strehl and Littman 2008], with 6 states and an episode length of 20. We repeat each experiment for 20 seeds, to produce a median and IQR. In Fig. 1(a) we see that both versions of NARL exhibit strong computational performance, while UCRL2 performs poorly. In Fig. 1(b) we explore why this is the case, and plot the approximate value function for UCRL2 and NARL. We see that the weak performance for UCRL2 likely comes from over-estimation, i.e. being overly optimistic, and it takes much longer to converge to the true value function. See the following link to run these experiments in a notebook: [https://bit.ly/3gVwsQP](https://bit.ly/3gVwsQP)

We also explore the choice of noise augmentation, using the Deep Sea environment [Osband et al. 2018] from bsuite [Osband et al. 2020]. This experiment shows the ability to scale with increasing problem dimension. We used ten

[https://github.com/osband/TabulaRL](https://github.com/osband/TabulaRL)
environments with \( N = \{10, \ldots, 28\} \). As we see in Fig. 1c) NARL solves all ten tasks. Interestingly, the Gaussian method is best, indicating promise for this approach.

Now we see NARL can compete empirically in the tabular setting, we next seek to demonstrate its scalability in the deep RL paradigm. Note that other methods, such as UCRL2, are intractable beyond tabular environments. Meanwhile, the noise augmentation we propose uses ingredients commonly found in state-of-the-art deep MBRL methods, such as bootstrap ensembles.

6 OPTIMISM IN DEEP RL

Despite being a popular theoretical approach, optimism is not prevalent in the deep RL literature. The most prominent theoretically motivated deep RL algorithm is Bootstrapped DQN [Osband et al., 2016a] which is inspired by PSRL. However, it is well-known that Q-functions generally overestimate the true Q-values [Thrun and Schwartz, 1993], therefore, many methods not using a lower bound (as used in TD3, Fujimoto et al. [2018]) may in fact be using an optimistic estimate. In recent times [Ciosek et al., 2019; Rashid et al., 2020] present model-free approaches using optimistic policies to explore by shifting Q-values optimistically based on epistemic uncertainty. However, as far as we are aware, optimism is not widely used w.r.t the dynamics in deep model based RL.

We know from our theoretical insights that an effective optimistic algorithm needs to balance the Optimism Decomposition. In the tabular setting we sought to add noise to boost the Optimism term, which led to too much variance in the case of UCRL2. For deep RL, the dynamics are very different, as we add significant noise from function approximation with neural networks. In this section we introduce a scalable implementation of NARL, which builds on top of the state-of-the-art continuous control (from states) MBRL algorithm. We also discuss the key factors influencing the Optimism Decomposition.

6.1 NOISE VIA BOOTSTRAPPED ENSEMBLES

We implement our algorithm in by using an ensemble, as is common in existing state-of-the-art methods [Janner et al., 2019; Clavera et al., 2018; Kurutach et al., 2018; Chua et al., 2018; Ball et al., 2020]. For our implementation, we focus on Janner et al. [2019], using probabilistic dynamics models [Nix and Weigend, 1994] and a Soft Actor Critic (SAC, Haarnoja et al., 2018a,b) agent learning inside the model.

Dyna-style approaches [Sutton, 1991], are particularly sensitive to model bias [Deisenroth and Rasmussen, 2011], which often leads to catastrophic failure when policies are trained on inaccurate synthetic data. To prevent this, state-of-the-art methods such as MBPO randomly sample models from the ensemble to prevent the policy exploiting an individual (potentially biased) model. Rather than randomly sampling, we follow the Noise Augmented UCRL approach (Equation 2) and pass the same state-action tuple through each model, and select the highest predicted reward, and assess which ‘hallucinated’ next state has the highest expected return according to the critic of the policy thus providing us with an optimistic estimate of the transition dynamics.

You’re only as good as your worst model: However, in the deep RL setting, we introduce a significant amount of noise due to function approximation with neural networks. It has
been observed in practice that this variance is sufficient to induce optimism [Osband et al. [2016a]]. A key consideration is the tendency for optimism to select the individual model with highest variance, resulting in over-exploitation of the least accurate models. In Fig. 2 we demonstrate this phenomenon, by training an ensemble of models (ordered here in increasing validation accuracy) and comparing the proportion each model was selected (red) against the average distance from the mean of the next state estimates (grey); we observe that these quantities are positively correlated. Thus, the optimistic approach selects (and exploits) the least accurate models.

**Reducing Estimation Error** To balance the Optimism Decomposition in deep RL, we must focus on Estimation Error. We introduce a “Model Radius Constraint” $\epsilon_M$: we calculate the empirical mean ($\mu_M$) of the expected returns, and exclude models that fall outside the permissible model sphere (defined as $\mu_M \pm \epsilon_M$) as being overly optimistic. Interestingly, an undocumented feature in MBPO [Janner et al., 2019] that mirrors this is the idea of maintaining a subset of “elite” models. Briefly, even though $K$ models are trained and maintained, in reality the top $E$ models are used for rollouts, where $E \leq K$. Even though models are sampled randomly in this approach, there is still a chance that “exploitable” samples are generated by these poorer performing models.

The introduction of $\epsilon_M$ allows us to reduce Estimation Error from optimism. We see in Fig. 2 that a wide radius ($\epsilon_M = 5$, blue) has a small impact on reducing usage of the worst model. However, when we set a small radius ($\epsilon_M = 0.1$, green), the models are selected almost uniformly. Details are in the Appendix.

### 6.2 Deep RL for Continuous Control

Now we evaluate the deep RL implementation of NARL. We focus on the InvertedPendulum task, as it is the simplest continuous environment and thus allows us to perform rigorous ablation studies. We run ten seeds for a variety of configurations, selecting the number of models $M$ from $\{3, 5, 10\}$ and $\epsilon_M$ from $\{0.1, 5, \text{None}\}$. These two parameters trade-off the amount of variance in the ensemble. Having more models means more noise. In addition, having a smaller $\epsilon_M$ will reduce variance. The results are presented in Table 2.

Interestingly, we see strong evidence for our hypothesis that too much variance is a problem in the deep RL setting. This results in the phenomenon whereby having fewer models (e.g. $M = 3$) actually gives better performance, which has an added benefit of reduced computational cost. This is in contrast to methods based on random ensemble sampling [Kurutach et al., 2018] [Osband et al., 2016a], where performance typically increases with the number of models. When using more models, the smaller model radius ($\epsilon_M$) is crucial.

Table 2: The mean number of timesteps to solve the InvertedPendulum task, with standard deviations.

<table>
<thead>
<tr>
<th>$\epsilon_M$</th>
<th>3</th>
<th>5</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>1850 ±166</td>
<td>2350 ±300</td>
<td>3300 ±953</td>
</tr>
<tr>
<td>5</td>
<td>2000±274</td>
<td>2575 ±251</td>
<td>3225 ±675</td>
</tr>
<tr>
<td>None</td>
<td>2000 ±353</td>
<td>2775 ±467</td>
<td>5850 ±2037</td>
</tr>
</tbody>
</table>

In Fig. 3 we compare NARL against the publicly released data from MBPO [Janner et al., 2019] on the InvertedPendulum, Hopper and HalfCheetah environments. For InvertedPendulum we show the results with $M = 3$ and $\epsilon_M = 0.1$, the strongest result from Table 2. With these parameters selected appropriately, we get meaningful gains against a very strong baseline, using an almost identical implementation aside from the model selection and $\epsilon_M$. For the larger Hopper and HalfCheetah tasks, we also used $M = 3$ and selected $\epsilon_M$ from $\{0.1, 0.5\}$. Again, we are able to perform favorably vs. MBPO, demonstrating the potential for our approach to scale to larger environments. This performance comes despite using over 50% fewer models than MBPO (3 models vs. 7).
We do not claim these results are state of the art, but highlight the design choices considered when using optimism for deep MBRL. In these settings we have been able to show that if variance can be controlled (e.g. by using $\epsilon_M$) then optimism can perform comparably well with the best random-sampling method.

**Author Contributions**

Aldo Pacchiano, Philip Ball, and Jack Parker-Holder provided an equal contribution to this paper.

**References**


