Supplementary Material for IJCAI Submission

Anonymous Authors

February 25, 2019

Abstract

1 Finite Horizon Learning-to-learn Framework

The setup of the problem begins with a given machine learning application with finite time or sampling budget, such that simply relying on asymptotic convergence may not work within the budget. We assume that the underlying application can be modeled as a discrete Markov decision process with finite time horizon T. We define the terminology for the underlying learning problem in Table 1. The goal of the underlying problem is to learn the optimal state-to-action mapping that maximizes the cumulative reward gained by the agent over a finite T time steps.

We assume that a machine learning algorithm is given to solve the underlying application, such that the algorithm asymptotically finds the optimal parameter θ^* that maximizes the finite horizon expected sum of reward function shown as:

$$\mathbb{E}\left[\sum_{t=0}^{T-1} C\left(S_t, X\left(S_t|\theta^*, \rho\right), W_{t+1}\right)\middle| S_0, \rho\right], \tag{1}$$

for any tunable parameter value ρ chosen from set $\mathcal{P}$. We use the term "learning to act" to designate the learning policy $\Theta\left(S|\rho\right)$ that updates the learnable parameter $\theta_t = \Theta\left(S_t|\rho\right)$ iteratively every t.

The focus of this paper is to learn ρ of (1) in a practical application scenario within a given budget. Let N be the number of different parameter settings to be tried, such that NT is the given budget, as each trial on the underlying application consumes T units of budget. We use the term "learning to learn" to designate the meta-learning policy that iteratively updates the tunable parameter ρ^n to try in n-th trial. We define the "learning to learn optimally" problem as a finite horizon sequential decision problem, such that the goal is to optimize the expected cumulative rewards from online meta-learning process of

Table 1: Terminology from the underlying problem modeled as an MDP

Symbol	**Meaning**
$\mathcal{S}$	state space from underlying MDP, $\forall s \in \mathcal{S}$
$\mathcal{X}$	decision space defined from underlying MDP, $\forall x \in \mathcal{X}$
T	finite time horizon from underlying objective function, $0 \leq t \leq T$
W_t	exogenous information from underlying MDP that becomes known at time t, $\mathfrak{F}_t$-measurable.
W_t^n	exogenous information that becomes known at time $nT + t$.
W^{n+1}	$:= \left\{W_0^n, W_1^n, \cdots W_{T-1}^n\right\}$, all exogenous information that becomes known at the end of iteration n.
$C(S, X, W)$	contribution function of decision X in state S, with randomness W
$\hat{C}_{t+1}$	$\sim C(s_t, x_t, W_{t+1})$, observed contribution given decision x_t in state s_t.
$T(S, X, W)$	state transition function of decision X in state S, with randomness W
$\hat{T}_{t+1}$	$\sim T(s_t, x_t, W_{t+1})$, observed state transition given decision x_t in state s_t. Note $s_{t+1} = \hat{T}_{t+1}$.
$X(s\|\theta, \rho)$	decision function given parameters θ, ρ. A function: $\mathcal{S} \mapsto \mathcal{X}$
$\Theta(S\|\rho)$	update rule of θ given tunable parameter ρ. A function: $\mathcal{S} \mapsto \Theta$
θ_t	$= \Theta^{\pi}(s_t)$. learnable parameter of X at time t
ρ	(non-learnable) tunable parameter of X
$\mathcal{P}$	Finite set of tunable parameters. $\rho \in \mathcal{P}$

length NT steps as:

$$\overbrace{\max_{\pi\in\Pi^{\rho}}\mathbb{E}\left[\overbrace{\sum_{n=0}^{N-1}\underbrace{\sum_{t=0}^{T-1}C\left(S_t^n,X\left(S_t^n|\theta_t^n,\rho^n\right),W_{t+1}^n\right)}_{\text{learning to act}}}^{\text{learning to learn}}\middle|S_0^{\rho,0}\right]}^{\text{learning to learn optimally}}, \tag{2}$$

where the outer finite horizon online optimization "learning to learn" problem encapsulates the online rewards from the underlying "learning to act" MDP that models the machine learning application. For every increment of n (or every T time steps), an LTLO policy $\pi \in \Pi^{\rho}$ will choose a new parameter ρ^n for the underlying adaptive bidding policy to use during next T time steps. $S_t^{\rho,n}$ stands for the LTLO state variable at time t of iteration n, which corresponds to $nT+t$ time steps from the beginning. W_t^n is the exogenous information that becomes known at time t of iteration n.

The LTLO policy $\pi \in \Pi^{\rho}$ determines the update rule $\boldsymbol{\rho}^{\pi}(\cdot)$ of ρ such that for $n \in \{0, 1, \ldots, N-1\}$:

$$\rho^n = \boldsymbol{\rho}^{\pi}\left(S_0^{\rho,n}|\lambda\right), \tag{3}$$

where λ stands for any tunable parameter for the LTLO policy.

2 Proofs

Lemma 1. *The marginal distribution of μ_2^n follows a truncated t-distribution with $\nu = 2a^n$ degrees of freedom.*

Proof. We integrate out μ_0^n, μ_1^n from the multivariate normal distribution to and get the normal inverse gamma distribution:

$$\mu_2^n, \left(\sigma^n\right)^2 \sim NIG\left(m_2^n, V_{33}^n, a^n, b^n\right), \tag{4}$$

where V_{33} is the $(3,3)$ element of V, and m_2^n is the mean parameter of μ_2^n, the third element of μ^n.

We then integrate out σ^2 from the normal inverse gamma distribution to get a truncated t-distribution:

$$\mu_2 \sim t_{\nu}\left(m_2, \Sigma\right), \tag{5}$$

where $\Sigma = \frac{b^n}{a^n} V_{33}^n$ and degree of freedom $\nu = 2a^n$. Due to the constraint on μ_2, the pdf for μ_2 is equal to that of the above t-distribution for $\mu_2 \leq 0$, otherwise it is 0. This results in a truncated t-distribution as the marginal distribution of μ_2.

□

Lemma 2. *Given $k \in \mathbb{R}$ and $\nu \geq 1$, function $f(z)$ for all $z \in \{0, 1, \cdots\}$, defined as*

$$f(z, k, \nu) := \nu^{-\frac{z+1}{2}} \int_{-\infty}^{k} x^z \left(1 + \frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} dx, \tag{6}$$

can be evaluated in closed form expressions as:

$$f(z, k, \nu) = \begin{cases} \frac{1}{2}(-1)^z B_{inc}\left(\frac{\nu-z}{2}, \frac{z+1}{2}, \frac{\nu}{\nu+k^2}\right) \\ -\frac{1}{2} B_{inc}\left(\frac{\nu-z}{2}, \frac{z+1}{2}, \frac{\nu}{\nu+k^2}\right) \\ B\left(\frac{\nu-z}{2}, \frac{z+1}{2}\right) - \frac{1}{2} B_{inc}\left(\frac{\nu-z}{2}, \frac{z+1}{2}, \frac{\nu}{\nu+k^2}\right) \end{cases}, \tag{7}$$

where the first case is for $k \leq 0$, the second case is for $k > 0$ and z odd, and the third case is for $k > 0$ and z even. Note that $B_{inc}(x, y, u) := \int_0^u t^{x-1}(1-t)^{y-1} dt$ is incomplete Beta function and $B(x, y) = B_{inc}(x, y, 1) = \int_0^1 t^{x-1}(1-t)^{y-1} dt$ is Euler Beta function.

Proof. We split the proof into two cases: $k \leq 0$ and $k > 0$, and evaluate the integral by substituting $y = \left(1 + \frac{x^2}{\nu}\right)^{-1}$.

In the first case of $k \leq 0$, the substitution of x with y leads to:

$$x = -\left(\left(y^{-1} - 1\right)\nu\right)^{\frac{1}{2}}, \tag{8}$$

since the integration range for $x \in (-\infty, k]$ where $k \leq 0$. Therefore, we get the substitution coefficient as:

$$\frac{dx}{dy} = -\frac{1}{2}\nu^{\frac{1}{2}}\left(y^{-1} - 1\right)^{-\frac{1}{2}} \frac{d}{dy}\left(y^{-1} - 1\right) \tag{9}$$

$$= -\frac{1}{2}\nu^{\frac{1}{2}}(1-y)^{-\frac{1}{2}} y^{\frac{1}{2}} \frac{d}{dy}\left(y^{-1} - 1\right) \tag{10}$$

$$= -\frac{1}{2}\nu^{\frac{1}{2}}(1-y)^{-\frac{1}{2}} y^{\frac{1}{2}}\left(-y^{-2}\right) \tag{11}$$

$$= \frac{1}{2}\nu^{\frac{1}{2}}(1-y)^{-\frac{1}{2}} y^{-\frac{3}{2}}. \tag{12}$$

The integration range after substitution is $y \in \left(0, \frac{\nu}{\nu+k^2}\right]$. For brevity, we let $u := \frac{\nu}{\nu+k^2}$ in the rest of this proof. We evaluate $f(z)$ using integration by

substitution as:

$$f(z,k,\nu) := \nu^{-\frac{z+1}{2}} \int_{-\infty}^{k} x^z \left(1+\frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} dx \tag{13}$$

$$= \nu^{-\frac{z+1}{2}} \int_{0}^{u} (-1)^z \left(y^{-1}-1\right)^{\frac{z}{2}} \nu^{\frac{z}{2}} y^{\frac{\nu+1}{2}} \left(\frac{1}{2}\nu^{\frac{1}{2}} (1-y)^{-\frac{1}{2}} y^{-\frac{3}{2}}\right) dy \tag{14}$$

$$= \nu^{-\frac{z+1}{2}} \int_{0}^{u} (-1)^z y^{-\frac{z}{2}} (1-y)^{\frac{z}{2}} \nu^{\frac{z}{2}} y^{\frac{\nu+1}{2}} \left(\frac{1}{2}\nu^{\frac{1}{2}} (1-y)^{-\frac{1}{2}} y^{-\frac{3}{2}}\right) dy \tag{15}$$

$$= \frac{1}{2}(-1)^z \int_{0}^{u} y^{\frac{\nu-z}{2}-1} (1-y)^{\frac{z+1}{2}-1} dy \tag{16}$$

$$= \frac{1}{2}(-1)^z B_{inc}\left(\frac{\nu-z}{2}, \frac{z+1}{2}, u\right). \tag{17}$$

In the second case of $k > 0$, the substitution of x with y where $k > 0$ leads to:

$$x = \left(\left(y^{-1}-1\right)\nu\right)^{\frac{1}{2}}, \tag{18}$$

for the integration range $x \in (0, k]$. Thus, in the corresponding range $y \in (1, u]$, we also get the substitution coefficient with opposite sign from the first case as:

$$\frac{dx}{dy} = \frac{1}{2}\nu^{\frac{1}{2}} \left(y^{-1}-1\right)^{-\frac{1}{2}} \frac{d}{dy}\left(y^{-1}-1\right) \tag{19}$$

$$= \frac{1}{2}\nu^{\frac{1}{2}} (1-y)^{-\frac{1}{2}} y^{\frac{1}{2}} \frac{d}{dy}\left(y^{-1}-1\right) \tag{20}$$

$$= \frac{1}{2}\nu^{\frac{1}{2}} (1-y)^{-\frac{1}{2}} y^{\frac{1}{2}} \left(-y^{-2}\right) \tag{21}$$

$$= -\frac{1}{2}\nu^{\frac{1}{2}} (1-y)^{-\frac{1}{2}} y^{-\frac{3}{2}}. \tag{22}$$

On the other hand, in the remaining integration range $x \in (-\infty, 0]$, the substitution is the same as the first case of $k \leq 0$. Finally, we evaluate $f(z)$ for $k > 0$

using integration by substitution for each of the two ranges as:

$$f(z,k,\nu) := \nu^{-\frac{z+1}{2}} \int_{-\infty}^{k} x^z \left(1+\frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} dx \tag{23}$$

$$= \nu^{-\frac{z+1}{2}} \int_{-\infty}^{0} x^z \left(1+\frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} dx + \nu^{-\frac{z+1}{2}} \int_{0}^{k} x^z \left(1+\frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} dx \tag{24}$$

$$= \frac{1}{2}(-1)^z \int_0^1 y^{\frac{\nu-z}{2}-1} (1-y)^{\frac{z+1}{2}-1} dy - \frac{1}{2}\int_1^u y^{\frac{\nu-z}{2}-1} (1-y)^{\frac{z+1}{2}-1} dy \tag{25}$$

$$= \frac{1}{2}(-1)^z \int_0^1 g(y)dy - \frac{1}{2}\int_1^u g(y)dy \tag{26}$$

$$= \frac{1}{2}(-1)^z \int_0^1 g(y)dy - \frac{1}{2}\left(\int_0^u g(y)dy - \int_0^1 g(y)dy\right) \tag{27}$$

$$= \frac{1}{2}\left((-1)^z+1\right)\int_0^1 g(y)dy - \frac{1}{2}\int_0^u g(y)dy, \tag{28}$$

where we abbreviate the expression using $g(y) := y^{\frac{\nu-z}{2}-1}(1-y)^{\frac{z+1}{2}-1}$. Note that the integrals can be evaluated using the definition of Beta function as $\int_0^1 g(y)dy = B\left(\frac{\nu-z}{2}, \frac{z+1}{2}\right)$ and the definition of incomplete Beta function as $\int_0^u g(y)dy = B_{inc}\left(\frac{\nu-z}{2}, \frac{z+1}{2}, u\right)$. Therefore, when $k > 0$ and z is odd, the first term disappears in equation (28), resulting in:

$$f(z,k,\nu) = -\frac{1}{2}B_{inc}\left(\frac{\nu-z}{2}, \frac{z+1}{2}, u\right) \quad (k>0 \text{ and } z \text{ odd}), \tag{29}$$

and when $k > 0$ and z is even, equation (28) becomes:

$$f(z,k,\nu) = B\left(\frac{\nu-z}{2}, \frac{z+1}{2}\right) - \frac{1}{2}B_{inc}\left(\frac{\nu-z}{2}, \frac{z+1}{2}, u\right) \quad (k>0 \text{ and } z \text{ even}), \tag{30}$$

where $u := \frac{\nu}{\nu+k^2}$ as introduced in the proof. □

Lemma 3. *The posterior mean of non-positive constrained μ_2^n is*

$$\mathbb{E}[\mu_2^n] = m_2^n + \sqrt{\nu}\frac{f(1,k,\nu)}{f(0,k,\nu)}\sqrt{\frac{b^n}{a^n}V_{33}^n}, \tag{31}$$

where $f(z,k,\nu)$ is evaluated using lemma 2, $k = \frac{m_2^n}{\sqrt{\frac{a^n}{b^n}V_{33}^n}}$, and $\nu = 2a^n$.

Proof. This proof is inspired by the original proof of Property 3 in [O'hagan, 1973], which contains a formula to compute generalized n-th moment estimator. We

correct several typos found in the original proof and present the proof in greater detail, adapted to iterative Bayesian inference setup.

In this proof, for brevity, we use shorthands: β instead of μ_2, and μ instead of $\bar{\mu}_2$, such that $\beta \sim t_\nu(\mu, \Sigma)$. We first transform β into another random variable X that follows standard t distribution with ν degrees of freedom:

$$X := \frac{\beta - \mu}{\sqrt{\frac{b}{a}V}} \sim t_\nu, \tag{32}$$

where $a = a^n, b = b^n, V = V_{33}^n$ and $\nu = 2a^n$.

Now we impose the nonnegative constraint $\beta \leq 0$ on X, which leads to $X \leq -\frac{\mu}{\sqrt{\frac{b}{a}V}}$. For brevity, let $k := -\frac{\mu}{\sqrt{\frac{b}{a}V}}$. With this constraint, X, which now follows a truncated standard t distribution, has a pdf that can be written as:

$$\mathbb{P}[X = x] = \begin{cases} \frac{1}{C}\left(1 + \frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} & (x \leq k) \\ 0 & (x > k) \end{cases}, \tag{33}$$

where the normalizing constant $C := \int_{-\infty}^{k}\left(1 + \frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} dx$.

Now we derive a closed form expression of the expectation of X.

$$\mathbb{E}[X] := \int_{-\infty}^{\infty} x\mathbb{P}[X = x]\, dx \tag{34}$$

$$= \frac{1}{C}\int_{-\infty}^{k} x\left(1 + \frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} dx \tag{35}$$

$$= \frac{\int_{-\infty}^{k} x\left(1 + \frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} dx}{\int_{-\infty}^{k}\left(1 + \frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} dx} \tag{36}$$

$$= \nu^{\frac{1}{2}}\frac{f(1, k, \nu)}{f(0, k, \nu)}, \tag{37}$$

where $f(z, k, \nu)$ can be evaluated in closed-form expression as shown in lemma 2.

According to the definition of X, the expectation of β can be computed from the expectation of X as:

$$\mathbb{E}[\beta] = \mu + \mathbb{E}[X]\sqrt{\frac{b}{a}V} \tag{38}$$

$$= \mu + \nu^{\frac{1}{2}}\frac{f(1, k, \nu)}{f(0, k, \nu)}\sqrt{\frac{b}{a}V}. \tag{39}$$

□

Lemma 4. *The conditional distribution of $\sigma^2|\mu_2$ follows a scaled chi-square distribution with $\eta+1$ degrees of freedom. The posterior mean of σ^2 given μ_2 is*

$$\mathbb{E}\left[\sigma^2\middle|\bar{\mu}_2^n\right] = \left(b^n + \frac{\left(\bar{\mu}_2^n - \mu_2^n\right)^2}{2V_{3,3}^n}\right)\frac{\Gamma\left(\frac{\eta-1}{2}\right)}{\Gamma\left(\frac{\eta+1}{2}\right)}, \tag{40}$$

where $\bar{\mu_2}$ is the marginal posterior mean of μ_2, $V_{3,3}^n$ is the $(3,3)$ element of V^n, and $\Gamma(z) := \int_0^\infty x^{z-1}\exp(-x)\,dx$ is Gamma function.

Proof. This lemma is shown as Property 4 in [O'hagan, 1973]. □

Lemma 5. *The conditional distribution of $\mu_0, \mu_1|\bar{\mu}_2, \bar{\sigma}^2$ follows a 2-dimensional normal distribution. The posterior means of μ_0 and μ_1 are:*

$$\mathbb{E}\left[[\mu_0,\mu_1]^\top\middle|\bar{\mu}_2^n, \bar{\sigma}^{2,n}\right] = m_{1:2}^n + \frac{\bar{\mu_2}^n - \mu_2^n}{V_{3,3}^n}V_{1:2,3}^n, \tag{41}$$

and the posterior variance matrix of μ_0 and μ_1 is:

$$\mathrm{Var}\left[[\mu_0,\mu_1]^\top\middle|\bar{\mu}_2^n, \bar{\sigma}^{2,n}\right] = \bar{\sigma}^{2,n}\left(V_{1:2,1:2}^n - \frac{V_{2:3,1}^n V_{1,2:3}^n}{V_{3,3}^n}\right). \tag{42}$$

Proof. This lemma is shown as Property 1 in [O'hagan, 1973]. □

3 Algorithm Subroutines

In this section, we present the details of the subroutines `Constrain` and `BU`, each in Algorithm 1 and 2.

Algorithm 1 `Constrain`: Belief Constraining

Require: B^n

1: Let $f(z,k,\nu)$ as defined in Lemma 2
2: $\bar{\mu}_2 \leftarrow m_2^n + \sqrt{\nu}\frac{f(1,k,\nu)}{f(0,k,\nu)}\sqrt{\frac{b^n}{a^n}V_{33}^n}$
3: $\bar{\sigma^2} \leftarrow \left(b^n + \frac{(\bar{\mu}_2-\mu_2^n)^2}{2V_{3,3}^n}\right)\frac{\Gamma\left(\frac{\eta-1}{2}\right)}{\Gamma\left(\frac{\eta+1}{2}\right)}$
4: $\bar{\mu}_1 \leftarrow m_1^n + \frac{\bar{\mu_2}-\mu_2^n}{V_{3,3}^n}V_{2,3}^n$
5: $\bar{\mu}_0 \leftarrow m_0^n + \frac{\bar{\mu_2}-\mu_2^n}{V_{3,3}^n}V_{1,3}^n$
6: Let $\bar{\mu} = [\bar{\mu}_0, \bar{\mu}_1, \bar{\mu}_2]^\top$ and $\bar{B}^n := \left(\bar{\mu}, \bar{\sigma^2}\right)$
7: **return** $\bar{B}^n$

4 Description of Test Environments

In this section, we provide detailed description on the test environments.

Algorithm 2 `BU`: Bayesian Update of Belief State

Require: $B^n, \rho^n, \hat{F}^{n+1}$

Let $\vec{\rho}^n := \left[1, \rho^n, (\rho^n)^2\right]^\top$

$V^{n+1} \leftarrow \left((V^n)^{-1} + \vec{\rho}^n \left(\vec{\rho}^n\right)^\top\right)^{-1}$

$m^{n+1} \leftarrow V^{n+1}\left((V^n)^{-1} m^n + \vec{\rho}^n \hat{F}^{n+1}\right)$

$a^{n+1} \leftarrow a^n + \frac{1}{2}$

$b^{n+1} \leftarrow b^n + \frac{1}{2}\left((m^n)^\top (V^n)^{-1} m^n + \left(\hat{F}^{n+1}\right)^2 - \left(m^{n+1}\right)^\top \left(V^{n+1}\right)^{-1} m^{n+1}\right)$

Let $B^{n+1} := \left(m^{n+1}, V^{n+1}, a^{n+1}, b^{n+1}\right)$

return B^{n+1}

4.1 Cliffwalking

The Cliffwalking environment models a navigation task described in [Sutton and Barto, 1998]. It is modeled as a 4×12 gridworld where the agent starts from the bottom left corner to reach the bottom right corner. The rest of the bottom row is a "cliff", entering which will reset the agent back to the starting point. The agent can move in four directions: up, down, left, and right. Each move incurs a loss of -1, and the agent can move up to 200 times until the end of episode. If the agent reaches the goal, then the agent rests there until the end of the episode. Otherwise, the episode ends with all the movement costs as the episode performance. The episode performance is the average per-step reward, divided by 100 and incremented by 1 to normalize the episode performance to other experiments.

4.2 WindyGrid

The WindyGrid environment models a navigation task with exogenous bias in transition. It is modeled as a 7×10 gridworld, in which the agent starts at $(3, 0)$ and the goal is located at $(3, 7)$. The agent attempts to move in one of the four directions with a loss of -1 per move, but this time, wind is present and the movement is adjusted by the wind. There is upward wind near the goal, such that the columns $3, 4, 5, 8$ in the gridworld has upward wind of strength 1 and the columns 6 and 7 has upward wind of strength 2. When an agent lands on any of such columns, its coordinates are adjusted upward by the wind strength. The goal of the agent is to reach the goal within 200 moves. As soon as the agent reaches the goal, the episode ends and no more losses due to additional movement are incurred in that episode. The episode performance is measured and normalized the same way as the Cliffwalking environment.

4.3 FrozenLake

The FrozenLake environment models a stochastic navigation task to reach the goal while avoiding falling into holes on a frozen lake. The movement may slip with significant probability such that the intended move is performed only one third of the time, and the two moves orthogonal to the intended move gets executed with probability $\frac{1}{3}$, respectively. So the control in this environment is very uncertain. The goal of the agent is to reach the goal without falling into any of the holes before 200 moves. When the agent falls into a hole, then the agent restarts from the starting point. There is no explicit cost in either moving or falling into a hole, and there is reward of 1 for reaching the goal. The episode performance is the average per-step reward, which favors agents reaching the goal in fewer steps, and it is normalized by multiplying with 20.

4.4 N-Chain

The N-Chain environment, described in [Strens, 2000], is a benchmark for sufficient exploration before exploitation. It models N states connected as a chain, where the agent starts from one end of the chain. The agent can move left or right, and a small amount of stochasticity is added by allowing the move to slip with probability 0.05, such that every move is executed as intended with 95% chance, and otherwise executed in opposite direction. In particular, we use a chain with 5 states, with reward of 10 for reaching the other end of the chain, and smaller reward of 1 for returning to the starting end of the chain. The intermediate nodes of the chain has zero reward, and if the agent reaches the far end of the chain, it collects the large reward and restarts at the starting end. As this task has no goal state that ends the episode early, all agents have 200 moves, during which it can repeatedly collect rewards. The episode performance is the average per-step reward scaled by dividing it with 1000, so it favors agents reaching out to the far end and collecting large reward over agents exploiting small reward by returning to the starting point immediately.

References

[O'hagan, 1973] O'hagan, A. (1973). Bayes estimation of a convex quadratic. *Biometrika*.

[Strens, 2000] Strens, M. (2000). A Bayesian Framework for Reinforcement Learning. In *Proceeedings of the Seventeenth International Conference on Machine Learning (ICML-2000)*.

[Sutton and Barto, 1998] Sutton, R. and Barto, A. (1998). Reinforcement Learning: An Introduction. *IEEE Transactions on Neural Networks*, 9(5):1054–1054.