Multiple Model Q-Learning for Stochastic Reinforcement Delays

Jeffrey S. Campbell
Carleton University
Ottawa, Ontario, Canada
Email: campbell.jeffrey.scott@gmail.com

Sidney N. Givigi
Royal Military College of Canada
Kingston, Ontario, Canada
Email: sidney.givigi@rmc.ca

Howard M. Schwartz
Carleton University
Ottawa, Ontario, Canada
Email: schwartz@sce.carleton.ca

I. INTRODUCTION

Reinforcement learning (RL) is a machine learning control scheme. It is useful for applications where an agent must learn from its interactions with an environment for which it does not have a complete model and lacks supervision. In RL, the agent learns to map actions to states based on feedback in the form of a reinforcement signal (reward) [1]–[3]. Intuitively, the agent is rewarded for desirable behaviour, and punished for behaving poorly.

II. THE PROBLEM

A. Q-learning

In this RL algorithm, from which our algorithm is extended, the agent updates a table Q with entries $Q(s,a)$. When the agent transitions from state s at time t to new state s' at time $(t+1)$, having taken action a at time t, the table is updated according to:

$$\begin{aligned} Q_{k+1}(s,a) \leftarrow\, & Q_k(s,a) \\ & + \alpha[r + \gamma \max_a Q_k(s',a) - Q_k(s,a)] \end{aligned} \tag{1}$$

where $0 \leq \alpha < 1$ is a learning rate, γ is a discount factor, and r is a reward received at time t. Over time, the Q-table will converge to estimate the true value of actions within various states, and thus find the optimal policy $a = \pi^*(s)$ [1].

B. Variably Delayed Markov Decision Processes

We define a Variably Delayed Markov Decision Process (VDMDP) to be a MDP where the reward signal suffers from a Poissonian time delay. This means that if the agent performs an action, it will not receive the corresponding reward until the variable delay has elapsed. In other words, rewards and actions are asynchronous. Normally, rewards are received immediately following an action. A VDMDP can be characterized by a 6-tuple (S,A,P,R,γ,λ) where S is the set of states within the environment, A is the set of actions the agent may choose, and P maps $S \times A \times S \mapsto [0,1]$ which is the probability that taking action $a \in A$ while in state $s \in S$ will lead to state $s' \in S$. R is defined as the reward signal which maps $S \mapsto \mathbb{R}$, γ is the discount factor to be applied to future rewards [4], and λ represents the mean and variance of the Poisson distribution for the time delay.

III. SIMULATION ENVIRONMENTS

Q-learning simulations were conducted in grid-world environments of differing sizes with variable delays introduced. In particular, square grid-worlds of size 3×3, 5×5, and 9×9 were used.

In these environments, the agent begins in a random state. The top left-most state yields a reward of 10 for any action and moves the robot to the bottom right-most state. Bumping into a wall yields a reward of -1. Diagonal moves are not allowed.

These simulations concluded that as the environment size increased, and as the variable delay increased, the performance of the learning agent decreased relative to that of an optimal undelayed policy for each respective environment. Thus, the problem of stochastically delayed rewards becomes more important for environments with more states and larger delays.

IV. MULTIPLE-MODEL Q-LEARNING

Building upon the Q-learning algorithm from [1], we proposed that the agent learn as detailed in Algorithm 1. The new idea is that we introduce a new learning dimension $\hat{\lambda}$ in addition to s and a. Think of the system as having many $Q(s,a)$ tables in parallel. At each time step, a candidate model $\hat{\lambda}$ is selected and its Q-table is made active for decision making during that time step. Next, all models (active and inactive) are updated according to their own respective delay assumptions, taking into account the action chosen by the active model.

V. CONVERGENCE

Multiple-model Q-learning converges in a stochastic sense, as shown in [5]. The proof is an extension of the work in [6], which states that if four assumptions are true, and if the update equation takes the form $x_i = x_i + \alpha(F_i(x) - x_i + w_i)$, then $x(t)$ converges to x^* with probability 1. Therefore, $Q_k(s,a,\hat{\lambda})$ converges to $Q(s,a,\hat{\lambda})^*$ with probability 1 as $k \to \infty$.

VI. RESULTS

Simulations were conducted in a 9 x 9 grid world to evaluate how well the novel algorithm generates a control policy. The simulations were performed in a grid world environments to demonstrate the significant effect of stochastic time delays even in a relatively small and simple environment. The solutions described are intended to be carried on to other more

Algorithm 1 - Multiple Model Q-Learning

```
Set $\hat{\lambda}_{\max}$
Initialize state-action memory of sufficient length
Initialize $\alpha, \gamma$ (e.g. $\alpha = 0.1, \gamma = 0.9$)
Initialize $Q(s, a, \hat{\lambda})$ arbitrarily (e.g. optimistic initialization)
for episode do
  Initialize $s$
  repeat
    select $\hat{\lambda}^*$ using policy from Q (e.g. $\varepsilon$-greedy)
    select $a^*$ using policy from Q assuming $\hat{\lambda}^*$ is correct (e.g. $\varepsilon$-greedy)
    execute action $a^*$
    observe $r_t$ and $s'$
    for $\hat{\lambda}$ from 0 to $\hat{\lambda}_{max}$ do
      $Q_{k+1}(s_{t-\hat{\lambda}}, a_{t-\hat{\lambda}}, \hat{\lambda}) \leftarrow (1-\alpha) Q_k(s_{t-\hat{\lambda}}, a_{t-\hat{\lambda}}, \hat{\lambda}) + \alpha \left[ r_t + \gamma \max_{\hat{\lambda}, a} Q_k(s'_{t-\hat{\lambda}}, a, \hat{\lambda}) \right]$
    end for
    $s \leftarrow s'$
  until $s$ is terminal or behaviour is acceptable
end for
```

complex applications, especially in mobile robotics, where grid world simulations are useful for testing algorithms.

During simulation, the mean delay is set to $\lambda = 13$, and the highest feasible delay estimate λ_{max} is set to 50 time steps. Other parameters are set to the example values in Algorithm 1.

Fig. 1 shows the improved performance from using the multiple-model Q-learning algorithm in a 9×9 grid world. The multiple-model Q-learning method achieves a performance of about 95% relative to normal Q-learning in an undelayed environment. In comparison, Q-learning achieves a performance of about 0% relative to the undelayed environment.

Next, we form a control policy using the maximally valuable entries in the Q-tables for both algorithms. The policies are shown in Fig. 2. Note that multiple-model Q-learning follows an optimal path while Q-learning has become stuck in a loop.

VII. CONCLUSION

By introducing parallel Q-tables along the new time delay learning dimension, multiple model Q-learning allows the agent to learn more from the same experience despite the presence of disruptive time delays in the reward signal.

Future work will explore new ways of processing the multiple delay estimate updates in parallel rather than in serial fashion, since the updates are independent. It may also be fruitful to investigate using a kernel to assign less reward to delay estimates which are far from the most valuable delay estimate, $\hat{\lambda}^*$. We hope that these ideas will make learning faster.

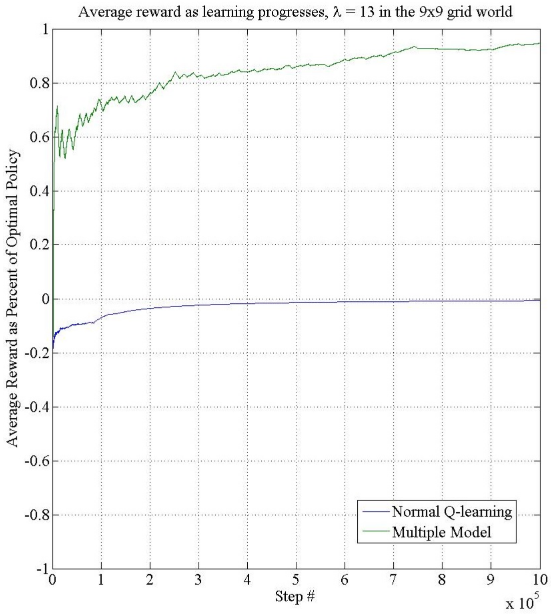

Fig. 1. Relative performance of multiple-model Q-learning vs single-model Q-learning in the 9×9 grid world with random delays present

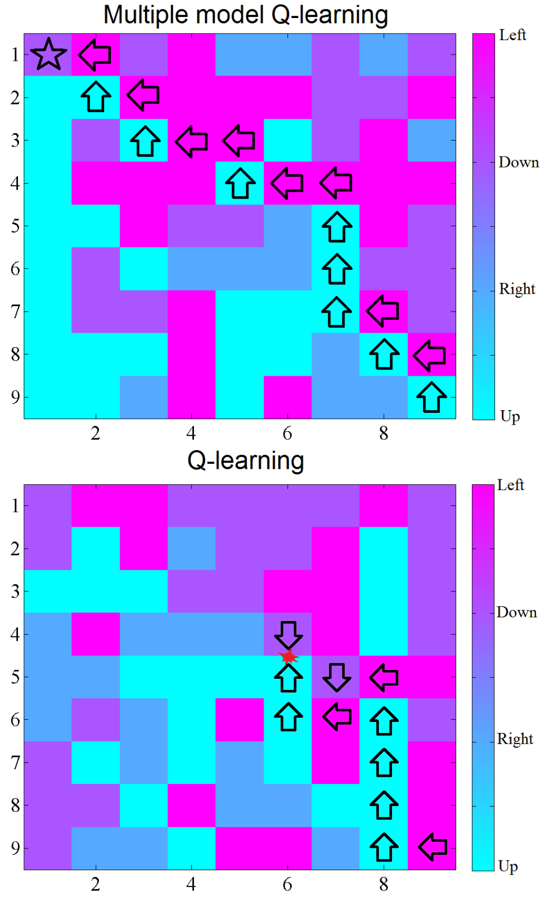

Fig. 2. Deterministic policy formed using each algorithm

VIII. THE TEAM

The Autonomous Robotics Research Group (ARRG) includes researchers in three institutions, the Royal Military College of Canada (RMCC), Carleton University and Queens University. The main interest of the group is in machine learning, especially Reinforcement Learning, and multiple robotics, from ground robots to Unmanned Aerial Vehicles (UAVs) and Autonomous Underwater Vehicles (AUVs). It counts with state-of-the-art lab facilities located at RMCC with more than a dozen ground robots and ten UAVs.

REFERENCES

[1] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction.* Cambridge, MA: MIT Press, 1998.

[2] P. Kulkarni, “Introduction to Reinforcement and Systemic Machine Learning,” Reinforcement and Systemic Machine Learning, pp. 1-21, 2012.

[3] M. P. Deisenroth, G. Neumann, J. Peters, “A Survey on Policy Search for Robotics,” Foundations and Trends in Robotics, 2011.

[4] T. Walsh, A. Nouri, L. Li, and M. Littman, “Planning and learning in environments with delayed feedback,” Machine Learning: ECML 2007, Jan. 2007.

[5] J. S. Campbell, S. N. Givigi and H. M. Schwartz, “Multiple Model Q-learning for Stochastic Time-Delayed Reinforcement Learning,” submitted to Journal of Intelligent & Robotic Systems, 2014.

[6] J. N. Tsitsiklis, “Asynchronous Stochastic Approximation and Q-learning”, Machine Learning, 16, 1994, pp. 185-202.

[7] J. S. Campbell, S. N. Givigi and H. M. Schwartz, “Multiple Model Q-learning for Stochastic Reinforcement Delays,” IEEE Systems, Man, and Cybernetics, 2014.

Bios

Second Lieutenant Jeffrey S. Campbell is a pilot in the Royal Canadian Air Force. He completed his undergraduate degree in 2012 at the Royal Military College of Canada in Kingston, Canada. While there, he specialized in robotic control and worked on quadrotor unmanned aerial vehicles. In 2014 he received his master’s degree in electrical engineering from Carleton University in Ottawa, Canada. His work there focused on unsupervised machine learning with applications in mobile robotics. Jeff is currently posted at Defence Research and Development Canada to work on over-the-horizon radar projects.

Sidney N. Givigi received his B.Sc. in Computer Science and an M.A.Sc. in Electrical Engineering from the Federal University of Esp*í*rito Santo, Brazil. He also received his Ph.D. in Electrical and Computer Engineering from Carleton University, Canada. In 2009, he joined the Department of Electrical and Computer Engineering of the Royal Military College of Canada (RMCC) as an Assistant Professor. Sidney’s research interests are mainly focused on autonomous systems, especially the decentralized control of multiple vehicles, learning and adaptation of autonomous robots and modeling of complex systems with Game Theory.

Professor H.M. Schwartz received his B.Eng. degree from McGill University, Montreal, Quebec and his M.S. degree and Ph.D. degree from M.I.T., Cambridge, Massachusetts. His research interests include adaptive and intelligent control systems, robotics, system modelling and system identification. His most recent research is in multi agent learning with applications to teams of mobile robots.