Compound Reinforcement Learning: Theory and An Application to Finance

Tohgoroh Matsui[1], Takashi Goto[2], Kiyoshi Izumi[3,4], and Yu Chen[3]

[1]Chubu University,
1200 Matsumoto-cho, Kasugai, 487-8501 Aichi, Japan
TohgorohMatsui@tohgoroh.jp, http://tohgoroh.jp

[2]Bank of Tokyo-Mitsubishi UFJ, Ltd.
2-7-1 Marunouchi, Chiyoda, 100-8388 Tokyo, JAPAN
takashi_6_gotou@mufg.jp

[3]The University of Tokyo
7-3-1 Hongo, Bunkyo, 113-8656 Tokyo, JAPAN
{izumi@sys.t, chen@k}.u-tokyo.ac.jp

[4]PRESTO, JST
Sanban-cho Building 5F, 3-5, Sanban-cho, Chiyoda, 102-0075 Tokyo, Japan

Abstract. This paper describes compound reinforcement learning (RL) that is an extended RL based on the compound return. Compound RL maximizes the logarithm of expected double-exponentially discounted compound return in return-based Markov decision processes (MDPs). The contributions of this paper are (1) Theoretical description of compound RL that is an extended RL framework for maximizing the compound return in a return-based MDP and (2) Experimental results in an illustrative example and an application to finance.

Keywords: Reinforcement learning, compound return, value functions, finance

1 Introduction

Reinforcement learning (RL) has been defined as a framework for maximizing the sum of expected discounted rewards through trial and error [14]. The key ideas in RL are, first, defining the value function as the sum of expected discounted rewards and, second, transforming the optimal value functions into the Bellman equations. Because of these techniques, some good RL methods, such as temporal difference learning, that can find the optimal policy in Markov decision processes (MDPs) have been developed. Their optimality, however, is based on the expected discounted rewards. In this paper, we focus on the compound return[1]. The aim of this research is to maximize the compound return by extending the RL framework.

In finance, the compound return is one of the most important performance measures for ranking financial products, such as mutual funds that reinvest their gains or losses. It

[1] Notice that the "return" is used in financial terminology in this paper, whereas the return is defined as the sum of the rewards by Sutton and Barto [14] in RL.

is related to the geometric average return, which takes into account the cumulative effect of a series of returns. In this paper, we consider tasks that we would face a hopeless situation if we fail once. For example, if we were to reinvest the interest or dividends in a financial investment, the effects of compounding interest would be great, and a large negative return would have serious consequences. It is therefore important to consider the compound returns in such tasks.

The gains or losses, that is, the rewards, would be increased period-by-period, if we reinvested those gains or losses. In this paper, we consider return-based MDPs instead of traditional reward-based MDPs. In return-based MDPs, the agent receives the simple net returns instead of the rewards, and we assume that the return is a random variable that has Markov properties. If we used an ordinary RL method for return-based MDPs, it would maximize the sum of expected discounted returns. However, the compound return could not be maximized.

Some variants of the RL framework have been proposed and investigated. Average-reward RL [6, 12, 13, 15] maximizes the arithmetic average rewards in reward-based MDPs. Risk-sensitive RL [1, 2, 5, 7, 9, 11] not only maximizes the sum of expected discounted rewards, it also minimizes the risk defined by each study. While they can learn risk-averse behavior, they do not take into account maximizing the compound return.

In this paper, we describe an extended RL framework, called "compound RL", that maximizes the compound return in return-based MDPs. In addition to return-based MDPs, the key components of compound RL are double exponential discounting, logarithmic transformation, and bet fraction. In compound RL, the value function is based on the logarithm of expected double-exponentially discounted compound return and the Bellman equation of the optimal value function. In order to avoid the values diverging to negative infinity, a bet fraction parameter is used.

The key contributions of this paper are: (1) Theoretical description of compound RL that is an extended RL framework for maximizing the compound return in a return-based MDP and (2) Experimental results in an illustrative example and an application to finance. Firstly, we illustrate the difference between the compound return and the rewards in the next section. We then describe the framework of compound RL and a compound Q-learning algorithm that is an extension of Q-learning [17]. Section 5 shows the experimental results, and finally, we discuss our methods and conclusions.

2 Compound Return

Consider a two-armed bandit problem. This bandit machine has two big wheels, each with six different paybacks, as shown in Figure 1. The stated values are the amount of payback on $1 bet. The average payback for $1 from wheel A is $1.50, and that from wheel B is $1.25. If we had $100 at the beginning and played 100 times with $1 for each bet, wheel A would be better than wheel B. The reason is simply that the average profit of wheel A is greater than that of wheel B. Figure 2 shows two example performance curves when we bet $1 on either wheel A or B, 100 times. The final amounts of assets are near the total expected payback.

However, if we bet all money for each bet, then betting on wheel A would not be the optimal policy. The reason is that wheel A has a zero payback and the amount of money

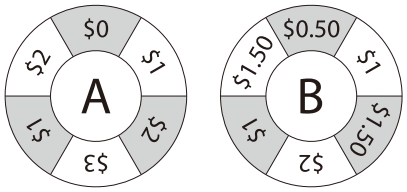

Fig. 1. Two-armed bandit with two wheels, A and B. The stated values are the amount of payback on a $1 bet. The average payback of wheel A is $1.50, and that of wheel B is $1.25.

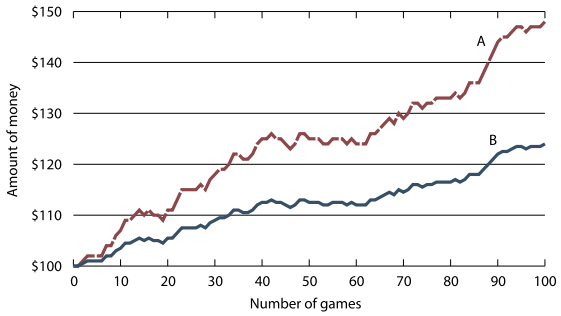

Fig. 2. Two example performance curves when we have $100 at the beginning, and we bet $1, 100 times, on either wheel A or B.

will become zero in the long term. In this case, we have to consider the compound return that is correlated to the geometric average rate of return:

$$G = \left(\prod_{i=1}^{n} (1 + R_i) \right)^{1/n} - 1, \tag{1}$$

where R_i is the i-th rate of return, and n represents the number of periods. Let P_t be the price of an asset that an agent has in time t. Holding the asset from one period, from time step $t-1$ to t, the "simple net return" or "rate of return" is calculated by

$$R_t = \frac{P_t - P_{t-1}}{P_{t-1}} = \frac{P_t}{P_{t-1}} - 1 \tag{2}$$

and $1 + R_t$ is called the "simple gross return." The compound return is defined as follows [3] :

$$(1 + R_{t-n+1})(1 + R_{t-n+2}) \dots (1 + R_t) = \prod_{i=1}^{n} (1 + R_{t-n+i}). \tag{3}$$

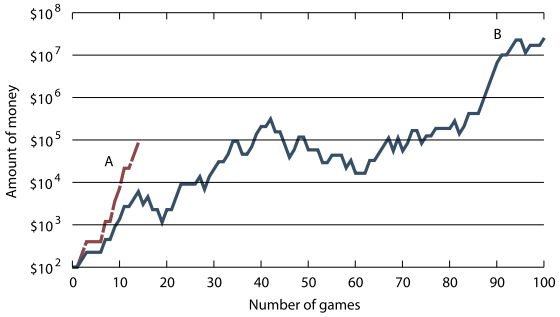

Fig. 3. Two example performance curves when we have \$100 at the beginning, and bet all money, 100 times, on either wheel A or B.

Whereas the geometric average rate of return of wheel A is -1, that of wheel B is approximately 0.14. Figure 3 shows two example performance curves when we have \$100 at the beginning, and bet all money, 100 times, on either wheel A or B. The reason the performance curve of wheel A stops is that the bettor lost all his or her money when the payback was zero. Note that it has a logarithmic scale vertically. If we choose wheel B, then the expected amount of money at the end would be as much as \$74 million.

As we see here, considering the compound return, maximizing the sum of expected discounted rewards is not useful. It is a general idea that the compound return is important in choosing mutual funds for financial investment [10]. Therefore, the RL agent should maximize the compound return instead of the sum of the expected discounted rewards in such cases.

3 Compound RL

Compound RL is an extension of the RL framework to maximize the compound return in return-based MDPs. We firstly describe return-based MDP, and then the framework of compound RL.

Consider the next return at time step t:

$$R_{t+1} = \frac{P_{t+1} - P_t}{P_t} = \frac{P_{t+1}}{P_t} - 1. \tag{4}$$

In other words, $R_{t+1} = r_{t+1}/P_t$, where r_{t+1} is the reward. The future compound return is written as

$$\rho_t = (1 + R_{t+1})(1 + R_{t+2}) \dots (1 + R_T), \tag{5}$$

where T is a final time step. For continuing tasks in RL, we consider that T is infinite; that is,

$$\begin{aligned}\rho_t =&(1+R_{t+1})(1+R_{t+2})(1+R_{t+3})\dots \\ =&\prod_{k=0}^{\infty}(1+R_{t+k+1}).\end{aligned} \tag{6}$$

In return-based MDPs, R_{t+k+1} is a random variable, $R_{t+k+1} \geq -1$, that has Markov properties.

In compound RL, double-exponential discounting and bet fraction are introduced in order to prevent the logarithm of the compound return from diverging. The double-exponentially discounted compound return with bet fraction is defined as follows:

$$\begin{aligned}\rho_t =&(1+R_{t+1}f)(1+R_{t+2}f)^{\gamma}(1+R_{t+3}f)^{\gamma^2}\dots \\ =&\prod_{k=0}^{\infty}(1+R_{t+k+1}f)^{\gamma^k},\end{aligned} \tag{7}$$

where f is the bet fraction parameter, $0 < f \leq 1$. The logarithm of ρ_t can be written as

$$\begin{aligned}\log\rho_t =&\log\prod_{k=0}^{\infty}(1+R_{t+k+1}f)^{\gamma^k} \\ =&\sum_{k=0}^{\infty}\log(1+R_{t+k+1}f)^{\gamma^k} \\ =&\sum_{k=0}^{\infty}\gamma^k\log(1+R_{t+k+1}f).\end{aligned} \tag{8}$$

The right-hand side of Equation (8) is same as that of simple RL, in which the reward, r_{t+k+1}, is replaced with the logarithm of simple gross return; $\log(1+R_{t+k+1}f)$. If $\gamma < 1$, then the infinite sum of the logarithm of simple gross return has a finite value as long as the return sequence $\{R_k\}$ is bounded. In compound RL, the agent tries to select actions in order to maximize the logarithm of double-exponentially discounted compound return it gains in future. It is equal to maximizing the double-exponentially discounted compound return.

Discounting is also a financial mechanism and it is called time preference in economics. Discounting in simple RL is called exponential discounting in economics. The double-exponentially discounted return can be considered as a kind of risk-adjusted returns in finance and it also can be considered as a kind of temporal discounting in economics. Figure 4 shows the difference between double exponential discounting and ordinary exponential discounting when $R_{t+k+1} = 1, 0.5$, and 0.1. The compound RL's double exponential discounting curve is very similar to the simple RL's exponential discounting curve when $|R_{t+k+1}|$ is small.

The bet fraction is the fraction of our asset that we place on a bet or in an investment. The Kelly criterion [8], which is well known in finance, is a formula used to determine the bet fraction that maximizes the expected logarithm of wealth when the accurate win

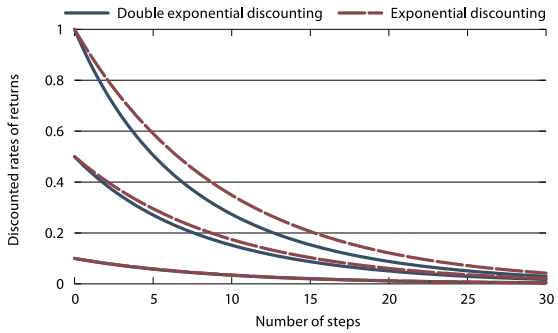

Fig. 4. Double exponential discounting and exponential discounting.

probability and return are known. Since we cannot know the accurate win probability and return a priori, we use a parameter for the bet fraction.

In compound RL, the value of state s under a policy π is defined as the expected logarithm of double-exponentially discounted compound return under π:

$$\begin{aligned} V^{\pi}(s) =& \mathrm{E}_{\pi}\left[\log \rho_t \middle| s_t = s\right] \\ =& \mathrm{E}_{\pi}\left[\log \prod_{k=0}^{\infty} (1 + R_{t+k+1} f)^{\gamma^k} \middle| s_t = s\right] \\ =& \mathrm{E}_{\pi}\left[\sum_{k=0}^{\infty} \gamma^k \log(1 + R_{t+k+1} f) \middle| s_t = s\right], \end{aligned}$$

this can be written in a similar fashion as simple RL:

$$\begin{aligned} =& \mathrm{E}_{\pi}\left[\log(1 + R_{t+1} f) + \gamma \sum_{k=0}^{\infty} \gamma^k \log(1 + R_{t+k+2} f) \middle| s_t = s\right] \\ =& \sum_{a \in \mathcal{A}(s)} \pi(s,a) \sum_{s' \in \mathcal{S}} \mathcal{P}^a_{ss'} \left(\mathsf{R}^a_{ss'} + \gamma \mathrm{E}_{\pi}\left[\sum_{k=0}^{\infty} \gamma^k \log(1 + R_{t+k+2} f) \middle| s_{t+1} = s'\right]\right) \\ =& \sum_{a} \pi(s,a) \sum_{s'} \mathcal{P}^a_{ss'} \left(\mathsf{R}^a_{ss'} + \gamma V^{\pi}(s')\right), \end{aligned} \tag{9}$$

where $\pi(s,a)$ is selection probability and $\pi(s,a) = \Pr\left[a_t = a \middle| s_t = s\right]$, $\mathcal{P}^a_{ss'}$ is transition probability, $\mathcal{P}^a_{ss'} = \Pr\left[s_{t+1} = s' \middle| s_t = s, a_t = a\right]$, and $\mathsf{R}^a_{ss'}$ is the expected logarithm of simple gross return, $\mathsf{R}^a_{ss'} = \mathrm{E}\left[\log(1 + R_{t+1} f) \middle| s_t = s, a_t = a, s_{t+1} = s'\right]$. Equation (9) is the Bellman equation for V^{π} in compound RL. Similarly, the value of action a in state s can be defined as follows:

$$\begin{aligned} Q^{\pi}(s,a) =& \mathrm{E}_{\pi}\left[\log \rho_t \middle| s_t = s, a_t = a\right] \\ =& \sum_{s' \in \mathcal{S}} \mathcal{P}^a_{ss'} \left(\mathsf{R}^a_{ss'} + \gamma V^{\pi}(s')\right). \end{aligned} \tag{10}$$

Algorithm 1 Compound Q-Learning

Input: discount rate γ, step size α, bet fraction f
Initialize $Q(s, a)$ arbitrarily, for all s, a
loop {for each episode}
  Initialize s
  repeat {for each step of episode}
    Choose a from s using policy derived from Q (e.g., ϵ-greedy)
    Take action a, observe return R, next state s'
    $Q(s,a) \leftarrow Q(s,a) + \alpha\left[\log(1+Rf) + \gamma \max_{a'} Q(s',a') - Q(s,a)\right]$
    $s \leftarrow s'$
  until s is terminal
end loop

4 Compound Q-learning

As we have seen above, in compound RL, the Bellman optimality equations are the same in form as simple RL. The difference in the Bellman optimality equation between compound RL and simple RL is that the expected simple gross return, $\log(1 + \mathsf{R}^a_{ss'})$, is used in compound RL instead of the expected rewards, $\mathcal{R}^a_{ss'}$. Therefore, most of the algorithms and techniques for simple RL are applicable to compound RL, by replacing the reward, r_{t+1}, with the logarithm of simple gross return, $\log(1 + R_{t+1}f)$. In this paper, we show the Q-learning algorithm for compound RL and the convergence in return-based MDPs.

Q-learning [17] is one of the most well-known basic RL algorithms, which is defined by

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left(r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)\right), \tag{11}$$

where α is a parameter, called step-size, $0 \leq \alpha \leq 1$. We extend the Q-learning algorithm for traditional RL to one for compound RL, which we have called "compound Q-learning." In this paper, traditional Q-learning is called "simple Q-learning," to distinguish it from compound Q-learning.

Compound Q-learning is defined by

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left(\log(1 + R_{t+1}f) + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)\right). \tag{12}$$

Equation (12) is same as Equation (11), replacing r_t with $\log(1 + R_t f)$. The procedural form of the compound Q-learning algorithm is shown in Algorithm 1.

In this paper, we focus on return-based MDPs; we assume that the rate of return R_{t+1} has Markov properties, that is, R_{t+1} depends only on s_t and a_t. In return-based MDPs, we can show the convergence of compound Q-learning. Compound Q-learning replaces the rewards, r_{t+1}, in simple Q-learning with the logarithm of simple gross return, $\log(1 + R_{t+1}f)$. On the other hand, the Bellman equation for the optimal action

value function Q^* in compound RL also replaces the expected rewards, $\mathcal{R}^a_{ss'}$, in simple RL with the expected logarithm of simple gross return, $\mathsf{R}^a_{ss'}$. Therefore, considering the logarithm of simple gross return in compound RL as rewards in simple RL, the action values Q approach to the optimal action values Q^* in compound RL.

More strictly, rewards are limited to be bounded in the Watkins and Dayan's convergence of simple Q-learning. We, therefore, have to limit the logarithm of the simple gross return to be bounded; that is, $1+R_{t+1}f$ is greater than 0, and has an upper bound, in a return-based MDP. Thus, we will prove the following theorem.

Theorem 1. *Given bounded return* $-1 \leq R_t \leq \mathrm{R}$*, bet fraction* $0 < f \leq 1$*, step size* $0 \leq \alpha_t < 1$*, and* $0 < 1 + R_t f$*,* $\sum_{i=1}^{\infty} \alpha_{t^i} = \infty$*,* $\sum_{i=1}^{\infty} [\alpha_{t^i}]^2 < \infty$*, then* $\forall s, a[Q_t(s,a) \rightarrow Q^*(s,a)]$ *with probability* 1*, in compound Q-learning, where* R *is the upper bound of* R_t*.*

Proof. Let $r_{t+1} = \log(1 + R_{t+1}f)$, then the update equation of compound Q-learning shown in Equation (12) is equal to that of simple Q-learning. Since $\log(1 + R_t f)$ is bounded, we can prove Theorem 1 by replacing r_t with $\log(1 + R_t f)$ in the Watkins and Dayan's proof [17] .

5 Experimental Results

5.1 Two-Armed Bandit

Firstly, we compared compound Q-learning and simple Q-learning, using the two-armed bandit problem described in Section 2. Each agent has \$100 at the beginning and plays 100 times. The reward for simple Q-learning is the profit for betting \$1, that is, the payback minus \$1. The rate of return for compound Q-learning is the profit divided by the bet value of \$1, that is, the same value as the rewards for simple Q-learning in this task. We set the discount rate of $\gamma = 0.9$ for both. The agents used ϵ-greedy selection, with $\epsilon = 0.1$, while learning, and chose actions greedily while evaluating them. The step-size parameter was $\alpha = 0.01$, and the bet fraction was $f = 0.99$. These parameters were selected empirically. For each evaluation, 251 trials were independently ran in order to calculate the average performance. We carried out 101 runs with different random seeds and got the average.

The results are shown in Figure 5. The left graph compares the geometric average returns, which means the compound return per period. The right graph compares the arithmetic average rewards. Compound Q-learning converged to a policy that chose wheel B, and simple Q-learning converged to one that chose wheel A. Whereas the arithmetic average return of the simple Q-learning agent was higher than that of compound Q-learning, the geometric average return was better from compound Q-learning than from simple Q-learning.

5.2 Global Bond Selection

Secondly, we investigated the applicability of compound Q-learning to a financial task: global government bonds selection. Although government bonds are usually considered

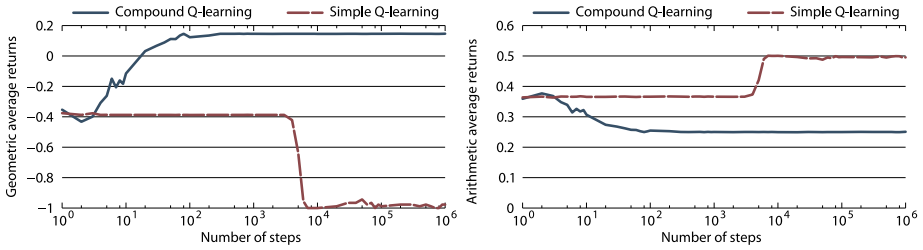

Fig. 5. Results for the two-armed bandit experiment.

Table 1. The yields and default probabilities in the global bond selection.

Country	Yields	Default Prob.
USA	0.01929	0.036
Germany	0.02222	0.052
UK	0.02413	0.064

as risk-free bonds, they still have default risks, that is, governments may fail to pay back its debt in full when economic or financial crisis strikes the country. Therefore, we have to choose bonds considering the yields and the default risks.

In this task, an agent learns to choose one from three 5-year government bonds: USA, Germany, and UK. The yields and default probabilities are shown in Table 1. We obtained the yields of 5-year government bonds on 31st December 2010 from the web site of the Wall Street Journal, WSJ.com. The 5-year default probabilities were obtained from the CMA global sovereign credit risk report [4], which were calculated based on the closing values on 31st December 2010 by CMA.

Because the interest of government bonds are paid every half year, we calculated the half-year default probabilities based on the 5-year default probabilities, assuming that it occurs uniformly. In this task, when a default occurs, the principal is reduced by 75% and the rest of the interest is not paid. For example, when you choose German government bond and its default occurs in the second period, the return would be $0.01111 - 0.75 = -0.73889$, where 0.01111 is the interest for the first half-year. For simplicity, time-varying of the yields, the default probabilities, and the foreign exchange rates are not considered. We thus formulated a global government bonds selection task as a three-armed bandit task. The parameters for compound RL and simple RL were $\gamma = 0.9$, $f = 1.0$, $\alpha = 0.001$, and $\epsilon = 0.2$.

Figure 6 shows the learning curves of the geometric average returns (left) and the arithmetic average returns (right). Although the geometric average return of simple Q-learning did not increase, that of compound Q-learning increased. The proportion of learned policies are shown in Figure 7. Simple Q-learning could not converge a definite policy because of the very nearly equal action values based on the arithmetic average returns. On the other hand, it shows compound Q-learning acquired policies that choose

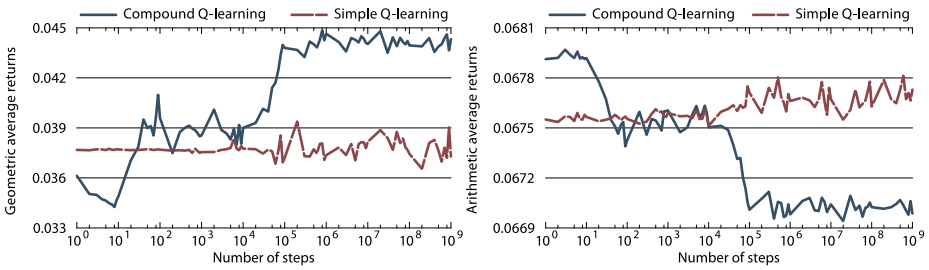

Fig. 6. Results in the global bond selection.

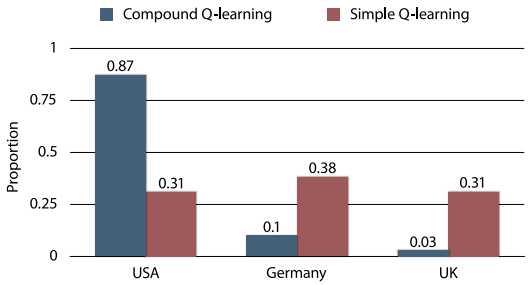

Fig. 7. Proportion of acquired policies in the global government bonds selection.

U.S. government bond in most cases. It was the optimal policy based on the compound return.

6 Discussion and Related Work

In compound RL, the simple rate of return, R, is transformed into the reinforcement defined by the logarithm of simple gross return, $\log(1 + Rf)$, where f is a bet fraction parameter, $0 < f \leq 1$. Compared with the reinforcement for simple RL, the logarithmic transformation suppresses positive reinforcement and increases negative reinforcement. Figure 8 shows the difference between the reinforcement of compound RL and that of simple RL. The effect of logarithmic transformation becomes larger when the bet fraction f increases.

On the other hand, the bet fraction is a well-known concept in avoiding over-investing in finance. The bet fraction that maximizes the investment effect can be calculated if the probability distribution is known [8]. It is called the Kelly criterion. Vince proposed a method, called "optimal f," which estimates the Kelly criterion and chooses a bet fraction based on the estimation [16]. It is, therefore, a natural idea to introduce a bet fraction parameter to compound RL.

There are some related work. Risk-sensitive RL [1, 2, 5, 7, 9, 11] not only maximizes the sum of discounted rewards, but it also minimizes the risk. Because the invest-

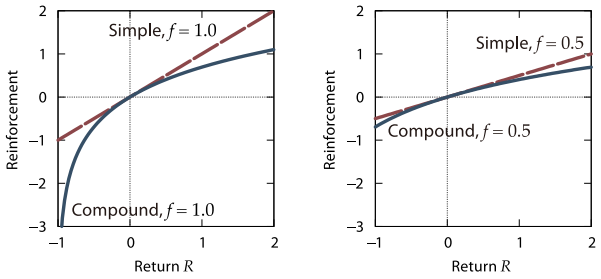

Fig. 8. The difference between the reinforcement for compound RL and that for simple RL.

ment risk is generally defined as the variance of the returns in finance, the expected-value-minus-variance criterion [7, 11] seems to be suitable for financial applications. Schwartz's R-learning [12] maximizes the average rewards instead of the sum of discounted rewards and Singh modified the Bellman equation [13]. Tsitsiklis and Van Roy analytically compared the discounted and average reward temporal-difference learning with linearly parameterized approximations [15]. Gosavi proposed a synchronous RL algorithm for long-run average reward [6]. However, risk-sensitive RL and average-reward RL are not effective for maximizing the compound return.

7 Conclusion

In this paper, we described compound RL that maximizes the compound return in return-based MDPs. We introduced double exponential discounting and logarithmic transformation of the double-exponentially discounted compound return, and defined the value function based on these techniques. We formulated the Bellman equation for the optimal value function using the logarithmic transformation with a bet fraction parameter. The logarithmic reinforcement results in inhibiting positive returns and enhancing negative returns. We also extended Q-learning into compound Q-learning and showed its convergence. Compound RL maintains the advantages of traditional RL, because it is a natural extension of traditional RL. The experimental results in this paper indicate that compound RL could be more useful in financial applications. Although compound RL theoretically works in general return-based MDPs, the both environments in this paper were single-state return-based MDPs. We have to investigate the performance of compound RL in multi-state return-based MDPs and compare with risk-sensitive RL in the next.

We aware that many RL methods and techniques, for example policy gradient, eligibility traces, and function approximation, can be introduced to compound RL. We plan to explore these methods in the future.

Acknowledgements

This work was supported by KAKENHI (23700182).

References

1. Arnab Basu, Tirthankar Bhattacharyya, and Vivek S. Borkar. A learning algorithm for risk-sensitive cost. *Mathematics of Operations Research*, 33(4):880–898, 2008.
2. Vivek S. Borkar. Q-learning for risk-sensitive control. *Mathematics of Operations Research*, 27(2):294–311, 2002.
3. John Y. Campbell, Andrew W. Lo, and A. Graig MacKinlay. *The Econometrics of Financial Markets*. Princeton University Press, 1997.
4. CMA. Global sovereign credit risk report, 4th quarter 2010. Credit Market Analysis, Ltd. (CMA), 2011.
5. Peter Geibel and Fritz Wysotzki. Risk-sensitive reinforcement learning applied to control under constraints. *Journal of Artificial Intelligence Research*, 24:81–108, 2005.
6. Abhijit Gosavi. A reinforcement learning algorithm based on policy iteration for average reward: Empirical results with yield management and convergence analysis. *Machine Learning*, 55(1):5–29, 2004.
7. Matthias Heger. Consideration of risk in reinforcement learning. In *Proc. of ICML 1994*, pages 105–111, 1994.
8. John Larry Kelly, Jr. A new interpretaion of information rate. *Bell System Technical Journal*, 35:917–26, 1956.
9. Oliver Mihatsch and Ralph Neuneier. Risk-sensitive reinforcement learning. *Machine Learning*, 49(2–3):267–290, 2002.
10. William Poundstone. *Fortune's Formula: The untold story of the scientific betting system that beat the casinos and wall street*. Hill and Wang, 2005.
11. Makoto Sato and Shigenobu Kobayashi. Average-reward reinforcement learning for variance penalized markov decision problems. In *Proc. of ICML 2001*, pages 473–480, 2001.
12. Anton Schwartz. A reinforcement learning method for maximizing undiscounted rewards. In *Proc. of ICML 1993*, pages 298–305, 1993.
13. Satinder P. Singh. Reinforcement learning algorithms for average-payoff markovian decision processes. In *Proc. of AAAI 1994*, volume 1, pages 700–705, 1994.
14. Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, 1998.
15. John N. Tsitsiklis and Benjamin Van Roy. On average versus discounted reward temporal-difference learning. *Machine Learning*, 49:179–191, 2002.
16. Ralph Vince. *Portfolio management formulas: mathematical trading methods for the futures, options, and stock markets*. Wiley, 1990.
17. Christopher J. C. H. Watkins and Peter Dayan. Technical note: Q-learning. *Machine Learning*, 8(3/4):279–292, 1992.