Policy Learning by GA using Importance Sampling

Chikao Tsuchiya, Hajime Kimura, Shigenobu Kobayashi
Tokyo Institute of Technology, 4259 Nagatsuta, Midori, Yokohama, Kanagawa Japan

Abstract. The most difficult problem of applying GA to a policy learning is that interactions with the environment require much time to evaluate the individuals. In this paper, we propose a new approach to estimate the individual's value using *importance sampling*. Importance sampling reuses the experiences obtained by some policy to estimate values of the other policies. The proposed technique cuts down the interactions with the environment in evaluating children, it can speed up optimization. In particular, it is effective in case GA is applied to a real robot's policy learning, because the load to the hardware accompanying trial and error can be mitigated. The proposed technique was implemented to the crawling robot, it was applied to obtain the control rules so that the robot is to walk. The experimental results show the strong affinity between GA and importance sampling, and also mean that GA using importance sampling can be a powerful tool for policy learning.

1 Introduction

Reinforcement learning(RL) learns the policy to maximize the average reward through interactions with the environment. That is, the problem is to search for mapping from the state space to the action space. However, because RL is basically a local search, it may lapse into a local minimum when applied to multimodal optimization problems. On the other hand, GA is a global search and thus it can deal with them. However, in the policy learning which requires interactions with the environment, it has been a serious obstacle in practical use that many trial and error are called for.

In recent years, the usefulness of importance sampling that reuses the data of the state transition series obtained by a certain policy for learning another policy, attracts attention[9][6][7]. In the domain of RL, several researchers use importance sampling to estimate Q-values for MDP. On the other hand, in the domain of GA, there is still little research on policy learning, and there is almost no research which sets the focus to importance sampling.

In this paper, in order to accelerate policy learning by GA, we introduce importance sampling. The policy learning by the naive GA requires many interactions with the environment to evaluate the children generated by crossover or mutation, and this is a practical obstacle. We evaluate the children by reusing the population's experiences using importance sampling. It is expected that our technique considerably cuts down the number of interactions with the environment.

In the proposed method, the agent can use plural policies held in the population, acts in the environment according to those policies, and accumulates new experiences. After some period, many children are generated by crossover between the parents chosen randomly from the

population. Processing the accumulated experiences using importance sampling, the rewards obtained by population are transformed into the children's rewards. Then let those rewards be their values. Because the children's policies aren't performed in the environment, the time to evaluate children turns into only processing time of importance sampling. Therefore, the optimization process is accelerated. We applied the proposed technique to the crawling robot to obtain control rules so that the robot is to walk. As the result, the usefulness of the proposed technique is verified.

2 Problem Formulation

2.1 Target Problem

The target problem is a reinforcement learning task in a Markov decision process(MDP). Even if it is partial observable Markov decision process(POMDP), the proposed technique can be applied. MDP is shown in the following. Let $\mathcal{S}$ denote state space, $\mathcal{A}$ be action space, $\mathcal{R}$ be a set of real number. At each discrete time t, the agent observes state $s_t \in \mathcal{S}$, selects action $a_t \in \mathcal{A}$, and then receives an instantaneous reward $r_t \in \mathcal{R}$ resulting from state transition in the environment. In general, the reward and the next state may be random, but their probability distributions are assumed to depend only on s_t and a_t in MDP. In MDP, the next state s_{t+1} is chosen according to the transition probability $\Pr(s_{t+1}|s_t, a_t)$, and the reward r_t is given randomly according to the expectation $r(s_t, a)$.

The learning agent does not know $\Pr(s_{t+1}|s_t, a_t)$ and $r(s_t, a)$ ahead of time. The objective of RL is to construct a policy that maximizes the agent's performance. A natural performance measure for a given task is the average reward per episode:

$$V = \frac{1}{M} \sum_{i=1}^{M} r_i$$

2.2 Policy Learning by Real Coded GA

We use UNDX[5] which is the well-known crossover operator in the real coded GAs, and MGG[8] which is one of the generation-alternation models excellent in diversity maintenance. The policy learning by the naive GA usually acquires the values of generated policies. Figure 1 shows the framework of policy learning by the naive GA.

Although interactions with the environment are required for evaluating the generated children, the efficiency of learning is improvable if the experiences of population are reusable.

3 Estimation of Policy's Value using Importance Sampling

3.1 Estimation using Importance Sampling

We assume that the policies can be parameterized by a vector θ. If we know the similarity between some policy θ and anothor policy θ', the value of policy θ' can be calculated by applying some corrections proportional to it.

In the policy parameterized by θ, let $\pi(s, a; \theta)$ be the probability an agent selects the action a in the state s.

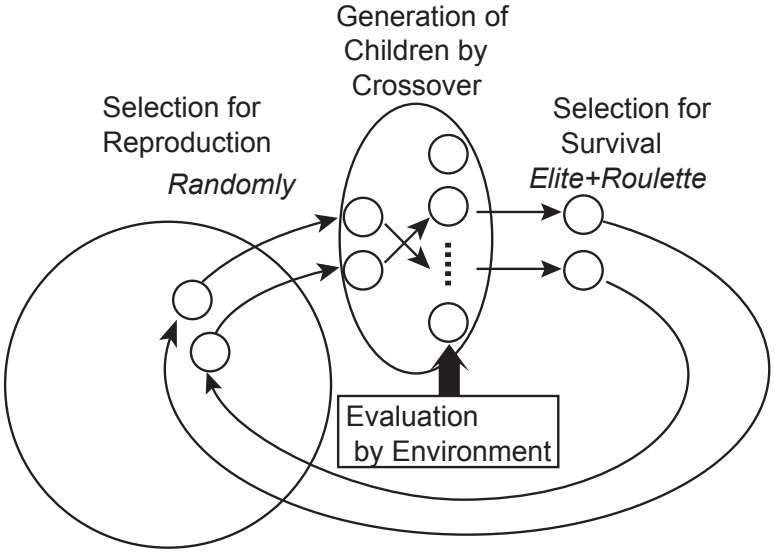

Figure 1: The framework of the policy learning by the naive GA.

The probability that an episode $h = \{s_1, a_1, r_1, s_2, a_2, r_2, ..., s_M, a_M, r_M, s_{M+1}\}$ occurs is described as follows:

$$\Pr(h|\theta) = \Pr(s_1)\Phi(h)\Psi(h) = \Pr(s_1)\prod_{i=1}^{M}\pi(s_i, a_i; \theta)\Pr(s_{i+1}|s_i, a_i)$$

where let $\Phi(h)$ be the action selection probability, let $\Psi(h)$ be the state transition probability, let $\Pr(s_1)$ be the probability the initial state is s_1.

Using importance sampling, the value of the policy parameterized by θ' can be estimated as follows[6]:

$$\begin{aligned}
\hat{V}(\theta') &= \frac{1}{N}\sum_{i=1}^{N} R_i \frac{\Pr(h_i|\theta')}{\Pr(h_i|\theta)} \\
&= \frac{1}{N}\sum_{i=1}^{N} R_i \frac{\Pr(s_1)\Phi'(h_i)\Psi(h)}{\Pr(s_1)\Phi(h_i)\Psi(h)} \\
&= \frac{1}{N}\sum_{i=1}^{N} R_i \frac{\Phi'(h_i)}{\Phi(h_i)} \\
&= \frac{1}{N}\sum_{i=1}^{N} R_i \prod_{j=1}^{M} \frac{\pi(s_j, a_j|\theta')}{\pi(s_j, a_j|\theta)}
\end{aligned}$$

where let R_i be the sum of rewards on the episode i, let N be the number of episodes. This formula means that the values can be calculated using only the ratio of the action selection probability in each policy.

Moreover, Precup et al.[6] show the weighted importance sampling. It has a lower variance estimate at cost of adding bias. It is described as follows:

$$\hat{V}(\theta') = \frac{1}{\sum_{i=1}^{N}\frac{\Phi'(h_i)}{\Phi(h_i)}}\sum\nolimits_{i=1}^{N} R_i \frac{\Phi'(h_i)}{\Phi(h_i)}$$

$$= \frac{1}{\sum_{i=1}^{N} \prod_{j=1}^{M} \frac{\pi(s_j,a_j|\theta')}{\pi(s_j,a_j|\theta)}} \sum_{i=1}^{N} R_i \prod_{j=1}^{M} \frac{\pi(s_j,a_j|\theta')}{\pi(s_j,a_j|\theta)}$$

3.2 Estimate the Value of Children

Importance sampling described in the previous section estimates the values of the other policies using the experience obtained by one policy. GA generates many children from two or more parents. Shelton et al.[9] proposed weighted importance sampling using the experiences obtained by policies $\theta_1, \theta_2, ..., \theta_N$. According to it, the value can be estimated as follows:

$$\begin{aligned}
\hat{V}(\theta') &= \frac{1}{\sum_{i=1}^{N} \frac{\Pr(h_i|\theta')}{\sum_{j=1}^{N} \Pr(h_i|\theta^j)}} \sum_{i=1}^{N} R_i \frac{\Pr(h_i|\theta')}{\sum_{j=1}^{N} \Pr(h_i|\theta^j)} \\
&= \frac{1}{\sum_{i=1}^{N} \frac{\prod_{k=1}^{M} \pi(s_k,a_k|\theta')}{\sum_{j=1}^{N} \prod_{k=1}^{M} \pi(s_k,a_k|\theta^j)}} \sum_{i=1}^{N} R_i \frac{\prod_{k=1}^{M} \pi(s_k,a_k|\theta')}{\sum_{j=1}^{N} \prod_{k=1}^{M} \pi(s_k,a_k|\theta^j)}
\end{aligned}$$

Using this technique, the children's values can be estimated from experiences and policies held within the population.

3.3 Algorithm of Policy Learning using Importance Sampling

The algorithm of the proposed technique is described as follows:

1. Generate N policies as an initial population, and obtain the experiences by these policies.
2. Choose 2+1 parents from N policies by random sampling, and generate C children by UNDX.
3. Estimate all children's values using importance sampling.
4. Choose two individuals from the family containing the parents and their children: one is the best individual and the other is selected from C+1 individuals other than the best one by the rank-based roulette wheel selection[1]. Replace the two parents with those two children. And, do away with the experiences obtained by the parents' policies.
5. Obtain the experiences by interactions with the environment using the policies of two newly added individuals. And, repeat the above procedures from step 2.

   Figure 2 shows the framework of the policy learning by the proposed technique.

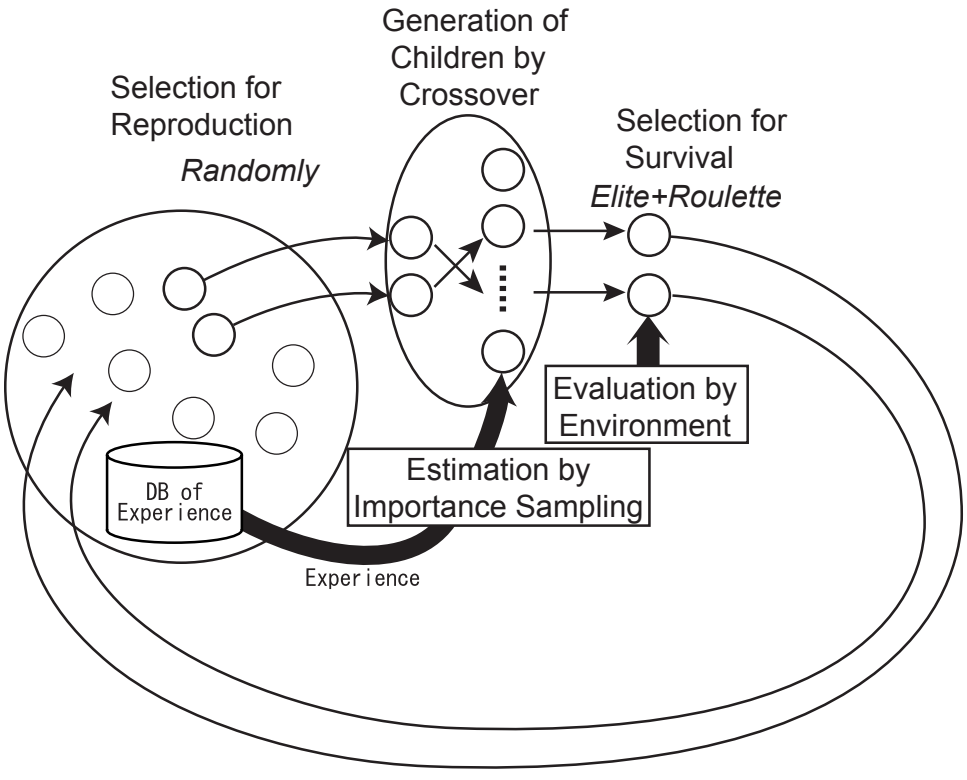

Figure 2: The framework of the policy learning by the proposed technique.

4 Application to the Crawling Robots

4.1 Crawling Robots

In this paper, we use the crawling robot shown in Figure 3. The crawling robot has one arm controlled by two servo motors and the touch sensor which investigates whether the tip of the arm is touching the ground or not.

We aim at obtaining control rules so that the robot is to walk through trial and error. However it is difficult to execute sufficient experiments using real robots for comparing several algorithms. We consider an imaginary crawling robot shown in Figure 4, and we evaluate the proposed technique from the experiments.

The robot has bounded continuous and discrete state variables. Continuous state variables are angular-position of the two joints, and discrete state variable represents for a touch sensor for the arm. The agent observes these state variables. The action the agent selects is an objective angular-position of two joint-motors. That is the same dimension of the continuous state. When the agent selects an action, the robot moves the motors towards the commanded positions. When the joint-angles move to the commanded position or the touch sensor's state changes, the reward is given as the result of the transition, and time step proceeds to the next step. The crawling robot's action stops when the motors reach to the objective angular-position or the touch sensor's state changes. That is, while an arm keeps contacting the ground or separating from the ground, the arm can move to the objective angular-position. Therefore, when the case of sensor variables changing in the way of moving joint-motors, the angular-position would not correspond to the selected objective position. Fot this reason, there exists

uncertainty of the state transition.

The reward signal reflects achievement of the given task. We want the robot to go forward, the immediate reward is defined as the speed of the body at each step.

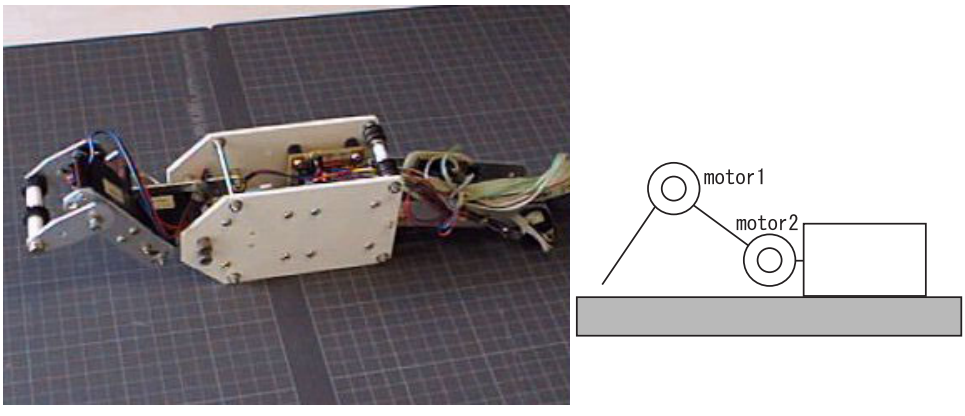

Figure 3: Crawling robot. The arm is controlled by two servo motors that react to angular-position commands.

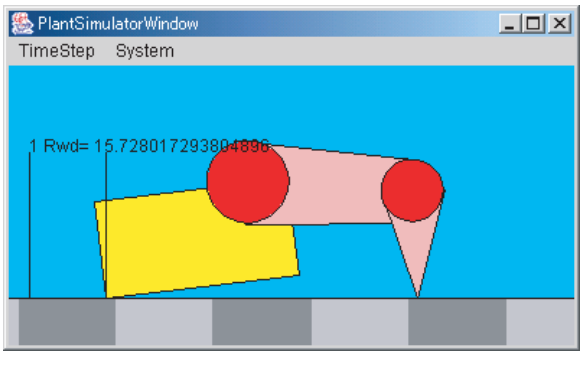

Figure 4: Imaginary crawling robot.

4.2 Implementation for the Robot

The action space has two dimensions, each element has range[0,1]. The state space has two dimensions for angular-position of the joints (each element has range[0,1]), one dimension for touch sensor (an element has 0 or 1). Therefore, the state space has a three dimensional vector $X = (x_1, x_2, x_3)$.

The policy is represented as follows. We construct a 7 dimensional feature vector $F = (x_1, x_2, x_3, x_4(= 1 - x_1), x_5(= 1 - x_2), x_6(= 1 - x_3), 1)$ based on $X = (x_1, x_2, x_3)$. The seventh element is always set to 1. Using weight vector $\Theta = (\theta_{1,i}, \theta_{2,i}, \theta_{3,i}, \theta_{4,i}, \theta_{5,i}, \theta_{6,i}, \theta_{7,i})$, the action at i-th dimension is selected from the normal distribution of the average of $\mu_i = 1/(1 + \exp(-\sum_{k=1}^{6} \theta_{k,i} x_k))$ and the standard deviation of $\sigma_i = 1/(1 + \exp(-\theta_{7,i})) + 0.1$. If a selected action is out of range, it is resampled[3]. The number of policy parameters is 14(=7x2), and the GA's search space has 14 dimensions.

4.3 Configuration of Experiment

In this experiment, the proposed method(`GA-IS`) holds 30 policies($N = 30$). At each episode, the agent performs 20 steps($M = 20$). In the generation-alternation, 10 children are generated by UNDX($C = 10$). The parameters of UNDX is based on Kita[4], Ono[5].

To verify the usefulness of the proposed technique, we compare the proposed method with the naive GA (`naive-GA-1`, `naive-GA-2`) and Stochastic Gradient Ascent(`SGA`)[2]. Both `naive-GA-1` and `naive-GA-2` don't use importance sampling. `Naive-GA-1`'s configuration is basically same to `GA-IS`'s($N = 30, M = 20, C = 10$). However, because the naive GA don't estimate the chidlren's values, the evaluation of children requires interactions with the environment using each child's policy. In this configuration, `naive-GA-1` requires 5(=$C/2$) times as many interactions as `GA-IS` does. On the other hand, `naive-GA-2` performs 600 steps at each episode($N = 30, M = 600, C = 10$). This configuration corresponds to that `GA-IS` uses the imaginary experiences of 600 steps. `SGA` is a RL approach which is based on gradient descent. But it has probable character so that a hill climbing search can be made.

4.4 Experimental Result

`GA-IS`, `naive-GA-1`, `naive-GA-2` and `SGA` are performed at 30000 steps. Figure 5 shows these performances.

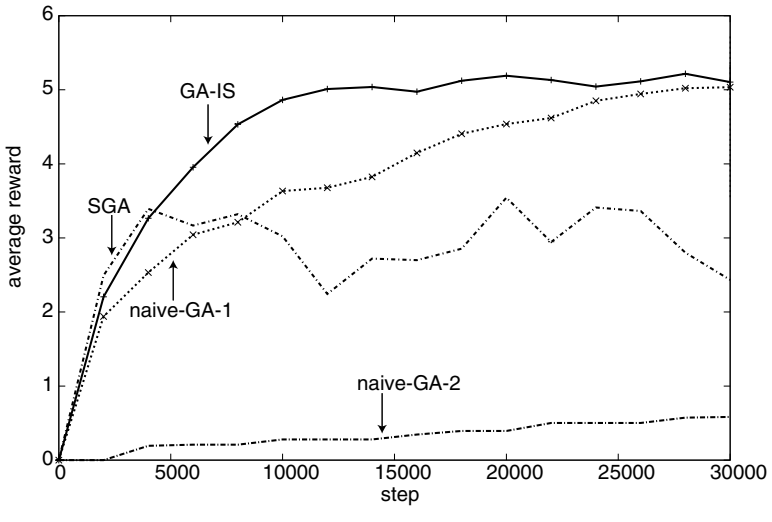

Figure 5: The perfoemance of learned policy averaged over 10 trials. `GA-IS` is the proposed method, `naive-GA-1` and `naive-GA-2` are the naive GA, `SGA` is the Stochastic Gradient Ascent for comparison.

5 Discussion

According to Figure 5, the number of steps taken for average reward to reach 5.0 is about 10000 by `GA-IS`, 30000 by the `naive-GA-1`. Therefore, the proposed technique made policy learning about three times as faster as the naive GA in this experiment. Considering

the number of interactions is reduced to 1/5 by the proposed technique, `GA-IS` should learn the policy five times as faster as the naive GA theoretically.

We think that one of cause of the gap between theory and practice is the period to accumulate experiences in first time. In the proposed method, all the individuals in a population need to hold the experiences interacting with the environment. But each individual holds no experiences in first time, so they need to interact with the environment. The episodes of the number of individuals are needed. In this configuration, it takes $MN = 600$ step to accumulate experiences. If we ignore this period, `GA-IS` can make policy learning about four times as faster as the `naive-GA-1`.

Moreover, we think that another cause of the gap between theory and practice is that the estimation by importance sampling is not completely correct. We investigated the estimation accuracy of importance sampling in this task. Figure 6 shows the children's values estimated by importance sampling and by maximum likelihood estimate method after 3000 steps; here, as to the maximum likelihood estimate method, the policies are actually performed in the environment. Because the proposed method uses experiences of 30 episodes of the parents to estimate the values, maximum likelihood estimate method also uses 30 episodes.

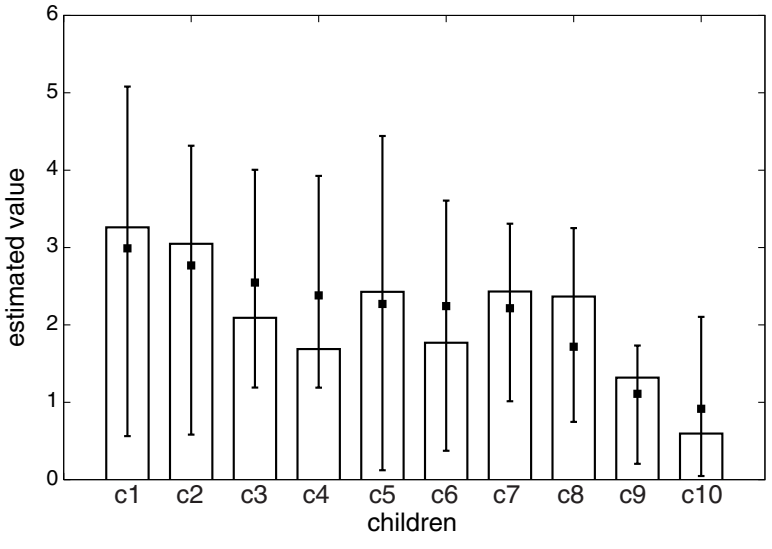

Figure 6: The children's values estimated by importance sampling and by maximum likelihood estimate method. The error-bars show the average rewards and the max/min rewards in 30 episodes, the boxes show the rewards estimated by importance sampling.

According to Figure 6, because the values estimated by importance sampling are between the max/min of maximum likelihood estimates and importance sampling can be said to estimate the true value well. In the proposed method, the rank-based roulette wheel selection is used in selecting for survival. Therefore, it is thought that there is no remarkable performance decrement of the proposed method resulting from the estimation accuracy of importance sampling even if some little errors are included in it.

Because `naive-GA-2` evaluates the children from their experience of 600 steps, its accuracy is equal or better than `GA-IS`. However, according to Figure 5, it turns out that `naive-GA-2` cannot converge within 30000 steps. Finally, `naive-GA-2` is converged at about 1300000 steps. This requires about 130 times as much time as the proposed method does.

Because SGA is basically a gradient descent method, it tends to lapse into local optima. Indeed, it lapsed into local optima whose value are about 3.0 in several trials. Contrarily, GA-IS can search globally, so it is stronger for multimodality than SGA. Moreover, SGA is sensitive to their parameters. To obtain a good performance stably, it needs hand tuning. On the other hand, because GA-IS has fewer parameters, a user need not worry about this. However, in speed of learning, a gradient descent method is generally more advantageous than the direct search method like GA. Also in this experiment, SGA's performance is better than GA-IS's until 4000 step. There exist trade-offs between the speed of gradient descent method and the global search capability of the proposed technique.

Then, we consider the population size(N), the number of generated children(C), and the length of episode(M). Because N is related to the number of samples used for importance sampling, it influences the estimation accuracy. The larger it becomes, the better the estimation accuracy of importance sampling improves.

C is related to the search capability of GA. The larger it becomes, the larger the number of individuals which must be estimated by importance sampling becomes. However, that processing is possible with only numerical computation; the interactions with the environment are not required. This means the strong affinity between the framework of MGG generates many children and the estimation by importance sampling.

M is related to the estimation accuracy of values of parents. In general, the culculation of the probability that an episode occurs becomes unstable if it is too long. Actually, we comfirmed that the limit of the length of an episode was about 30 steps by experiments. To deal with the longer episode is the future work.

6 Conclusion

In this paper, we proposed a method reduces the number of interactions with the environment, using importance sampling in policy learning by GA. There is a strong affinity between GA and importance sampling for policy learning. The experimental result shows the proposed method can learn policies about three times faster than the naive GA. This means that GA can be a powerful tool for policy learning.

The future work is to deal with the longer episode tasks and the higher dimensional tasks.

References

[1] Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning, AddisonWsley Publishing Company Inc. (1989).

[2] Kimura, H. and Kobayashi, S.: Reinforcement Learning for Continuous Action using Stochastic Gradient Ascent, Intelligent Autonomous Systems (IAS-5), pp.288–295 (1998).

[3] Kimura, H., Yamashita, T. and Kobayashi, S.: Reinforcement Learning of Walking Behavior for a Four-Legged Robot, 40th IEEE Conference on Decision and Control (CDC2001), pp.411–416 (2001).

[4] Kita, H., Ono, I. and Kobayashi, S.: Theoretical Analysis of the Unimodal Normal Distribution Crossover for Real-coded Genetic Algorithms, Proc. 1998 IEEE ICEC, pp.529–534 (1998).

[5] Ono, I. and Kobayashi, S.: A Real-coded Genetic Algorithm for Function Optimization Using Unimodal Normal Distribution Crossover, in Proc. 7th ICGA, pp.246–253 (1997).

[6] Precup, D., Sutton, R.S., and Singh, S.: Eligibility Traces for Off-Policy Policy Evaluation, Proc. 17th International Conf. on Machine Learning (ICML2000), pp.759–766 (2000).

[7] Precup, D., Sutton, R.S., and Dasgupta, S.: Off-Policy Temporal-Difference Learning with Function Approximation, Proc. 18th International Conf. on Machine Learning (ICML2001), pp.417–424 (2001).

[8] Satoh, H., Yamamura, M. and Kobayashi, S.: Minimal Generation Gap Model for GAs considering Both Exploration and Exploitation, Proceedings of IIZUKA'96, pp.494–497 (1996).

[9] Shelton, C.R.: Policy Improvement for POMDPs using Normalized Importance Sampling, Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence (UAI2001), pp.496–503 (2001).