감독 지식을 융합하는 강화 학습 기법들에 대한 비교 연구

김성완○, 장형수
서강대학교 컴퓨터공학과
{inaina21○, hschang}@sogang.ac.kr

A Comparison Study on Reinforcement Learning Method that Combines Supervised Knowledge

S. W. Kim○, H. S. Chang
Department of Compute Science and Engineering, Sogang University

요 약

최근에 제안된 감독 지식을 융합하는 강화 학습 기법인 potential-based RL 기법의 효용성은 이론적 최적 정책으로의 수렴성 보장으로 증명되었고, policy-reuse RL 기법의 우수성은 감독지식을 융합하지 않는 기존의 강화학습과 실험적인 비교를 통하여 증명되었지만, policy-reuse RL 기법을 potential-based RL 기법과 비교한 연구는 아직까지 제시된 바가 없었다. 본 논문에서는 potential-based RL 기법과 policy-reuse RL 기법의 실험적인 성능 비교를 통하여 potential-based RL 기법이 policy-reuse RL 기법에 비하여 더 빠르게 수렴한다는 것을 보이며, 또한 policy-reuse RL 기법의 성능은 재사용하는 정책의 optimality에 영향을 받는다는 것을 보인다.

1. 서 론

로봇과 같이 지능을 가진 에이전트(agent)가 매 의사결정시간에서 환경의 상태를 인지하고, 그 상태에서 취할 행동을 결정하여 그에 따른 피드백을 환경으로부터 받아 주어진 보상(reward)을 최적화하는 문제를 순차적 의사결정 문제(sequential decision making problem)라 한다. 많은 순차적 의사결정 문제는 마르코프 의사결정 과정(Markov Decision Process)[1]로 형식화되어 그 최적 정책을 value iteration(VI), policy iteration(PI)을 사용하여 구할 수 있다[1]. 하지만 MDP 모델의 보상 및 상태 전이함수를 에이전트가 모를 때, 즉 hidden MDP 에서의 에이전트들은 매 시간 스텝마다 직접 행동을 선택하고 피드백을 관찰(observe)함으로써 최적 정책을 학습하는데, 강화 학습(reinforcement learning, 이하 RL)[1] 알고리즘이 그 대표적인 학습 방법이다. 하지만 강화 학습은 최적 정책에 느리게 수렴한다는 단점이 있다. 강화 학습의 수렴 속도를 향상시키려는 연구는 현재까지 활발하게 이루어지고 있다.

이 연구(논문)는 산업자원부 지원으로 수행하는 21세기 프론티어 연구개발사업(인간기능 생활지원 지능로봇 기술개발사업)의 일환으로 수행되었습니다.

최근 “감독” 지식(supervised knowlegde)을 강화 학습의 과정에 융합하여 수렴 속도를 향상시키려는 연구가 진행되어 왔다[2][3][4][5]. 특히 Chang [6] 은 “potential-based reinforcement function”을 이용하여 감독 지식을 융합하는 방법을 제시하고, 이전의 연구들과 다르게 그 이론적인 최적 정책으로의 수렴성을 확립하였다. 또한, Fernández와 Veloso [7] 는 이전에 학습된 정책(policy)들을 확률적으로 현재의 학습에 사용하는, policy-reuse RL 기법을 소개하고 그 기법이 감독 지식을 융합하지 않은 Q-learning에 비해 더 좋은 학습 성능을 보인다는 것을 로봇 내비게이션 실험을 통하여 입증하였다. 앞의 두 가지 학습 기법은 그 성능의 우수성을 각각 이론적, 실험적으로 증명하였고, 다른 학습 기법과의 유연한 조합이 가능하다는 점에서 기존의 연구들과의 차별성을 갖는다. 하지만 potential-based reinforcement를 이용하여 감독 지식을 강화 학습에 융합한 학습 기법(이하 potential-based RL 기법)의 성능을 실험적으로 보인 연구는 현재까지 이루어지지 않았다. Policy-reuse RL 기법을 potential-based RL 기법과 비교한 실험적 결과 역시 아직까지 제시된 바가 없었다.

본 논문에서는 [5] 의 potential-based RL 기법과 [7] 의 policy-reuse RL 기법과의 실험적인 성능 비교를 통하여 potential-based RL 기법이 policy-reuse RL 기

법에 비하여 더 빠르게 수렴한다는 것을 보이며, 또한 policy-reuse RL 기법의 성능은 재사용하는 정책의 optimality에 영향을 받는다는 것을 보인다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 강화 학습에 대한 간략한 소개가 이루어진다. 3장에서는 감독 지식을 강화 학습에 융합하는 두 가지 방법인 potential-based RL 기법과 policy-reuse기법에 대해서 보다 자세히 설명한다. 4장에서는 로봇 내비게이션 실험을 통하여 앞의 두 학습 기법의 성능을 비교하고 5장에서 본 논문에서 이야기하고자 하는 내용을 결론짓고, 앞으로의 연구 방향을 제시한다.

2. 강화 학습(Reinforcement Learning)

학습 에이전트는 MDP 모델로 표현되는 환경과 상호작용한다. MDP $M=(X,A,P,R)$이 있다고 하자. 각 은 다음과 같이 정의된다. X는 상태들의 집합. A는 행동들의 집합, P는 집합 $\{(x,a)|x\in X, a\in A\}$를 X에 가능한 모든 확률분포로 mapping하는 상태전이함수이다. 상태 x에서 어떤 행동 a를 선택하여 상태 y로 전이할 수 있는 확률을 $P(y|x,a)$라 하자. R은 $X\times A\times X$를 실수집합 $\Re$로 mapping하는 보상함수라 하고, 상태 x에서 어떤 행동 a를 선택하여 상태 y로 갔을 때의 보상(reward)을 $R(x,a,y)$라 하자.

정책(policy) π는 $\pi: X\rightarrow A$로 정의되며, Π를 정책들의 집합이라고 하자. 초기 상태 x에서 매 상태마다 $\pi\in\Pi$인 정책 π에 따라 행동을 선택했을 때 얻어지는 값을 V^{π}라 하고 다음과 같이 정의한다:

$$V^{\pi}(x)=E\left[\sum_{t=0}^{\infty}\gamma^{t}R(X_t,\pi(X_t),X_{t+1})|X_0=x\right] \quad (1)$$

X_t는 시간 t에서 의 상태를 나타내는 random variable이며, $\gamma\in(0,1)$은 수렴을 위한 고정된 discount factor이다. $V^{*}(x)=\max_{\pi\in\Pi}V^{\pi}(x)$, $x\in X$ 라 할 때 $V^{*}(x)$는 Bellman's optimality principle에 의하여 다음과 같이 정의된다[1]:

$$V^{*}(x)=\max_{a\in A}\left\{\sum_{y\in X}P(y|x,a)\big(R(x,a,y)+\gamma V^{*}(y)\big)\right\} \quad (2)$$

$V^{*}(x)$, $x\in X$는 오직 하나의 값을 가지며 최적 정책(optimal policy) π^{*}는 모든 $x\in X$에 대해 다음과 같다.

$$\pi^{*}(x)\in\arg\max_{a\in A}\left\{\sum_{y\in X}P(y|x,a)\big(R(x,a,y)+\gamma V^{*}(y)\big)\right\} \quad (3)$$

$X\times A$에 대한 함수 Q^{*}를 $x\in X, a\in A$에 대해서 다음과 같이 정의하면.

$$Q^{*}(x,a)=\sum_{y\in X}P(y|x,a)\big(R(x,a,y)+\gamma V^{*}(y)\big) \quad (4)$$

이 때 Q^{*}는 다음의 식을 만족한다:

$$Q^{*}(x,a)=\sum_{y\in X}P(y|x,a)\big(R(x,a,y)+\gamma\max_{a'\in A}Q^{*}(y,a')\big) \quad (5)$$

그러면 Q^{*}함수를 이용하여 최적 정책 π^{*}를 다음과 같이 표현할 수 있다.

$$\pi^{*}(x)\in\arg\max_{a\in A}Q^{*}(x,a) \quad (6)$$

강화 학습은 Q^{*}함수의 근사 값을 "학습"함으로써 최적 정책 π^{*}를 구하며 그 종류로는 Q-learning, SARSA(λ) 등이 있다[1][8]. Q-learning과 SARSA(λ)(여기서는 SARSA(0)를 사용한다)는 본 논문에서 다루게 될 potential-based RL 기법과 policy-reuse RL 기법에 사용되기 때문에 이 장에서 간략하게 언급된다.

Q-learning은 discrete한 시간 스텝 $t\geq 0$마다 에이전트 상태 x_t에서 exploration-exploitation rule(이하 EE rule)[1]에 따라 행동 a_t를 선택한다. 이 과정에서 얻어지는 보상(reward) $R(x_t,a_t,x_{t+1})$과 다음 상태 x_t를 이용하여 $Q^{*}(x_t,a_t)$의 추정 값을 다음의 식에 의해 업데이트한다.

$$Q_{t+1}(x_t,a_t)\leftarrow Q_t(x_t,a_t)+\beta[R(x_t,a_t,x_{t+1})+\gamma\max_{a\in A}Q_t(x_{t+1},a)-Q_t(x_t,a_t)] \quad (7)$$

β는 에이전트의 학습을 Q함수 값의 업데이트 식에 반영하는 정도를 나타내는 계수이며, learning rate라고도 한다.($\beta\geq 0$)

SARSA(0) 역시 Q-learning과 마찬가지로 에이전트가 EE rule에 따라 행동을 선택함으로써 얻어지는 피드백을 이용하여 Q함수 값의 추정 값을 구한다. SARSA(0)의 업데이트 식(update rule)은 다음과 같다.

$$Q_{t+1}(x_t,a_t)\leftarrow Q_t(x_t,a_t)+\beta[R(x_t,a_t,x_{t+1})+\gamma Q_t(x_{t+1},a_{t+1})-Q_t(x_t,a_t)] \quad (8)$$

몇 가지 일반적인 수렴 조건을 만족할 경우 Q-learning과 SARSA(0)은 $t\rightarrow\infty$일 때 $Q_t\rightarrow Q^{*}$로 수렴한다[8][9].

3. 감독 지식과 강화학습의 융합

3.1 Potential-based RL 기법

이 절에서는 감독 지식을 potential-based reinforcement function을 이용하여 강화 학습에 융합한 학습 기법을 간단히 기술한다(자세한 내용은 [6] 참조).

MDP $M=(X,A,P,R)$이 주어져 있을 때 X에 대한 potential function $\Phi: X\rightarrow \Re$ 에 의하여 $M'=(X,A,P,R')$로 다음과 같이 변환되었다고 하자.

$$R'(x,a,y)=R(x,a,y)+\gamma\Phi(y)-\Phi(x) \quad (9)$$

위의 식에 사용된 $X\times X$에 대한 함수 $F(x,y)=\gamma\Phi(y)-\Phi(x)$를 potential-based reinforcement function이라 한다. Ng *et al.* [10]은 potential-based reinforcement에 의해 변형된 M'의 최적 정책(optimal policy)이 M의 최적 정책과 동일하다는 것을 보였다. 이를 SARSA(0)에 융합하면 Q 값의 업데이트 공식은 다음과 같이 변형되어질 수 있다[6].

$$\begin{aligned} Q_{t+1}(x_t, a_t)\leftarrow\ & Q_t(x_t, a_t)+\alpha_t(x_t, a_t)[R(x_t,a_t,x_{t+1}) \\ & +\gamma\Phi(x_{t+1})-\Phi(x_t) \\ & +\gamma Q_t(x_{t+1}, a_{t+1})- Q_t(x_t, a_t)\,] \end{aligned} \quad (10)$$

이는 (8)의 식에 potential-based reinforcement function을 추가한 것이다. Potential function Φ는 감독 지식을 다수 학습(multiple learning), 또는 expert advice의 형태로 SARSA(0)에 반영한다. potential-based 융합 기술에서는 (10)의 SARSA(0)-업데이트 공식을 사용하는 기본 에이전트(base agent)와, Q-learning이나 SARSA(λ), model-based 강화학습 [11] 등을 사용하는 서브에이전트(subagent)들이 있다. m개의 서브에이전트들이 있고, t_i를 서브에이전트 i의 시간 스텝이라고 할 때 Φ는 다음과 같이 정의된다.

$$\begin{aligned} &\Phi(x_t;t_1, \ldots, t_m) \quad (11)\\ &=\sum_{a\in A}\left(\frac{1}{m}\sum_{i=1}^{m} Q^i_{t_i}(x_t, a_t)\times\theta(x_t, a;t_1, \ldots, t_m)\right) \end{aligned}$$

$Q^i_{t_i}$-함수는 서브에이전트 i가 자신의 학습 기법을 사용하여 학습하는 Q 함수의 t_i에서의 추정 값을 말한다. 여기서 $\theta(x_t, a;t_1, \ldots, t_m)$는 다음과 같이 주어진다.

$$\begin{aligned} &\theta(x_t, a;t_1, \ldots, t_m) \quad (12)\\ &=\frac{\sum_{i=1}^{m} I\left(a\in \arg\max_{b\in A} Q^i_{t_i}(x_t, b)\right)}{\sum_{a'\in A}\sum_{i=1}^{m} I\left(a'\in \arg\max_{b\in A} Q^i_{t_i}(x_t, b)\right)} \end{aligned}$$

potential function Φ 를 통하여 m개의 서브에이전트들 각각의 Q 함수의 추정 값을 기본 에이전트의 업데이트 공식에 반영할 수 있으며, $t\rightarrow\infty$에 따라 SARSA(0)의 Q_t값은 Q^*에 수렴하게 된다[6].

3.2 Policy-reuse RL 기법

Policy reuse RL 기법은 현재 풀고자 하는 문제와 유사한 문제들의 policy들을 참조하여 에이전트의 학습 성능을 향상시키고자 하는 기법이다. Fernàndez와 Veloso [7] 는 policy reusing을 위한 세 가지 방법을 제시하였는데, 먼저 강화 학습의 과정에서 확률적으로 pre-learned policy를 사용하게 해 주는 "π-reuse (exploration) strategy"와 여기에 여러 개의 pre-learned policy들 중 어떤 policy를 사용할 것인지에 대한 방법을 추가한 "PRQ-learning algorithm", 마지막으로 유한한 수의 policy들을 저장하는 "policy library"를 유지하는 방법인 "PLPR algorithm"이 그것이다.

"π-reuse (exploration) strategy" 는 일정 확률로 기존의 policy를 재사용하고 나머지 확률로 기존의 EE rule을 사용한다. 이전에 현재 풀고자 하는 문제와 다른 유사한 문제에 대해 학습한 policy를 π_{past}라 하고, 지금 학습하고자 하는 새로운 policy를 π_{new}라 하자. 그리고 에이전트는 SARSA(0)를 통한 학습을 하고 있다고 하자. 그러면 π-reuse strategy에서 $x\in X$인 상태 x에서 $a\in A$인 행동 a는 다음과 같이 선택된다.

$$a=\begin{cases}\pi_{past}(x) & \psi\text{의 확률}\\ selected\ action\ by\ EE\ rule & (1-\psi)\text{의 확률'}\end{cases} \quad (13)$$

계수 ψ ($\psi\in[0,1]$)의 크기가 학습이 진행됨에 따라 점점 작아지면 학습 초기에는 기존의 pre-learned policy들에 의존한 학습을 하게 되는 반면 학습이 진행될수록 점점 현재의 학습 기법에 의한 학습을 하게 된다.

다수의 pre-learned policy들 중 어떤 policy가 π_{past}가 될 지에 대한 결정은 매 시간 스텝마다(또는 일정 시간 스텝마다) 이루어지며, 그 결정 방법은 다음과 같다. 에이전트가 학습을 통해 현재 해결하고자 하는 문제를 task Ω로 표현하자. Task Ω와 유사한 task n개가 있고$(\Omega_1, \Omega_2, \ldots, \Omega_n)$, Ω_i, $i=1, \ldots, n$에 대한 pre-learned policy들은 π_i이라 하자. 이 때 π_j가 π_{past}로 선택될 확률은 다음과 같다:

$$P(\pi_j=\pi_{past})=\frac{e^{\tau W_j}}{\sum_{i=1}^{n} e^{\tau W_i}} \quad (14)$$

W_i는 Ω에 대해 학습하는 과정에서 π_i를 참조할 때마다 얻은 보상(reward)의 평균이며 "reuse gain" 이라고 한다[7]. π_i가 위 식에 의한 확률로 π_{past}로 선택될 때마다 Reuse gain W_i는 다음의 식에 의해 업데이트 된다:

$$W_i=\frac{W_i U_i+R}{U_i+1} \quad (15)$$

R은 policy π_i를 π_{past}로 선택되었을 때 얻은 보상(reward)를 가리키고, U_i는 Ω에 대해 학습하는 과정에서 지금까지 π_i가 π_{past}로 선택된 횟수이다. (14)의 식에 의하면 reuse gain이 큰 policy일수록 π_{past}로 선택될 확률도 크다. τ ($\tau \geq 0$)는 temperature parameter이며 그 값이 작을수록 임의의 policy가 선택되는 경향이 크고, 클수록 reuse gain W_i ($i=1,2,...,n$)이 큰 policy가 선택되는 경향이 크다. 이를 이용하여 처음에는 τ의 값을 작게 설정하고 학습이 진행됨에 따라 점점 증가시키면, policy의 선택에 대한 exploration-exploitation이 가능하다.

이처럼 여러 개의 pre-learned policy들 중 어떤 policy를 reuse할 것인지 식 (14)에 의한 확률로 선택하고 π−reuse strategy를 사용하여 Q-learning을 적용한 알고리즘을 PRQ-learning 알고리즘이라 하는데, 이는 policy-reuse RL 기법의 핵심 아이디어이다. Fernàndez와 Veloso [7] 는 로봇 내비게이션 실험을 통하여 PRQ-learning 알고리즘이 Q- learning만을 사용한 학습보다 성능이 좋아짐을 보였다.

Pre-learned policy들은 policy library에 저장되는데 이를 유지하는 알고리즘이 PLPR 알고리즘이라 한다. 이 알고리즘은 최근에 학습된 policy π_Ω의 average reward의 weighted value와 기존에 있던 policy들의 reuse gain을 각각 비교하여 π_Ω를 policy library에 넣을 것인지의 여부를 결정한다.

PLPR 알고리즘까지 모두 구현한 policy-reuse RL 기법을 다른 RL 기법과 비교하기 위해서는, 서로 다른 task들에 대한 다수의 실험이 다른 RL 기법과의 비교실험을 하기 전에 이루어진 상태이어야 하기 때문에 객관적인 비교가 어렵다. 뿐만 아니라 policy-reuse RL 기법의 성능은 지금 학습하고자 하는 문제와 기존에 학습했던 문제와의 유사성에 큰 영향을 받기 때문에, 매번 서로 다른 문제를 학습하는 상황에서 PLPR 알고리즘을 여러 번 수행하는 것이 반드시 좋은 성능을 보장하지는 않는다.

그러므로 본 논문의 실험에서는 PLPR 알고리즘을 배제한, PQR-learning 알고리즘까지 구현된 policy-reuse RL 기법이 사용되었다. 다음 장에서 이 실험에 관한 자세한 내용이 기술된다.

4. 실험 및 분석

4.1 실험 환경

Potential-based RL 기법과 policy-reuse RL 기법의 성능 비교를 위하여 Fernández 와 Veloso [7] 의 로봇 내비게이션 실험을 이용하였다. [그림 1]과 같은 24×21 크기의 grid-based domain이 있다고 하자.

로봇이 선택할 수 있는 행동(action)은 “동”, “서”, “남”, “북” 으로 한 grid공간만큼 이동할 수 있는 네 가지이다. 로봇의 다음 행동이 벽에 부딪치면 원래 위치로 되돌아온다. 이 실험의 목적은 Grid 공간 내의 임의의 위치에서 시작하여 최대 H 번의 행동을 선택할 때까지 로봇이 'G' 의 위치에 도달하는 것이며, 성공할 경우 ‘1’의 보상(reward)을 받게 된다. 그렇지 못할 경우 보상은 ‘0’이 되고, 다시 임의의 시작 지점에서 탐색을 계속하게 되는데 위의 과정을 episode라고 한다. 본 논문에서는 K 번의 episode 동안 potential-based RL 기법과 policy-reuse RL 기법을 적용하여 학습하고 그 성능을 비교한다.

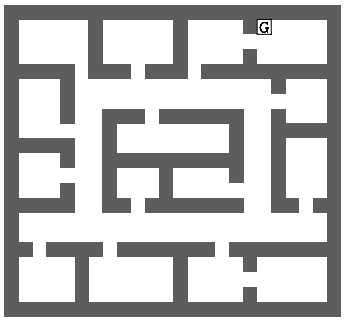

그림 1.
Grid-based domain (24×21)

학습 기법의 성능 척도는 average reward per episode, E를 사용하는데[7], 이는 다음과 같다:

$$E = \frac{1}{K}\sum_{k=0}^{K}\sum_{h=0}^{H}\gamma^h r_{k,h} \qquad (16)$$

$\gamma \in (0,1)$이며, $r_{k,h}$는 k번째 episode의 h 시간 스텝에 얻은 보상을 말한다. 학습 기법이 optimal policy에 수렴할수록 E 값은 커지게 된다[7]. 따라서 E 값을 척도로 학습 기법의 optimality를 측정할 수 있다.

Potential-based 융합 기술과 policy-reuse의 EE rule로는 ϵ_t−greedy strategy를 사용하였다[6]. 시간 스텝 t에서 $1-\epsilon_t$의 확률로 $a_t \in \arg\max_{a\in A} Q_t(x_t,a)$ 인 행동, 즉 greedy action이 선택되며, ϵ_t의 확률로 임의의 행동이 uniform하게 선택된다. 이 실험에서는 ϵ_t의 값을 다음과 같이 정의하였다.

$$\epsilon_t = \frac{c}{n_t(x_t)}, \quad c \in (0,1) \qquad (17)$$

$n_t(x_t)$는 t 시간 스텝까지 에이전트가 상태 x_t에 방문한 횟수를 말한다. 따라서 ϵ_t-greedy strategy는 어떤 상태의 방문 횟수가 증가할수록 ϵ_t의 값이 증가하여 그 상태에서 greedy action을 선택할 확률이 커진다.

Potential-based RL 기법은 SARSA(0)을 사용하는 기본 에이전트와 Q-learning을 사용하는 5개의 서브에이전트로 이루어져 있다. Potential-based RL과의 정확한 비교를 위하여 policy-reuse RL 기법에서는 $\pi-$reuse strategy를 SARSA(0)에 적용하고, 현재 학습하려는 task와 매우 유사한, 즉 Goal의 위치가 [그림 1]에서의 Goal의 위치와 1~2 grid 공간 이내에 있는, 5개의 task에 대해 Q-learning으로 학습한 5개의 pre-learned policy들을 사용하였다.

4.2 결과 및 분석

위의 절에서 정의한 문제와 성능 척도 E, $\epsilon_t-greedy$ rule을 사용하여 potential-based RL 기법과 policy-reuse RL 기법의 성능을 실험을 통하여 비교했다. $K=10000$ 번($H=100$으로 설정)의 episode를 거치면서 얻어지는 average reward per episode E 를 통하여 학습 성능을 측정하였다.

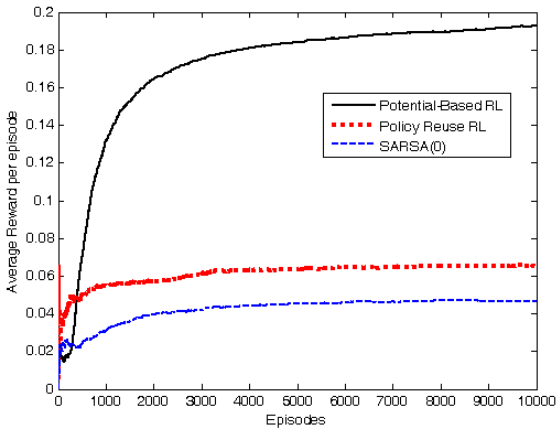

그림 2.
Potential-Based RL 기법과 Policy-Reuse RL 기법, SARSA(0) 의 성능 비교

[그림 2]는 potential-based RL 기법과 policy-reuse RL 기법, 그리고 SARSA(0)의 성능을 보여 준다. Potential-based RL 기법과 policy-reuse RL 기법은 모두 SARSA(0)에 비해 좋은 성능을 보였다. 특히 Potential-based RL 기법은 policy-reuse RL 기법에 비해서도 좋은 성능을 보임을 확인할 수 있다. 반면 policy-reuse RL 기법은 SARSA(0)보다 약간 우수한 정도의 성능을 보이는데 이는 Fernández와 Veloso [7]의 Q-learning과의 비교 실험에서 보였던 성능에 못 미치는 것이다. 이는 [7]에서 pre-learned policy로 이미 얻어진 optimal policy들을 사용한 반면 이 실험에서는 이전에 Q-learning을 사용하여 이전에 학습한 policy들을 pre-learned policy들로 사용한 것에 기인한 것이다. 즉 policy-reuse RL 기법의 성능은 pre-learned policy들의 optimality에 많은 영향을 받는다.

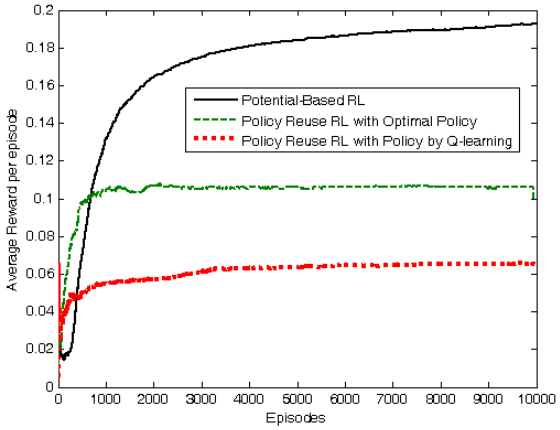

그림 3.
Potential-Based RL 기법과 pre-learned policy로 optimal policy를 사용한 Policy-Reuse RL 기법, pre-learned policy로 Q-learning을 사용한 Policy-Reuse RL 기법의 성능 비교

[그림 3]에서는 policy-reuse RL 기법의 성능이 pre-learned policy들의 optimality에 영향을 받는다는 사실을 확인할 수 있다. 이 실험에서는 먼저 현재의 task와 매우 유사한 task의 optimal policy를 heuristic한 방법으로 구해서 policy-reuse RL의 pre-learned policy로 사용하였다. 그 결과 앞의 실험에서 policy-reuse RL이 Q-learning을 통한 pre-learned policy들을 사용하여 보였던 성능보다 좋은 성능을 보였다. 이처럼 policy-reuse RL 기법은 pre-learned policy들의 optimality에 따라 그 성능이 달라지는 단점이 있다.

Optimal policy를 사용한 policy-reuse RL 기법의 성능도 potential-based RL 기법의 성능에 미치지는 못했다. 결과적으로 potential-based RL 기법은 서로 다른 pre-learned policy를 사용한 두 policy-reuse RL 기법들에 비해서 좋은 성능을 보였다.

한 가지 주목할 점은 episode 1000 이하에서의 초기 학습 속도, 즉 초기 수렴 속도는 policy-reuse RL 기법이 potential-based RL 기법에 비해서 빠르다는 점이다. [그림 4]는 potential-based RL 기법과 optimal policy를 사용한 policy-reuse RL 기법, Q-learning을 통해

pre-learned된 ploicy들을 사용한 policy-reuse RL기법의 성능을 episode 1000 이하에서 비교한 결과를 보여준다. 두 policy-reuse RL 기법은 episode 1000 이하인 학습 초기에는 potential-based RL 기법에 비해서 좋은 성능을 보였다. 이는 학습의 초기 단계에 큰 확률로 이전의 policy에서 near-optimal한 힌트를 제시해 주기 때문이다.

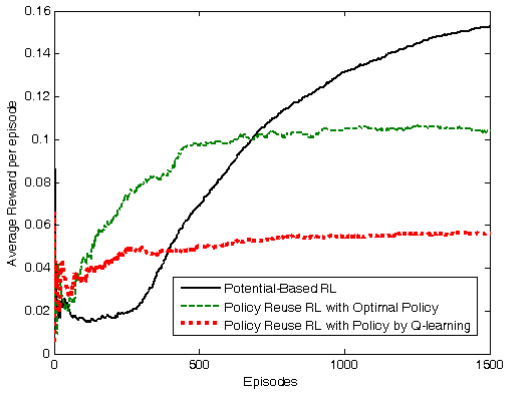

그림 4.
Potential-Based RL 기법과 Policy-Reuse RL 기법들의 episode 1000 이하에서의 성능 비교

5. 결 론

지금까지 감독 지식을 강화 학습에 적용하기 위한 두 가지 기법, potential-based 융합 기술과 policy-reuse 기법에 대해 알아보고, 아직까지 실험적으로 입증되지 않은 potential-based 융합 기술의 학습 성능을 policy-reuse RL 기법과의 비교를 통하여 확인하였다.

결과적으로, potential-based RL 기법이 policy-reuse RL 기법에 비해 좋음을 확인하였으며, policy-reuse RL 기법은 그것이 사용하는 pre-learned policy들의 optimality에 따라 그 성능에 큰 영향을 받는다는 사실과, episode 1000 미만의 초기 수렴 속도는 policy-reuse RL 기법이 potential-based RL 기법보다 빠르다는 것을 확인하였다.

앞으로의 연구는 위의 두 가지 학습 기법을 조합한 새로운 강화 학습 프레임워크를 개발하는 방향으로 진행될 것이다. 먼저 potential-based RL 기법의 서브에이전트들 중 일부가 policy-reuse기법을 사용하게 하는 방법과 potential-based RL 기법의 EE rule에 policy-reuse의 아이디어를 적용하여 특정 확률로 이전의 policy를 재사용하는 방법을 실험적으로 구현하여 그 성능을 평가할 것이다. Potential based 기법에서 서브에이전트 수의 증가, 감소가 학습에 미치는 영향도 확인해야 할 문제이다. Q-learning으로 학습하는 5개의 서브에이전트를 사용했던 이번 실험과는 별개로, 서버에이전트 수를 증가 혹은 감소함에 따라서 학습 성능이 어떻게 변화하는지를 실험을 통하여 확인할 것이다.

참 고 문 헌

[1] R. Sutton and A. Barto, *Reinforcement Learning*. MIT Press, 2000.

[2] M. N. ahmadabadi and M. Asadpour, "Expertness based cooperative Q-learning," *IEEE Trans. on Systems, Man, and Cybernetics*, part B. vol. 32, no. 1, pp. 66-76, 2002.

[3] A. G. Barto and M. T. Rosentein, "Supervised Actor-Critic Reinforcement Learning," *in Handbook of Learning and Approximate Dynamic Programming*, J. Si, A. G. Barto, W. B. Powell, and D. Wunsch (eds.), pp. 359-380, Wiley-IEEE Press, Piscataway, NJ, 2004.

[4] M. Rosentein and A. G. Barto, "Reinforcement learning with supervision by a stable controller," in *Proc. of the American Control Conf.*, 2004, pp. 4517-4522.

[5] K. Driessens and S. Dzeroski, "Integrating experimental and guidance in relational reinforcement learning," in *Proc. of the 19th Int. Conf. on Machine Learning*, 2002, pp. 115-112.

[6] H. S. Chang, "Reinforcement Learning with Supervision by Combining Multiple Learnings and Expert Advices", in *Proc. of the 2006 American Control Conference*, June, 2006, pp. 4159-4164.

[7] F.Fernández and M. Veloso, "Probabilistic Policy Reuse in a Reinforcement Learning Agent", In The Fifth International Joint Conference on Autonomous Agents and Multiagent Systems, May, 2006.

[8] S. Singh, T. jaakkola, M. Littman, and C. Szepesvari, "*Convergence results for single-step on-policy reinforcement learning algorithms,*" Machine Learning, vol. 38, pp. 287-308, 2000.

[9] T. Jaakkola, S. Singh, "On the Convergence of Stochastic Iterative Dynamic Programming Algorithms" Neural Computation, 6: 1185-1201, 1994.

[10] A. Y. Ng, D. Harada, and S. Russel. "Policy invariance under reward transformations: theory and application to reward shaping," in *Proc. of the 16th Int. Conf. on Machine Learning*, 1999, pp. 278-287.

[11] V. Gullapalli and A. G. Barto. “Convergence of indirect adaptive asynchronous value iteration algorithms.” In J. D. Cowan, G. Tesauro, and J. Alspector, editors, Advances in Neural Information Processing Systems 6, 1994, pages 695--702, San Mateo, CA. Morgan Kaufmann.