Expert Systems with Applications 38 (2011) 1565–1574

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Multi-goal Q-learning of cooperative teams

Jing Li [b,*], Zhaohan Sheng [a], KwanChew Ng [a]

[a] School of Management Science and Engineering, Nanjing University, Nanjing, China
[b] School of Engineering, Nanjing Agricultural University, Nanjing, China

ARTICLE INFO

Keywords:
Q-learning
Cooperative team
Multi-agent learning
Multi-goal learning

ABSTRACT

This paper studies a multi-goal Q-learning algorithm of cooperative teams. Member of the cooperative teams is simulated by an agent. In the virtual cooperative team, agents adapt its knowledge according to cooperative principles. The multi-goal Q-learning algorithm is approached to the multiple learning goals. In the virtual team, agents learn what knowledge to adopt and how much to learn (choosing learning radius). The learning radius is interpreted in Section 3.1. Five basic experiments are manipulated proving the validity of the multi-goal Q-learning algorithm. It is found that the learning algorithm causes agents to converge to optimal actions, based on agents' continually updated cognitive maps of how actions influence learning goals. It is also proved that the learning algorithm is beneficial to the multiple goals. Furthermore, the paper analyzes how sensitive the learning performance is affected by the parameter values of the learning algorithm.

1. Introduction

In cooperative teams, team members adopt knowledge to improve their ability and teams' performances. They have more than one learning goal in cooperative teams. In this paper, team members' learning goals consist of the size of the team, the performance of the team and individuals. A multi-agent system is used to simulate cooperative teams. The model of virtual cooperative team is based on Gilbert and Ahrweiler's research (Ahrweiler, Pyka, & Gilbert, 2004; Gilbert, Ahrweiler, & Pyka, 2007; Gilbert, Pyka, & Ahrweiler, 2001). In their research, the "KENE" was used to describe the knowledge of members. Suppliers and customers were generated by the computation of the KENE. Based on the research, this paper proposes a virtual cooperative team for the experiments skeleton of the learning algorithm. Details of the virtual cooperative team are proposed in Section 3.1.

To solve the multi-goal problem, Gadanho (2003) presented the ALEC agent architecture which has both emotive and cognitive decision-making capabilities to adapt the multi-goal survival task. Gadanho's research was beneficial to deal with the multi-goal task (the goals may conflict with each other). An improved reinforcement learning algorithm was proposed to learn multi-goal dialogue strategies (Cuayáhuitl, 2006). Zhou and Coggins (2004) presented an emotion-based hierarchical reinforcement learning (HRL) algorithm for environments with multiple goals of reward.

The multi-goal Q-learning algorithm is proposed to improve the multi-goal learning ability of the agents (the virtual team members). The tendency of agents for exploring unknown actions is discussed in the learning algorithm. Agents with the learning algorithm can decide what knowledge to adopt and how much to learn (choosing learning radius) by themselves for multiple goals. Experimental results show that the multiple goals can be achieved by agents with the learning algorithm. Moreover, two sets of sensitivity experiments are conducted in the paper.

2. Review of the related research

The learning algorithm is one of the key issues in the agent based system. Vriend (2000) considered that an agent was said to employ individual-level learning (if it learned from its own past experiences) and to employ population-level learning (if it learned from other agents). This paper focuses on both the population-level and individual-level learning in cooperative teams. The agent in this paper learns the learning radius from its own experiences and learns other knowledge from other agents. Many algorithms can be used for the individual-level and population-level learning, such as reactive reinforcement learning, belief-based learning, anticipatory learning, evolutionary learning, and connectionist learning.

Reinforcement learning was learning what to do and how to map situations to actions, so as to maximize a numerical reward signal. The learner must found which actions yielded the most reward by trying them in each state (Sutton & Barto, 1998). Compared with other algorithms, reinforcement learning models make

* Corresponding author at: School of Management Science and Engineering, Nanjing University, Nanjing, China. Tel.: +86 2583597501.
E-mail addresses: doctorlijing@gmail.com (J. Li), zhsheng@nju.edu.cn (Z. Sheng), kwanchew_ng@hotmail.com (K. Ng).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.07.071

few assumptions about both the information available to an agent and the cognitive abilities of an agent. Wang proposed a two-layer multi-agent reinforcement learning algorithm to improve the performance of the agents (Wang, Gao, Chen, Xie, & Chen, 2007). Reinforcement learning model was also used in supply chain for the ordering management (Chaharsooghi, Heydari, & Zegordi, 2008). Tuyls investigated the reinforcement learning in multi-agent systems from an evolutionary dynamical perspective (Tuyls, Hoen, & Vanschoenwinkel, 2006). The incremental method for learning in a multi-agent system was proposed with reinforcement learning (Buffet, Dutech, & Charpillet, 2007).

Q-learning is one of the reinforcement learning models that have been studied extensively by researchers. *Q*-learning was a simple way for agents to learn how to act optimally in controlled Markovian domains (Watkins, 1989). It was a famous anticipatory learning approach. Watkins presented and proved in detail a convergence theorem for *Q*-learning based on the outlined in 1992 (Watkins, 1992). Many researchers improved the learning model in their paper, such as Even-Dar and Mansour (2003) and Akchurina (2008).

In the literature, *Q*-learning has been used in many fields. Cheng (2009) investigated how intelligent an agent used the *Q*-learning approach to make optimal dynamic packaging decision in the e-retailing setting. Park employed modular *Q*-learning in assigning a proper action to an agent in the multi-agent system (Park, Kim, & Kim, 2001). Waltman and Kaymak (2008) studied the use of *Q*-learning for modeling the learning behavior of firms in repeated Cournot oligopoly games. Based on *Q*-learning algorithm, Distante presented a solution to the problem of manipulation control: target identification and grasping (Distante, Anglani, & Taurisano, 2000). Tillotson, Wu, and Hughes (2004) proposed a multi-agent learning model to control routing within the Internet.

Based on the former learning algorithm, the paper proposes a multi-goal *Q*-learning algorithm which is implemented in a virtual cooperative team. In the algorithm, agents can adjust their learning radius and knowledge adaptively. The remainder of this paper is organized as follows. Section 3 proposes the model of the virtual cooperative team and the multi-goal *Q*-learning algorithm. Section 4 describes the five experiments used to test the availability of our approach and the results obtained. In Section 5, the paper conducted two sets of sensitivity experiments with respect to learning parameters. Future directions and conclusive remarks end the paper in Section 6.

3. Model

In this paper, the multi-goal learning algorithm is based on *Q*-learning. The experiments of the algorithm are manipulated on a virtual cooperative team. The model of the virtual cooperative team is proposed in Section 3.1 and the multi-goal *Q*-learning algorithm is considered in Section 3.2.

3.1. The virtual cooperative team

The cooperative team consists of several team members who meet some others' demands. All team members cooperate to accomplish some work with their knowledge. Each team member is simulated by an agent in the NetLogo 4.0.2. The virtual team G ($G = \langle V, E \rangle$) consists of N agents ($V = \{v_1, v_2, v_3, \ldots, v_N\}$), where each agent can be considered as a unique node in a cooperative team. The relation in the cooperative team is modeled by an adjacency matrix E, where an element of the adjacency matrix $e_{ij} = 1$ if the agent v_i uses his knowledge to support v_j to accomplish its task (M_{v_j}) and $e_{ij} = 0$ otherwise. The relation among the agents are directed, so $e_{ij} \neq e_{ji}$. The relation between v_i and v_j is shown in Fig. 1 with an arrow. In the model, if v_i supports v_j to do something, v_i is called as the follower in the relation of v_i and v_j. Meanwhile, v_j is called as the leader.

3.1.1. Agent state

The state of v_i is defined as $S_{v_i} = \{k_{v_i}, r_{v_i} f_{v_i}\}$, where k_{v_i} is the knowledge of the agent, r_{v_i} is the learning radius and f_{v_i} is the fitness of the agent. If $f_{v_i} \leqslant 0$, v_i will be deleted from the virtual cooperative team. If the agent v_i gets the biggest reward in last period and the reward $f_{v_i}^{last\text{-}reward}$ is more than $f_{threshold}^{reward}$, v_i will create $\left\lfloor \log\left(f_{v_i}^{last\text{-}reward}\right) \right\rfloor$ agents. In the virtual team, the agent is a team member with an individual knowledge base. This knowledge of v_i is represented as $k_{v_i} = \{\left\{k_{v_i}^F, k_{v_i}^T, k_{v_i}^E\right\}, \left\{k_{v_i}^F, k_{v_i}^T, k_{v_i}^E\right\}, \ldots, \left\{k_{v_i}^F, k_{v_i}^T, k_{v_i}^E\right\}\}$, where $k_{v_i}^F \left(k_{v_i}^F \in [1, 100]\right)$ is the research field, $k_{v_i}^T \left(k_{v_i}^T \in [1, 10]\right)$ is the special technology in the field of $k_{v_i}^F$ and $k_{v_i}^E$ ($k_{v_i}^E \in [1, 10]$) is the experience of using $k_{v_i}^T$ in the field of $k_{v_i}^F$. The length of k_{v_i} is between $kl_{v_i}^{min}$ and $kl_{v_i}^{max}$.

In the model, the agent (v_i) adjusts learning radius (r_{v_i}) from his own experiences and learns k_{v_i} from other agents in the scope of r_{v_i}. An agent with a sampling radius of 2 takes data on the two levels to his followers and leaders. Fig. 1 shows the learning targets of the agent with $r_{v_i} = 2$. The agent's leaders in level $l + 1$ and $l + 2$ are shown with the black circles. The agent's followers in two levels are shown with the gray circles. In the figure, the arrows mean the agent (at the end of the arrow) is the follower of the pointed agent.

The agent's performance in the model is presented as the fitness (f_{v_i}). The fitness can be explained by the sum of rewards in the all last periods. In the paper, all revenues and costs are in fitness units. Each new agent's fitness is $f_{initial}$.

3.1.2. Adjacency matrix

In the paper, an adjacency matrix E is used to model the relation between agents. Since the task for each agent must be supported

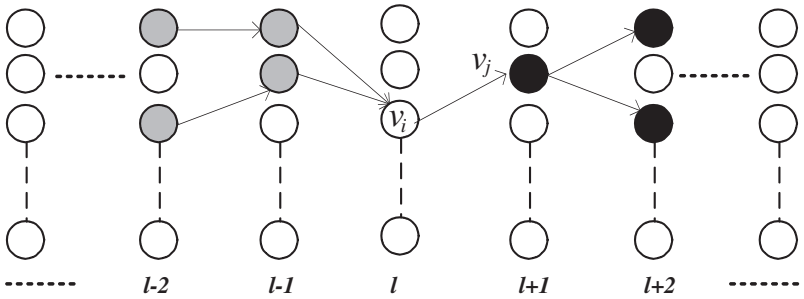

Fig. 1. The learning targets ($r_{v_i} = 2$).