Co-operative Reinforcement Learning By Payoff Filters
(Extended Abstract)

Sadayoshi Mikami[1 2], Yukinori Kakazu[1], and Terence C. Fogarty[2]

[1] Hokkaido University, Kita-13, Nishi-8, Sapporo, 060, Japan
[2] University of the West of England, Coldharbour Lane, Frenchay, Bristol, BS16 1QY, UK

Abstract. This paper proposes an extension of Reinforcement Learning (RL) to acquire co-operation among agents. The idea is to learn filtered payoff that reflects a global objective function but does not require mass communication among agents. It is shown that the acquisition of two typical co-operation tasks is realised by preparing simple filter functions: an averaging filter for co-operative tasks and an enhancement filter for deadlock prevention tasks. The performance of these systems was tested through computer simulations of n-persons prisoner's dilemma, and a traffic control problem.

1 Introduction

Reinforcement Learning (RL) is widely used in robot learning fields [1][2]. One reason for the feasibility to robotic applications is that it requires minimum information to develop policies; only state observation and real-valued payoff feedback are necessary, and these two types of information are always guaranteed in real-world robotic applications. This paper is aiming at extending single RL to acquire co-operation among multiple-agents, preserving the simplicity of RL that it can learn from minimal information.

The approach is to acquire global co-operation from locally exchanging payoff signals. Since communication of payoff is realised by asynchronously broadcasting and listening to scalar values, it is a minimal realistic way of co-ordinating co-operation. The main idea proposed here is to apply a filter to gathered payoffs to generate a payoff that will guide the agent to behave co-operatively. It is shown that spatial averaging and enhancement filters will give global co-operation over agents, and that the type of acquired co-operation is predicted by the type of the filter.

2 Related Works

Achieving co-operation among robots is actively studied in the field of Decentralised AI (for example[3]) and Artificial Life (for example [4]). Although there are some works that use RL to acquire robot co-operation (for example [5][6]), these works do not include communication amongst agents, which is necessary to learn complicated interactions, or they require huge real-time communication capabilities so that physical implementation into distributed robots is difficult. This paper is the first attempt to realise communication-based co-operative RL under realistic constraints.

The idea of applying filter to payoff has been widely used in RL Temporal filters, such as TD algorithms [1], are well studied. However, spatial filters amongst agents, such as the one proposed in this paper, have not been investigated. The learning of an iterated N-persons prisoner's dilemma game is employed for testing co-operative RL, since it is one of the commonly used co-operative problems and there are many GA based approaches to acquire co-operation such as [7].

3 Payoff Filters

We define a minimal co-operative agent (robot) as the one that performs the following trial-communication-learn cycle:

1. get sensory data as a state vector,
2. perform an action according to the state,
3. evaluate immediate local payoff r_i, where i is the index of an agent,
4. broadcast the normalised payoff to neighbours,
5. monitor the broadcasted message, and
6. apply filter to the broadcasted message, and invoke RL by using the filtered payoff.

The objectives of co-operation are generally classified into two types: one is to maximise one common objective function, and another is to maximise each objective function where they interfere with each other as the result of sharing an environment.

For the former type of co-operation, we consider the case where the global objective function is a summation of local payoffs $\sum_{\forall i} r_i$ and where each agent has an identical local payoff at the state where the global function is maximised. In this case, the payoff filter should easily be implemented by averaging local payoffs as

$$r' = \sum_{\forall j} r_j / N , \qquad (1)$$

where r' is the payoff given to RL and N is the number of neighbouring agents. It is proven that if neighbours of agents overlap with each other, the filter maximises the global objective function.

A more important case is the latter type of co-operation. We deal with the case where each agent's objective is to achieve its goals one after another, that are specified as the tops of a payoff landscape. In this case, a dead-lock over many agents may take place at states where climbing up directions for payoff functions differs for each agent (Fig.1). Thus, to co-operate corresponds to exiting this dead-lock situation. This is achieved by encouraging one agent to achieve its goal, and the other agents to give way. Representing this strategy into a payoff filter, it should be written as the following spatial enhancement filter:

$$r' = \begin{cases} r + \alpha|r|, & \text{if } (\sum_{\forall j} r_j / N) \le r; \\ r - \alpha|r|, & \text{otherwise,} \end{cases} \tag{2}$$

where α is a positive enhancement factor.

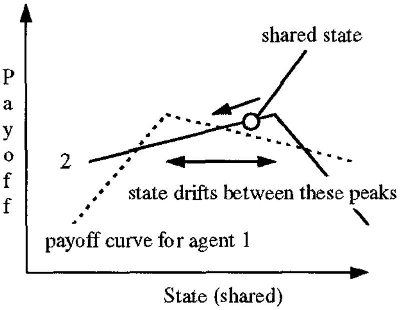

Fig. 1. Deadlock on two agents.

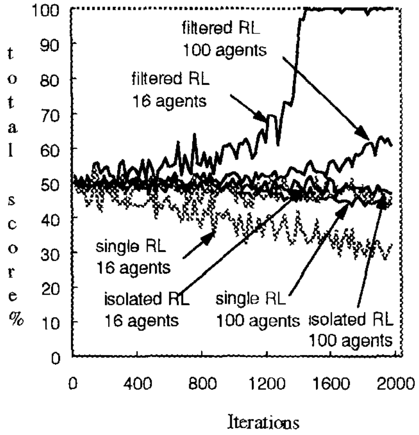

Fig. 2. Score curve by the simulation 1.

4 Simulation 1: Learning N-Persons Prisoners Dilemmaa

This experiment was conducted to illustrate the effect of an averaging filter (Eq.1). Each agent is allocated a cell in a NxN lattice. Each agent performs either an action 0 (co-operative) or 1 (selfish). Action 0 gives score $10/N^2$ to the agent whereas action 1 gives score $30/N^2$ to the agent and subtracts score $10/N^2$ from all the agents. The objective is to maximise the total scores over agents.

Three simulations were conducted: (1) single agent RL, (2) average filtered RL with 4 overlapping neighbours, and (3) average filtered RL with 4 isolated neighbours. TD(0) was used for the RL algorithm and its learning parameter was set to 0.02. Fig.2 shows the percentage of the global score plotted against iteration times. It is shown that the overlapping average filtered RL could achieve co-operation. It should be pointed out that the convergence speed will be faster if the size of the scope of the neighbourhood is relatively bigger.

5 Simulation 2: Conflict Resolution in Traffic Signal Control

To test the enhancement filter (Eq.2), a simple traffic control simulation was employed, where each traffic signal was controlled by RL. It should be noted that an optimisation of the traffic flow at a junction may sometimes cause congestion at the other junctions. This is the typical deadlocking situation and it is therefore a good example of testing the ability of conflict resolution.

The simulation contains 3x3 lattice roads. A car arrives at the edge of randomly chosen road approximately once every 2 seconds, and it runs 33km/h. At a junction, it decides its new direction according to the probability associated with that junction.

The number of cars that are allowed to stay on the roads is limited, by which we can specify the traffic density. More precise descriptions are found in [7].

Each junction is associated with an agent that controls two phases (go or stop) of its 4 signals. At each unit time (around every 15 seconds), it decides whether to change its signal phase or not, according to the output from RL. The state provided to RL is the signal phases of its 4-neighbouring junctions. Immediate local payoff is provided by the multiplication of -1 by the number of cars waiting at the junction during a unit time. TD(0) with learning factor 0.5 was used for RL.

Pure random controllers, single RL controllers, and enhancement filtered RL controllers communicating with 4 neighbourhood, were simulated. Table 1 summarises the degree of congestion during 40 minutes in simulation time. The number in the table shows total time during which cars were waiting for signals. The number in parenthesis shows the degree of improvement over random controllers. It is shown that the enhancement filter improved the quality of control over a wide range of congestion. From Fig.3, it is shown that the filter is parameter α sensitive, and the appropriate α lies around 1.5.

Table 1. Total waiting time (minutes) over 40 minutes simulation.

Maximum cars	Random	Single RL (%)	Filtered RL (%)
10	108	79 (137)	78 (138)
30	343	213 (161)	200 (171)
50	551	346 (159)	329 (167)
60	654	405 (162)	391 (168)

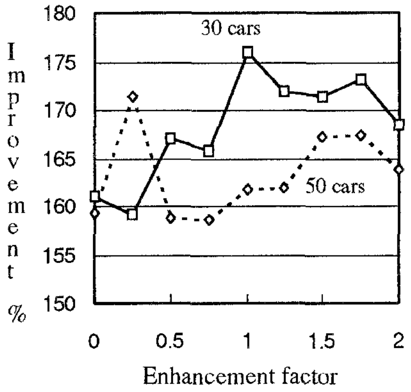

Fig. 3. Performance improvement against enhancement parameter.

References

1. Sutton, R.S., ed.: Reinforcement Learning, Kluer Academic (1993)
2. Connell, J.H., Mahadevan, S.: Robot Learning, Kluer Academic (1993)
3. Steels. L.: Co-operation between Distributed Agents through Self-Organisation, Decentralised AI, North-Holland (1990)
4. Arkin, R.C.: Integration of Reactive and Telerobotic Control in Multi-agent Robotic System, From Animals to Animats 3, MIT Press (1994) 473 478
5. Mataric, M.J.: Learning to Behave Socially, From Animals to Animats 3, MIT Press (1994) 453 462
6. Tan, M.: Multi-Agent Reinforcement Learning, Proc. Machine Learning (1993) 330 337
7. Fogel, D.B.: Evolving Behaviors in the Iterated Prisoner's Dilemma, Evolutionary Computation, **1** 1 (1993)
8. Mikami, S., Kakazu, Y.: Genetic Reinforcement Learning for Co-operative Traffic Signal Control, IEEE World Congress on Computational Intelligence (1994) 223 228