ECAI 2014
T. Schaub et al. (Eds.)

doi:10.3233/978-1-61499-419-0-1063

Coordinated Team Learning and Difference Rewards for Distributed Intrusion Response

Kleanthis Malialis[1] and **Sam Devlin** and **Daniel Kudenko**

Abstract. Distributed denial of service attacks constitute a rapidly evolving threat in the current Internet. Multiagent Router Throttling is a novel approach to respond to such attacks. We demonstrate that our approach can significantly scale-up using hierarchical communication and coordinated team learning. Furthermore, we incorporate a form of reward shaping called difference rewards and show that the scalability of our system is significantly improved in experiments involving over 100 reinforcement learning agents. We also demonstrate that difference rewards constitute an ideal online learning mechanism for network intrusion response. We compare our proposed approach against a popular state-of-the-art router throttling technique from the network security literature, and we show that our proposed approach significantly outperforms it. We note that our approach can be useful in other related multiagent domains.

1 Introduction

One of the most serious threats in the current Internet is posed by distributed denial of service (DDoS) attacks, which target the availability of a system. Such an attack is designed to exhaust a server's resources or congest a network's infrastructure, and therefore renders the victim incapable of providing services to its legitimate users. Multiagent Router Throttling [4] is a novel throttling approach where multiple reinforcement learning (RL) agents are installed on a set of upstream routers and learn to throttle traffic towards the victim.

In this paper, we incorporate hierarchical communication (Comm) and coordinated team learning (CTL) with a form of reward shaping called difference rewards [5], to improve scalability and enable online learning. Difference rewards are introduced to tackle the credit assignment problem encountered in RL. The difference reward (D_i) is a shaped reward signal that helps an agent learn the consequences of its actions on the system objective by removing a large amount of the noise created by the actions of other agents active in the system. It is defined as:

$$D_i(z) = R(z) - R(z_{-i}) \tag{1}$$

where z is a general term representative of either states or state-action pairs depending on the application, $R(z)$ is the reward function used, and $R(z_{-i})$ is $R(z)$ for a theoretical system without the contribution of agent i.

Our contributions in this paper are the following. We show that our approach can significantly scale-up by incorporating difference rewards. We demonstrate this in experiments involving over 100 RL agents. We evaluate our approach against AIMD [6], a popular throttling technique, and we show that our proposed approach outperforms it. The addition of difference rewards into our system lays the foundation for online learning. The system with difference rewards, not only performs better, but it learns remarkably quickly.

[1] Department of Computer Science, University of York, UK, email: {malialis,devlin,kudenko}@cs.york.ac.uk

2 Multiagent Router Throttling

Basic Design (MARL)
The basic design of our approach is referred to as MARL and is based on [4]. The underlying idea is to have multiple RL agents installed on a set of upstream routers and learn to throttle traffic towards the victim server. Each router applies throttling via probabilistic traffic dropping. The system has two important goals, which are directly encoded in the global (G) reward function. The first goal is to keep the server operational, that is, to keep its load below the upper boundary U_s. When this is not the case, each agent receives a punishment in the range $[-1, 0)$. When this is the case, each agent receives a reward of $L \in [0, 1]$, where L denotes the proportion of the legitimate traffic that reached the server during a time step. Difference rewards for agent i are calculated using:

$$D_i(z) = G(z) - G(z_{-i}) \tag{2}$$

Hierarchical Communication (Comm)
The first step towards scalability is to form teams of agents. Teams can either be homogeneous or heterogeneous. Each team consists of its leader, an inner layer of intermediate routers, and the throttling routers. Note that only the throttling routers are RL agents. The number of teams and their type depend on the network topology and model. The second step towards scalability involves communication. We propose a hierarchical uni-directional communication scheme. The victim's router signals its local load reading to the team leaders. The team leaders signal both their local load reading and the received reading from the victim's router to their intermediate routers. Similarly, the intermediate routers signal their local load reading and the two received readings to their throttling routers. The state space of each RL agent therefore consists of four features. Comm uses the same reward functions as the basic MARL approach.

Coordinated Team Learning (CTL)
CTL keeps the Comm functionality and further applies task decomposition and team rewards. It is now assumed that instead of having a big DDoS problem at the victim, there are several smaller DDoS problems where the hypothetical victims are the team leaders. Assuming a defence system of homogeneous teams, with respect to their sources (i.e. host machines), the hypothetical upper boundary for each team leader is given by $U_s/\#teams$. Moreover, agents are now provided with rewards at the team level rather than the global level. The coordinated team (CT) reward function allows a team's load to exceed its hypothetical upper boundary as long as the victim's

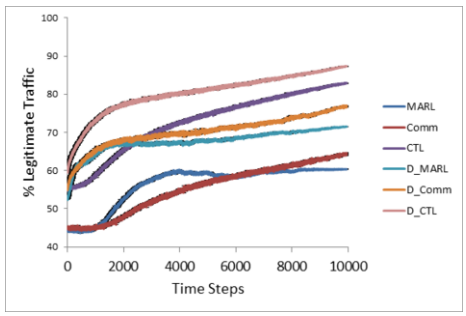

Figure 1. Performance for 30 RLs

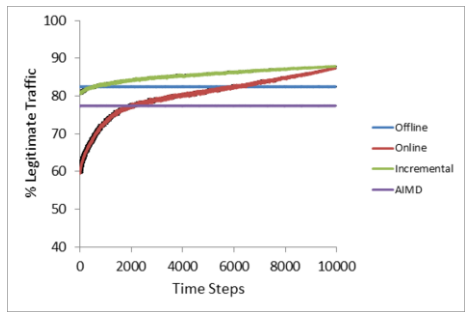

Figure 3. Incremental learning (D_CTL, 30 RLs)

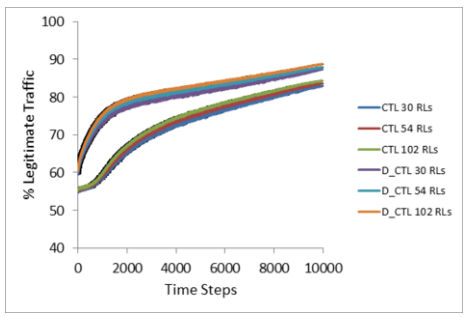

Figure 2. Scalability results for CTL and D_CTL

router load remains below the global upper boundary. The difference rewards signal for agent i in team j is calculated using:

$$D_{ji}(z) = CT_j(z) - CT_j(z_{-i}) \quad (3)$$

3 Experiments and Results

Our approach uses the SARSA algorithm, a linear decreasing ϵ-greedy strategy with $\epsilon = 0.2$, $\alpha = 0.05$ and $\gamma = 0$. Each agent has ten actions: $0.0, 0.1, ..., 0.9$ which correspond to $0\%, 10\%..., 90\%$ traffic drop probabilities. To calculate difference rewards we use the action 0.9. Experiments are conducted using an abstract network emulator. We use tree network topologies consisting of homogeneous teams of agents. Each team of agents contains two intermediate routers and six throttling routers (i.e. RL agents, three for each intermediate router). There are 12 hosts corresponding to each team.

MARL and Comm use the global (G) reward and CTL uses the coordinated team (CT) reward. D_MARL, D_Comm and D_CTL use difference rewards. Figure 1 shows how all the approaches compare to each other in terms of the percentage of legitimate traffic that reaches the server for 30 RL agents. Each episode runs for 10000 time steps. The values are averaged over 500 different episodes and error bars showing the standard error around the mean are plotted. Difference rewards significantly improve the system's performance.

Figure 2 shows how the approaches CTL and D_CTL compare to each other in topologies of 30, 54 and 102 learning agents. The performance of both CTL and D_CTL remains unaffected by the number of learning agents. Most importantly, D_CTL not only performs better but it learns much faster than CTL.

Offline learning attempts to learn a universal policy, that is, the "best" policy for all the instances of the network model. The policies learnt during offline learning are now used to initialise the Q-tables of the agents in order to facilitate the online learning process; we call this incremental learning. Incremental learning for D_CTL[2] is depicted in Figure 3 for 30 RL agents; error bars showing the standard error around the mean are plotted. The universal policy learnt during offline learning outperforms the AIMD approach. Most importantly, the incremental learning approach performs better than the online approach throughout the duration of an episode.

4 Conclusion

In this paper we demonstrate that the combination of difference rewards with hierarchical communication (Comm) and coordinated team learning (CTL) creates a scalable mechanism; this is demonstrated in experiments involving over 100 RL agents. The proposed approach also lays the foundations for online learning as it learns remarkably fast. We note that it can be useful in other related domains. This work assumes the availability of a reward function. For further details please see [3].

ACKNOWLEDGEMENTS

The authors would like to thank Logan Yliniemi and Kagan Tumer.

REFERENCES

[1] Sam Devlin and Daniel Kudenko, 'Theoretical considerations of potential-based reward shaping for multi-agent systems', in *AAMAS*, (2011).

[2] Sam Devlin, Logan Yliniemi, Kagan Tumer, and Daniel Kudenko, 'Potential-based difference rewards for multiagent reinforcement learning', in *AAMAS*, (2014).

[3] Kleanthis Malialis, Sam Devlin, and Daniel Kudenko, 'Intrusion response using difference rewards for scalability and online learning', in *AAMAS Workshop on Adaptive and Learning Agents (ALA)*, (2014).

[4] Kleanthis Malialis and Daniel Kudenko, 'Multiagent router throttling: Decentralized coordinated response against ddos attacks', in *Proceedings of the 25th Conference on Innovative Applications of Artificial Intelligence (AAAI / IAAI)*, (2013).

[5] Kevin R. Wheeler Wolpert, David H. and Kagan Tumer, 'Collective intelligence for control of distributed dynamical systems', *Europhysics Letters*, **49**(6), (2000).

[6] David K. Y. Yau, John C. S. Lui, Feng Liang, and Yeung Yam, 'Defending against distributed denial-of-service attacks with max-min fair server-centric router throttles', in *IEEE/ACM TON*, pp. 29–42, (2005).

[2] Given that Q-table initialisation has been proved to be equivalent to multiagent potential-based reward shaping [1], incremental D_CTL is an instance of DRiP [2].