Multi-Policy Optimization in Decentralized Autonomic Systems

(Extended Abstract)

Ivana Dusparic and Vinny Cahill
Lero – The Irish Software Engineering Research Centre
Distributed Systems Group, School of Computer Science and Statistics
Trinity College Dublin
{ivana.dusparic, vinny.cahill}@cs.tcd.ie

ABSTRACT

This paper addresses the challenge of multi-policy optimization in decentralized autonomic systems. We evaluate several multi-policy reinforcement learning-based optimization techniques in an urban traffic control simulation, a canonical example of a decentralized autonomic system. Our results indicate that W-learning, which learns separately for each policy and then selects between nominated actions based on current action importance, is a suitable approach for optimization towards multiple policies on non-collaborating agents in heterogeneous autonomic environments.

Categories and Subject Descriptors

H.3.4 [Systems and Software]: Distributed systems; I.2.11 [Distributed Artificial Intelligence]:Multiagent systems

General Terms

Algorithms, Design, Experimentation

Keywords

Autonomic Computing, Reinforcement Learning, Decentralized Systems

1. AUTONOMIC SYSTEMS

Autonomic computing systems are those capable of self-management and self-adaptation to varying circumstances without human intervention [4]. They need to be capable of learning how to meet their objectives and how to maintain optimal performance even when their operating conditions change. Rather than being centrally managed, the components of an autonomic system can be modelled as autonomic elements, that are capable of sensing their environment and making their own local decisions [4]. As these capabilities of autonomic elements map to the capabilities of autonomous agents, it has been proposed that multi-agent systems approaches are well suited to the implementation of decentralized autonomic systems [9]. Several agent-based techniques have already been successfully applied to decentralized optimization of large-scale systems (e.g. in [5]). In particular, Reinforcement Learning (RL) has been shown to be a suitable basis for the implementation of autonomic elements as it requires no domain knowledge and can be used to learn optimal policies for meeting high-level goals purely based on the element's interactions with the environment [8].

Cite as: Multi-Policy Optimization in Decentralized Autonomic Systems, (Extended Abstract), Ivana Dusparic, Vinny Cahill, *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009),* Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10–15, 2009, Budapest, Hungary, pp. 1203–1204

2. MULTI-POLICY OPTIMIZATION

Autonomic systems have often focused on optimizing system performance with respect to only a single high-level policy. However, system policies rarely exist in isolation and autonomic systems might be required to optimize their performance with respect to multiple policies simultaneously. Any given agent in the system may be responsible for contributing to the implementation of all of the policies present in the system, or of only a subset of them. This heterogeneity of policies leads to heterogeneity of agents in the system. Since RL has been successful as a learning technique for optimization towards a single policy in decentralized systems (e.g. in [2, 8]), as well as a learning technique for multiple policies on a single agent (e.g. in [1, 3]), we hypothesise that RL-based implementations of agent-based systems can be used to optimize towards multiple policies in decentralized autonomic systems. In this paper, we focus on evaluating RL-based approaches to multi-policy optimization on non-collaborating agents.

3. EXPERIMENT DESIGN

To evaluate our hypothesis, we have implemented several scenarios in a simulation of an Urban Traffic Control (UTC) system, a canonical example of a decentralized autonomic system. RL has been previously applied in UTC optimization (e.g. in [10]) but these implementations deal with a single policy only. We implemented two single-policy scenarios, to evaluate the impact that policies targeted at one vehicle type have on other vehicle types, and two multi-policy based scenarios, to compare different multi-policy approaches. The single-policy scenarios address a global, continuous, standard-priority policy that aims to optimize waiting time for all the vehicles in the system (Global Waiting time Only policy - GWO) and - a regional, temporary, high-priority policy that aims to prioritize emergency vehicles only (Emergency Vehicles Only policy - EVO). These policies are combined in two ways to implement the following multi-policy scenarios: using combined state space (GWEV-c), where GWO and EVO are combined into a single learning process over a single state space, similar to multi-policy

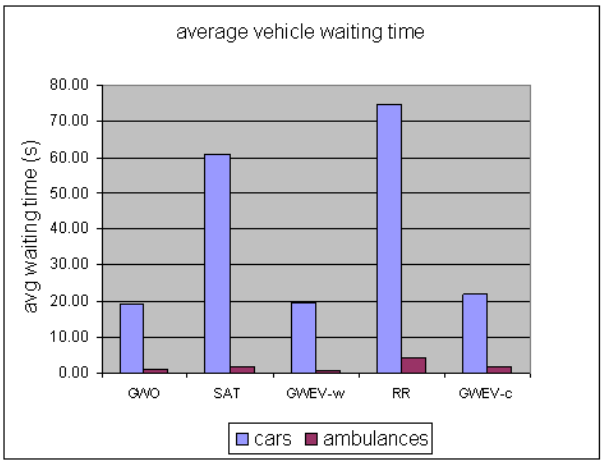

Figure 1: Average waiting time per vehicle type

implementation in [1], and using W-Learning (GWEV-w), where GWO and EVO learn the best actions separately as two separate learning processes, and W-learning [3] is used to determine which action is to be executed. Note that in both multi-policy approaches agents act individually, i.e. do not communicate or cooperate with each other. As baselines for comparison we implemented a round-robin (RR) controller (that loops through all available phases giving equal duration to each phase), and a SAT controller (a simple adaptive SCATS-like algorithm as defined in [7]).

In our experiments we use an urban traffic simulator developed in Trinity College Dublin [6]. We simulate 2000 minutes of car and emergency vehicle traffic on a road network corresponding to Dublin's main street, O'Connell Street, and several of its side roads. We perform experiments for 3 traffic loads: low (˜28k vehicles), medium (˜56k vehicles), and high (˜100k vehicles). Traffic is directed by 5 traffic-light junctions that are controlled by an agent each, implementing one of the RR, SAT, GWO, EVO, GWEV-w, and GWEV-c algorithms, depending on the experiment.

4. RESULTS AND ANALYSIS

We compared the performance of the agents based on traffic density and average waiting time per vehicle. We observed that both of the multi-policy approaches, GWEV-w and GWEV-c, outperform SAT and RR, both in terms of emergency vehicle and car waiting times (see Figure 1 for the results observed at medium load).GWEV-w performs better than GWEV-c, indicating that W-learning based approaches are more suitable for multi-policy optimization. We also observed a high dependency between the performance of the policies; EVO, which addresses only emergency vehicles, performs very badly both in terms of car and emergency-vehicle waiting times, as it fails to clear non-emergency traffic (as indicated by an increasing density in EVO in Figure 2), while GWO, which addresses only cars, performs well in terms of emergency vehicle waiting times (see Figure 1).

5. CONCLUSIONS AND FUTURE WORK

This paper evaluated several RL-based approaches to multi-policy optimization in UTC, and found W-learning to be a

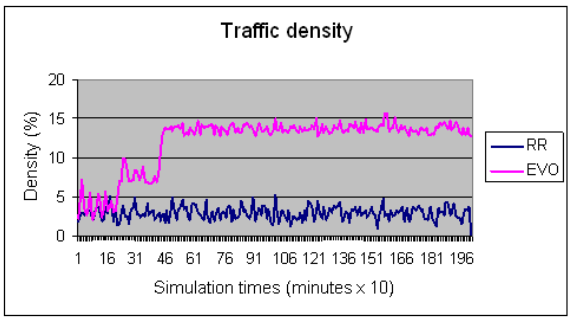

Figure 2: Density during low load

suitable approach for optimization towards multiple policies in decentrazied autonomic environments implemented as a group of non-cooperating agents. We plan to further investigate W-learning's applicability by evaluating its performance for additional policy types, as well as developing a collaborative version of the algorithm to investigate the potential for performance improvement by agent collaboration.

6. ACKNOWLEDGEMENTS

This work was supported, in part, by Science Foundation Ireland grant 03/CE2/I303_1 to Lero - the Irish Software Engineering Research Centre (www.lero.ie).

7. REFERENCES

[1] H. Cuayáhuitl, S. Renals, O. Lemon, and H. Shimodaira. Learning multi-goal dialogue strategies using reinforcement learning with reduced state-action spaces. In *Int. Journal of Game Theory*, 2006.

[2] J. Dowling. *The Decentralised Coordination of Self-Adaptive Components for Autonomic Distributed Systems*. PhD thesis, Trinity College Dublin, 2005.

[3] M. Humphrys. *Action Selection methods using Reinforcement Learning*. PhD thesis, University of Cambridge, 1996.

[4] J. O. Kephart and D. M. Chess. The vision of autonomic computing. *Computer*, 36(1):41–50, January 2003.

[5] A. Montresor, H. Meling, and O. Baboglu. Messor: Load-balancing through a swarm of autonomous agents. In *AP2PC'02*.

[6] V. Reynolds, V. Cahill, and A. Senart. Requirements for an ubiquitous computing simulation and emulation environment. In *InterSense '06*.

[7] S. Richter. Learning traffic control - towards practical traffic control using policy gradients. Technical report, Albert-Ludwigs-Universitat Freiburg, 2006.

[8] G. Tesauro. Reinforcement learning in autonomic computing: A manifesto and case studies. *IEEE Internet Computing*, 11(1), 2007.

[9] G. Tesauro, D. M. Chess, W. E. Walsh, R. Das, A. Segal, I. Whalley, J. O. Kephart, and S. R. White. A multi-agent systems approach to autonomic computing. In *AAMAS '04*.

[10] M. Wiering. Multi-agent reinforcement learning for traffic light control. In *ICML '00*.