RMASBench: Benchmarking Dynamic Multi-Agent Coordination in Urban Search and Rescue

(Extended Abstract)

Alexander Kleiner
Linköping University
58183 Linköping, Sweden
alexander.kleiner@liu.se

Alessandro Farinelli
University of Verona
Verona, I-37134, Italy
alessandro.farinelli@univr.it

Sarvapali Ramchurn
University of Southampton
Southampton, SO17 1BJ, UK
sdr1@soton.ac.uk

Bing Shi
Wuhan University of Tec.
Wuhan, China
bingshi@whut.edu.cn

Fabio Maffioletti
University of Verona
Verona, I-37134, Italy
fabio.maffioletti.uni@gmail.com

Riccardo Reffato
University of Verona
Verona, I-37134, Italy
refax88@gmail.com

Categories and Subject Descriptors: I.2.11 Distributed Artificial Intelligence: Multiagent Systems. **Keywords:** DCOP; Agent Coordination; Agent Simulation.

1. INTRODUCTION

The development of benchmarking platforms for agent-based systems is crucial to evaluate the performance of state-of-the-art algorithms and mechanisms in controlled settings and then determine the best ones to use under a wide range of realistic conditions. To this end, these platforms need to pose realistic challenges that mimic those of the real world and at the same time should allow for easy implementation of new algorithms without requiring researchers to implement low-level elements irrelevant to their algorithm. To date, very few benchmarking platforms have been developed according to such principles (and adopted by the artificial intelligence community). In particular, we note the dearth of realistic testbeds for agent-based coordination mechanisms such as distributed constraints optimization (DCOP), task allocation (TA), and coalition formation (CF). Those testbeds that do aim to evaluate such algorithms (e.g., DCOPolis and CATS) typically provide inputs that are drawn from fixed distributions or define coordination problems that are usually static or require significant extensions to create dynamic settings and large-scale problems. As a result, there are currently no well defined realistic benchmarks for DCOP, TA, and CF.

Against this background, we develop and evaluate a novel testbed for multi-agent coordination algorithms called RMASBench. Our work builds upon the existing RoboCup Rescue simulation platform (RSP) that simulates an urban search and rescue scenario and has also been used by emergency responders and planners to both train and plan for emergencies. In the RSP, agent designers have to code police agents to unblock roads, fire brigade agents to extinguish fires, and ambulance agents to rescue trapped civilians. More importantly, the problems that the RSP requires agent designers to solve for, are coalition formation (e.g., forming teams of ambulances or fire brigades to save civilians and extinguish fires respectively) [6], distributed constraint optimization (e.g., to ensure police agents unblock roads in an optimal way) [1], and task allocation (e.g., to schedule allocation of tasks according to levels of priority) in general[5].

We significantly extended the RSP in order to allow for the easy implementation of DCOP algorithms and develop a large-scale crowd simulator that allows users to run simulations involving thousands of agents in real-time on a single standard computer. In more detail, our work advances in the following ways. First, we formalize the fire brigade coordination problem as a DCOP and develop a test suite for DCOP algorithms. In particular, our test suite provides a simple communication protocol for the agents and provides facilities to compute standard metrics for evaluating DCOP approaches. Second, we implemented standard DCOP algorithms such as DSA [4] and MaxSum [3] on our test suite, and show how they perform under different simulation settings. Third, we develop a large-scale crowd simulator based on the use of Graphics Processing Units (GPUs) that allows us to scale the simulation to thousands of agents. Fourth, we present estimates of crowd evacuations for different maps for 3000 civilians given different conditions of road blockage. In so doing, we establish benchmark scenarios, and thus the initial steps towards building a complete testbed for multi-agent coordination algorithms.

2. RMASBENCH

RMASBench was developed as part of the RSP to introduce a generic API for multi-agent coordination that essentially provides facilities for exchanging messages among agents and for making coordinated decisions. RMASBench provides a library implementing state-of-the art solvers for DCOPs such as *DSA* and *MaxSum*, as well as facilities for comparing coordination algorithms. At the current stage there are two agent benchmarks, which are the *Fire Brigade Coordination Problem* and *Large Scale Crowd Evacuation Problem.* The coordination problem is about assigning fire brigades to fires so to mitigate damages to buildings. We focus on this problem because it poses crucial challenges for coordination: i) it is a dynamic problem: the set of fires to be extinguished varies over time because fires may spread to neighboring buildings and grow in intensity as time proceeds ii) it has a strong spatial aspect: a fire brigade agent can operate on a given fire only if it is physically located in the proximity of that fire, hence agents must consider their travel time when coordinating and iii) it includes coalitional effects: several agents can work on the same fire, and the more agents work

Appears in: *Proc. of the 12th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2013)*, Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.

Metrics/Algs	DSA	DSA-R	MaxSum
Exti. Time	273.85 [±8.63]	299 [±0]	**134.6** [±13.12]
onceburned	1132.95 [±65.42]	1110.15 [±81.57]	**168.25** [±81.47]
Violated Const.	89.62 [±2.26]	126.41 [±2.72]	**46.2** [±1.03]
Msgs num.	8853.25 [±147.43]	**143.18** [±5.28]	3767.4 [±153, 33]
Msgs bytes	70826 [±1179.5]	**1145.47** [±42.26]	99880 [±1995.2]
NCCC	**240.28** [±23.49]	341.19 [±30.52]	263608 [±6218.4]

Table 1: Statistics for DSA, DSA-R and MaxSum averaged over 30 runs for the Kobe scenario. For DSA-R and MaxSum $k = 4$. The best result for each metric is in bold. Numbers in brackets report the standard error of the mean

on the same fire the faster the fire can be extinguished. However if too many units are allocated to the same fire they might hinder each other or ignore other fires.

Furthermore, RMASBench offers large scale crowd simulation that implements individual civilians as agents running their commands in parallel. Crucially, RMASBench implements agents' path planning and social behaviors in terms of a parallel programs that can be run on standard Graphics Processing Units for easy deployment on any PC [2]. By so doing, we are able to exploit the highly parallelized memory and processing architecture of the GPU to speed up the simulations by orders of magnitude and therefore create realistic situations for decentralized algorithms to solve in real time.

3. EMPIRICAL RESULTS

We now report empirical results for the comparison of MaxSum and DSA in the fire fighting coordination problem. In particular, we represent our problem by using a factor graph [3], and we employ a number of pruning heuristics to make the factor graph sparse and hence reduce computation and communication associated to MaxSum and DSA. As for DSA we consider both a version working on a fully connected factor graph, and another version of where we employ our pruning heuristics (DSA-R). In DSA-R two agents are neighbors if they can work on the same fire according to the restricted factor graph representation. We empirically tuned the activation probability of DSA and in all the following experiments we use the value 0.7. To evaluate algorithms' performance we consider standard metrics for the RoboCup scenario such as data about the *onceburned* (the number of buildings that caught fire at least once) and the extinguish time,i.e. the time step at which all fires have been extinguished. As for coordination overhead we use standard measures for evaluating DCOPs, such as the number of messages exchanged per simulation step, the size of messages exchanged (in bytes) per simulation step and the Non Concurrent Constraint Checks (NCCCs), which is a standard measure for computation overhead in DCOPs.

We focused on one reference scenario, which is the map of the Kobe with 3 ignition points and 12 fire brigade units. Each simulation is run for 300 steps and fire brigades remain idle for 40 time steps before they become aware of all fires. Table 1 reports a comparison of the three coordination algorithms. Results show that MaxSum outperforms both DSA and DSA-R for the *onceburned* metric, extinguish time and violated constraints. The results related to coordination overhead (message number, message size and NCCC) confirm that MaxSum requires significantly more computation and communication than both versions of DSA (especially for what concerns the NCCCs). However, comparing the DSA and DSA-R we can see that by using the factor graph pruning heuristics we can achieve similar performance while significantly reducing coordination overhead (e.g., we reduce the size of total message exchanged by more than one order of magnitude). This suggests that our factor graph pruning approach is not limited to MaxSum but could be successfully applied to other DCOP solvers. We now demonstrate that our crowd simulator can create realistic dynamic behaviors for large scale simulations. The scale factor is particularly important to address scenarios that can be useful for rescue operators in real applications. In more detail, we simulated the evacuation of 3000 agents 3000 on maps of Berlin and Paris. In order to evaluate the simulation, we generate artificial roadblocks across the maps to create settings where police agents may be successful at unblocking roads to different degrees. Then, for each map and each certain called blocked ratio, we run the simulation for 10 times. We record the mean total evacuation time. The simulation results are shown in Figure 1. As can be seen, the evacuation times for the Berlin map are nearly three times (1700 time steps compared to 400 for 0 blockage ratio) those of the Paris map. Moreover, a high percentage of civilians (nearly 30% in Paris) tend to completely trapped in parts of the city (with Paris being worse than Berlin) and never manage to evacuate given the blocked roads. A closer inspection at the maps reveals that the Berlin map has many more junctions than Paris and, while this causes regular congestions (leading to long evacuation times), it also allows many more civilians to escape since it has more alternatives at every junction.

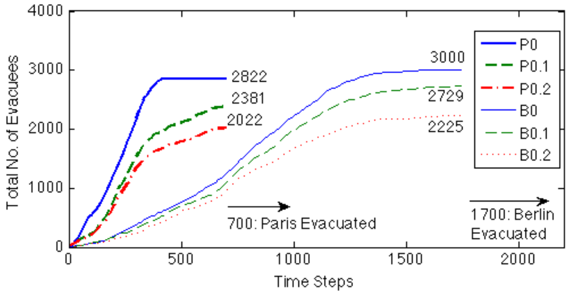

Figure 1: Simulation results from runs on Berlin and Paris maps with 3000 civilians. Bk and Pk represent Berlin and Paris for different blockage ratios k.

4. CONCLUSIONS

We introduced RMASBench, a testbed for multi-agent coordination based on the RoboCup Rescue Simulation Platform. RMASBench provides extremely challenging problems for multi-agent coordination in the USAR domain, and, in contrast to existing testbeds, it offers highly dynamic and large-scale benchmarking environments. We demonstrated the evaluation of state-of-the-art algorithms for the fire fighting problem and we simulated large scale evacuation scenarios on realistic maps, setting key benchmarks in this domain. This work was partially supported through the RoboCup federation and the Excellence Center at Linköping and Lund in Information Technology (ELLIIT).

5. REFERENCES

[1] A. Chapman, R. A. Micillo, R. Kota, and N. Jennings. Decentralised dynamic task allocation: A practical game-theoretic approach. In *The Eighth Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS '09)*, pages 915–922, May 2009.

[2] J. Sanders and E. Kandrot. *CUDA by Example: An Introduction to General-Purpose GPU Programming*. Addison Wesley, 2010.

[3] A. Farinelli, A. Rogers, A. Petcu, and N. R. Jennings. Decentralised coordination of low-power embedded devices using the max-sum algorithm. In *Proc. of the Seventh Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, pages 639–646, 2008.

[4] S. Fitzpatrick and L. Meetrens. *Distributed Sensor Networks A multiagent perspective*, chapter Distributed Coordination through Anarchic Optimization, pages 257–293. Kluwer Academic, 2003.

[5] A. Kleiner, M. Brenner, T. Bräuer, C. Dornhege, M. Göbelbecker, M. Luber, J. Prediger, J. Stückler, and B. Nebel. Successful search and rescue in Simulated Disaster Areas. *Robocup 2005: Robot Soccer World Cup IX*, volume 4020 of *Lecture Notes in Computer Science*, pages 323–334. Springer, 2005.

[6] S. D. Ramchurn, M. Polukarov, A. Farinelli, C. Truong, and N. R. Jennings. Coalition Formation with Spatial and Temporal Constraints. In *Proc. of the 9th Int. Conf. on Autonomous Agents and Multiagent Systems*, pages 1181–1188, 2010.