Decision-Support for Real-Time Multi-Agent Coordination *

Rajiv T. Maheswaran, Craig M. Rogers, Romeo Sanchez and Pedro Szekely
University of Southern California - Information Sciences Institute
4676 Admiralty Way #1001, Marina Del Rey, CA 90292

1. INTRODUCTION

Teams of people need to coordinate in real-time in many dynamic and uncertain domains. Examples include disaster rescue, hospital triage, and military operations. It is possible to develop plan *a priori*, but many parts of these plans must be left unspecified because people won't know exactly what needs to be done until they are executing the plan in the field. Additionally, requirements and tasks can evolve during execution. Our work addresses a fundamental multi-agent systems endeavor of creating decision support systems that help humans perform better in these domains. The technical challenges to compute good solutions for these problems have been well documented [1, 2, 3].

2. PROBLEM DOMAIN

Our system was developed for field exercises were based on a simulated disaster rescue domain. The first two exercises were held in the city of Rome, NY, and the second three were in Stanton Wood Park in Herndon, VA. Images of the field exercise in Rome, NY are shown in Figure 1 and a map of the sites and road network of Stanton Wood Park are shown in Figure 2. They were organized and evaluated by independent parties contracted by the DARPA Coordinators program. The rules of the field exercise were created collaboratively by the teams building coordinator agents, the independent evaluation team, and subject matter experts. The specific instances or *scenarios* that comprised the test problems were chosen by the independent evaluation team.

Various locations were selected as *sites* and a feasible road network was constructed. If the site was *populated*, it could have injured people in either *critical* and *serious* condition. Populated sites would also have gas, power and water substations which may have been damaged. In addition, any site could have *facilities* such as a *hospital*, *clinic*, *warehouse*, *gas main station*, *power main station* and *water main station*. A team would obtain points by rescuing injured to hospitals or operational clinics (before a deadline associated with each injured person) and by repairing main stations and substations. The goal of a scenario was to accumulate as many points as possible before the scenario deadline.

Figure 1: Field Exercise Images from Rome, NY

The teams were composed of 8 field agents and 2 command agents. Each agent had a different set of skills. Three *specialists* in *gas*, *power* and *water* could perform *major* and *minor* repairs in their respective skill area. The *medical specialist* could load any type of injured person by themselves. The remaining four *survey specialists* could have any collection of skills involving minor repairs. The field agents could move throughout the field exercise area and perform actions. The command agents were located at a base where they helped to coordinate the activities of the team. The *Radio Team* communicated only with radios. Our *CSC Team* had ruggedized tablet computers on which our agents were loaded, in addition to radios. The tablets had cell modems and GPS.

A *survey for damage* at a main station or substation revealed the number and type of problems chosen from a set of known possible problems. A *survey for injured* at a populated site revealed the number, types and deadlines for the injured at that site. As the result of a survey, any team member might be injured, forcing them to go to an operational medical facility to recover before proceeding with any other action. A survey could also reveal that the vehicle of the agent doing the survey had failed and would require a vehicle repair before the agent could travel to any other site. While traveling, agents could encounter*road blocks* which could not be passed until fixed. Travel and repair times could vary and repairs could fail. Furthermore, most of these outcomes were only observable by the agent encountering the outcome.

There were many rules and couplings that forced agents to coordinate. To do surveys, gas and power substations at the site had to

*The work presented here is funded by the DARPA COORDINATORS Program under contract FA8750-05-C-0032. The U.S.Government is authorized to reproduce and distribute reports for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of any of the above organizations or any person connected with them. Approved for Public Release, Distribution Unlimited.

Cite as: Title, Author(s), *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, van der Hoek, Kaminka, Luck and Sen (eds.), May, 10–14, 2010, Toronto, Canada, pp. XXX-XXX.

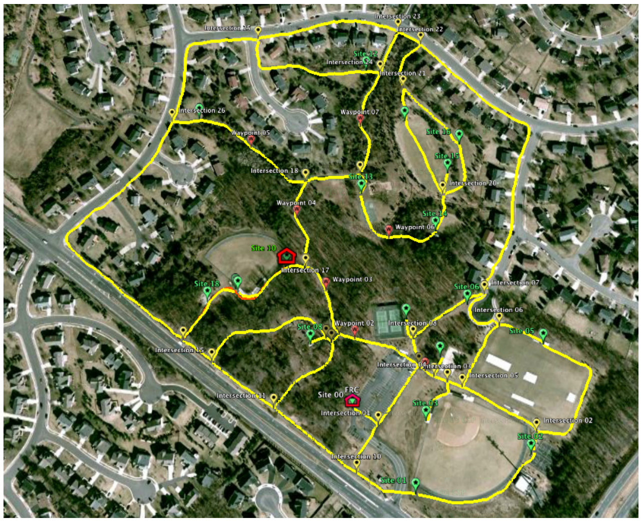

Figure 2: Stanton Woods Park, Herndon, VA

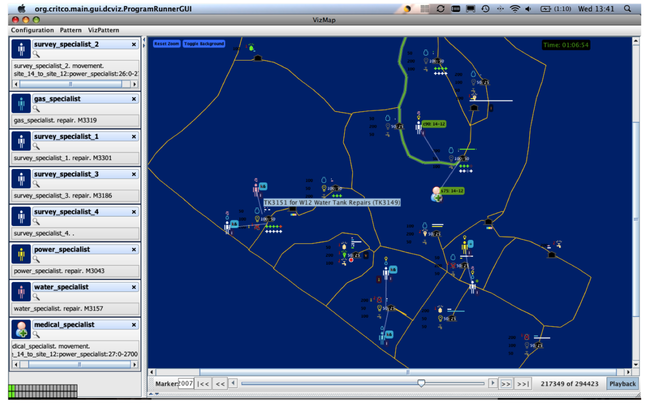

Figure 3: Screenshot of Coordinator Agent

be off, which required agents with those skills. Two agents had to be at the same location simultaneously to load a critically injured person or repair a road block. Repair options could involve multiple tasks and require two agents with certain skills to act in synchrony or in a particular sequence. Some repair options required kits which guaranteed their success, but kits were available only at warehouses. Agents could transport at most one entity, i.e, either a repair kit or a single casualty. A substation was considered repaired only if the corresponding main station was also repaired. A clinic was not operational until all substations at the site and all corresponding main stations were repaired. These are examples of rules that, along with the dynamism and uncertainty in outcomes mentioned earlier, created challenging real-time real-world distributed coordination problems.

The goal was to see if humans operating with radios and a multi-agent decision-support system could outperform humans operating with only radios. Although the field exercises still abstracted some aspects of a real-world disaster scenario, we believe they closely approximated the challenges of helping a human team solve difficult real-world problems.

3. DEMONSTRATION

Agents run on several laptops and allow participants to play a simpler and shorter version of the field exercise. Participants attempt to rescue injured and repair substations as efficiently as possible. Human-guided strageties are encoded and participants execute the strategy with and without the help of the decision-support agents. A screenshot of one view of the coordinator agent can be seen in Figure 3.

4. TECHNICAL CONTRIBUTIONS

We developed a new approach named STaC (Subteams-Tasks-Constraints) based on the premise that people have good intuitions about how to solve problems in each domain. The idea is to enable users to encode their intuition as guidance for the multi-agent system and to use this guidance to vastly simplify the problems that the system needs to address. The approach is related to heuristic planning, but differs in two important aspects. First, the goal is to capture intuition about solving specific instances of the problem rather than providing heuristics that apply to many instances in the domain. End-users rather than domain experts or developers encode heuristics for the system. Second, in STaC, the intuition is not captured by rules of what actions to take in specific situations, but rather as a decomposition of the problem into simpler problems that can be solved independently.

The key to STaC is using the model and guidance to produce sufficiently smaller task structures that can be centralized so that a single agent can determine who does what, when and where with respect to these significantly simpler task structures. This mitigates the distribution challenge and enables using auxiliary solvers based on established techniques which produce good solutions at a smaller scale. These smaller task structures are solved independently assuming that the human guidance has addressed any significant dependencies.

STaC addresses tracking the dynamism in these task structures, the transitioning of agents assignment between these smaller task structures and the invocation of auxiliary solvers. Given that the task structures are treated independently and sufficiently small to be centralized, we call them sandbox reasoners. The sandbox reasoners required in each domain are different, so custom code must be written for each domain. However, the benefit of the approach is that sandbox reasoners are significantly simpler than the custom solvers required to produce a custom solution for a domain. Our sandbox reasoners for this domain include both Belief-Desire-Intention reasoning and sampling-based algorithms built on decision-theoretic models such as Markov Decision Processes.

5. REFERENCES

[1] C. Boutilier. Multiagent systems: Challenges and opportunities for decision-theoretic planning. *AI Magazine*, 20(4):35–43, 1999.

[2] F. C. Groen, M. T. Spaan, J. R. Kok, and G. Pavlin. *Real World Multi-agent Systems: Information Sharing, Coordination and Planning*, volume 4363/2007 of *Lecture Notes in Computer Science. Logic, Language, and Computation*. Springer-Verlag, 2007.

[3] R. R. Murphy. Human-robot interaction in rescue robotics. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 34(2):138–153, 2004.