Enable Automated Emergency Responses Through An Agent-Based Computer-Aided Dispatch System

Demonstration

Jihang Zhang, Minjie Zhang, Fenghui Ren
School of Computing and Information Technology,
University of Wollongong
NSW, Australia
jz718@uowmail.edu.au,{minjie,fren}@uow.edu.au

Weicheng Yin, Aden Prior, Claudio Villella, Chun-Yu Chan
School of Computing and Information Technology,
University of Wollongong
NSW, Australia
{derrickyinx,aden.prior,claudiovillella2,chancy1120}@gmail.com

ACM Reference Format:
Jihang Zhang, Minjie Zhang, Fenghui Ren and Weicheng Yin, Aden Prior, Claudio Villella, Chun-Yu Chan. 2018. Enable Automated Emergency Responses Through An Agent-Based Computer-Aided Dispatch System. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 3 pages.

1 INTRODUCTION OF THE APPLICATION DOMAIN

In modern emergency responses, computer-aided dispatch (CAD) systems play a critical role in rescue unit management. Currently, majority of CAD systems that used by emergency departments around the world could provider three major functionalities for emergency operators, which are (1) sending dispatch orders to rescue units; (2) recording the information of emergency incidents and dispatched units into databases and (3) tracking dispatched rescue units [5, 9]. Although using CAD systems could simplify the management of rescue units, emergency operators still need to manually select appropriate units and deploy them for emergency incidents. According to our analysis of the open database of San Francisco Fire Department [1], the time used for the unit dispatch of an emergency incident usually occupy 35% to 45% of the total unit deployment time, which is a time-consuming process for life-threatening incidents such as house fires or vehicle accidents. Besides, with the rapid growth of emergency service demands in metropolitan regions [2, 3], emergency operators might need to simultaneously handle the emergency responses of multiple incidents with different severities, contents and rescue unit requirements.

Over the years, agent and multi-agent technologies have provided promising solutions for handling emergency responses, due to their abilities of addressing critical needs in high-speed, task-critical applications and systems, where uncertainty, dynamic environments, mutual interdependencies and sophisticated control play a significant role [4]. In this demonstration, we present an agent-based computer-aided dispatch (ACAD) system to organise multiple emergency departments to automatically dispatch appropriate rescue units for different types of emergency incidents. The proposed system contains a server-side application to manage emergency incidents and dispatch rescue units on GoogleMaps and a mobile application to provide interactions between incidents and rescue units during an emergency response process.

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

2 PROBLEM DESCRIPTION

Emergency incidents that occur in urban areas are usually characterised by their unpredictability, uncertainty and variability, which require the collaboration of different types of rescue units to handle their emergency responses, such as ambulances, police cars and fire engines. Normally, different emergency departments might have their own unit management systems with different regulations to control their units. Using traditional centralised systems [6, 7] to manage all units that come from different emergency departments could be extremely difficult to be realised in real-world environments. Besides, some emergency departments might have sensitive resource information that they do not want to share with other departments. Furthermore, centralised systems are usually vulnerable to the single point of failure [8]. It refers to when one system fault or malfunction might results in the breakdown of the entire system, which could cause devastating effects in emergency response. Therefore, to automatically dispatch rescue units from emergency facilities, the critical problem needs to address is to design an effective and reliable solution to manage the rescue units from different emergency departments.

3 OVERVIEW OF THE SYSTEM DESIGN

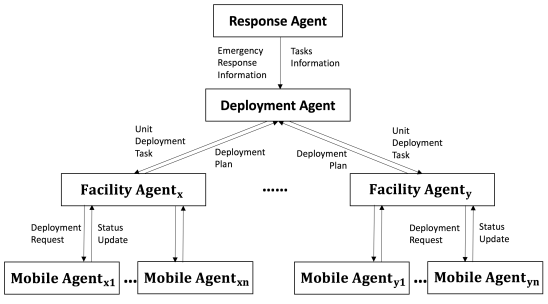
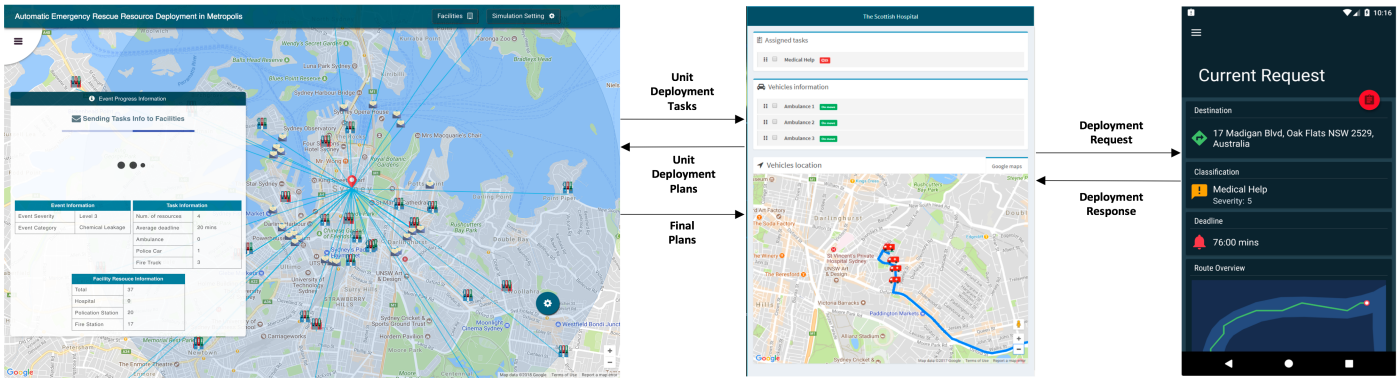


Figure 1: The Interactions between Agents for a Single Emergency Response



(a) Emergency Response Control Interface (b) Facility Agent Interface (c) Mobile Agent Interface

Figure 2: Agent-Based Computer-Aided Dispatch System

The ACAD system is designed as a multi-threads system with a decentralised system architecture to handle concurrent emergency incidents simultaneously. For each incident that is received by the system, a processing thread is generated to handle the response of the incident, which composes of two components — the emergency response control and the rescue unit management. The emergency response control component contains two types of agents, which are *response agents* and *deployment agents*, while the rescue unit management component also contains two types of agents, which are facility agents and mobile agents. The interactions between agents for a single emergency response are depicted in Figure 1.

In detail, a response agent has the responsibility to analyse and monitor the emergency response of an incident. A deployment agent has the responsibility to handle unit deployment tasks and generate unit deployment plans for an emergency response. A facility agent has the responsibility to process the tasks that come from a deployment agent and manage a set of mobile agent in an emergency facility (i.e. hospital, police station, etc.), while a mobile agent has the responsibility to manage a mobile rescue unit. Generally, facility agents could be implemented as customised unit management applications to work together with emergency facilities or APIs to be integrated into the facilities' own unit management systems. By doing so, our system could overcome the unit dispatch difficulty cross heterogeneous unit management systems from different departments. Furthermore, since rescue units are managed and controlled by independent facility agents with a decentralised manner, it means the system failure of one facility agent will not affect the operations of other facility agents and their relevant mobile agents.

4 DEMONSTRATION FEATURES OF THE SYSTEM

The proposed ACAD system is implemented as a web-based system, which contains a server-side application and a mobile application that can interact with the server application. In detail, the server-side application is designed as the emergency response control centre (see Figure 2 (a)) based on GoogleMaps, which can access the real-world information to demonstrate the system's functionalities, such as emergency facilities, roads and traffic conditions. The server-side application allows users to randomly generate emergency incidents with different attribute settings on GoogleMaps. When the information of an emergency incident is received by the ACAD system, a processing thread will be automatically generated to handle the emergency response of this incident. First, the emergency response control component assigns a response agent to analyse the incident information (i.e. incident content, severity, required units, etc.) and generate a set of unit deployment tasks. Each unit deployment task only requires one type of rescue units. Then, these tasks are passed to a deployment agent, who will send task information to relevant emergency facilities near the incident location.

In the unit management component, a facility agent, who received a unit deployment task, needs to propose a unit deployment plan based on the availability of the facility units and sends the plan back to the deployment agent. Users can click an emergency facility icon on GoogleMaps to expand a facility agent's interface (see Figure 2 (b)), which provides the information of assigned unit deployment tasks, on-mission rescue units on GoogleMaps and available units in the facility.

After the deployment agent receives all unit deployment plans from selected facility agents, it needs to combine these plans and uses a predefined cost function to generate an optimal plan for each unit deployment task. In our system, the cost function, used for the plan calculation can be customised with different cost attributes and weightings according to local emergency departments' response codes and protocols. Finally, each of the unit deployment plans will be sent to relevant emergency facilities, in which facility agents will inform mobile agents to execute the unit deployment requests, which is simulated by our mobile application (see Figure 2 (c)). The mobile application will display the details of the emergency response. If users choose to accept the unit deployment request, the mobile application will calculate the shortest route from the unit's facility to the incident location and simulate the unit movement on GoogleMaps based on the live traffic information. Users can track a rescue unit through our mobile application, and the movement of the unit will synchronise the movement on the facility agent interface and emergency response control interface.

REFERENCES

[1] [n. d.]. Fire Department Calls for Service, https://data.sfgov.org/Public-Safety/Fire-Department-Calls-for-Service/nuek-vuh3. ([n. d.]). https://data.sfgov.org/Public-Safety/Fire-Department-Calls-for-Service/nuek-vuh3

[2] Patrick Aboagye-Sarfo, Qun Mai, Frank M Sanfilippo, and Daniel M Fatovich. 2016. Impact of population ageing on growing demand for emergency transportation to emergency departments in Western Australia, 2005–2020. *Emergency Medicine Australasia* 28, 5 (2016), 551–557.

[3] Patrick Aboagye-Sarfo, Qun Mai, Frank M Sanfilippo, David B Preen, Louise M Stewart, and Daniel M Fatovich. 2015. Growth in Western Australian emergency department demand during 2007–2013 is due to people with urgent and complex care needs. *Emergency Medicine Australasia* 27, 3 (2015), 202–209.

[4] Federico Bergenti and Agostino Poggi. 2012. Developing smart emergency applications with multi-agent systems. In *Emerging Communication Technologies for E-Health and Medicine*. IGI Global, 31–44.

[5] Richard A Comroe, Richare H Coe, Katherine B Roane, and Lewis H Rosenthal. 1990. Computer aided dispatch system. (May 15 1990). US Patent 4,926,495.

[6] Aygul Gabdulkhakova, B Konig-Ries, and Dmitry A Rizvanov. 2011. An agent-based solution to the resource allocation problem in emergency situations. In *2011 Ninth IEEE European Conference on Web Services (ECOWS)*. 151–157.

[7] Glenn I Hawe, Graham Coates, Duncan T Wilson, and Roger S Crouch. 2015. Agent-based simulation of emergency response to plan the allocation of resources for a hypothetical two-site major incident. *Engineering Applications of Artificial Intelligence* (2015).

[8] Gary S Lynch. 2009. *Single point of failure: The 10 essential laws of supply chain risk management*. John Wiley and Sons.

[9] Richard Saalsaa. 2006. Method and system for integrating a computer aided dispatch system with an emergency medical dispatch protocol. (Sept. 12 2006). US Patent 7,106,835.