DrillSim: A Simulation Framework for Emergency Response Drills

Vidhya Balasubramanian, Daniel Massaguer, Sharad Mehrotra, and Nalini Venkatasubramanian

Donald Bren School of Information and Computer Science
University of California, Irvine
Irvine, CA 92697, USA
{ `vbalasub, dmassagu, sharad, nalini`}`@ics.uci.edu`

Abstract. *Responding to natural or man-made disasters in a timely and effective manner can reduce deaths and injuries, contain or prevent secondary disasters, and reduce the resulting economic losses and social disruption. Appropriate IT solutions can improve this response. However, exhaustive and realistic validation of these IT solutions is difficult; proofs are not available, simulations lack realism, and drills are expensive and cannot be reproduced. This paper presents DrillSim: a simulation environment that plays out the activities of a crisis response (e.g., evacuation). It has capabilities to integrate real-life drills into a simulated response activity using an instrumented environment with sensing and communication capabilities. IT solutions can be plugged in the simulation system to study their effectiveness in disaster management and response. This way, by using a simulation coupled with an on-going drill, IT solutions can be tested in a less expensive but realistic scenario.*

[1]

1 Introduction

Organized crisis response activities include measures undertaken to protect life and property immediately before (for disasters where there is at least some warning period), during, and immediately after disaster impact. Depending upon the scale of the disaster, crisis response may be a large-scale, multi-organizational operation involving many layers of government, utility companies, commercial entities, volunteer organizations, media, and the public. In a crisis, these entities work together as a virtual organization to save lives, preserve infrastructure and community resources, and reestablish normalcy within the community. During a crisis, responding organizations confront grave uncertainties in making critical decisions. They need to gather situational information (e.g., state of the civil, transportation and information infrastructures) and information about available

---

[1] This research has been supported by the National Science Foundation under award numbers 0331707 and 0331690.

resources (e.g., medical facilities, rescue and law enforcement units). Timely and accurate information can radically transform the ability of organizations to gather, manage, analyze and disseminate information when responding to man-made and natural catastrophes.

While innovations in information technology are being made to support awareness in a crisis, a key issue that must be addressed is evaluating the efficacy of the developed solutions in an actual response process. In other words, strategies must be developed to translate IT metrics into meaningful emergency response metrics that help us analyze the cost-effectiveness of the technologies before deployment. One approach is to mimic a crisis by conducting emergency drills over a sample region, incorporating the IT innovations in the process of response during the drill, and measuring the improvements in the process as compared to some pre-existing baseline. Organizing and conducting emergency drills is a challenging proposition due to several reasons. Drills are expensive in terms of time and resources, significant planning is required to instrument drills. Participation of multiple response organizations, businesses and individuals might need to be coordinated making frequent spot-testing of technology solutions impossible. Furthermore, drills are often carefully scripted making it impossible to test out response to a range of scenarios using "what-if" analysis. In addition, scalability testing of solutions (e.g. evacuation of an entire city) is close to impossible to test via drills.

An alternate solution is to use simulation tools and techniques to understand disasters, their evolution, and the potential impact of IT solutions on the outcome of the response. A simulation based approach allows us to create what-if scenarios dynamically and determine the ability of the response to adapt to the changing disaster landscape. Simulations are also useful for training response personnel and in evaluating solutions and plans for emergency response. The need for sophisticated tools and techniques for the modeling, simulation, and visualization of emergency activities has been articulated in recent reports [17]. Much of the efforts in the area of crisis simulations have focused on modeling disasters and their effects–techniques for radioactive plume modeling, modeling the spread of biological agents, fire modeling, earthquake impact modeling help in understanding the characteristics of the specific type of disaster itself.

What has not been studied in as much detail are simulation tools that help understand the response activity itself (e.g. evacuation, medical triaging, fire fighting, rubble removal) as it unfolds. Commercial evacuation simulators [5, 8, 2] help establish bounds on evacuation of regions and buildings under ideal knowledge conditions; these tools model movement and behavior of people during evacuation. However, they do not capture the interactions between people or the impact of technology in such scenarios.

In this paper, we discuss the design and implementation of a simulation environment for crisis response that addresses the aforementioned gap. A simulation framework for crisis response activities must address the modeling of human behavior (and decisions made by humans) in a changing environment. The proposed simulation framework is a multi-agent system for crisis response activities that mainly (a) embodies agents that drive the simulation in different roles and make decisions and (b) captures the environment under which agents make decisions through the use of a pervasive infrastructure. In addition, the framework incorporates a variety of models that drive the activity and decision making, captures information flow between different entities/agents and integrates the abilities of the infrastructure with respect to the information flow. In addition to scalability (supporting large numbers of entities) and flexibility (extensibility to various crises), the emphasis of our simulation framework is on being able to accurately calibrate the crisis environment and behavior of agents in that environment.

A key aspect of our approach is the incorporation of real-world instrumented framework [6], i.e. a pervasive environment, that can capture physical reality during an activity as it occurs. The instrumented smart space consists of a variety of sensing technologies (video, audio, RFID people sensors) and is used to conduct and monitor emergency drills. This extends the scope of the simulation framework into an augmented reality environment for IT testing in the context of a crisis. Such an integration allows humans to assume specific roles in the multi-agent simulator (e.g. first responder in an evacuation process within a building) and capture decisions made by humans (evacuees, response personnel) involved in the response process. Capturing people's behavior during a drill also allows calibrating the simulation both in terms of physical and cognitive agent response (e.g., speed of movement and decisions taken).

The rest of the paper is organized as follows. Section 2 presents the design details and concepts behind the system. A prototype is presented in Section 3. Given the implementation framework we describe how this system serves to test IT solutions and disaster response methodologies in Section 4. Conclusions and future research directions are discussed in Section 5.

2 Design Details of DrillSim

Figure 1 illustrates the basic methodology of the DrillSim environment which consists of a multi-agent simulator driven by an instrumented smart space. An activity in DrillSim occurs in a *hybrid world* that is composed of (a) the **simulated world** generated by a multi-agent simulator and (b) a **real world** captured by a smart space.

The purpose of this environment is to play out a crisis response activity where agents might be either computer agents or real people playing diverse roles (actors). In order to capture real actors in the virtual space we utilize a sensing infrastructure that monitors and extracts information from real actors that is needed by simulator (such as agent location, agent state, etc.); in other words, we infuse actions and state of human actors in the virtual space. Likewise, to enable a real-actor to participate in a simulated reality, we use appropriate display devices and modalities to provide the real-actor with awareness of the virtual world. While our main goal is to evaluate new techniques for crisis response, one important byproduct of the hybrid world approach is that it becomes an immersive training environment for first responders. Another design consideration of DrillSim is to be able to run DrillSim with other simulators (e.g., communication simulators, crisis simulators). Such simulators are often developed by domain experts and the ability to integrate relevant input to DrillSim from these simulators allows us to model a wider range of aspects more accurately. For instance a crisis simulator can be plugged into DrillSim by translating the crisis parameters to the impact. In the remainder of this section, we describe our approach to modeling the various components of the DrillSim world and in enabling interactions between the two worlds (real to simulated and vice-versa).

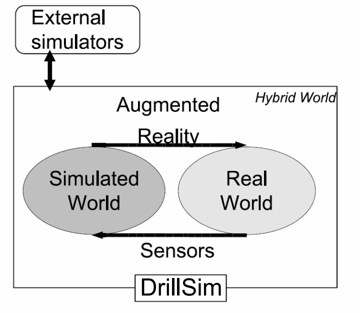


Fig. 1. Methodology

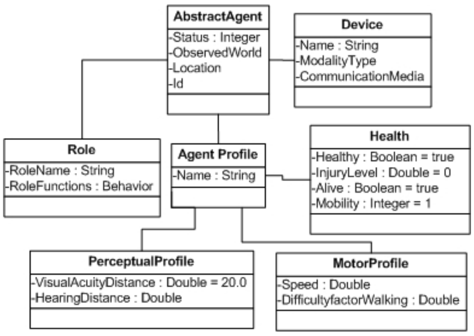


Fig. 2. UML of Agent

2.1 Conceptual Modeling of DrillSim

Two key concepts that drive the design and implementation of DrillSim are *entities* and *scenario*. Together these concepts capture the overall activity over space and time as well as the observed world of each agent (real or virtual) participating in the activity.

DrillSim Entities: The primary entities modeled in DrillSim are the agents, space, resources, crisis, and infrastructure. In our model, entities can either be real or virtual.

Agents are autonomous participants that drive the crisis response simulation and can be real or virtual. Agents may represent an evacuee, a firefighter, a building captain, etc. Every agent has a set of properties associated with it. A UML representation of some sample agent properties is shown in Figure 2. The properties of an agent include its role (e.g., evacuee, fire fighter), the agents physical and perceptual profile (e.g., range of sight, speed of walking), the current health status of the agent(e.g. injured, unconscious), and the devices carried by the agent (e.g. cellphone). At any given time, agents are associated with a given location in the geographical space.

Space is where the response activity is played out and includes both indoor space and outdoor space. Indoor space consists of floors, rooms, corridors, stairways, elevators, etc., while outdoor space consists of buildings, roads, walkways, open spaces of different terrains, parking lots and other external structures.

The *Crisis* models physical phenomena such as spreading fire and spread of a hazardous material. Instead of modeling a crisis, our simulator represents crisis via its impact. Specifically, it represents at any time, for each location the impact of the crisis to other entities (the intensity of fire, the toxicity of chemical spill at the location, etc).

Infrastructure represents the sensing and communication components used to capture the context in which an activity occurs. The components can be a fire alarm within a building that is used by an agent. Again, infrastructure is modeled via its impact to the pervasive space. For instance, instead of modeling WiFi communication using a set of access points, we model whether WiFi communication is available to a particular agent at a particular location.

Scenario Representation: A scenario is essentially the state of the real and virtual entities modeled at a given point of time. We specifically represent scenario as a snapshot taken at every time unit. This snapshot is represented using a grid based representation which is expanded to include information about obstacles, hazards, and occupancy. In this representation we divide the space into equal sized grid cells and every cell contains a tuple of the form $G_{i,j} =< Obstacle, Occupied, Hazard >$. *Obstacle* is a value between 0 and 1 and represents the difficulty an agent faces in traversing a cell. *Occupied* contains a list of agents occupying the cell. Hazard contains a list of hazards present on that cell. While the cell-based representation is simplistic it is rich enough to capture a variety of crisis activities. Our future plans are to expand the representation further to capture other entities like cell tower, fire extinguishers etc.

2.2 Virtual Reality/Augmented Reality Integration

Virtual reality/augmented reality integration requires projection of the simulated world to the real world and vice versa. To augment the real world with the simulated world, the necessary Virtual Reality/Augmented Reality (VR/AR) interfaces have to be designed, such that real persons taking part in a drill can interact with the hybrid world and take decisions based on their observation of this hybrid world. This immersion is achieved via the appropriate GUIs on portable devices (cellphone, PDA, etc) carried by the person. There are several challenges in bringing the simulated world to the real world. The first set of challenges arise from limitations in the device such as processing, storage, and energy capabilities. To address the restrictions imposed by hand-held devices, the interface at the hand-held device is simplified to contain a minimal set of functions that enable the user to effectively communicate with the agents in the simulation. Furthermore, synchronization issues also arise due to communication delays between server and clients and due to the different processing capabilities of different clients. The second set of challenges emphasize the need for a customized view of the simulation based on user location and orientation. This has to be done in real-time so that appropriate contextual information is sent to the user as and when needed. Key technologies to enable this include quality-aware localization [12] and adaptive content customization.

To capture a real agent participating in the activity we must have a mechanism to observe the real agents and their actions in space. We have instrumented a *smart space* within our campus that allows us to capture phenomena taking place in the real world. The smart space consists of integrated sensing and communication infrastructure that includes video and audio sensors, people counters, built-in RFID technology, powerline, ethernet, and wireless communications. Specifically, real sensors and communication devices are used to capture physical space and phenomena, and to monitor people positions and actions. This instrumentation also enables calibrating both the agent's action parameters (e.g., speed of movement) and decision-making mechanisms. Furthermore, this smart space provides an infrastructure for IT researchers to test their solutions (e.g., 802.11-based localization [10]). One of the key issues in the collection of dynamic multimodal data is in aggregating and interpreting the collected data in real time. Also, incompatibility and unreliability of multi-sensor data has to be addressed using suitable sensor fusion (e.g. probabilistic) techniques. Additionally there are privacy issues regarding collected data about real people participating in a drill. Our research within the RESCUE [18] project addresses several of these challenges.

2.3 DrillSim Operational Dynamics

We begin the simulation by generating the current scenario on a chosen geographical space using the pervasive infrastructure, i.e. the current state of real world entities (e.g. location of people, state of resources) is used to calibrate an initial scenario generator. The current crisis, which is generated by the crisis generator, represents both the disaster as well as the changes that occur as an effect of the disaster. These changes are reflected both in the geographical space and the scenario, and updated as the simulation progresses through the shared cell-based representation of space.

The main entities that drive the simulation are the agents. Agents wake up every t time units and execute some actions. Specifically, on waking up, an agent acquires awareness of the world around it, transforms the acquired data into information relevant to decision making, and makes decisions based on this information. The acquired awareness is stored as the observed world of the agent (and may be augmented using information from the instrumented smart space). An agent takes decisions using the awareness acquired. Based on the decisions, it (re)generates a set of action plans. These plans dictate the actions the agents attempt before going to sleep again. For example, hearing a fire alarm may result in the decision of exiting a floor, which in turn may result in a navigation plan to attempt to go from the current location to an exit location on the floor. Given this navigation plan, the agent executes one action of the plan every time, e.g. it walks one step towards the exit of the room. An impact of changing global environment, crisis, and actions of other agents is the possible change of plan or even a change of decision. Decisions, plans, and actions are logged by the agents as and when they change.

3 DrillSim: System Implementation

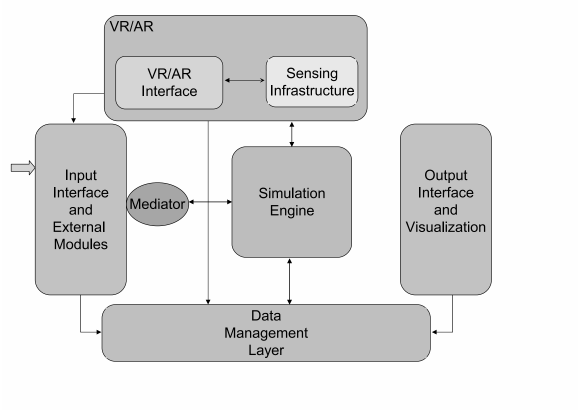


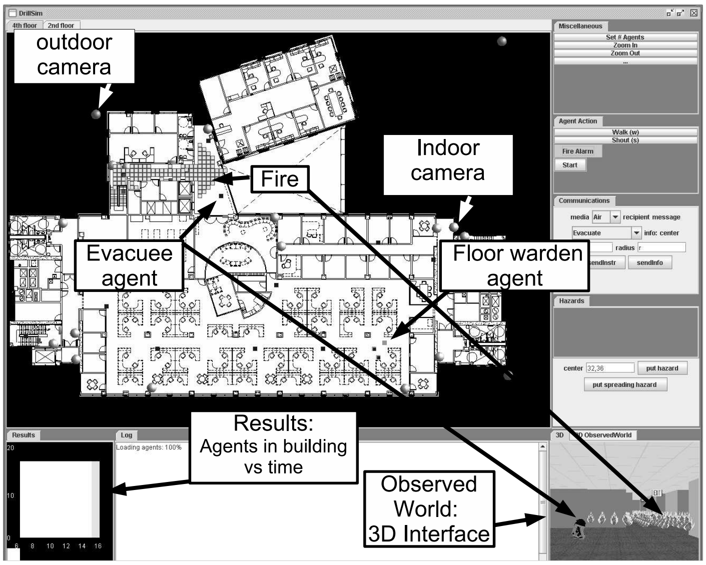
Fig. 3. Software architecture of DrillSim

We now describe the architecture and implementation of the DrillSim simulation framework. The prototype of the simulator models a multi-agent evacuation activity at a campus level. In this system there are two types of agents–the evacuee and the floor warden–whose role is to evacuate the floor during an emergency. The architecture of DrillSim is shown in Figure 3. The primary components are the I/O interfaces, simulation engine, data management module, and the VR/AR modules. In addition there is a database server which holds the spatial data. These components are described in the sections below.

3.1 I/O Interfaces of DrillSim

Figure 4 shows a sample screenshot of the I/O interface of the system. The interface allows a set of inputs to be provided to the simulation, outputs the results of the simulation, and also allows users to interact with the simulation environment. The user can start the simulation, start the crisis when needed, turn on the fire alarm once a crisis starts, communicate with other agents, and control an agent. The user interfaces are built using Java and Java3D.

Inputs to DrillSim: allow users to initialize parameters to the properties of DrillSim entities, specified in Section 2.1, and the initial scenario. Agent inputs include the definition of roles and their behavior, definition of profiles to be associated with the different agents, location of different people at the beginning of the simulation, the devices agents carry, etc. In the current implementation the geography, roles of agents, agent behavior definitions and profiles are predefined. Location of agents at the beginning of the simulation is entered by the



media Air

Evacuate

radius r

center 32,36

Loading agents: 100%

Fig. 4. Snapshot of DrillSim User Interface

user. Inputs pertaining to resources, crisis and infrastructure are entered by the user as a spatial layer on a geographical map. Users can also input response procedure plans to the system. The inputs to DrillSim come both from users who instantiate parameters and from external models that relate to the actual event (crisis and its response) and the entities in the event (the resources, people, devices, etc). The inputs can be dynamic as in the scenario generated by external modules (e.g., crisis model, scenario generator) or static parameters initializing the scenario (e.g, time of start of simulation, total number of people in building, location of resources). In addition, inputs can be derived from the sensors in the pervasive space, which dynamically feed back user counts, user locations, resource locations, and so on. There is a mediator for every external model that translates the data to the grid representation format understood by the simulator. The inputs are either sent directly to the engine or stored in the database. The geography is stored in the database. Also embedded within the DrillSim are the geography (in the database) and the response plan which outlines the basic response rules followed in our campus.

Outputs of DrillSim: While the other modules in DrillSim model and generate the response activity, it is also essential to generate output of the activity so that the end user can observe the simulation and evaluate the results. The output modules capture the results of the simulation through a 2D/3D visualization or as statistics. The visualization is a hybrid output with both the real world and simulated world overlapping and can be both in 2D and 3D. While the 2D visualization provides a birds-eye view of the entire activity or of the observed reality of a specific agent, the 3D observed reality is the view of a camera set in any arbitrary position or on top of any agent. The latter is useful both to understand what a simulated agent decides based on its observed world and to be able to take the role one of the agents by controlling it. The statistics of the simulation include disaster response metrics such as speed of evacuation and number of people injured. These metrics help to study the effectiveness of the solutions plugged into our system. Recall the primary purpose of this simulator is to evaluate IT solutions by translating IT metrics to disaster metrics. At every instance of time, the simulator updates a graph of *time* versus *agents in the building*. This graph is an example of a method to show the impact of different IT solutions in the context of an evacuation.

3.2 DrillSim Simulation Engine and Data Management Module

The *simulation engine* is the principal component that drives DrillSim by playing out an evacuation activity. The simulation engine is driven by different agents and, in our current implementation, is developed using Java and JADE [11]. It consists of the simulated geographic space, the current evacuation scenario (i.e. where people are and what they are doing), and the current crisis as it unfolds. The functionality of the engine is supported by different agents that represent the human population involved in the response. In our engine the two agents represented are evacuees and floor warden. Their representation also conveys when they are receiving messages from other agents and when the floor warden agent is activated. The agents follow the operational dynamics as described in Section 2.4. Decision-making in an agent is modeled as a stochastic neural network. In particular, it is a recurrent neural network [13] that outputs the probability of taking each decision (e.g., evacuate the floor, exit the building). The input to the neural network is the agent's information, the probability of taking each decision, and the decisions made. The weights of the neural network (i.e. the weights given to every input) are set according to the agent's profile and calibrated by running real emergency response drills within the smart space. Modeling decision-making as a neural network allows for explicitly modeling the importance of each piece of information on each decision, setting the emphasis on the impact of information on decision making rather than on the reasoning process itself. The engine is the most computationally intensive module in DrillSim which results in scalability issues (i.e., as more agents are added, they compete for limited resources). We are working on strategies to distribute the computation uniformly to improve the efficiency of the system.

The data management module manages the data exchange between other components. The data relevant to the simulator is stored in a database. The geography of both indoor and outdoor spaces are converted to the GIS format and stored in the database. There is a JDBC interface available for agents to access this spatial information in order to make decisions. The data management module is responsible for 1.- managing continuous queries from agents regarding the environment, other agents etc, 2.- managing highly frequent updates, and 3.- logging the events as they happen. An important aspect of this module is the representation of the spatial and temporal data so as to adequately support functioning of the simulated activity.

3.3 Virtual Reality/Augmented Reality Integration in DrillSim

In order to allow the projection of the virtual world to real people, the visualization interface is extended to allow the user both to observe the simulation and to participate in it. Specifically there are three versions of the interface implemented. The EOC version of this interface is running on a laptop or a PC and can be connected to a 5x11 multi-tier display. The EOC version also allows a user to manage the entire simulation, see what an agent (real or virtual) is seeing, send messages to other agents, and control the actions of an agent. In the current implementation the PDA version is a low resolution version of the same interface. The output of the computer is connected to a pair of MicroOptical SV-6 VR glasses [1]. In order to provide the customized output for the user carrying the PDA we need to track and localize the user. This is achieved using the instrumented smart space covering one quarter of our campus. Indoor localization is achieved via WiFi based localization technologies such as the Ekahau Positioning Engine [3]. Sensor fusion based techniques using WiFi, Bluetooth, RFID triangulation and GSM are also being implemented and tested. In addition, people counting technologies (video-based counting) also help feed the real time data of location of people to generate more realistic dynamic scenarios.

4 Testing and Validation

An important purpose of the simulator is for advancing research in human behavior, emergency response processes, and developing IT solutions in the context of emergency response. This section describes how DrillSim can be used to test impact of IT techniques and response procedures on disaster response. Note that the experiments are here included for illustration purposes only, since the validity of the results depends on the validity of the behavior model. Studying the validity of the behavior model and calibrating the behavior model remains part of our future work.

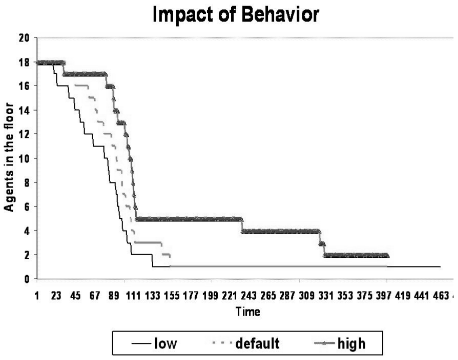


Fig. 5. Impact of Human Behavior

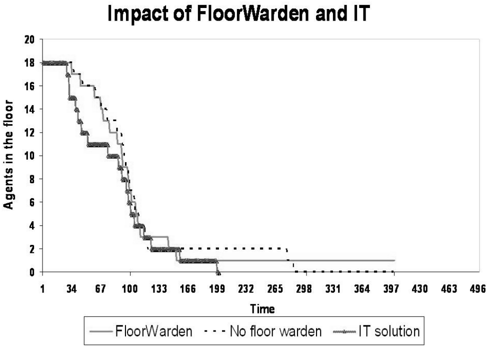


Fig. 6. Impact of FloorWarden and IT

The prototype simulates an evacuation activity in one building with four floors. Specifically, the experiments are performed on the 4th floor of the building, with 20 agents evacuating the floor due to a fire. There are two roles taken by agents: floor warden and evacuees. The floor warden is responsible for evacuating the floor by going to every room and making sure the evacuees leave the building, and is the last to leave the floor. The evacuee agents' decisions include exiting the building, and telling others to exit the building. The relevance each evacuee gives to the fire alarm or to each other agent requesting to evacuate (including the floor warden) depends on the obstinacy level–the higher the obstinacy level the less likely the agent will evacuate.

In the first set of experiments given the activity chosen (evacuation in the presence of fire) we study the impact of human behavior by studying the impact of obstinacy of agents in such situations. Figure 5 shows the relationship between evacuation time and agent obstinacy levels. As expected, the more obstinate the agents, the longer time it takes to evacuate them. The second set of experiments demonstrate the impact of floor warden on the process of evacuation as shown in Figure 6. We can notice that those agents who are obstinate even after hearing the fire alarm leave earlier due to the presence of fire alarm. However we can see that the floor warden is in the building long after people have left in order to search the entire floor. An IT solution that can tell the floor warden when the floor has been evacuated, he can leave earlier The plot labelled IT Solution indicates the impact of this IT solution in the evacuation process, and shows the floor warden leaving earlier.

5 Related Work and Concluding Remarks

Evacuation simulation tools like Myriad, Simulex, and Egress and others [5, 8, 2] model movement and behavior of people during evacuation. Multi-agent simulators like the efforts in Robocup-Rescue Simulation Project [7] and the evacuation simulator developed as part of the Digital city project in Kyoto, Japan [19] simulate not only the civilian movement but also the activities of the response personnel. While the efforts mentioned address individual aspects of disaster management these tools do not address the overall emergency response activity. A few efforts have been directed towards integrating different tools [9, 4, 14]. For instance the SOFIA project at Los Alamos National Laboratory is aimed at developing actor-based software for analyzing infrastructures that are interdependent. The Integrated Emergency Response Framework from National Institute of Standards and Technology [14–16] targets integration of different simulation tools to address overall emergency response. Our simulator not only simulates activities but also integrates the actual instrumented infrastructure, enabling us to immerse real people in the simulation. The primary goal is to use this emergency response activity view in order to integrate IT solutions at appropriate interface points in the simulator and test the effectiveness of IT solutions in disaster response.

In this paper, we described DrillSim. Such a simulation framework that merges reality and simulation opens up opportunities to recreate more realistic response activities and test solutions and models in this context. Designing such a simulator opens many research challenges related to modeling and data management. Mixing of real with virtual worlds, dynamic data management, modeling the geographical space and representing events on it, and modeling agents to mimic human behavior in crisis are some of these challenges. We have addressed a few of them in this paper. We have also demonstrated how DrillSim can be used to test solutions for disaster response. We are focussing on modeling spatial data and agent behaviors as part of ongoing work. There are scalability and synchronization issues when designing a large scale crisis response simulator. Our eventual goal is to support scalable plug and play of crisis response activities over a variety of scenarios and geographies.

References

1. MicroOptical-SV-6 PC Viewer specification, 2003.
2. EGRESS. http://www.aeat-safety-and-risk.com/html/egress.html, 2005.
3. Ekahau Positioning Engine. http://www.ekahau.com/, 2005.
4. Interdependent Infrastructure Modeling, Simulation, and Analysis Project (SOFIA), Los Alamos National Laboratory. http://www.lanl.gov/orgs/d/d4/infra/sofia.html, 2005.
5. Myriad. http://www.crowddynamics.com, 2005.
6. Responsphere. http://www.responsphere.org, 2005.
7. Robocup-Rescue Simulation Project. http://www.rescuesystem.org/robocuprescue/, 2005.
8. Simulex: Simulation of Occupant Evacuation. http://www.iesve.com, 2005.
9. The Urban Security Initiative, Los Alamos National Laboratory. http://www.lanl.gov/orgs/d/d4/infra/urban.html, 2005.
10. Bahl, P., and Padmanabhan, V. N. RADAR: An In-Building RF-Based User Location and Tracking System. In *INFOCOM (2)* (2000), pp. 775–784.
11. Bellifemine, F., Poggi, A., Rimassa, G., and Turci, P. "An object oriented framework to realize agent systems". In *WOA 2000* (May "2000").
12. Cristoforetti, J. Multimodal Systems in the Management of Emergency Situations. Master's thesis, Universita de Bologna, 2005.
13. Haykin, S. *Neural Networks - A Comprehensive Foundation.* Prentic Hall, 1999.
14. Jain, S., and McLean, C. R. A Framework for Modeling and Simulation for Emergency Response. *Winter Simulation Conference* (2003).
15. Jain, S., and McLean, C. R. An Integrating Framework for Modeling and Simulation of Emergency Response. *Simulation Journal: Transactions of the Society for Modeling and Simulation International* (2003).
16. Jain, S., and McLean, C. R. An Architecture for Modeling and Simulation of Emergency Response. *Proceedings of the 2004 IIE Conference* (2004).
17. John L. Hennessy, D. A. P., and S.Lin, H.
18. Mehrotra, S., Butts, C., Kalashnikov, D., Venkatasubramanian, N., Rao, R., Chockalingam, G., Eguchi, R., Adams, B., and Huyck, C. Project rescue: Challenges in responding to the unexpected. *SPIE Journal of Electronic Imaging, Displays, and Medical Imaging*, 5304 (2004), 179–192.
19. Yohei Murakami, Kazuhisa Minami, e. a. Multi-Agent Simulation for Crisis Management. *KMN* (2002).