# COLLABORATIVE MODELING, SIMULATION, AND VISUALIZATION FRAMEWORK FOR AIRPORT EMERGENCY

**Shafagh Jafer**

Department of Electrical, Computer, Software, and Systems Engineering
Embry-Riddle Aeronautical University, Daytona Beach, FL
jafers@erau.edu

**ABSTRACT**
Airport is a highly dynamic environment, turning emergencies into extensively costly and disruptive events with cascaded effects that will adversely affect the entire aviation system. Airports' management and safety divisions exercise precise and case-based procedures to accommodate any possible emergency scenarios. With this regards, airport emergencies have a number of challenges that require better and more adaptive way of addressing. Challenges such as lack of adaptive and customizable simulation-based decision support tools, infeasibility of conducting frequent physical emergency exercises, and the time-critical nature of emergencies in general raise an immediate need for a technology that collaboratively performs modeling, simulation, and visualization to guide first responders throughout the decision-making process. Here we propose a collaborative emergency modeling, simulation and analysis software system that provides an all-in-one simulation technology for enhancing emergency response at airports.

Keywords: emergency simulation, airport security, risk analysis, disaster analysis, what-if scenarios

## 1. INTRODUCTION

Aviation safety and security has always been recognized as one of the top challenges of the Department of Homeland Security. Previous CREATE project (SAAS) has attempted simulating "in-the-air" emergency events and their risk/cost analysis focusing only on aircrafts behavior under attack situations. Since many of aviation related emergencies are affected by attacks originating from the airport ground, it is crucial to be able to simulate emergencies at the airport itself. The recent shooting in LAX is one catastrophic event that disrupted not only the normal operation of the airport itself, but also the entire aviation system (1,550 flights, 167,050 passengers, air cargo system, car rentals and hotel reservations, sales at airport restaurant sand store, and all related services supporting the aviation domain were affected). Efficient crowd guidance methods are one of key concerns of airports' personnel under emergency evacuations. Handling such events is only possible through precise estimation and training. To mitigate the enormous financial and human loss caused by airport emergencies, here we propose a full-scale modeling, simulation, and visualization framework that provides an all-in-one solution to airport security and emergency response planning. The tool addresses multiple interdependent aspects of airports' emergency scenario by integrating different aspects of an emergency situation, providing situational awareness in the form of the entire picture to planners, trainers, and first responders. The proposed framework provides real-time modeling and analysis capabilities for the airport risk analyst. The tool allows the user to quickly define a microscopic model of the airport emergency scene by selecting the airport area (e.g. gate, terminal, or security checkpoint), disaster type, crowd reaction behavior, and crowd guidance strategy. It will then takes in virtual- and real-time data feed, performs rapid emergency response analysis by incorporating live and historic information. Simultaneously, the tool will generate real-time 3-D visualization of the entire disaster scenario (airport environment, progress of the disaster event, and crowd behavior), allowing airport security officers to obtain live feedback and analysis of the emergency situation. As an emergency simulating and analysis tool, the framework will significantly improve the DHS's capability in the emergency response and risk management field. Specifically, the FLETC (Federal Law Enforcement Training Center) and TSA (Transportation Security Administration) could benefit from this tool to optimize their emergency guidance strategies. Requirements elicitation for the project will be conducted from the personnel (a current Director of Public Safety and a former Airport Manager) in Daytona Beach International Airport in Florida. The first prototype of the framework will be tested by security personnel and risk analyst from this airport.

## 2. BACKGROUND AND MOTIVATION

The existing airport simulation technologies (RescueSim, paxport, and CAST) are mostly used for training purposes, providing augmented virtual environments for practicing response to emergency situations. These technologies mainly focus on a single aspect of the emergency scenario (e.g. modeling only one type of disaster event or simulating only the disaster and not the environment) and usually lack real-time data processing and analysis capabilities, thus, fail to provide a holistic and effective representation of the emergency situation. This research attempts to provide a full-scale simulation-based decision support technology that provides a real-time modeling, simulation, and analysis benchmark to aid emergency analysts and personnel in identifying optimal tactical strategies for handling disaster scenarios at airports. With integrated risk/cost assessment capabilities, the tool helps emergency responders make optimized decisions under multi-aspects and dynamically changing situations. The tool supports both real-time and virtual-time simulations and is operated under two modes: learning/training (processing historic data in advance of actual crisis), and live response (processing live data as the emergency is happening). Providing a holistic emergency response framework, it attempts to address the following goals:

- Providing full-scale, adaptive, and configurable models of natural (earthquake, flood, fire, etc.) and manmade disasters (terrorist attacks, explosives, chemical spill, etc.);
- Modeling disaster impact to study and estimate the impact of a disaster event (e.g. analyzing crowd guidance strategies, as well as evacuation time and speed);
- Analyzing the emergency response activities (e.g. crowd guidance) by monitoring the disaster event in real-time using the live 3-D visualization feedback of the scene;
- Estimating disaster progress based on historic and real-time data feed, allowing significant disaster cost reduction or even prevention;
- Logging large-scale data of disaster management and emergency response for use as database for estimating and analyzing current and future scenarios;
- Offering emergency assistant to responders by quickly providing optimal evacuation strategies for efficient evacuation of the crowd from the emergency scene;
- Allowing for collaborative interfacing and interoperability with existing and future disaster simulation technologies for extending the tool to simulate first responders' activities.

Aiming for the goals stated above, the proposed tool will be implemented to provide the user a framework to define disaster events over an airport area, thus, dynamically analyzing the crowd evacuation behavior, the progress of the disaster event, and make on-demand and real-time tactical adjustments to the emergency response activities.

## 3. ARCHITECTURE OVERVIEW

The proposed airport emergency simulation and analysis system is based on discrete-event modeling and simulation theory. The airport area, crowd behavior, and the disaster models are constructed using the Discrete-Event System Specification (DEVS) formalism (Zeigler, Praehofer, and Kim 2000) and its cellular extension (Cell-DEVS) (Wainer 2009). The tool will be implemented on top of an existing open-source discrete-event modeling, simulation, and visualization framework called CD++ (Wainer 2002). The CD++ toolkit implements DEVS theory and supports model creation and real-time/virtual-time execution, as well as dynamic generation of 3-D scenery. The proposed system is composed of three main subsystems as shown in Figure1:

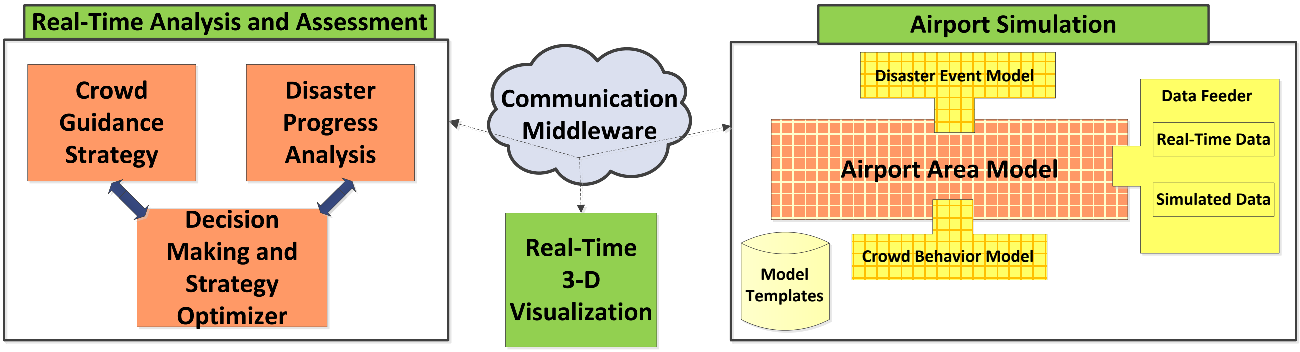


Figure 1: System Architecture

**Airport Simulation:** to model and simulate the complete disaster event and the surrounding environment. This subsystem is further divided into six components:

- *Airport Area Model*: a grid of discrete-event cellular entities that represent the environment model, depicting an actual airport area (gate, terminal, TSA checkpoint, etc.).
- *Disaster Event Model*: a cellular discrete-event model representing the disaster event (for instance, spread of fire at the gate due to an explosion).
- *Crowd Behavior Model*: a cellular discrete-event model representing the flow of the crowd at the disaster scene (people are modeled as independent agent entities).
- *Data Feeder*: a data source unit providing simulated (based on historic information) or real-time data such as spread of fire, crowd flow direction, or building collapse.
- *Model Templates*: the user has the option of selecting various models from the templates repository. These templates provide various already-built models than can be used "as-is", extended, or even modified. The templates

repository includes various models for: airport area layout (area layout for gates, terminals, checkpoints, etc.), natural and manmade disaster types (earthquake, fire, explosion, terrorist attack, shooting, etc.), and crowd emergency behavior (herding behavior, multi-directional flow, uneven density distribution, etc.). Given the duration of this project, only a number of sample models will be implemented for each category. Some crowd evacuation and behavioral models already exist from model repository available from CD++ users community (http://cell-devs.sce.carleton.ca/). Additionally, the user will be given instructions on how to define their specific model and save it in the repository for future reuse. The CD++ toolkit already supports this and provides an easy and user-friendly model development environment.

**Real-Time Analysis and Assessment**: to provide real-time and on-demand feedback to airport risk analyst and security personnel. Statistical cost/risk analyses are provided which evaluate alternative evacuation strategies in the form of text and graph.

- *Crowd Evacuation Guidance*: suggests crowd evacuation guidance strategies (egress models (Tsai and et. al. 2011)) that get integrated dynamically with the Crowd Behavior Model. The suggested guidance method will affect crowd behavior. The resulting behavior is instantly observed through the visualizations generated from the Crowd Behavior Model.
- *Disaster Progress Analysis*: provides current and estimated progress of the disaster event (e.g. percentage of terminal area affected by the emergency event, fire spread rate, etc.). This entity is fed dynamically by the Crowd Behavior and Disaster Event Model.
- *Decision Support and Strategy Optimizer*: evaluates current evacuation process, provides assessment on alternative evacuation strategies, and searches for optimal strategy. This entity provides decision making support based on historical and live data to guide emergency responders in identifying optimal tactical strategies. The framework uses two metrics to evaluate the response strategies: 1) evacuation speed, 2) number of people evacuated. The tool also includes a logging facility that records all events that occur during a simulation. These logs can be used to search or replay a simulation, allowing researchers to conduct post emergency analyses.

**Real-Time 3-D Visualization**: to render 3D visualization images of the disaster event and the airport environment sceneries in real-time. This component provides an interactive environment to visually track the disaster propagation and crowd evacuation behavior. The 3-D scenes are generated at run-time as the models execute. This entity is already implemented and provided by the CD++ toolkit. The required interfacing mechanism to the visualization engine is also already available.

This proposed research is conducted with the collaboration of Daytona Beach International Airport and the existing facilities at Embry-Riddle Aeronautical University (ERAU). Personnel (Director of Public Safety, Airport Manager, and Security Officers) form the Daytona Beach Airport will provide data resources and consultation for defining airport environment, disaster models, and testing scenarios. Since the framework will be built on top of the existing CD++ modeling and simulation tool, a large portion of the underlying infrastructure is already available. The CD++ tool also allows web service-based parallel and distributed execution. This feature can be combined with the computing cluster available at the researchers' institution (ERAU's Zeus computing cluster - 256 Xeon nodes @ 3.2 GHz processor with Myrinet Interconnect) to allow for very large-scale emergency simulations.

## 4. UNDERLYING TECHNOLOGY: EXISTING AND NEW MODULES

Aiming for technology reuse and benefiting from open-source resources, the framework will integrate existing and new modeling/simulating modules. As discussed in the previous section, the core computing power of the framework is provided by the open-source CD++ M&S engine (Wainer 2002). Figure 2 provides an overview of how various modules will be incorporated into the proposed collaborative technology. As illustrated on Figure 2, there are two different building blocks of the framework: ***existing components***, and ***proposed components***.

The *existing components* are a set of publicly available and open-source technologies that are ready to be used. These include:

Model Templates: various cellular models implemented in CD++, based on Cellular DEVS formalism, are freely available from a model repository available on http://cell-devs.sce.carleton.ca. Among them, there are various Evacuation and Natural Disaster sample models that can be used for creating an emergency scenario by slightly modifying them and customizing them to reflect the user's needs. The proposed emergency framework will package such models along with newly created models matching airport emergency scenarios (such as airport specific areas and sceneries) and provide a well-established model repository to the user. The airport models generated for this research will also be made available to other researchers as open-source and free products;

Visualization Capabilities: Beside models, a DEVS-based modeling and simulation toolkit (created as an eclipse plug-in) is also available for reuse. The tool allows 3D rendering of the simulation results by sending the output to open-source visualization engines (such as Blender or VegaPrime) through an interface.

The *proposed components* will provide the unique features of the framework, which include:

Real-time Analysis and Assessment: this component provides "what-if" analysis to guide decision makers to take optimal paths in evacuating the crowd under emergencies. Given an emergency scenario, the component will run several simulations of the same scenario but with varied metrics (e.g. dictating difference evacuation methods such as adding more patrolling agents, evacuating through different routes, blocking a specific area or exit door, etc.). These various simulation instances will be executed simultaneously to provide an optimal solution. While such simulations are in execution, real-time data can be fed into them to reflect updated emergency data. Given the nature of emergencies, time is a critical factor and the framework thrives to make these simulations as efficient and precise as possible by incorporating fast simulation techniques. This is made possible through using cellular discrete-event simulation mechanism which allows to simulation thousands of independent agents in a matter of seconds. Cell-DEVS not only will allow to simulate each individual person on the scenery, but also allows simulating the environment of the airport and integrating the two concepts seamlessly into one single simulation scenario. At the same time, the mathematical backbone of Cell-DEVS provides a powerful calculation mechanism for analyzing risk and other sensitive statistics.

Data Feeder: As emergency is evolving, metrics change and new data arrive continuously (status of the building, number of people injured or blocked, new fire locations, arrival of first responders and patrolling staff, etc.). These data must be captured in real-time and fed into the "already-running" simulations to reflect realistic scenarios. The Data Feeder component is in charge of capturing and streaming these information to the simulation entity to update the scenarios as information becomes available. Not only real-time data can alter emergency responses, but historic data can also be used to predict situations. Thus, the data feeder, provides a storage mechanism as well for extracting of data from old emergencies.

New Models: Generic airport layouts will be generated and stored in the framework Model Template repository to speed up the scene generation process. Building areas such gats, terminals, baggage claim, parking, etc. will be created in Cell-DEVS allowing the user to reuse or customize them accordingly.

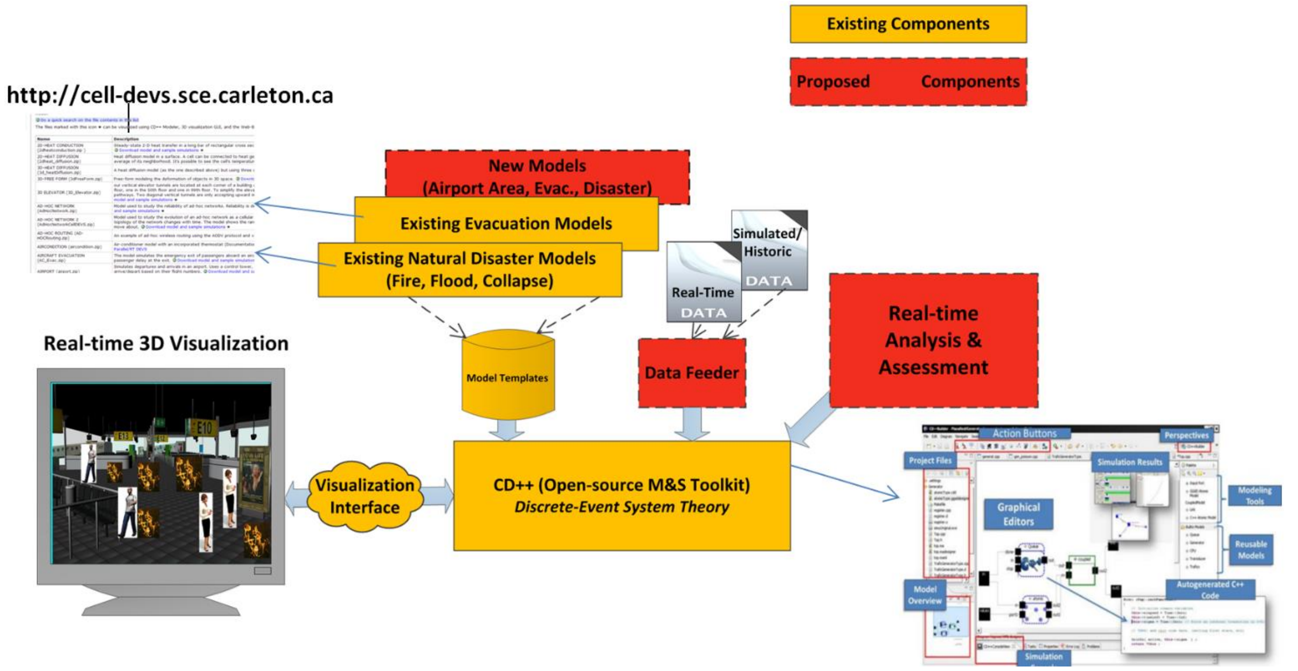


Figure 2. Framework Existing and New Modules

## 5. TECHNOLOGY OUTCOMES

The tool aims at providing a "plug-and-play" framework where various components can be easily integrated into the system. From the user's stand point, the framework is executed in the following steps:

1. Define Airport Area Model: User can select a pre-existing model from the templates repository or define a new one using the tool's development environment (the new model can be added to the templates repository for future use).
2. Define Disaster Event Type: User can select one or more disaster types from a Graphical User Interface (GUI) listing various disaster types.

The user will need to specify the location of the disaster and pin it on the Airport Area Model.

3. Define Crowd Behavior Model: User selects among various crowd emergency behavior. Similar to the above models, the template repository contains various options that can be selected by the user from a GUI window.
4. Select Data Source: User defines the data source (real-time data that will be injected into the simulation at run-time, historical data from data repository, or a mixture of these two).
5. Visualization Generation: Given the above required inputs, the simulation launches and real-time data are fed into the visualization engine. 3-D scenery of the airport emergency area, disaster progress, and crowd flow behavior are rendered and presented to the user in real-time.
6. Analysis and Assessment Evaluation: Statistical data (text and graphs) are presented to the user providing cost analysis of current scenario and alternative strategies. These data are dynamically logged for later investigation and analysis of emergency situations.
7. Run-time Strategy Switching: Based on the feedback obtained from the analysis, the user can dynamically switch to other alternatives at run-time. For instance, the user can select a different crowd evacuation strategy and observe the resulting behavior (the tool will provide a number of most efficient egress strategies that can be selected by the user and implemented on top of the Crowd Behavior Model as the simulation is in progress or prior to its start).

Advancing the tool by incorporating other subcomponents (e.g. Simulation of First Responders Activities) and intelligence is in the future milestone for the framework. Also, the tool can be potentially expanded to allow for modeling larger airport areas (i.e. ultimately the entire airport facility including all buildings and lots).

## 6. CELLULAR DISCRETE EVENT APPROACH

The selected modeling and simulation methodology for this work is the Cellular Discrete-Event System Specification (Cell-DEVS) formalism (Wainer 2009). A model defined by Cell-DEVS is a cellular grid where each cell represents an independent agent. Agents communicate and interact with each other throughout the simulation by sending/receiving messages. The behavior of each agent is expressed using a state-machine. Agents can be stationary or mobile meaning they can change their position within the grid. Each cell defines a surrounding neighborhood that affects its state value. Whenever a computation is performed and the cell's value is modified, all its neighboring cells get an update to re-evaluate their states accordingly. A cell's value changes as a result of a simple local computation based on the current state of the cell and its immediate neighbors as dictated by the Cell-DEVS specification. Figure 3 illustrates a typical Cell-DEVS grid and the neighboring definition. Each cell can choose to have as many direct or indirect neighbors.

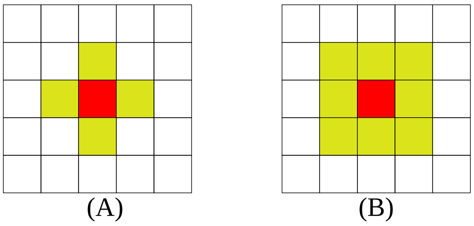


Figure 3: Cell Neighborhoods: (A) Von Neumann, (B) Moore.

A Cell-DEVS model is defined by:

$$TDC = < X, Y, I, S, \theta, N, d, \delta_{int}, \delta_{ext}, \tau, \lambda, D >,$$

where $X$ is a set of external input events; $Y$ is a set of external output events; $I$ represents the model's modular interface; $S$ is the set of sequential states for the cell; $\theta$ is the cell state definition; $N$ is the set of states for the input events; $d$ is the delay for the cell; $\delta_{int}$ is the internal transition function; $\delta_{ext}$ is the external transition function; $\tau$ is the local computation function; $\lambda$ is the output function; and $D$ is the state's duration function. The modular interface ($I$) represents the input/output ports of the cell and their connection to the neighbor cell. Communications among cells are performed through these ports. The values inserted through input ports are used to compute the future state of the cell by evaluating the local computation function $\tau$. Once $\tau$ is computed, if the result is different from the current cell's state, this new state value must be sent out to all neighboring cells informing the state change. Otherwise, the cell remains in its current state and therefore no output will be propagated to other cells. This will happen when the time given by the delay function expires. Finally, the internal, external transition functions and output functions ($\lambda$) define this behavior. Cell-DEVS improves execution performance of cellular models by using a discrete-event approach. It also enhances the cell's timing definition by making it more expressive.

CD++ (Wainer 2002) is an open-source object-oriented modeling and simulation environment that implements Cell-DEVS theories in C++. The tool provides a specification language that defines the model's coupling, the initial values, the external events, and the local transition rules for Cell-DEVS models. CD++ also includes an interpreter for Cell-DEVS models. The language is based on the formal specifications of Cell-DEVS. The model specification includes the definition of the size and dimension of the cell space, the shape of the neighborhood and the border. The cell's local computing function is defined using a set of rules with the form *postcondition delay {precondition}*. These indicate that when the *precondition* is met, the state of the cell changes to the designated *postcondition* after the duration specified by *delay*. If the precondition is not met, then the next rule is

Proceedings of the International Defense and Homeland Security Simulation Workshop 2014,
ISBN 978-88-97999-41-6; Bruzzone, Frydman, Longo, Sokolowski and Sottilare Eds.

evaluated until a rule is satisfied or there are no more rules. CD++ also provides a visualization tool, called *CD++ Modeler*, which takes the result of the Cell-DEVS simulation as input and generates a 2-D representation of the cell space evolution over the simulation time. This feature of the tool provides an interactive environment allowing for visual tracking of the mode's evolution.

## 7. CELL-DEVS MODEL OF AIRPORT

To demonstrate a cellular representation of an airport area, here we present a Cell-DEVS model depicting check-in, lobby, and baggage claim area of the Daytona Beach International Airport (on Figure 4 and 5).

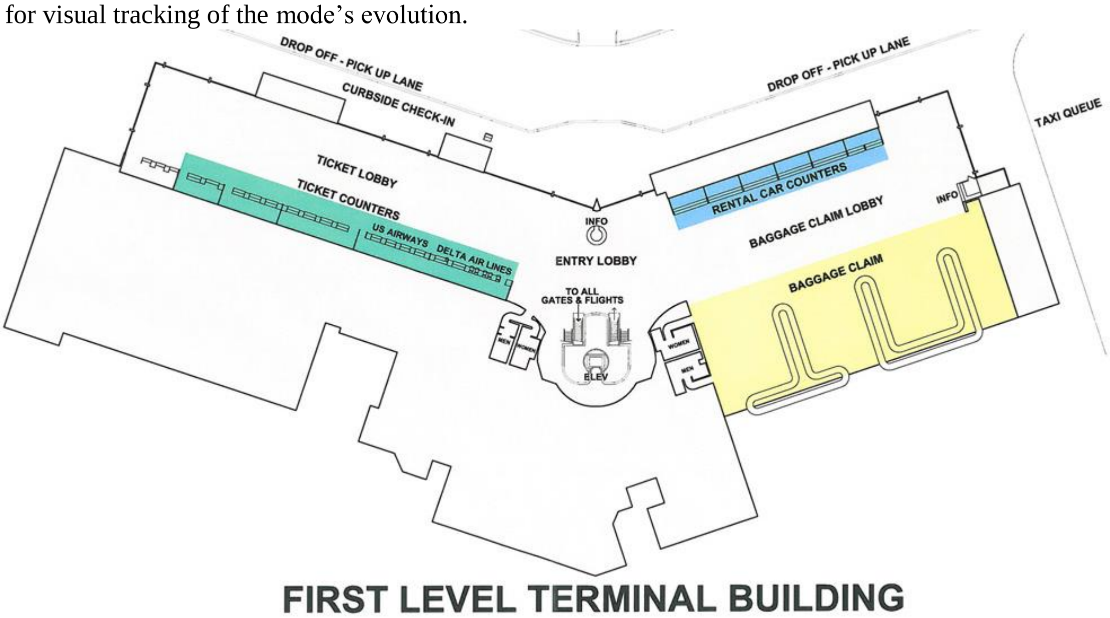


Figure 4: Daytona Beach Airport, First level terminal

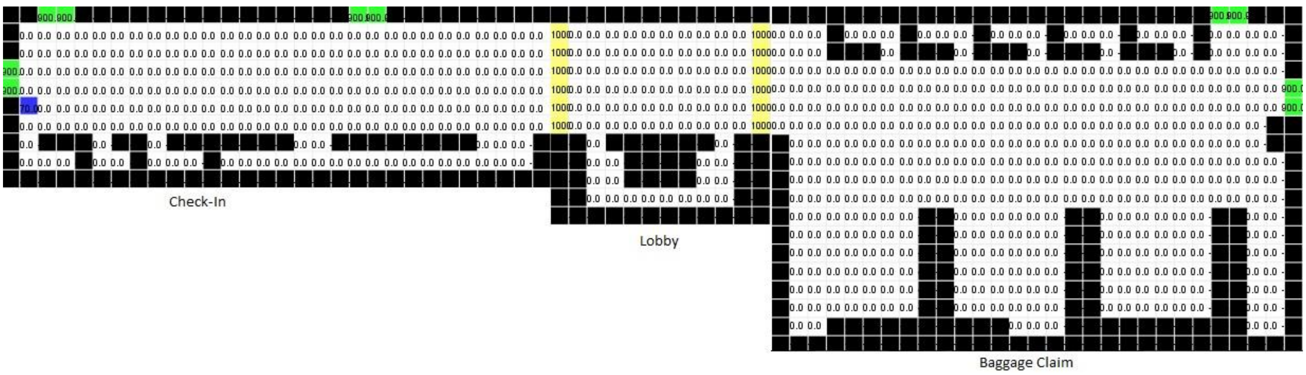


Figure 5: Cellular Representation of First Level Terminal Layout

When comparing Figure 4 and Figure 5 it can be seen that there are a number of differences between the two images as a result of design decisions. The first design decision was to remove the angular aspect of the terminal design by stitching together three separate cellular designs of the terminal. The Check-In, Lobby and Baggage Claim each represent separate components within the top-level Cell-DEVS model, which have been coupled together at the seams represented by yellow colored cells. The second design decision was to remove the areas that are not accessible by the general public. The third design decision was to get the cellular representation as close to scale as possible, and only modify the design when reaching the limits of the Cell-DEVS visualizer. Note that green cells represent exits, black cells are walls/obstacles, blue cells are crowd, while white cells represent vacant space.

Using the airport layout model, various egress strategies can be applied to study and analyze two important emergency metrics: speed of evacuation, number of evacuated persons. Using the same cellular structure, multiple airport egress models (e.g. random movement, directional movement, guided evacuation, follow the herd, etc.) illustrating crowd evacuation behavior are studied. Given the scope of this article, here we will only present one evacuation strategy and demonstrate the preliminary results.

The simulation was conducted by placing crowd on various locations on the cellular grid. Cell's rules was written in such a way that crowd would move in a random behavior not knowing the exists (simulating panic scenario where crowd is panicked and can not determine where the nearest exit door is located). Each cell representing crowd was modeled to have access to twelve surrounding neighbors (to determine direction and avoid collapsing with walls/ moving people, etc.). The model executed based on discrete-event steps (arrival of events determined the simulation time advancement). The airport model of Figure 5 included total of 995 cells (Check-in: 10x30 cells, Lobby: 12x12 cells, Baggage: 19x29) was modeled where 80 people were placed at different locations on the three areas of Lobby, Check-in, and Baggage. The simulation was executed till the entire crowd was evacuated, which took almost 2600 seconds (201 minutes). Given the nature of random movement, it was expected to take the longest time compared to guided/directional crowd evacuation. The detailed simulation scenarios and other statistical metrics will be published in near future. Figure 6 presents three screenshots of the airport random evacuation simulation at *initial*, *half-way*, and *near final* stages of the execution.

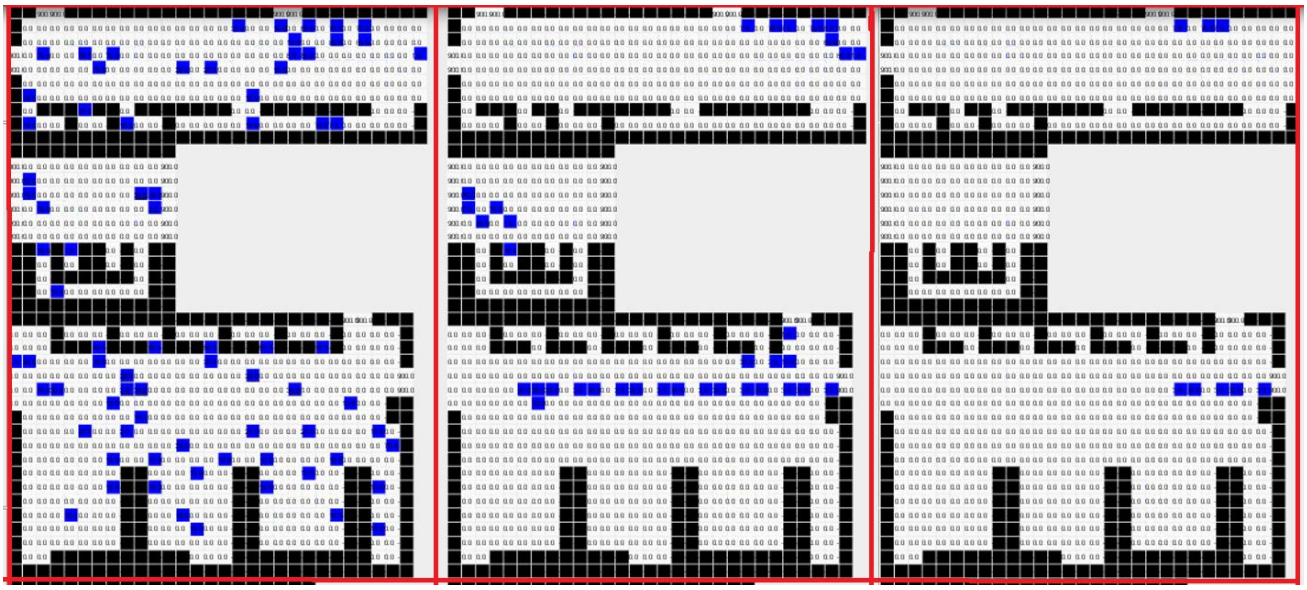

Figure 6: Airport Random Evacuation at Initial, Half-Way, and Near Final Stages.

## 8. SAMPLE APPLICATION OF CELL-DEVS TO EMERGENCY MANAGEMETN – A SUCCESS STORY

DEVS and Cell-DEVS has been successfully used in the past for emergency management (Moallemi and et. al. 2011). The initial idea of the proposed framework was actually driven by that work. For demonstration purposes, a simple emergency scenario depicted from (Moallemi and et. al. 2011) will be presented here to show the use of Cell-DEVS and 3D visualization in managing fire emergency.

The work illustrates a real-time robotic firefighter that is place on the field and is updated about fire locations. The first responder officer sitting in his fire department office is getting real-time and simulated (expected) fire locations and reports them to the robotic firefighter. The fire simulation (Cell-DEVS model) running on the first responder's computer is warning about the speed and direction of the fire and predicts the fire progress using lively-fed data (from the firefighter acting on the scene) and historic/simulated fire data. The first responder officer is also viewing 3-D live scenery of the fire location and the activities of his firefighters. The robotic firefighter periodically reports back his fire extinguishing results, updating the officer's visualization scenery. This collaborative work provides highly precise and dynamic information of the fire progress, allowing for much faster emergency management and an optimal supervisory control experience to first responders. Figure 7 is a snapshot of the 3-D real-time scenery, illustrating a robotic firefighter approaching fire location, while the first response officer is watching his activities and the simulated fire progress.

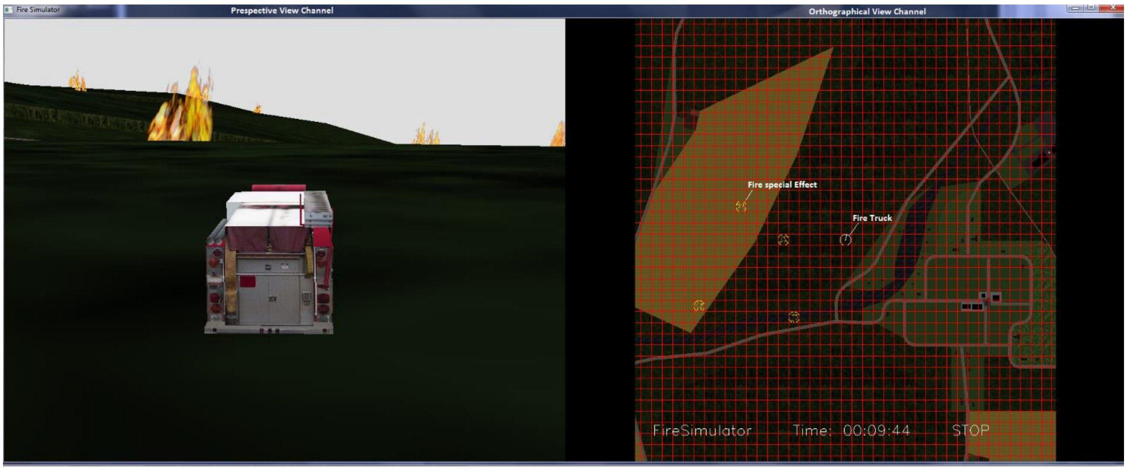


Figure 7: Cell-DEVS 3-D Modeling of Fire Emergency Management (Moallemi and et. al. 2011)

## 9. CONCLUSION

This work attempts to propose a collaborative modeling, simulation, and visualization architecture to enhance airport emergency preparation and management. A plug-and-play framework was presented that integrates various airport emergency components. Such adaptive and scalable system not only could be used for training and supervisory control, but it could also play a great deal at real-time disaster management. Although the work is still under development, but successful projects have been implemented in the literature that use modeling, simulation, and visualization for disaster engineering. The framework discussed in this work allows first responders to define the airport area model (user fine-tuned), the disaster type (natural and manmade), and crowd behavior (various egress strategies) and simulate the entire disaster scenario in virtual or real-time. Constructed on top of an open-source M&S environment (CD++), the proposed framework supports live 3D visualization, what-if analysis, as well as run-time strategy switching. The work presented here outline the research goals and the software architecture, while the core implementation activities are still under development. The outcome of this research will be a tool that can be easily used by non-experts (airport security officers and first responders) to enhance emergency exercises and complement table-top and field-based trainings.

## REFERENCES

Moallemi, M., Jafer S., Ahmed, A., Wainer, G. 2011. Interfacing DEVS and Visualization Models for Emergency Management. *Proceedings of SpringSim'11*. Boston, USA.

Zeigler, B., Praehofer, H., Kim, T. 2000. Theory of Modeling and Simulation. *Academic Press*.

Wainer, G. 2009. Discrete-event modeling and simulation; a practitioner's approach. *Taylor & Francis*.

Tsai, J., et. al. 2011. ESCAPES: evacuation simulation with children, authorities, parents, emotions, and social comparison. *International Conference on Autonomous Agents and Multi-agent Systems*, May 02-06, 2011, Taipei, Taiwan.

RescueSim. http://www.rescuesim.com. *Accessed: July 2014.*

paxport. http://www.paxport.net. *Accessed: July 2014.*

C3Pathways. http://c3pathways.com. *Accessed: July 2014.*

CAST. http://www.airport-consultants.com. *Accessed: July 2014.*

Kuligowski, E. D., Peacock, R. D. "A Review of Building Evacuation Models". National Institute of Standards and Technology Technical Note 1471, 2005.

Wainer, G. 2002. CD++: A Toolkit to Develop DEVS Models. *Software – Practice and Experience, 32(13), pp. 1261- 1306*.

## AUTHORS BIOGRAPHY

**Dr. Shafagh Jafer** is an Assistant Professor of Software Engineering at Embry-Riddle Aeronautical University, FL. She received her Ph.D. Degree in Computer Engineering from Carleton University, Ottawa, Canada in 2011 where she was awarded the *Senate Medal for Outstanding Academic Achievement* for her doctoral thesis on large-scale parallel simulation. For the past eight years she has been conducting research on discrete-event modeling and simulation, distributed and parallel simulation, and integrated modeling environments. Dr. Jafer has been previously involved in projects dealing with modeling and simulation of natural disasters as well as emergency response to natural fire. She is currently conducting research in disaster engineering, modelling and simulation in aviation, and large-scale NAS (National Airspace System) data analysis. Dr. Jafer has served as committee member and organizer of the Spring Simulation Multi-conference.