Proceedings of the 2006 Winter Simulation Conference
L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, eds.

CROWD SIMULATION FOR EMERGENCY RESPONSE USING BDI AGENT BASED ON VIRTUAL REALITY

Ameya Shendarkar

Dept. of Computer Science
University of Arizona
Tucson, AZ 85719, U.S.A.

Karthik Vasudevan
Seungho Lee
Young-Jun Son

Dept. of Systems and Industrial Engineering
University of Arizona
Tucson, AZ 85719, U.S.A.

ABSTRACT

This paper presents a novel VR (Virtual Reality) trained BDI (belief, desire, intention) software agent used to construct crowd simulations for emergency response. The BDI framework allows modeling of human behavior with a high degree of fidelity. The proposed simulation has been developed using AnyLogic software to mimic crowd evacuation from an area under a terrorist bomb attack. The attributes that govern the BDI characteristics of the agent are studied by conducting human in the loop experiments in VR using the CAVE (Cave Automatic Virtual Environment). To enhance generality and interoperability of the proposed crowd simulation modeling scheme, input data models have been developed to define environment attributes. Experiments are also conducted to demonstrate the effect of various parameters on key performance indicators such as crowd evacuation rate and densities.

1 INTRODUCTION

Management of emergency response both for manmade and natural incidents has become a key research field in today's world after the September 11th attack. The scenario considered in this paper is the management of crowd evacuation under a truck bomb attack in an open public area. Effective incident management presents a number of challenges to the responsible agencies (Jain and McLean 2006). The effective crowd management requires accurate prediction of impact of such incidents on the crowd as well as the environment (which will affect the crowd behavior). The involvement of human lives demands high accuracy of prediction. For these purposes, simulation is an ideal technique. An accurate simulation model would allow the responsible agencies to evaluate different evacuation and damage control policies beforehand, which will allow the execution of the most effective crowd evacuation scheme during the actual terror scenario. Also, it can be used for helping train response personnel, and for vulnerability analysis.. Use of such techniques will allow training of responders and emergency managers at a fraction of cost of the live training exercises (Jain and McLean 2006).

All large cities face the problem of managing the flow of people trying to accomplish their own objectives (Page et al. 1999). Obtaining data to model a crowd is a challenge in itself. The data so obtained can be so diverse that it can never be organized. Crowd simulations involve the simulation of an environment and the people in the crowd. Most crowd simulations model the crowd as groups of people with common characteristics or objective. We attempt to an agent with individual characteristics to create a crowd.

Constructing an accurate crowd simulation is a challenging task. If crowd simulations are constructed based on the traditional discrete event simulation paradigm, where entities are passively driven by processes or event, their validity may be questionable. To enhance the level of detail in the crowd simulation, we propose to build a simulation model with the intelligent agents paradigm, where active, detailed agents mimic humans in the crowd. An intelligent agent has its own characteristics such as autonomy, social ability, reactivity, pro-activeness, cooperativeness, learning ability, and adaptivity (Laughery 1998). These features allow modeling of realistic human behavior, and therefore, a crowd. In this paper, we employ extended BDI (belief, desire, intention) agent framework which is explained in the next paragraph.

The realistic representation of human behavior in crowd simulations require rich models or architecture that is able to represent partly, the complex human behavior. Modeling complicated human behavior involves a certain degree of abstraction backed up by a rich architecture. This architecture of a human's mental functions is very well supported by BDI(Belief-Desire-Intention) architecture. The behavior of each person in the crowd is modeled using the BDI architecture extended in our research work and a well defined data model. To this end, the intention module in the traditional BDI framework is expanded to include 1) deliberator, 2) planner (reasoning processor), and 3) decision-executor sub-modules. Furthermore, a confidence state is also considered, which affects as well as is affected by three other mental modules (BDI). But it must also be noted that there is an obvious trade-off between the improved accuracy obtained by introduction of BDI framework and computational speed. We plan to tackle this issue as a part of future research work.

The data required to drive the development of the BDI agent is extracted from human in the loop experiments in a CAVE environment. These attributes that are determined are fitted to a distribution that gives each agent unique and near realistic characteristics. Virtual reality is used as a tool to simulate dangerous environments and scenarios which cannot be simulated otherwise in reality. Hence human in the loop experiments of potentially dangerous situations can be conducted with very minimal or no risk.

In this paper, scenario of a bombing near the National mall area in Washington DC is considered. We attempt to evaluate various scenarios of first responder deployment and the rate of crowd evacuation in specific sections of the area. We also estimate the number of casualties and injuries. We employ a commercial agent-based simulation software (AnyLogic), which integrates the power of the discrete event simulation with the versatility of the agent based modeling. To enhance generality and interoperability of the proposed crowd simulation modeling scheme, characteristics of each agent is defined by a data model with attributes that mimic the human. Each agent in the model then gets unique characteristics based on the attributes assigned to it.

The rest of the paper is divided as follows. Section 2 explains various modules of the extended BDI architecture in brief. Section 3 illustrates the required data extraction using virtual reality. Section 4 describes the complete process of development of simulation. Section 5 contains explains different computational aspects of developing this crowd simulation. Section 6 contains experimental results and the final section lists the conclusions drawn.

2 EXTENDED BDI FRAMEWORK

As mentioned before we propose to use extended BDI framework to model human behavior in the crowd simulation. Section 2.1 gives an introduction to the BDI framework and section 2.2 explains how we have extended the BDI framework to model human behavior

2.1 Background Information on BDI Framework

Modeling of human behavior for a general situation is extremely difficult, if not impossible. One of the commonly used ways has been designing an intelligent agent that will mimic the overall (abstract) characteristics of a human. Such intelligent agents operate within an environment and act autonomously in that environment (Fikes and Nilsson 1971). Such agents are autonomous, cooperative, learnable, adaptable and equipped with social ability, reactivity, pro-activeness (Laughery 1998). Autonomy in this case means that the agent operates without direct, continuous supervision. Among several techniques available for modeling such human characteristics, the best-known and most mature type of intelligent agent is the belief-desire-intention (BDI) modeling. The ensuing text explains in brief the BDI architecture and enhancements being developed at the University of Arizona in brief. The detailed explanation of the extended BDI architecture is well beyond the scope of this paper.

A BDI agent is characterized by its "mental state" with three major components: *beliefs, desires, and intentions* (Rao and Georgeff 1998). Beliefs correspond to information the agent has about the world. It may be incomplete or incorrect. Desires represent state of affairs that the agent would wish to be brought about. Intentions represent desires that the agent has committed to achieve. Following are some of the advantages offered by the BDI framework.

- First, the BDI paradigm is based on folk psychology, where the core concepts of the paradigm map easily to the language people use to describe their reasoning and actions in everyday life (Norling, 2004).
- The BDI paradigm is a relatively mature framework and has been successfully used in a number of medium to large scale software systems.
- The development of such agents is well supported by simulation software packages like AnyLogic, Jack.

2.2 Extended BDI Architecture

The framework described in figure 1 consists of the following key components:

- *Perceptual processor* observes the environment and tries to interpret the data coming from the sensors/external environment. The information gathered by the perceptual processor is then converted into beliefs. The choice of the technique for transformation can be made as per the real life scenario under consideration. Based on the current beliefs and initial intentions, the agent decides what state of affairs to achieve (*desires*) via the cognitive processor.

  The current state (explosion occurred or not) affects the agent in the crowd simulation setting its desire to pick either a goal destination such as an office, mall etc. (during normalcy) or an exit (during evacuation).

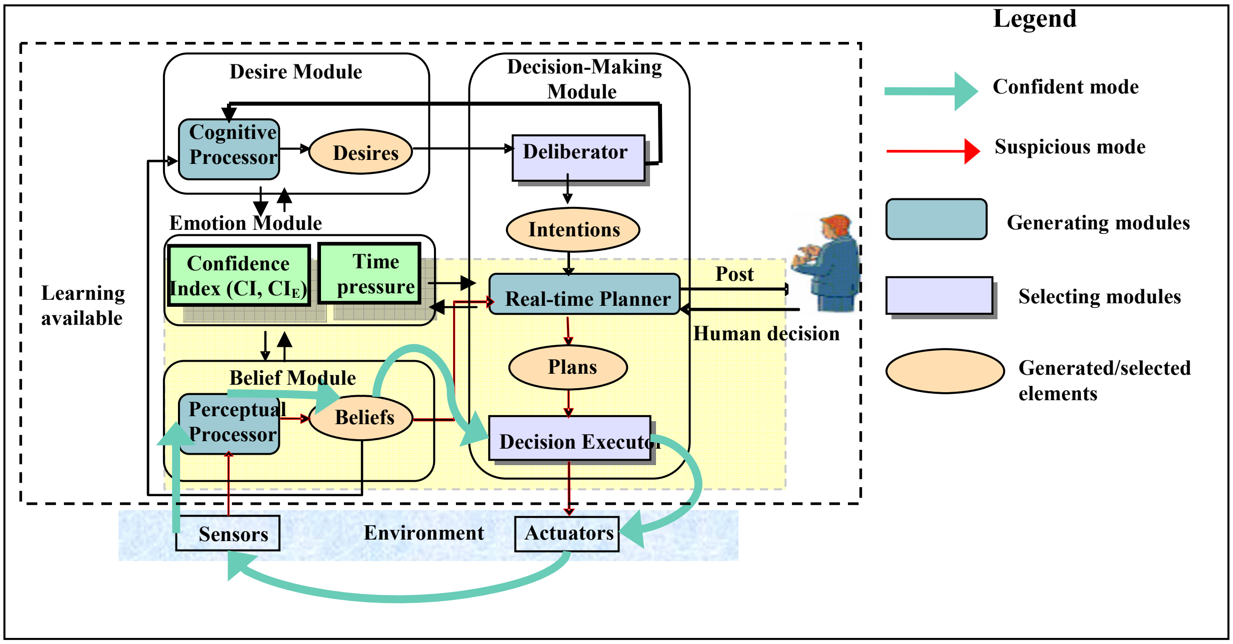

Figure 1 Extended BDI Architecture

- *Deliberator* filters desires to select one intention.

  During evacuation the deliberator can choose an intention such as reaching a policeman, one of the exits, the metro station or any intermediate intersection for finding help (policeman).
- *Real time planner* (RTP) generates different plausible plans to achieve the selected intention based on the current beliefs. A plan is defined as a sequence of ordered actions. The generation of plans is heavily influenced by the current set of intentions. It should also be noticed that there can be more than one plan generated for the given case.

  The real time planner uses the knowledge of area (KOA) rating of the agent to provide the shortest path to an exit (KOA =1), or a number of possible paths to intermediate intersection (KOA<1).
- *Decision executor* selects one of several plans generated by the real time planner. The policy to select the plan varies from case to case. In some cases an optimal plan will be selected. On the contrary, in some cases any satisfactory plan will be selected by the decision executor.

  The data from the VR experiments is used in the decision executer to pick one of the paths (generated by the RTP) to an intermediate intersection for the agent to move into, in the agent's quest to reach an exit.
- *Confidence index* denotes the agent's optimism about achieving it's intentions. For example, if the agent performs some major actions or plans success-fully, it will be in the *confident* mode. Otherwise, it will be disappointed and will be in the *suspicious* mode. Under the confident mode the agent continues to execute the current plan. Otherwise, in the suspicious mode, the agent tends to be cautious and will reconsider it's intentions and reevaluate plans via the real-time planner before performing any actions.

The simulated human model(BDI) continually executes this cycle of observing the environment, deciding which intention(s) to achieve next, determining plan(s) to achieve these intention(s), and then finally executing this plan. In case the simulated human model comes across a situation that cannot be resolved, the human in the loop can be consulted in order to determine how to proceed next.

The confidence index in this crowd simulation is represented as an aggregation of various attributes such as age, leadership, knowledge of area, velocity, panic state, injury level etc. The higher the confidence index the better the chance for the agent to evacuate earlier.

3 DATA EXTRACTION USING VIRTUAL REALITY

Virtual reality is a powerful tool that can be used to mimic real world scenarios that are either too expensive or too difficult to conduct in live exercises. Virtual reality immersive environments allow us to conduct human in the loop experiments. The data extracted from these experiments are significantly more accurate to build agents that mimic the action of humans in real-world scenarios than experiments involving only PCs. These agents then drive the agent based simulations to test the consequence of various contingency plans.

The current application calls for building a model of a section of the Washington mall area. The model is used to determine the trend of human decisions with respect to the choice of exit routes. This data collected is used to populate the BDI agent data model.

The VR model is built using CAVELib, a library of functions built for the CAVE system. The programming is done in Visual C++. We use OpenGL Performer libraries to create the required graphic detail. The hardware system is a CAVE simulator manufactured by FakeSpace Inc. We use stereoscopic glasses to create an immersive effect supported by a 6 speaker sound system. A wand (joystick) is used for navigation through the VR space. More explanation of the VR experiments is provided in the following and Experiments sections.

4 SIMULATION DEVELOPMENT OVERIVIEW

This section consolidates the work mentioned in the sections above and presents the sequence of the various stages in developing this crowd simulation (Figure 3). The first step in building the agent based simulation model is to collect the data on which the agent is modeled. The VR CAVE is used to setup an immersive environment to conduct human in the loop experiments. The agent data model constructed from this data is then used to build the state charts to define agent (human) thought process.

Once one agent is built, they are replicated to create multiple agents (crowd) as demanded by the scenarios. These agents are then placed with their initial intents into the environment as defined by the input files. Anylogic is used to implement the agents and in turn the crowd. The agents then derive their beliefs from this environment. This crowd simulation model with intelligent agents can be used to run multiple replications as per design of experiments to obtain the required outputs and statistics. The model is verified by completing several successful runs. Validation is done by comparing crowd evacuation time from the model to that obtained through calculations.

5 CROWD SIMULATION MODEL

5.1 Overview of Crowd Simulation

In this model, an evacuation of the crowd at the National Mall area in city of Washington DC is considered. The basic functional characteristics of this simulation are stated below. One of the most likely area of explosion/terrorist-attack is considered to be near the Washington mall area circled in figure 2. Before the explosion occurs, people in different areas of the city are discretely distributed based on the type of goal (business, shopping etc.). After the explosion, people try to exit as quickly as possible from the exit points available.

Depending on the Knowledge of Area characteristics,

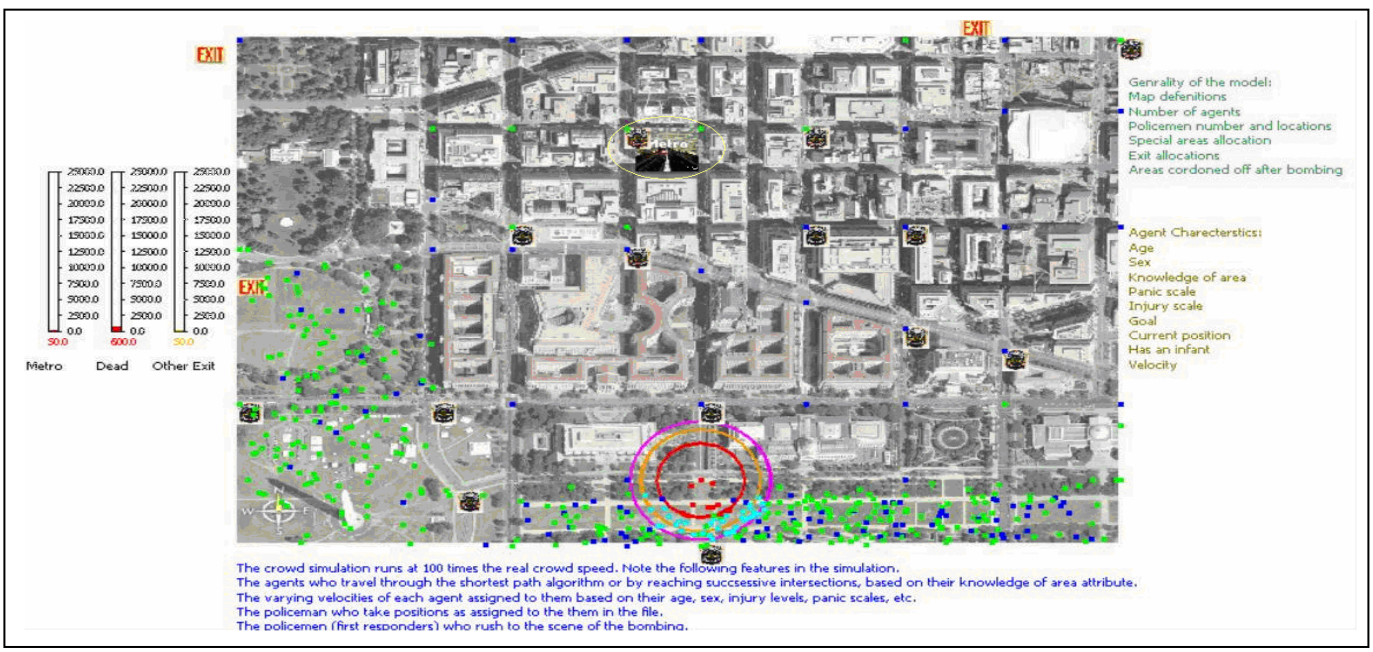

Figure 2: Extended BDI Architecture

some people become group leaders and follow the shortest path to the exit. Others move from intersection to intersection and may encounter a policeman who guides them to nearest exit. One example of an exit point is a Metro station which is located beyond the radius of the intensity of the explosion. After the explosion, some policemen (first responders) approach the area around of explosion and guide the people in the area to the closest exit point possible to hasten the process of evacuation.

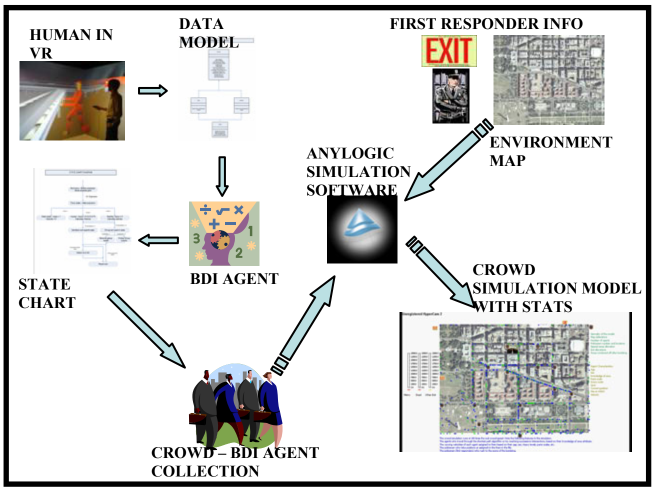

Figure 3: Different Components of the Simulation

As seen in figure 2 and later in detail in figure 5, the map of Washington DC is divided into a network of horizontal and vertical roads. It is important to note that when people move along the roads in the network, they do not violate the boundary of roads. Apart from a grid of intersection of such roads, the map also consists of some recreational areas like a garden and mall areas present in the lower left and bottom part of the map in figure 2. In such areas people can wander without restrictions of road boundaries. Such areas can also have exit points from where people can safely exit. After the explosion, the areas near the explosion are cordoned off and people are not allowed to move through this area.

5.2 Agent Characteristics

The agent in the model is given a number of different attributes to give each and every one an unique character set. Attributes such as age, sex, knowledge of area, panic scale, leadership, independence, injury scale, current positions, etc. The values are set on a relative basis. For e.g. Injury scale of 1 refers to dead, 0.75 to severe injury, 0.5 to minor injury and 0 to healthy. The class diagram used to represent the data model formally is omitted from here because of lack of space.

5.3 Behavior of an Agent using State-charts

State charts are the critical part of agent based modeling. State chart diagram is a generic way of representing the behavior of an agent in response to the external events(beliefs like explosion) or internal events (like achieving an intention). When an agent reaches the end state it could be said to have achieved it's intentions.

The Figure 4 represents the various states that each agent (person) goes through in our crowd simulation. The arrows in the diagram represent the transition from one state to another (triggered by an internal or external event).

5.4 Generic and Interoperable Nature of the model

The major focus while designing this simulation model was to keep it generic to such an extent that if the map of Washington DC was replaced by any other map then the changes to the model should be minimal (like new intersection and path information). Hence all the environment characteristics were represented in a declarative nature using simple text files to describe input that drives the model. The following files were used to provide the required input data.

- **Intersection file:** File describing the map of Washington DC as a weighted graph of interconnected network of roads.
- **Recreational area file:** File describing the recreational areas that allow random movement instead of a predefined path(in case of roads). Such areas are approximated to rectangles.
- **Environment information file:** File describing the various parameters that characterize the given scenario. e.g. Density of people in various areas of the map.
- **Exit information file:** File containing information about the exit points for the area under scrutiny of DC map.
- **Police information file:** File containing information about policemen such as their initial positions, total number of policemen etc.

The output produced by this simulation may be used by other simulations. Hence the results of the simulation can be dumped into a simple text file with format agreed upon previously or if required, some well known formats like XML. Use of such files facilitates frictionless communication between different simulations and thus increasing interoperability.

5.5 Major Data Structures

In order to enable the movement of the population, the map of DC is defined in terms of roads and recreational areas.

Looking at it from a different perspective, the network of roads is nothing but a weighted graph with the vertices of the graph being intersections where two roads meet and the distance between two intersections as the weight of the edge. The adjacency matrix data structure is used or representing a weighted a graph. Hence the network of roads in the Washington DC map is defined in terms of an adjacency matrix. e.g. the figures 5 and 6 explain representation of a part of map in terms of an adjacency matrix.

Other areas such as recreational areas are represented as approximated rectangles. In this simulation, the movement of the population in such areas does not follow any well defined path, thus eliminating the need data structures like adjacency list. The major data structures used to represent most of the input data are maps and vectors. A map data structure simply maps keys to values. While the Vector data structure is used to represent a dynamically extensible array of values. The implementation of these data structures is provided by the built-in java package java.util.

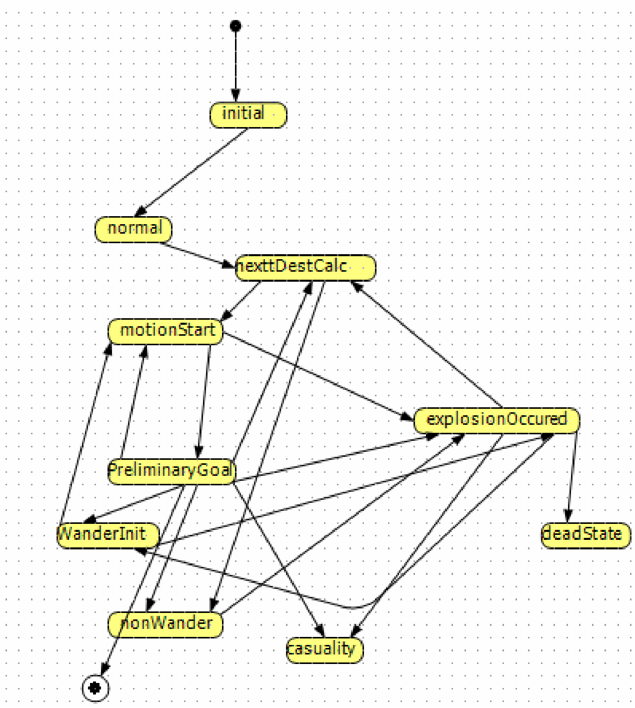

Figure 4: State Chart Diagram of an Agent

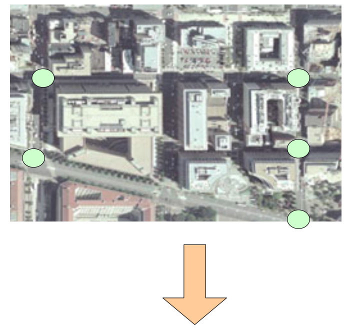

Figure 5: Part of Washington DC

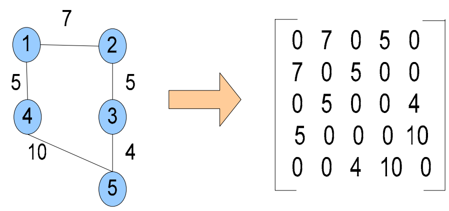

Figure 6: Adjacency Matrix Representation of the Sample Map

5.6 Major Algorithms used in Crowd Simulation

5.6.1 Shortest path algorithm

As explained in the section 5.1, some people find the shortest path to the exit point they want to reach. Dijkstra's algorithm is used to calculate the shortest path. Dijkstra's algorithm is one of the most commonly used type of algorithms because of it's simplicity and quadratic computational cost. The time complexity of Dijkstra's algorithm used in our case is $O(|V|^2)$ where |V| is the number of vertices in the graph.

5.6.2 Algorithm followed by every agent

Taking into account all the factors mentioned so far, here are the various steps that an agent goes through during it's lifetime in the simulation.

- Read in the required input files and create in-memory data structures.
- Instantiate an object of Agent class.
- Agents have random point of entry (current intersection), a goal intersection, crowd density values for each area on the map is the one obtained from VR experiments.
- As per the current beliefs and desire, each agent tries to achieve the intention(destination intersection here) either in suspicious or confident mode.

- At a random time, explosion occurs
- Panic values are assigned to agents depending on their proximity to the location and their characteristics.
- Each agent, after understanding the current situation, reassesses beliefs and changes goals in order to adapt to the current situation.
- Calculate the leader of the group based on the leadership ability.
- If the agent is part of the group and there is already a leader leading the group, then just follow the group. If the agent itself is the leader then calculate the shortest path using Dijkstra's shortest path algorithm to the nearest exit point.
- Calculate the velocity and start moving towards the exit.
- If agent is not part of the group and KOA = 1 then call shortestPath() to find the shortest path to the exit point. Otherwise move through the intersection till an exit point is reached or policeman informs about the path to the exit point.
- End of path for every agent when the agent reaches an exit point, including metro.

5.6.3 Software Tools

The crowd simulation for the emergency response scenario was developed using AnyLogic version 5.4. This choice was made because of the following features of AnyLogic:

- AnyLogic has a built-in library called AgentBase that facilitates the development of agent based models.
- AnyLogic also allows the developer to enhance the existing models by allowing Java extensions. In our opinion, AnyLogic provides a good balance between simplicity(by providing built in libraries) and flexibility(by Java based extensions).
- AnyLogic supports state-chart based modeling which facilitates the modeling human behavior.

6 EXPERIMENTATION AND RESULTS

The experiments are conducted in two phases. Phase 1 consists of data extraction experiments conducted in VR. These experiments are setup and run as follows. For the purpose of this paper, pilot experiments were conducted to obtain sample human response data.

- An intersection model of a section of the map is developed in VR. The immersive environment is essential to run human in the loop experiments to extract information about instantaneous human behavior and response to the virtual emergency.
- The paths leading to and from intersection are built to mimic the real world emergency situation (with fire, crowd, exit etc.). The paths features are then assigned weights based on VR results.
- Experiments are conducted immersing subjects in the scenario 15 times each with randomized path graphics.
- The aggregate data for number of times each path is chosen is recorded in the table below.

Table 1a: Weight Assignments

	Weight(15)		Weight(15)
Danger paths	6	Short paths	4
Exit paths	4	Medium paths	9
Police paths	3	Long paths	2
Crowded paths	2		

Table 1b: Velocity Validation using VR

	Avg Velocity
Human1	5.08 m/s
Human2	4.12 m/s
Human3	4.36 m/s

The flowing conclusions are reached through VR:

- Weights for each of the path features (Table 1a). Figure 7 shows a sample junction and its probabilities. The diagram shows the characteristics of each road from a junction. For example, a strong human could choose a short yet dangerous path if he is confident he can get through. On the other hand, an old man would choose a path which is less crowded and has less danger even though the exit may not be in sight.
- Emergency evacuation velocities are validated and formula for it is set accordingly (Table 1b).

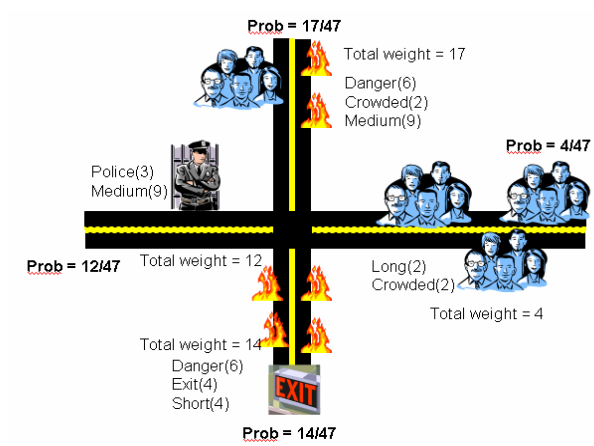

Figure 7: Junction Probabilities as Obtained from VR

Phase 2 consists of the experiments with the simulation model to obtain necessary results and conclusions from the model. In this section we demonstrate the working of the crowd simulation and determine the effect of the following factors on a) number of casualties, b) crowd evacuation time and c) the number of people exiting through the metro station. Following factors can be used to conduct experiments to see their effect on the aforementioned parameters:

1. The number of people in the area (population),
2. The number of exits,
3. The number of policeman in the area,
4. The intensity of the explosion, and
5. The distribution of policemen.

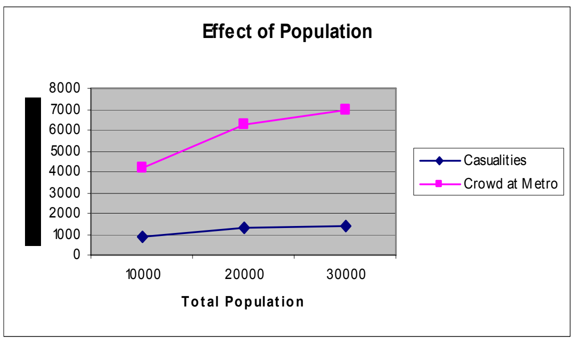

Figure 8: Effect of Population

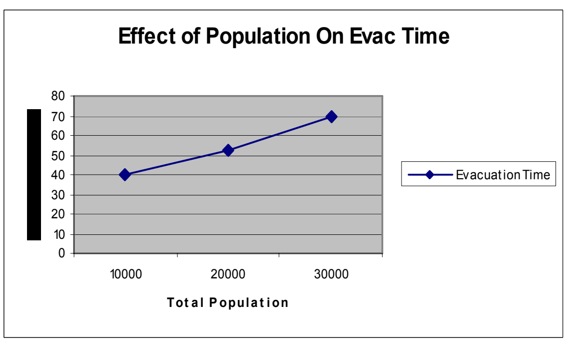

Figure 9: Population vs Evacuation Time

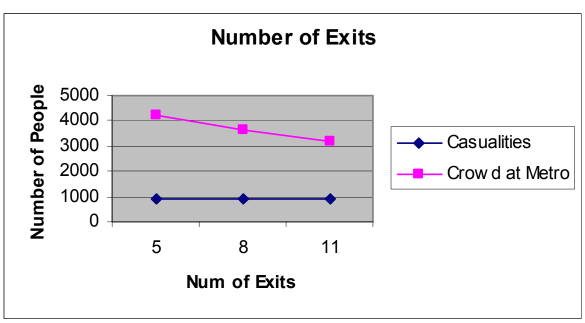

Figure 10: Effect of Number of Exits

We demonstrate here the results of the simulation for 3 scenarios of factor 1 and factor 2 to determine their effect on the crowd behavior. The base model is setup with 10000 people, 10 policemen, Low explosion intensity, 5 exits and with policemen deployed in Config1.

The experiment results for the first two factors shown in Figures 8, 9, and 10.

7 CONCLUSION AND FUTURE WORK

The BDI agents driven by the data extracted from VR experiments are used to build a crowd simulation to study evacuation in event of a terror attack at the National Mall area in Washington D.C. The base model is run to verify and validate the model. Further experiments are conducted to explore evacuation scenarios.

The simulation results help us to establish the direct relation between population of the crowd and casualties. It also provides us with an estimate of the number of people entering the metro station for every scenario considered. The best exit routes and congestion areas are also extracted from the model. The intersections to the bottom left of the explosion in figure 2 are found to be most congested. As expected number of policemen and number of exits do not affect number of casualties. However, further research must be undertaken involving transportation of injured people to the hospitals, where the role of policemen may affect the number of casualties, The number of policemen in the area is found to have no major relation to the number of people exiting through the Metro, but it is found to reduce the evacuation time significantly. Many other conclusions can be drawn from this BDI-agent based crowd simulation model. The versatility of the model makes it suitable for extension to other cities, scenarios and situations. The extended BDI agent can be used to mimic human behavior in a variety of other applications such as agents for driverless cars, predicting sporting outcomes etc.

Our future work involves addressing computational issues involving more accurate models that contain usage of complex architectures like BDI, further increasing the scope of application of VR to consider other factors and also scenarios of multiple bombings and detailed first responder configurations and resource utilizations.

REFERENCES

Cormen T. H., C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*, Second Edition. MIT Press and McGraw-Hill, 2001.Chapters 24,25:pp.580–642.

Fikes, R., and N. Nilsson. STRIPS: a new approach to the application of theorem proving. *Artificial intelligence*, Vol. 2, pp. 189-208, 1971.

Ingrand, F.F., M. P. Georgeff, and A. S. Rao. An architecture for real-time reasoning and system control. *IEEE expert magazine*, 7(6), pp. 33-44, 1992

Jain, S., and C. R. McLean 2006. An Integrated Framework for Modeling and Simulation for Incident Management. *Journal of Homeland Security and Emergency Management(Volume 3 Issue 1)*

Laughery, R., 1998. Computer simulations as a tool for studying human-centered systems. In *Proceedings of the 2001 winter simulation conference*, pp. 61-65, 1998

Macal, C. M., and M. J. North 2005. Tutorial on Agent-Based Modeling and Simulation. In *Proceeding of the 2005 Winter Simulation Conference*, ed. M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines,

Norling, E. Folk psychology for human modeling: extending the BDI paradigm. In *International conference on autonomous agents and multi agent system (AAMAS)*, New York, 2004.

Zhao X. , Y. Son. BDI Based Human Decision Making Model in Automated Manufacturing System. Department of Systems and Industrial Engineering, University of Arizona.

AUTHOR BIOGRAPHIES

AMEYA SHENDARKAR is a master's student in the department of Computer Science at the University of Arizona. He received his Bachelor of Engineering degree in Computer Engineering from Government College of Engineering, Pune, India. His research areas include distributed systems , computational aspects of simulation and database systems. He can be reached by email at <ameya@cs.arizona.edu>.

KARTHIK VASUDEVAN is a master's student in the department of Systems and Industrial Engineering at the University of Arizona. He received his Bachelor of Engineering degree in Electrical and Electronics from Sathyabama Deemed University, Chennai, India in 2005. His research focuses on applications of virtual reality in computer simulation, distributed simulation and simulation based shop floor control. He is a student member of IIE. He can be reached by email at <karthikv@email.arizona.edu>.

SEUNGHO LEE is a Ph.D. student in the department of Systems and Industrial Engineering at the University of Arizona. He received his Bachelor of Engineering degree in Industrial Engineering from Korea University, Seoul, Korea in 1999 and his M.S. degree in Industrial Engineering from Texas A&M University in 2005. His research focuses on application of distributed simulation and simulation of human decision making. He is a student member of IIE. He can be reached by email at <mountlee@email.arizona.edu>.

YOUNG JUN-SON is an associate professor in the Department of Systems and Industrial Engineering at The University of Arizona. Dr. Son received his Ph.D. in Industrial and Manufacturing Engineering from Penn State University in 2000. His research focuses primarily on the application of distributed and hybrid simulation to the analysis and control of automated manufacturing system and integrated enterprise. He is a recipient of 2004 SME Eugene Merchant Outstanding Young Manufacturing Engineer Award and 2005 IIE Outstanding Young Industrial Engineer Award. He is an associate editor of the International Journal of Modeling and Simulation and the International Journal of Simulation and Process Modeling, and a professional member of ASME, IEEE, IIE, INFORMS, and SME. He can be reached by email at <son@sie.arizona.edu>.