Proceedings of the 2018 Winter Simulation Conference
M. Rabe, A.A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, eds.

MODELING SAFEST AND OPTIMAL EMERGENCY EVACUATION PLAN FOR LARGE-SCALE PEDESTRIANS ENVIRONMENTS

Muhammad Haris
Imran Mahmood
Maryam Badar
Muhammad Saad Qaisar Alvi

Center for Research in Modeling, Simulation & Vision (Crimson)
School of Electrical Engineering and Computer Science
National University of Sciences & Technology
Sector H-12, Islamabad, 44000, PAKISTAN

ABSTRACT

Large-scale events are always vulnerable to natural disasters and man-made chaos which poses great threat to crowd safety. Such events need an appropriate evacuation plan to alleviate the risk of causalities. We propose a modeling framework for large-scale evacuation of pedestrians during emergency situation. Proposed framework presents optimal and safest path evacuation for a hypothetical large-scale crowd scenario. The main aim is to provide the safest and nearest evacuation path because during disastrous situations there is possibility of exit gate blockade and directions of evacuees may have to be changed at run time. For this purpose run time diversions are given to evacuees to ensure their quick and safest exit. In this work, different evacuation algorithms are implemented and compared to determine the optimal solution in terms of evacuation time and crowd safety. The recommended framework incorporates Anylogic simulation environment to design complex spatial environment for large-scale pedestrians as agents.

1 INTRODUCTION

Large-scale events such as: religious obligations, sports events, cultural shows, annual dinners and political protests are very common these days. These special events are attended by millions of people and are quite vulnerable to catastrophic incidents, due to natural or manmade disasters. Hence, these events require contingency planning and emergency support. Therefore emergency evacuation planning for crowded areas holds imperative position and thus plays vital role to alleviate the risk of casualties. In many cases, we can’t afford to find the appropriate solution by experimenting with the real world. Experimentation in the real world can be too expensive, dangerous, or is just impossible. It is therefore intractable, risky, and difficult to experiment emergency situations in real to formulate the evacuation plans for public safety (Aguiar 2010). As compared to the real experiments, simulation models are low cost, time independent and provides harmless experimental environment, thus providing risk-free replication of real scenarios, where modelers can perform different tests to gain the insights of the system and compare a variety of strategies to determine the optimal solutions (Banks et al. 2013). Modeling and simulation is widely used in solving crowd safety and evacuation problems (Zhou et al. 2010; Ijaz et al. 2015; Zaharia et al. 2009). Crowd simulation has been found effective in modeling and analyzing behavior and movements of dense crowds in different areas such as: shopping malls, airports and parks thus providing provisions for adapting better strategies in managing the crowd and reducing number of causalities. We propose an agent-based crowd simulation framework which incorporates large-scale pedestrians as agents in a complex spatial environment and provides safest path evacuation during disastrous situations. Hence, provides run time diversions from the assigned exits to the safest exits. We have implemented and compared two evacuation strategies, Harmony search and shortest distance, under different scenarios to provide the optimal solution for crowd safety. The proposed method has the capability to be used for case studies other than the presented one to provide safe and feasible evacuation plans. The Anylogic simulation environment is used to develop the proposed framework, which provides space markup and pedestrian library to design complex spatial environment and simulate large-scale pedestrians.

The rest of the paper is organized as follows: Section 2 describes related work; Sections 3 illustrates the background of concepts used in this paper; Section 4 formulates the main problem which has been solved in this paper; Section 5 describes the proposed methodology; simulation studies and results are given in Section 6; finally section 7 presents the conclusion.

2 RELATED WORK

The literature has witnessed several approaches for simulating crowd evacuation models. A simulation framework comprising of agent-based model and Genetic Algorithm (GA) has been proposed for crowd evacuation (Abdelghany et al. 2014). They used GA to determine the optimal gate assignments whereas crowd simulation model has been used to evaluate the GA's result. Another evolutionary-algorithm based framework has been proposed for crowd evacuation planning (Zhong et al. 2015). The recommended framework includes agent-based simulation model and optimization approach - Cartesian Genetic Programming (CGP) to obtain the heuristic rule (evacuation plan). The main idea is to divide the whole area into sub-areas and to allocate the exit gates to each region using heuristic rule dynamically. Pedestrians of each region will follow the assigned exit gate to evacuate from the scene. The framework has been trained and tested on different evacuation scenarios to obtain the generic heuristic rule. Frescha and Zia proposed the use of a wearable device named life belt for crowd evacuation. This device contains sensors which are used to detect the current location of each pedestrian and recommend exit gates at run time to each pedestrian according to their position (Ferscha et al. 2009). Moreover, a leader-follower approach based evacuation model has been recommended by JI and GAO. In this approach, leaders are decided and assigned to crowd dynamically during emergency whereas A* algorithm is used to find the exit paths. The exit paths are conveyed to each leader according to their location and rest of the people follow their assigned leaders to evacuate from the hall (Ji et al. 2006). Lee et al. 2010 suggested evacuation model using Belief-Desire-Intention (BDI) architecture. The extended BDI framework is demonstrated through agent-based modeling and simulation. Their work is dedicated to representing the characteristics of the human decision-planning and decision-making process during selected evacuation scenarios. The developed simulation enables us to observe crowd behaviors under evacuation scenarios. An extensive study (Mahmood et al. 2017) suggested evacuation framework comprising of crowd simulation model and different evacuation strategies such as random evacuation, shortest regional distance and GA. They scrutinized their proposed work using the case study of evacuating pilgrims from The Great Mosque of Makkah. The comparison of their algorithms showed that GA has performed remarkably better than the other two strategies. An INCROWD framework has been proposed for crowd interaction management. It is an integrated framework for crowd interaction, mining and making decision purpose for crowd management (Wijermans et al. 2016).

The so far proposed methodologies have mostly presented static evacuation plans. Moreover, they just focused on finding the evacuation plan which offers nearest-gate immediate evacuation irrespective of the safety of pedestrians.

In this paper, we propose an agent-based crowd simulation framework which mainly focuses on providing safest path for evacuation during disastrous situations. It provides diversions dynamically from the assigned exits to the safest exits. We have implemented and compared different evacuation algorithms to provide the optimal solution for crowd safety.

3 BACKGROUND

3.1 Anylogic Simulation Environment

AnyLogic (Anylogic Simulation Software 2018) is a versatile simulation tool with graphical interface that allows modeler to quickly model complex environments. It supports different modelling techniques such as Discrete Event, Agent Based and System Dynamics (Borshchev 2013).

We propose the use of Anylogic simulation software for development of our proposed solution. Anylogic provides different libraries such as Pedestrian, Agent-based and Space markup to build complex pedestrian models. Different primitive elements of Space Markup Library such as: walls, polylines and pathways can help to make rich spatial environments for pedestrian models (Space Markup Palette 2018). Moreover, AnyLogic Pedestrian Library (APL) which is basically an agent-based modeling, is used to simulate pedestrians flow and their behavior in a physical environment (Anylogic Pedestrian Simulation 2018). Table 1 describes the use of APL components which have been employed to develop a pedestrian model. APL also provides user friendly interface for robust development of pedestrian models in the form of flowcharts. Pedestrian models allow the user to collect statistics on pedestrian density in different areas, measure safety precautions and compute time estimations on services and queues.

Table 1: Pedestrian Library Blocks.

	PedConfiguration	Used to configure important parameters for pedestrian objects
	PedGround	Used to draw the floor where pedestrians can walk and also used to define walls and obstacles
	PedArea	Represents walking area for pedestrians
	PedAttractors	It Allows to control pedestrians' location in an area. Attractors define the waiting points where pedestrians will go and wait inside an area
	PedSource	It is the starting point of pedestrian model and is used to generate agents with specified rate
	PedSink	It is the final block of pedestrian model and is used to dispose the pedestrians
	PedGoto	PedGoto is used to direct the crowd towards specified location which can be defined using area, target line or attractor
	PedWait	It causes pedestrians to wait for a specified time at a specified location
	PedSelectOutput	It is used to define the conditional statements in the model. It routes the pedestrians to one of several output ports according to the specified conditions or probabilities

3.2 Harmony Search Algorithm (HSA)

Harmony Search (HS) is a metaheuristic approach inspired from musical instruments and focuses on finding the best state of harmony. In musical terminology, harmony can be defined as multiple pitches being played at a time. The top notch musical performances obtain the best state of harmony which is determined by aesthetic principles, similarly optimization problems obtain the global optimum solution which is determined by objective function (Lee et al. 2005). We have used this correspondence between harmony and optimization to determine the optimal solution for our problem. Detailed working of HS is given in the methodology section.

4 PROBLEM FORMULATION

This section illustrates the design of the spatial environment and also describes the problem to be solved.

4.1 Spatial Environment

Spatial environment has been developed using different space markup elements such as area, walls, attractors and enter/exit gates. Figure 1 shows an area where pedestrians can walk and wait, and this area is divided into 20 logical sub-regions (R1, R2,…, R20). Each sub-region will be assigned an exit gate and people of that region will evacuate to gate which they have been directed to follow. Black border lines are denoting the walls and obstacles in an area and there are total 8 entrance/exits (Exit1, Exit2,…, Exit8), depicted in green color.

4.2 Optimal Safe Route

During emergency situation it is possible that different gates can be blocked or unsafe according to the circumstances. This uncertainty can be harmful for pedestrians and number of causalities can increase. Our main goal is to choose the optimal and safest exit for crowd evacuation at run time. The optimal gate assignment for each region has been determined using HSA and for safest exit, run time diversions (exit gate number) are defined for the regions whose initially assigned gates are blocked.

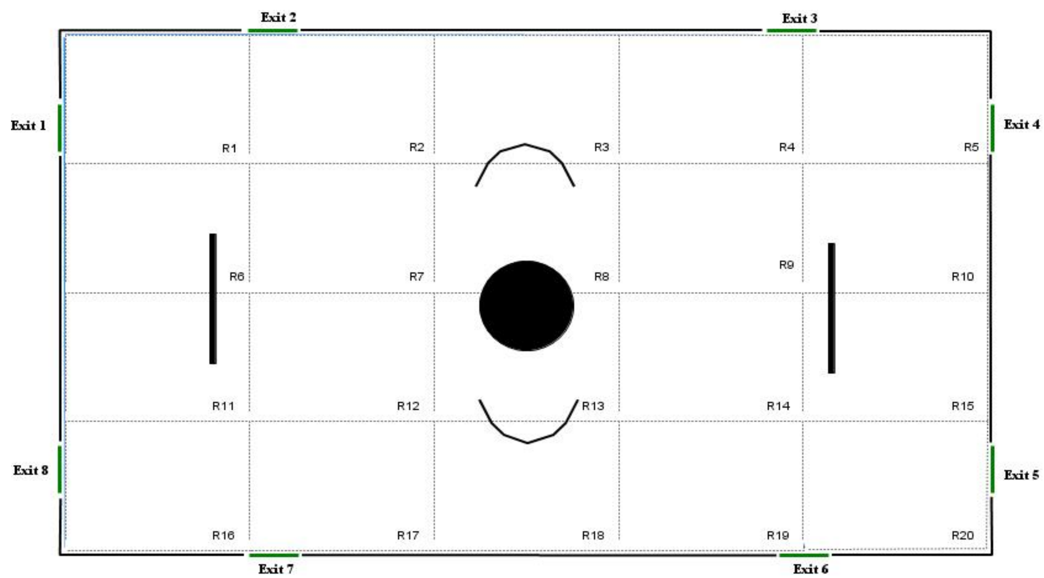

Figure 1: Anylogic Spatial Environment.

5 PROPOSED METHOD

5.1 Shortest Distance Evacuation

This approach uses the minimum distance from exit gates for pedestrians to evacuate. At first, the whole region is partitioned into logical sub-regions and the crowd is divided into regions based on their {x, y} positions. According to the region {x, y} coordinates, nearest distance has been calculated and the closest gate is allocated to each region. Thus, people of a particular region will move towards the gate which has been allocated to their region during evacuation.

```
Procedure: Shortest Distance
      for each R i ∈ Region do
         RXY ← (R.X, R.Y)
         Gate ← Nearest gate from (R.X, R.Y)
         Gets the gate with shortest distance from the {x,y} coordinates to a region
         if Current Gate is not Safe then
            Gate ← Get next immediate shortest distance gate for that region
         end if
      end for
Procedure end
```

During evacuation, if a gate is not safe to move, which has been allocated to a particular region then another gate will be assigned to that region. As this approach focuses on evacuating people with shortest distance so, the next immediate nearest gate will be allocated to that region.

5.2 Harmony Search Algorithm

As mentioned earlier we have implemented HSA to obtain the optimal evacuation solution for each sub-region. Initially, the optimization problem and main parameters of algorithm are defined which are used to find optimized results. These parameters include harmony memory size (HMS) (also known as the number of solution vectors in harmony memory), harmony memory consideration rate (HMCR), pitch adjusting rate (PAR), number of improvisations (NI) and stopping criteria. Harmony memory (HM) is a memory where all solution vectors are stored. These solution vectors represent gate assignments used for crowd evacuation whereas HMCR and PAR are used to improve the solution vectors to find the optimal solution. Figure 2 shows harmony memory matrix example according to the spatial environment described in Figure 1. HMM is defined as randomly generated solution vectors according to the harmony memory size. Where each matrix entity is representing the gate against the sub-region.

$$\begin{bmatrix} 1 & 3 & 4 & 5 & 6 & 7 & 8 & 8 & 2 & 1 & 4 & 3 & 2 & 6 & 4 & 5 & 7 & 5 & 4 & 3 \\ 3 & 5 & 1 & 2 & 4 & 6 & 7 & 8 & 2 & 5 & 1 & 3 & 2 & 5 & 6 & 5 & 4 & 8 & 7 & 1 \\ 4 & 5 & 6 & 1 & 3 & 4 & 8 & 7 & 6 & 3 & 1 & 5 & 4 & 2 & 1 & 7 & 8 & 4 & 2 & 1 \end{bmatrix}$$

Figure 2: Harmony Memory Matrix.

After defining the HM matrix, fitness is calculated of each vector according to the objective function. Once the fitness of each vector is determined HM should be improvised according to the parameters defined in step 1. A new harmony vector can be improvised in 3 ways: (i) memory consideration (ii) pitch adjustment (iii) random selection. In memory consideration, value of decision variable for new vector is chosen from already defined HM range. The value of HMCR is varied from 0 to 1. HMCR defines ratio of selecting the value from HM and (1-HMCR) is the rate of randomly selecting value from possible range of values. Each value determined by memory consideration should be pitch-adjusted. This operation can be done using PAR parameter. Third way of improvising harmony is to select random values from defined range and add them to the newly generated vector. After generating the new vector, its fitness is calculated and compared with the worst harmony in HM. If newly generated vector is better, it is replaced in HM and worst harmony is excluded. The stopping criteria needs to be checked after each iteration. If maximum number of improvisations has been reached, then it should be stopped otherwise the HM will be improvised.

```
   Procedure: Harmony Search
        Input: N ⊳ No of vectors/harmonies in a harmony memory
      Output: Harmony with Optimal Evacuation Time

      Initialize HS parameters HMS, HMCR, PAR, NVAR, NI
      {Harmony Memory} ← Generate Harmony Memory (N)
         while Iteration < Maximum Iterations
            for i = 1 to NVAR do
               if Random Value < HMCR
                     NewHarmony[i] ← memoryConsideration
                        if Random Value < PAR
                                          NewHarmony[i] ← PitchAdjustmentRate
                        end if
               end if
            Else
                  NewHarmony[i] ← randomSelection
            end else
            Current Fitness ← calculate fitness(NewHarmony)
               ⊳ Run simulation model for each harmony with assigned gates for each
region and returns      Evacuation Time as fitness value
         end while
            ⊳ get fittest harmony i.e. harmony having minimum value
      Return Fittest Harmony
Procedure end

Procedure: Generate Harmony Memory
      Input: HMS, NVAR
      Output: {Harmony Memory Matrix}

      for k = 1 to HMS
         for j = 1 to NVAR
               HM[k][j] ← Get Gate(length)
               HM[k][NVAR] ← Calculate Fitness(HM[k])
                     ⊳ Run simulation model for each harmony with assigned gates
for each region and returns              Evacuation Time as fitness value
         end for
      end for
           Return {Harmony Memory Matrix}
Procedure end
```

HS has been integrated with simulation model to obtain the optimal evacuation plan. Figure 3 describes the workflow of HS and its integration with simulation model. In order to obtain the safest path for evacuation, run time diversions will be made. So in this case original plan which is obtained using HS will be modified according to the conditions. Regions which have closed gates will be assigned new gates according to the current situation. New gates assignments will be determined using two approaches, either calculate next nearest exit or obtain the gate which is less crowded. If next nearest gate is less crowded than other gates, then this gate will be replaced with the blocked gate. Whereas, if another exit which is less crowded than shortest exit, then less crowded gate will be preferred over nearest exit. According to these rules new gate assignments will be determined.

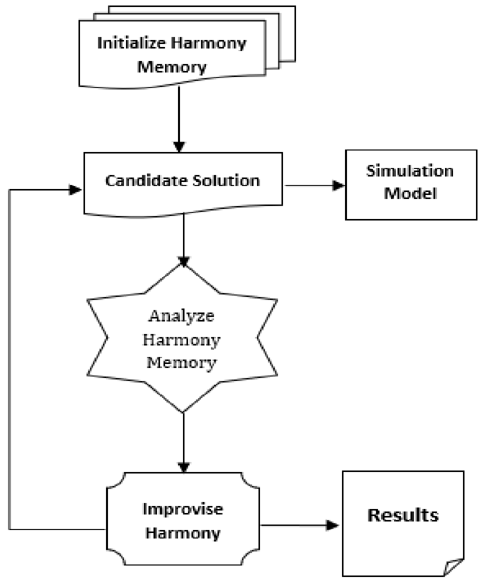

Figure 3: Harmony Search Flow.

6 SIMULATION AND RESULTS

6.1 Model Development

After the development of spatial environment, APL is used to simulate the crowd in courtyard. The working of APL components for crowd simulation model have been depicted in Figure 4. Initially PedConfiguration is used to configure some important parameters related to objects of pedestrian library. Then PedGround is used to define the floor where pedestrians walk. After configuring the PedGround, PedArea and PedAttractors are used to control the walking speed and movement of pedestrians in the courtyard. When all the configuration blocks have been defined, PedSource, which is the starting point of the simulation model, is used to generate the pedestrians in the courtyard. Then PedWait is used to keep the pedestrian in the courtyard for large time and they wait or walk there until the occurrence of emergency event. Once the emergency event occurs, PedWait.cancell function causes the pedestrians to move to the PedSelectOutPut (PS1, PS2 and PS3). PedSelect is a gate selection block which directs the crowd towards their PedGoto block, representing the exit gates (Exit1, Exit2…. Exit8), and finally the PedSink is called to dispose the crowd.

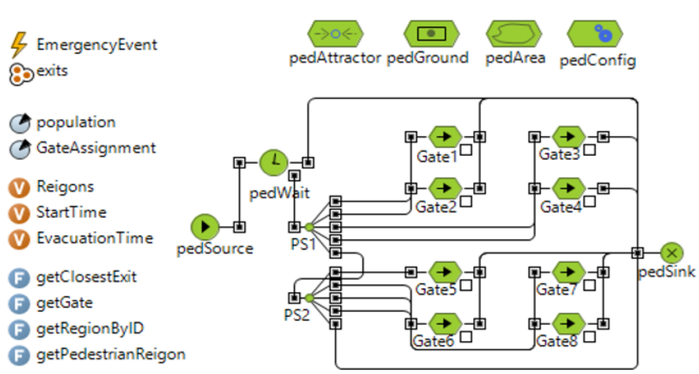

Figure 4: Anylogic Pedestrian Model.

6.2 Simulation Results

The simulation has been evaluated with 15000 population and each simulation is executed 20 times then mean evacuation time has been plotted for each algorithm.

6.2.1 Scenario 1 (One gate closed)

This scenario depicts the situation when one gate has been closed which pedestrians can't cross during evacuation. When emergency event occurs one gate among 8 gates has been randomly closed at run time. Figure 7 describes the scenario where gate 3 is closed and depicted in red color. The people of the sub-region to whom gate 3 was allocated, will not follow this gate and their route will be diverted at run time according to the evacuation strategy. Figure 8 shows the mean evacuation time of SD and HS, where SD takes mean 7.9 minutes and HS takes mean 6.8 minutes to evacuate people from the scene. Figure 5 and Figure 6 shows the evacuation plans obtained using HS and SD respectively.

$$\begin{bmatrix} R1 & R2 & R3 & R4 & R5 & R6 & R7 & R8 & R9 & R10 & R11 & R12 & R12 & R14 & R15 & R16 & R17 & R18 & R19 & R20 \\ E1 & E2 & E2 & E4 & E4 & E1 & E1 & E5 & E4 & E2 & E8 & E7 & E7 & E5 & E5 & E8 & E7 & E6 & E6 & E6 \end{bmatrix}$$

Figure 5: Optimal Gate Assignment Using HS.

$$\begin{bmatrix} R1 & R2 & R3 & R4 & R5 & R6 & R7 & R8 & R9 & R10 & R11 & R12 & R12 & R14 & R15 & R16 & R17 & R18 & R19 & R20 \\ E1 & E2 & E2 & E3 & E3 & E1 & E1 & E7 & E3 & E4 & E8 & E7 & E7 & E6 & E6 & E8 & E7 & E7 & E6 & E6 \end{bmatrix}$$

Figure 6: Shortest Distance Gate Assignment.

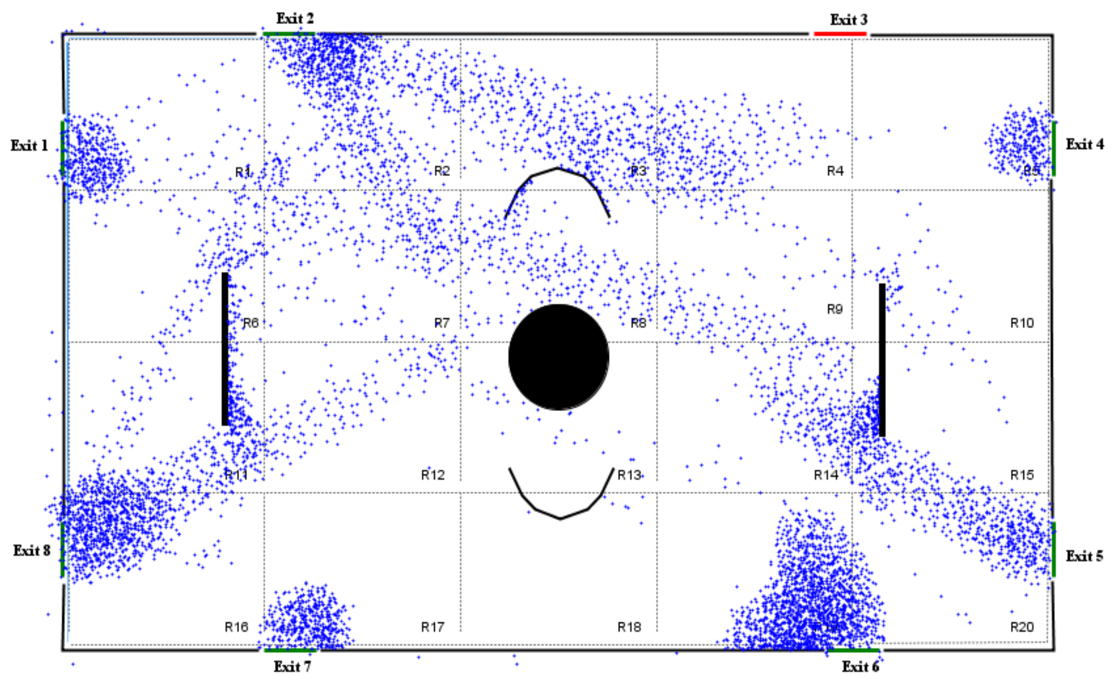

Figure 7: Evacuation Scenario with One Closed Gate.

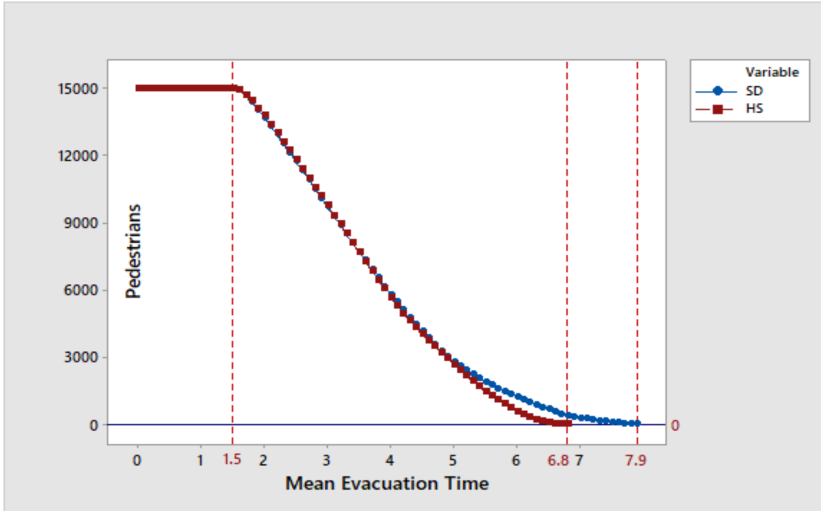

Figure 8: Mean Evacuation Time of One Closed Gate.

6.2.2 Scenario 2 (Two gates closed)

This scenario depicts the situation when two out of 8 gates have been closed. When emergency event occurs two gates are randomly closed at run time. Figure 9 describes the scenario where gate 1 and 5 are closed and denoted in red color. So, run time diversions will be made for people of the sub-regions to whom these gates were previously assigned. Figure 10 shows the mean evacuation time of SD and HS with two closed gates scenario, where SD takes mean 8.3 minutes and HS takes mean 7.2 minutes to evacuate people from the scene.

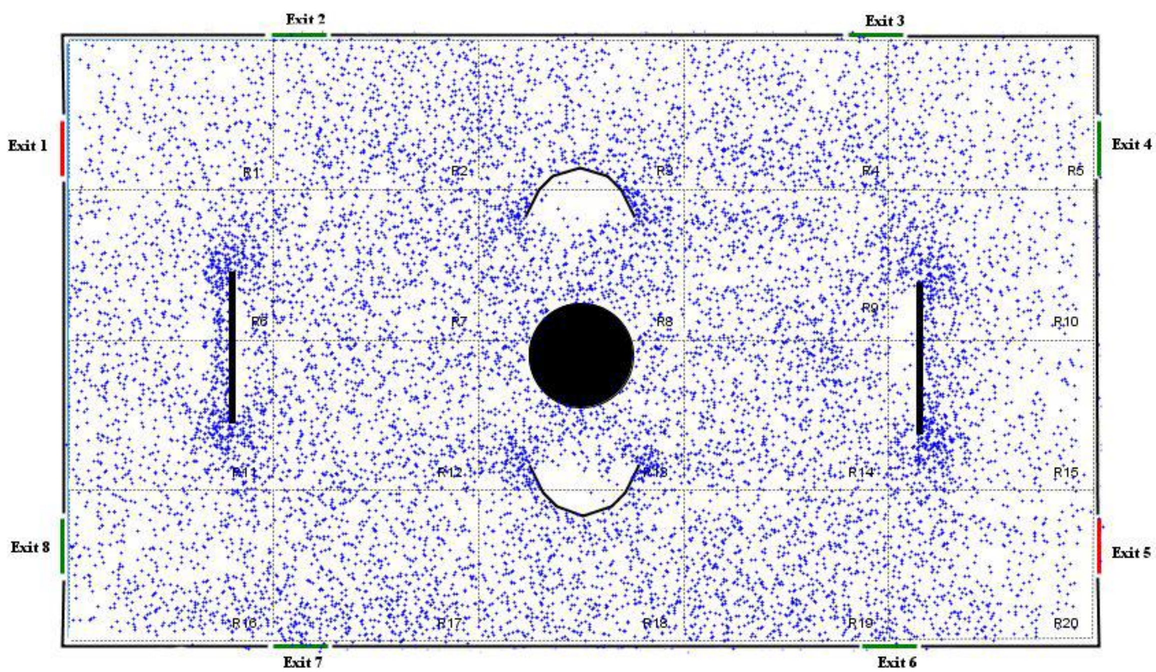

Figure 9: Evacuation Scenario with Two Closed Gates**.**

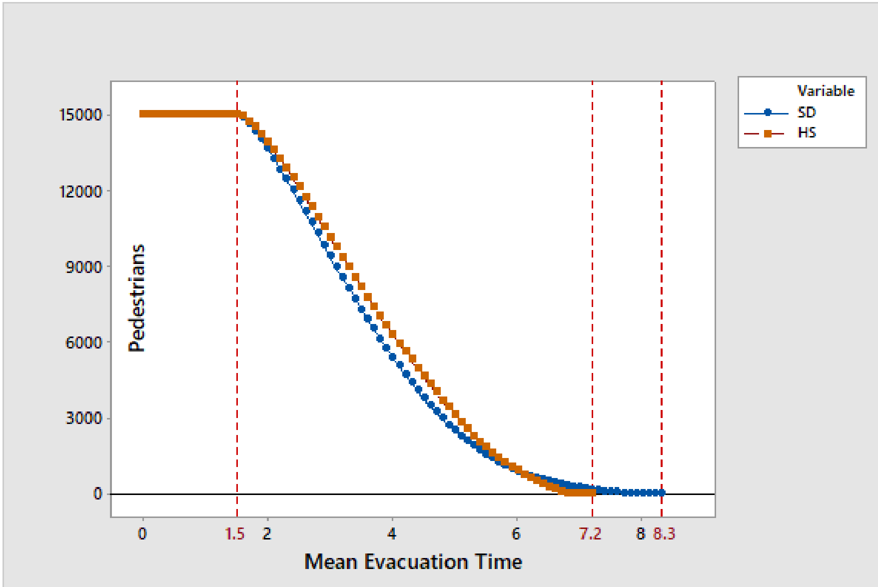

Figure 10: Mean Evacuation Time for Two Closed Gates**.**

7 CONCLUSION

In this research we have presented simulation based safest evacuation strategy for overcrowded areas which can confront emergency situations either because of natural disasters or due to man-made chaos. Most of the methods proposed in the literature are focused only on quick evacuation of masses from disastrous area from the nearest exit, irrespective of the risk posed by clogged door due to overcrowding. To the best of our knowledge, this is the first attempt to provide optimal paths in terms of safety and exit time to people for their safety and security during large events. We have proposed an agent-based crowd simulation framework to determine the optimal evacuation strategy for ensuring crowd safety.

We provided key steps to build the spatial environment using AnyLogic's libraries: APL, and space markup. Proposed framework adopts SD and HS strategies for crowd evacuation. These approaches have been tested on a hypothetical scenario with different simulation runs by varying parameters to analyze and obtain the optimal evacuation plan. We have also incorporated recommendation of safest evacuation path at run time in order to further reduce the number of causalities.

There are several future directions of proposed work. This work can be used to develop simulation model and evacuation plans for the Holy City. Moreover, our crowd simulation model is extendable to micro level in order to incorporate the attributes and physical states of each pedestrian which can provide detailed insights of the human behavior during evacuation.

REFERNCES

Abdelghany, A., K. Abdelghany, H. Mahmassani, and W. Alhalabi. 2014. "Modeling framework for Optimal Evacuation of Large-scale Crowded Pedestrian Facilities". *European Journal of Operational Research, 237* (3): 1105-1118.

Aguiar, F. H. M. 2010. Crowd simulation Apllied to Emergency and Evacuation Situations.

Anylogic Pedestrian Simulation. 2018. Retrieved 2018, from https://en.wikibooks.org/wiki/Simulation_with_AnyLogic/Pedestrian_Simulation

Anylogic Simulation Software. 2018. Retreived 2018, from http://www.anylogic.com/

Banks, J., J.S. Carson, B.L. Nelson, and D.M. Nicol. 2013. *Discrete-event System Simulation: Pearson new international edition*. 5th ed. Pearson USA.

Borshchev, A. 2013. *The Big Book of Simulation Modeling: Multimethod Modeling with AnyLogic 6*: AnyLogic North America Chicago.

Ferscha, A., and K.Zia. 2009. *Lifebelt: Silent Directional Guidance for Crowd Evacuation. In proceedings of the 2009 International Symposium on Wearable Computers*, September 4th-7th, Linz, Austria.

Ijaz, K., S. Sohail, and S.Hashish. 2015. "A Survey of Latest Approaches for Crowd Simulation and Modeling using Hybrid Techniques". *In Proceedings of the 17th UKSIMAMSS International Conference on Modelling and Simulation*, March 25th-27th, Washington, DC, USA.

Ji, Q., and C.Gao. 2006. "Simulating Crowd Evacuation with a Leader-follower Lodel". *IJCSES, 1* (4): 249-252.

Lee, K. S., and Z.W.Geem. 2005. "A New Meta-heuristic Algorithm for Continuous Engineering Optimization: Harmony Search Theory and Practice". *Computer methods in applied mechanics and engineering, 194* (36-38): 3902-3933.

Lee, S., Y.J. Son, and J.Ji. 2010. "An Integrated Human Decision Making Model for Evacuation Ecenarios Under a BDI Framework". *ACM Transactions on Modeling and Computer Simulation (TOMACS), 20* (4): 23.

Mahmood, I., M. Haris, and H.Sarjoughian. 2017. "Analyzing Emergency Evacuation Strategies for Mass Gatherings using Crowd Simulation And Analysis framework: Hajj Scenario." *In Proceedings of the 2017 ACM SIGSIM Conference on Principles of Advanced Discrete Simulatio*n, May 24th-26th, Singapore, Republic of Singapore.

Space Markup Palette. 2018. Retreived 2018, from https://help.anylogic.com/index.jsp?topic=%2Fcom.anylogic.help%2Fhtml%2Fui%2FSpace+Markup+palette.html

Wijermans, N., C. Conrado, M. Van Steen, C. Martella, and J.Li. 2016. "A landscape of Crowd-Management Support: An Integrative Approach". *Safety science, 86*, 142-164.

Zaharia, M. H., F. Leon, C. Pal, G. Pagu, N. Baykara, and N.Mastorakis. 2009. "Agent-based Simulation of Crowd Evacuation Behavior*". In Proceedings at the WSEAS International Conference. Proceedings. Mathematics and Computers in Science and Engineering*, May 30th-June 1st, Istanbul, Turkey.

Zhong, J., W. Cai, and L.Luo. 2015. "Crowd Evacuation Planning using Cartesian Genetic Programming and Agent-based Crowd Modeling*". In Proceedings of the 2015 Winter Simulation Conference*, 127-138, edited by L.Yilmaz et al., Piscataway, New Jersey:IEEE.

Zhou, S., D. Chen, W. Cai, L. Luo, M.Y.H. Low, F. Tian, and B.D.Hamilton. 2010. "Crowd Modeling and Simulation Technologies". *ACM Transactions on Modeling and Computer Simulation (TOMACS), 20* (4): 1-34.

AUTHOR BIOGRAPHIES

MUHAMMAD HARIS is currently pursuing his Masters in Information Technology at the Department of Computing, School of Electrical Engineering and Computer Science, National University of Sciences and Technology (Pakistan). His current research interests are Agent-Based Modeling and Simulation. His email address is mharis.msit15seecs@seecs.edu.pk.

IMRAN MAHMOOD is currently working as an Assistant Professor at the Department of Computing, School of Electrical Engineering and Computer Science, National University of Sciences and Technology, (Pakistan). Imran earned his Masters and doctoral degrees in Computer Systems at the School of Information and Communication Technology (ICT), KTH-Royal Institute of Technology Sweden in 2007 and 2013 respectively. He is serving as the Director of the Center for Research in Modeling and Simulation (CRIMSON). His current research interests are in applied modeling, simulation, analysis and formal verification of complex systems. He can be reached at imran.mahmood@seecs.edu.pk.

MARYAM BADAR is currently pursuing her Masters in Computer Science at the Department of Computing, School of Electrical Engineering and Computer Science, National University of Sciences and Technology (Pakistan). Her current research interests are computer vision and Agent-Based Modeling and Simulation. Her e-mail address is mbadar.mscs16seecs@seecs.edu.pk.

MUHAMMAD SAAD QAISAR ALVI is currently pursuing his Masters in Information Technology at the Department of Computing, School of Electrical Engineering and Computer Science, National University of Sciences and Technology (Pakistan). His current research interests are Agent-Based and System Dynamics Modeling and Simulation. His email address is alvi.msit15seecs@seecs.edu.pk.