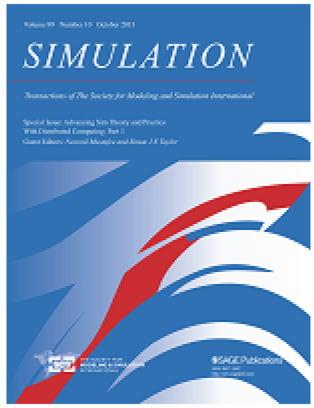

Analyzing Effect of Crowd Formations on Evacuation Plans using Crowd Simulation and Analysis Framework

Journal:	*Simulation: Transactions of the Society for Modeling and Simulation International*
Manuscript ID	T-20-0211
Manuscript Type:	2 Methodology
Date Submitted by the Author:	10-Aug-2020
Complete List of Authors:	Haris, Muhammad ; National University of Sciences and Technology, School of Electrical Engineering & Computer Science (SEECS) Badar, Maryam; National University of Sciences and Technology, School of Electrical Engineering & Computer Science (SEECS) Saboor, Abdul; National University of Sciences and Technology, School of Electrical Engineering & Computer Science (SEECS) Khan, Muneed Ahmed; Institute of Southern Punjab, Department of Computer Science Askari, Syed Hassan; The University of Lahore, Department of Software Engineering Alvi, Muhammad Saad Qaisar ; National University of Sciences and Technology, School of Electrical Engineering & Computer Science (SEECS)
Key Words and Phrases:	Tools and Technology -> Agent-based Systems , Tools and Technology -> Modeling and Simulation Environments , Tools and Technology -> Simulation System Architecture
Abstract:	Mass gatherings are often vulnerable to calamitous situations such as stampede, fire explosions and terrorist attacks. Cogent and robust pre-planning of crowd management and emergency evacuation is required to mitigate the risk of casualties. For efficient pre-planning, crowd evacuation formations (floor division) play a pivotal role in ensuring the safe and early evacuation of masses from the emergency area. Similarly, the evacuation algorithms are necessary to improve the evacuation time. In this context, we propose an agent-based large-scale pedestrians evacuation framework that examines the combined effect of crowd formations using evacuation algorithms on evacuation time. We have presented two case studies for the assessment of our proposed framework. From the results of both case studies, we observe that crowd

	formations have a significant effect on overall evacuation time. The simulation model has also been validated by performing real and simulation experiments on a local building to ensure the real-time applicability and reliability of our model.

Analyzing Effect of Crowd Formations on Evacuation Plans using Crowd Simulation and Analysis Framework

Muhammad Haris[1], Maryam Badar [1], Abdul Saboor [1], Muneeb Ahmed[2], Sayed Hassan Askari[3] Muhammad Saad Qaisar Alvi [1]
[1] School of Electrical Engineering and Computer Science, National University of Sciences and Technology (NUST), Islamabad, Pakistan
[2]Department of Computer Science, Institute of Southern Punjab (ISP), Multan, Pakistan
[3]Department of Software Engineering, University of Lahore, Lahore, Pakistan
Email:[mharis.msit15seecs, mbadar.mscs16seecs, asaboor.msit15seecs, malvi15seecs]@seecs.edu.pk, muneebahmad@isp.edu.pk, syed.hassan@se.uol.edu.pk

***Abstract*—Mass gatherings are often vulnerable to calamitous situations such as stampede, fire explosions and terrorist attacks. Cogent and robust pre-planning of crowd management and emergency evacuation is required to mitigate the risk of casualties. For efficient pre-planning, crowd evacuation formations (floor division) play a pivotal role in ensuring the safe and early evacuation of masses from the emergency area. Similarly, the evacuation algorithms are necessary to improve the evacuation time. In this context, we propose an agent-based large-scale pedestrians evacuation framework that examines the combined effect of crowd formations using evacuation algorithms on evacuation time. We have presented two case studies for the assessment of our proposed framework. From the results of both case studies, we observe that crowd formations have a significant effect on overall evacuation time. The simulation model has also been validated by performing real and simulation experiments on a local building to ensure the real-time applicability and reliability of our model.**

***Keywords*—Agent-based Modeling, Emergency Evacuation, Risk Reduction, Evacuation Formations, Genetic algorithm**

I. Introduction

Crowd evacuation planning has gained attention of a lot of researchers in the past few years [1], [2], [3]. With the increase in socio-economic development, pre-planning of evacuation strategies during panic situations has become an indispensable part of managing the crowd and ensuring safety during large events [4], [5]. The large events or mass gatherings can be categorized as sports events, religious obligations, and political gatherings. Due to progressively increasing crowd size, such large events have become more vulnerable to many disastrous situations such as stampede, terrorist attacks, and fire explosions [6]. Event management teams and security departments need to handle emergencies efficiently to mitigate the risk of casualties. It requires the event hosts and management teams to properly coordinate with international or local Public Safety & Security (PSS) organizations. PSS is usually expressed as a governmental responsibility that primarily aims at the protection of public from dangers like disasters, health risks, crimes, or terrorism. Public Safety refers to the welfare and protection of the general public against natural or man-made disasters. In contrast, Public Security ensures the protection of communities against threats related to crimes, law & order, terrorism, and mass incidents (riots) [7]. These services bring value to society by creating a stable and secure environment and protection to the people and assets at the local, regional, and national levels.

From the above discussion, we can deduce that mass gatherings require enormous evacuation planning to mitigate the risk of causalities [8]. In order to determine the evacuation plan for public safety and security, we cannot perform speculative experiments in the real world. The primary reason is that such experiments can be risky, uneconomical and life-threatening. However, Modeling and Simulation (M&S) provides economical, risk-free, and harmless way to experiment and analyze hazardous situations [9]. In this context, crowd simulation is an important aspect to study the behaviour of pedestrians in large events for efficient analysis of different factors to ensure crowd safety and management [10]. M&S enables modellers to perform a diverse range of experiments to gain deep insights into the models and results to determine the better solutions under different conditions. The literature provides many simulation-based methods for crowd evacuation. However, to the best of our knowledge, this is the first attempt that focuses on determining the effect of crowd evacuation formations or floor plans on ensuring minimum evacuation time along with reduced clogging effect.

In this paper, we propose an agent-based crowd simulation and analysis framework to simulate hypothetical and real pedestrian environments with different crowd evacuation formations. After that, we evaluate and analyze the effect of these formations on overall evacuation time. Our proposed framework incorporates the use of the Anylogic Pedestrian Library (APL) to simulate the agents in a spatial environment and integrates external modules to formulate the evacuation plans. Our contributions in this paper based on the proposed framework are listed as follows:

- Our proposed framework incorporates the design of different spatial environments such as buildings, stadiums and mosques. Besides, it can incorporate obstacles and constraints of the crowd movements such as walls and doors.
- Our framework simulates large-scale crowd behaviour using agent-based modeling paradigm in the spatial environments, such as the evacuation in the Hajj scenario.
- We have used three different evacuation formations: Grid-based, Sector-based and Pentagonal-based for the evacuation during an emergency. Hence, we examine the impact of different evacuation formations on evacuation time.
- We investigate the combined effect of evacuation formations using different evacuation algorithms such as Shortest Distance (SD), Adaptive Shortest Distance (ASD) and Genetic Algorithm (GA) on the evacuation time.
- We validate the proposed simulation model by carrying out real and simulation experiments on a local building to check the real-time applicability of our model.

The rest of the paper is organized as follows: Section II illustrates the related work; Section III describes the proposed framework for crowd evacuation, and Section IV discuss the evacuation formations. Crowd evacuation algorithms are described in section V. Simulation results of case study 1 and 2 are presented in section VI and section VII, respectively. In sections VIII, we validate the presented model, along with the conclusion in section IX.

II. Related Work

Several techniques have been presented in the literature for crowd evacuation modeling. Mahmood et al. [11] use an agent-based modeling framework for crowd evacuation. The authors take a case study of the Hajj scenario and perform evacuation using three approaches; random evacuation, shortest regional distance, and GA. The results show that the evacuation carried out using GA is remarkably better than the other two methods. Abdelghany et al., [12] also use GA in their agent-based modeling framework to determine the optimal evacuation plan. An Evolutionary Algorithm (EA) based methodology has been adopted by Zhong et al., [13] for ensuring safety during emergency crowd evacuation. The EA tries to obtain an optimized discriminant function for dividing the evacuation area into formations or sub-regions. Each sub-region directs the masses to the corresponding exit gate for the efficient evacuation. Another EA-based method has been employed by Zhong et al. [14] for evacuation planning. It incorporates Cartesian Genetic Programming (CGP) to obtain the generic evacuation plans for multiple scenarios. Yang et al., [15] provide an approach named as Crowd-separated Allocation of Routes, Exits and Shelters (CARES) to avoid anomalies during crowd evacuation from Makkah. Their proposed methodology resolves issues like shelter allotment and transportation-network bottleneck during an emergency evacuation. Santosh et al. [16] uses an evacuation system based on soft computing for guiding people to the safest location during the fire explosion. They trained and tested an Artificial Neural Network (ANN) with multiple scenarios to prove the reliability of the proposed method.

From the literature review, we observe that the suitable evacuation model based of the environment plays a pivotal role in ensuring minimum evacuation time during mass-gathering events. An evacuation scene turns into chaos when the evacuees panic and haphazardly try to evacuate from the emergency without knowing an environment. Such a situation can be handled by using optimized evacuation algorithms. In this context, the authors in [17] have introduced an information transmission mechanism along with a collision avoidance strategy to optimize evacuation time during emergencies. Similarly, [18] proposed a modeling framework for mass evacuation with the safest path approach. They have used a concept of run time diversions in which the path diversions are instructed to evacuees dynamically during clogging that ensures the safest exit of evacuees. Chiu et al. [3] presents another evacuation model using a migration tool EXODUS and unity to simulate the pedestrians' evacuation under different conditions.
The second important aspect of the evacuation model is the use of evacuation formations. Most of the environments are divided into specific formations/sub-regions to evacuate pedestrians accordingly. Hence, the crowd evacuates from the gate, depending on their region to reduce clogging at a specific gate. These formations could be of any geometry, either Grid-based or Sector-based. Most of the proposed methodologies in literature have focused on single evacuation formation, i.e. Grid-based [12], [13] [14], [18], [19] or Sector-based [11]. However, the existing literature has not explored the effect of using different evacuation formations on evacuation time. In general, the suitable crowd formations help in reducing the evacuation time and minimize the clogging at exit gates. Therefore, in this paper, we propose an agent-based crowd simulation and analysis framework that combines the optimized evacuation algorithms with the evacuation formations. The aim is to explore the combined effect of different algorithms and evacuation formations on the evacuation time. We have implemented different algorithms such as SD, ASD, and GA in our framework and compare their results for different evacuation formations. In addition to the simulation, the model's real-time applicability is ensured by carrying out real experiments on a local building.

III. Proposed Framework

In this section, we present our 3-layer agent-based crowd simulation and analysis framework. The model consists of a simulation layer, an interface layer, and an optimization layer, as shown in Figure 1. The detail of these layers is provided in the following sub-sections.

A. Simulation Layer

This layer allows modelers to construct spatially explicit agent-based simulation models of crowd at large-scale. In this layer, we have used an Anylogic-based simulation environment for the development of agent-based models [20].

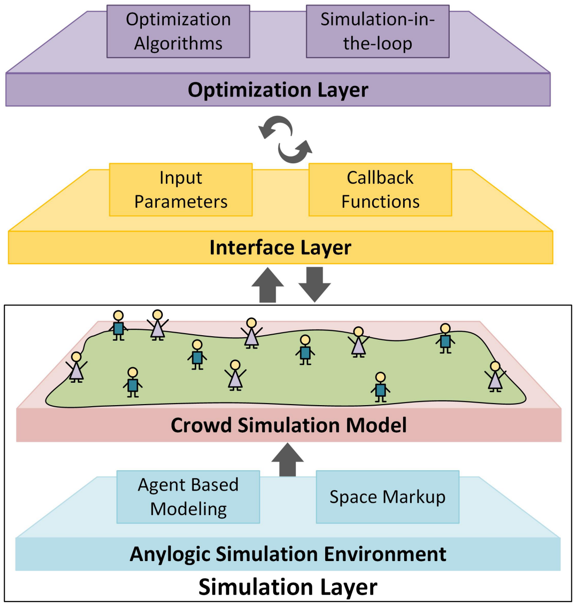

Fig. 1. Crowd Simulation & Analysis Framework

Anylogic provides agent-based modeling, space markup tools, and pedestrian library to build agents' complex behavior in spatial environments. Agent-based modeling is an appropriate paradigm to describe the individuals' nature and their interactions within the crowd by defining attributes such as age, gender, health status etc. The Space markup library provides necessary elements such as walls, doors, and pathways for developing a real-scale pedestrian environment in a continuous space. Moreover, the modeler can create a logical model of the crowd using APL blocks [21]. This library allows modelers to simulate the behavior of pedestrians, the distance between pedestrians, speed, density, and movement based on predetermined rules. By default, APL follows the rules of the social force model [22]. This model simulates the pedestrians' behavior, such as walk in a continuous space, reaction to the obstacles, and collision avoidance. Following are the objects of the pedestrian library used for the development of proposed crowd model:

- **PedSource:** Generates agents in the defined area
- **PedGoto:** Directs the crowd towards the specified gate
- **PedWait:** Causes pedestrians to wait for a pre-determined time at a particular location
- **PedSelectOutput:** Defines the conditional statements in the model and directs the pedestrians to one of several output ports according to the particular conditions
- **PedSink:** Disposes the pedestrians

B. Interface Layer

The interface layer is a Java-based middleware that interacts with the simulation layer and configures the parameters to run the model. Hence, it enables the integration of the simulation layer with external programs so that the model can be co-simulated with external environments. After building the simulation model, we can integrate different optimization algorithms with the developed model to find the optimal evacuation plan. During the model development, the interface layer allows the modeler to make specific parameters accessible, such as total population and gate assignment. This accessibility allows modelers to input varying values of these parameters and analyzes results from the simulation experiment.

C. Optimization Layer

The optimization layer is concerned with the implementation and integration of optimization algorithms with the simulation layer. Moreover, this layer analyzes the optimization scenario with the Simulation-in-the-loop concept. It interacts with the interface layer, which further inter-operates with the simulation layer to run the model in a loop, using a set of given input parameters, and retrieves the results. Furthermore, It provides the flexibility of integrating different optimization algorithms and heuristics such as simulated annealing, GA, and harmony search with the simulation layer. Thus, using this layer, we can execute our simulation model with different evacuation algorithms to obtain the optimized evacuation plan for crowd safety.
We aim to investigate the implementation of different optimization algorithms on diverse evacuation formations and their combined effect on evacuation time. Therefore, the next section discusses the evacuation formations used in this study.

IV. Evacuation Formations

Evacuation formation refers to the floor division in different sub-regions. It helps in classifying and guiding the crowd towards the designated gates to avoid clogging at a specific exit gate. We present a simulation model based on three different evacuation formations: Grid-based, Sector-based, and Pentagonal-based. In order to provide the proof of concept, we present a case study of hypothetical scenario as an example to demonstrate the functionality of our framework.The working of each layer is given as follows.

A. Simulation Layer (Model Development)

This layer describes how the model has been developed. It has two primary constituents (i) Spatial Environment (ii) Pedestrian Behavior.

1) Spatial Environment: We have tested Grid-based, Sector-based, and Pentagonal-based formations on a single hypothetical scenario for crowd evacuation, as shown in Figure 2, Figure 3 and Figure 4 respectively. The pedestrian physical environment is developed using different space markup elements like walls and enter/exit gates. After that, we integrate these three formations in this environment. Figure 2 depicts the scenario of Grid-based formation where the whole region is divided into twenty logical sub-regions (SR1, SR2,..., SR20). For the Sector-based evacuation scenario, the whole area is divided into eighteen logical sub-regions (SR1, SR2,..., SR18), as represented in Figure 3. Similarly, Figure 4 shows a Pentagonal-based evacuation scenario with twenty six sub-regions. In each figure, black border lines denote walls, whereas exit/entrance gates (Exit1, Exit2,..., Exit6) are

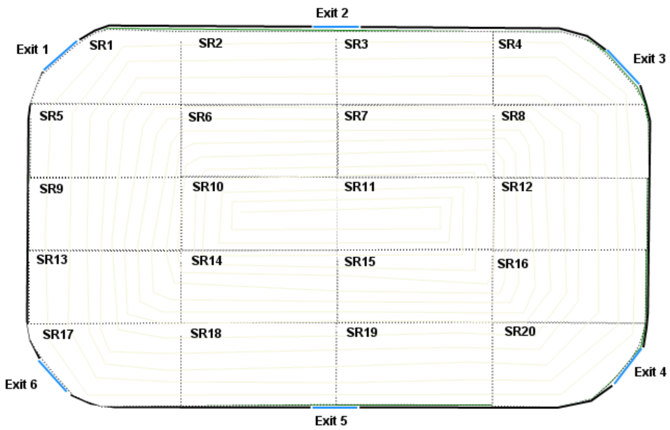

Fig. 2. Grid-based Evacuation Formation

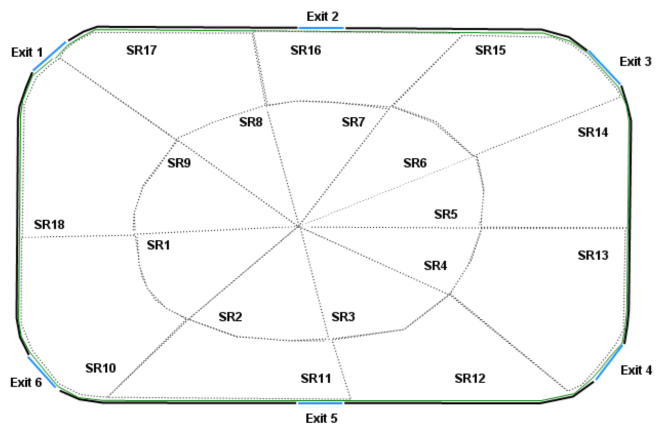

Fig. 3. Sector-based Evacuation Formation

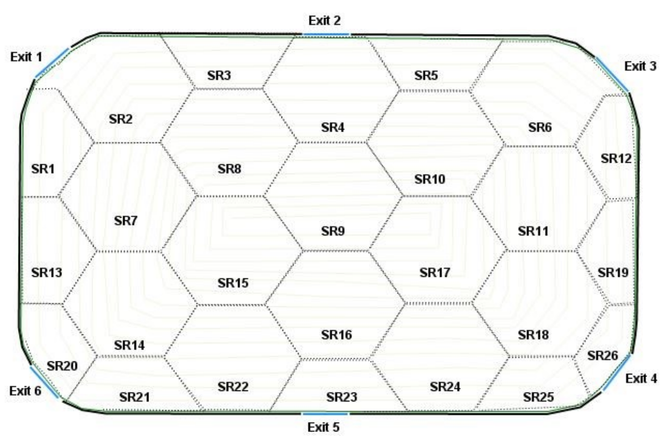

Fig. 4. Pentagonal-based Evacuation Formation

highlighted in blue. In this model, the orientation of exit gates and the boundary is kept the same in all three scenarios so that the significance of region-based evacuation formations can be analyzed effectively. The key idea behind the division of the whole region into sub-regions is to allocate specific exit gate to people present in a particular sub-region. During an emergency strike, the masses will be instructed to evacuate from the gate based on their sub-region.

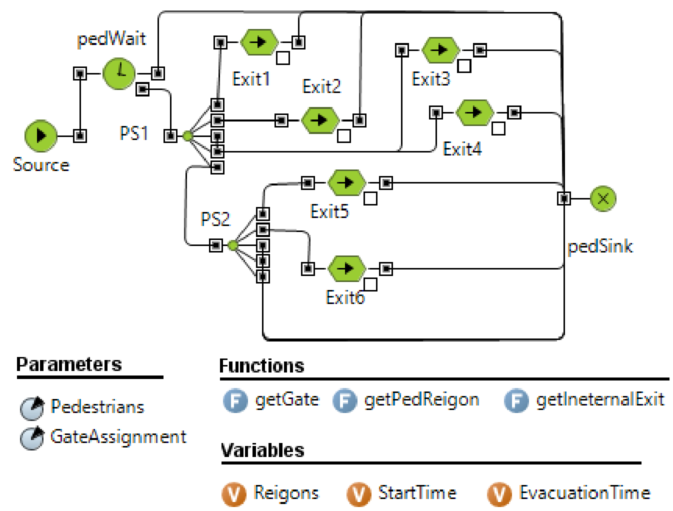

Fig. 5. Pedestrians Flow Control

2) Modeling Pedestrian Behavior: On top of the physical environment, the modeler can use APL objects to simulate the pedestrians in the courtyard with continuous space by following the social force model rules. Figure 5 shows the workflow of pedestrian objects. The *PedSource*, which is the starting point of the simulation model, generates the pedestrians in the courtyard. Depending on the model requirements, a modeler can adjust *PedSource* parameters such as pedestrians' walking speed and the total number of arrivals. The average walking speed of pedestrians is defined by triangular distribution [23] with parameters: min=1.0 m/s, most likely=1.4 m/s, max=1.8 m/s. The *PedWait* keeps the pedestrians moving in the area until the emergency occurs. Once the emergency event occurs, it moves the pedestrians towards the *PedSelectOutPut* (PS1 and PS2) block. The *PedSelectOutPut* is a gate selection block that directs the crowd towards their exit gates, and finally, the *PedSink* is called to dispose of the crowd.

B. Interface Layer

As mentioned earlier, during the model development, the interface layer allows the modeler to declare some parameters as the public that can be accessible in external environments. In order to integrate simulation model with external environments, a Simulation Experiment was developed which consists of the two parts, parameters and callback functions. These parameters are *Population* and *GateAssignment* as show in Figure 5. They are used by Interface layer and adjusted at run time during the optimization process. These parameters will be assigned varying values each time the simulation runs in the loop.

The interface layer also defines the *onFinish()* as a callback that can access model variables (EvacuationTime) and objects at any particular event or throughout the simulation lifetime. Once the simulation is finished, the callback function is called to retrieve the simulation results. The structure of the callback function is given below. The *onFinish()* function is called after the completion of each simulation run whereas the *root*

```
simulation.callback = new Callback() {
@Override
public void onFinish(Main root) {
System.out.println(root.EvacuationTime);
}
};
```

parameter is used to gain access to the total evacuation time when all the pedestrians exit from the courtyard.

C. Optimization Layer

The primary purpose of this layer is to implement the optimization algorithm. It enables us to implement any optimization algorithm in order to get the optimized exit plan to vacate the crowd in minimum time. This layer generates combinations of simulation parameters (gate assignments) and executes the simulation model with these parameters. Hence, it finds the optimized solution, which ensures minimum evacuation time. Figure 6 describes the working of the optimization layer.
Evacuation formation helps in dividing the crowd into different regions during the emergency. However, the optimal evacuation during such scenarios depends on the optimization algorithms. Therefore, the next section discusses such algorithms for the efficient expulsion of the crowd.

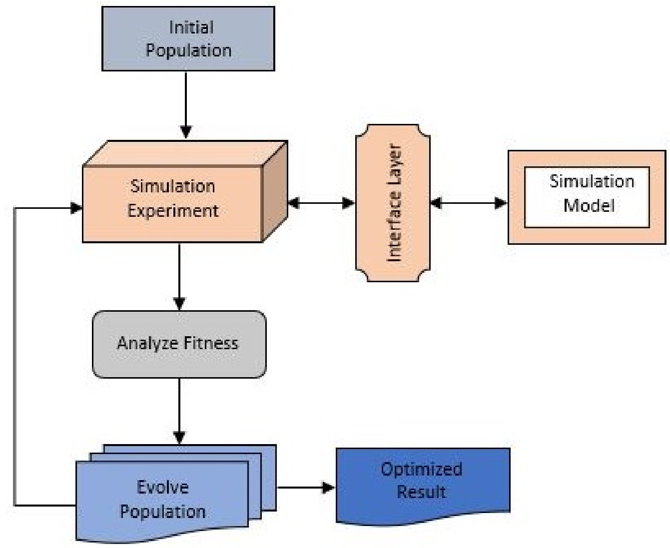

Fig. 6. Working of Optimization Layer

V. Crowd Evacuation Algorithms

We have implemented SD, ASD, and GA in the proposed framework for finding optimal evacuation plans. We aim to analyze the performance of these algorithms for three crowd formation scenarios i.e., Sector-based, Grid-based, and Pentagonal-based. The following sub-sections illustrate the implementation details of the above-mentioned algorithms.

A. Shortest Distance (SD)

This algorithm utilizes the mean position of sub-regions and location of exit gates for assigning the exit gates to masses belonging to a particular sub-region. The pedestrians are distributed into sub-regions based on their x,y coordinates. Equation 1 calculates the mean position for each sub-region by using these coordinates. In the equation, n denotes the total number of pedestrians in a region, and (Ped.X, Ped.Y) are the x,y coordinates of the pedestrian. The SD uses this mean position to calculate the distance between each sub-region withe each exit gate. Based on the calculated distance, the closest gate is allocated to each sub-region. Hence, the SD directs the people to evacuate from the nearest gate to their sub-region based on the formation.

$$RegionAverage = \frac{1}{n}\sum(Ped.X, Ped.Y) \qquad (1)$$

B. Adaptive Shortest Distance (ASD)

The ASD also uses the shortest distance parameter to assign the exit gate to each sub-region based on the evacuation formation. However, unlike SD, ASD equalizes the load on each gate to minimize the clogging at exit gates. It uses a balance factor (bf) to assign the load on the exit gates uniformly. The bf for an exit gate is the ratio of the total number of sub-regions(N_S) to the total number of exit gates(N_G) in the scenario, as illustrated in Equation 2. The first part of an equation represents the case when N_S and N_G are divided evenly. In this case, the balancing factor will be the same for each region. For example, if there are twenty regions with five exit gates in a scenario. A set of four regions will be directed towards each exit gate. In contrast, the second case of the equation deals with the uneven distribution of N_S and N_G. For instance, in a Grid-based scenario, there are a total of six gates and twenty sub-regions. Therefore, each gate will have minimum of three regions, whereas the division of the remaining regions will depend on each gate's load. The gates which are incorporating fewer pedestrians will bear the burden of remaining regions.
In ASD, the first and second shortest distances are calculated for each sub-region using their respective mean positions. If any exit gate bears more load than other gates i.e., bf is disturbed, then gate allocation is recalculated for some sub-regions according to their second shortest distance. The selection of the sub-regions for reassignment of exit gates is based on their distance from the previously allocated overloaded gate. For example, if two sub-regions are assigned to an overloaded gate, then the sub-region farther from the overloaded gate will be selected for gate reassignment. Hence, equalizing the load at each gate will decrease the clogging effect leading to reduced overall evacuation time.

$$bf = \begin{cases} \frac{N_s}{N_g} & 0 \\ \left\lfloor \frac{N_s}{N_g} \right\rfloor + 1 & otherwise \end{cases} \qquad (2)$$

C. Genetic Algorithm (GA)

GA is a meta-heuristic search algorithm for solving optimization problems. It is inspired by the theory of natural evolution in which the best variables (chromosomes) reproduces the next generation. The GA process starts from an initial

E1	E3	E4	E5	E6	E2	E2	E2	E5	E3	E5	E6	E1
R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13

Fig. 7. GA Chromosome Structure

population of chromosomes, also known as the generation. A generation comprises of multiple chromosomes, and every chromosome consists of a set of genes [24]. After that, the GA evaluates each chromosome's fitness based on the objective function. An objective function aims to calculate each chromosome's fitness score. The fitness score determines the probability of a chromosome to take part in the next reproduction. Hence, the chromosomes with the higher fitness score serve as parents in the next mutation to produce offspring with better quality. The same process continues until the maximum number of generations have been produced, or a termination condition has reached.

We have also used GA to obtain the optimal evacuation solution for the three selected formations. It is integrated with the simulation model through an interface layer that iteratively searches for the optimal solution. Figure 7 shows the structure of a chromosome for the evacuation scenario. This chromosome represents the sub-regions and their assigned gates. In the chromosomes, all the sub-regions (R1, R2..., R13) are fixed, and gates (E1, E2..., E6) are randomly assigned to each sub-region. Figure 8 shows the parent chromosomes and their crossover operation to generate the offspring. After performing crossover, the mutation is performed on the offspring to maintain the diversity from one generation to another. It is done by replacing the older genes with new genes. Figure 9 illustrates the mutation procedure in which values of highlighted genes are randomly changed in offspring 2.

For the evaluation of evacuation algorithms, we have considered two different evacuation scenarios. Each algorithm is applied to different evacuation formations, and the result is analyzed. The explanation of evacuation case studies is presented in section VI and section VII.

VI. Simulation Scenario 1

The first hypothetical scenario consists of ground with six exit gates. In this scenario, the evacuation plans for three types of crowd formations are developed using the above-mentioned evacuation algorithms (SD, ASD, and GA), as shown in the Figure 2, Figure 3 and Figure 4. Each experiment has been performed 20 times, and the mean evacuation time is calculated for each algorithm. To ensure the robustness of our findings, we have conducted a comparative analysis by performing multiple experiments with varying population sizes i.e., 5000, 10000, 15000, and 20000 in the ground region. Table I presents the mean evacuation time obtained by each algorithm with 5000, 10000,15000, and 20000 population sizes for the three evacuation formations.

From the results shown in Table I, we can recognize that the evacuation time of GA is better than the SD and ASD in a Grid-based evacuation formation. For example, the mean

Parent Chromosome 1

E1	E3	E4	E5	E6	E2	E2	E2	E5	E3	E5	E6	E1
R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13

Parent Chromosome 2

E6	E4	E3	E1	E2	E3	E5	E5	E4	E2	E3	E3	E6
R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13

Offspring 1

E1	E3	E4	E5	E6	E2	E2	E5	E5	E2	E3	E3	E6
R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13

Offspring 2

E6	E4	E3	E1	E2	E3	E5	E2	E5	E3	E5	E6	E1
R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13

Fig. 8. Crossover Operation to Generate Two Offspring

E6	E4	E3	E4	E2	E3	E5	E3	E5	E3	E5	E6	E1
R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13

Fig. 9. Mutation of Offspring 2

TABLE I
Mean Evacuation Time (Minutes) for Case Study 1

Crowd Evacuation Formations	Population	Evacuation Algorithms		
		SD	ASD	GA
Grid-based	5000	2.4	2.2	2.1
	10000	3.6	3.3	3.1
	15000	5.2	4.6	4.2
	20000	6.1	5.8	5.3
Sector-based	5000	2.3	2.1	2.0
	10000	3.3	3.2	2.9
	15000	4.5	4.3	3.9
	20000	5.6	5.4	5.1
Pentagonal-based	5000	2.1	2.1	1.9
	10000	3.1	3.1	2.9
	15000	4.2	4.0	3.7
	20000	5.4	5.1	4.8

evacuation time in Grid-based evacuation formation using GA is 2.1 minutes, which is better than SD (2.4 minutes) and ASD (2.2 minutes) for the population of 5000. This difference starts becoming more significant in the presence of a higher population. At the population size of 20000, the GA shows 13.11% and 4.92% improvements against SD and ASD, respectively. The same trends are observed in the Sector-based and Pentagon-based evacuation formations with the varying population. However, based on the result, the Pentagon-based evacuation formation provides better evacuation time as compared to the rest of the evacuation formations. On average, a Pentagon-based evacuation formation using GA offers a 21.31% improvement as compared to the Grid-based evacuation using SD. It validates our stance that evacuation formations and algorithms have a significant effect on overall evacuation time.

VII. Simulation Scenario 2

To further validate our claim and results of the first case study, we have used the extended version of Hajj scenario from

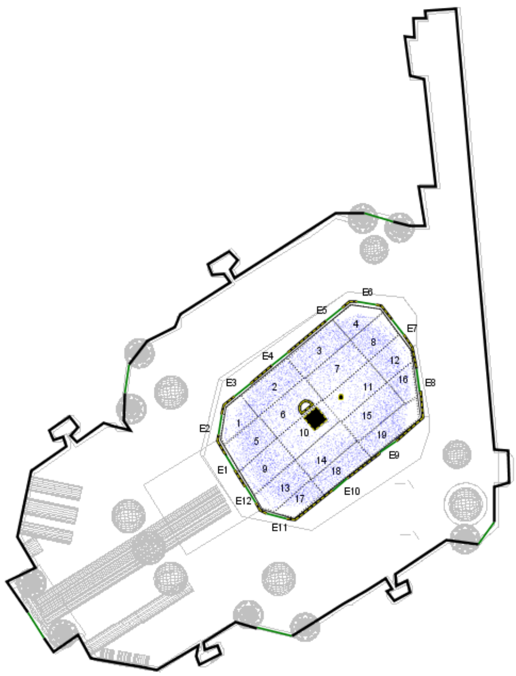

Fig. 10. Case Study 2: Grid-based Evacuation Spatial Environment

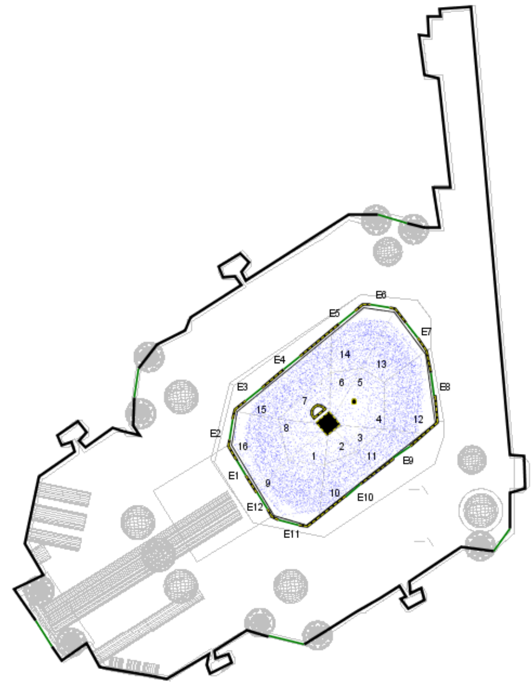

Fig. 11. Case Study 2: Sector-based Evacuation Spatial Environment

[11] as a second case study. Figure 10 depicts the scenario of Grid-based evacuation formation where the courtyard is divided into 19 logical sub-regions (1, 2,..., 19). For Sector-based formation, 16 logical sub-regions (1, 2,..., 16) have been defined as shown in Figure 11. Figure 12 shows Pentagonal-based evacuation scenario with 25 sub-regions. In these figures, black border lines denote walls, and exit/entrance gates (Exit1, Exit2,..., Exit12) are depicted in green. This case study is also simulated using all the three layers of our proposed framework presented in section III.

Simulation results of the first case study and [11] shows that GA is performing better than the other algorithms. Therefore, for this case study, we have only used GA to examine the impact of evacuation formations on the evacuation time. In order to determine the effectiveness of our findings, we have conducted experiments with 5000, 10000, 15000 and 20000 population. From the results mentioned in Table II, we observe that a population of size 5000 took 1.9 minutes with Grid-based formation whereas 1.7 minutes with Sector-based and Pentagonal-based formations to evacuate. However, the Pentagon-based evacuation formation starts showing better mean evacuation time with the increase in population size. These results are in line with the results of the previous case study. It shows that the Pentagon-based evacuation formation offers better results when compared to the Grid-based and Sector-based evacuation formation. However, our research is conducted in a rectangular evacuation area that limits the findings of this study. In future, we will analyze more evacuation formations using different environmental shapes such as circular, square and Pentagon.

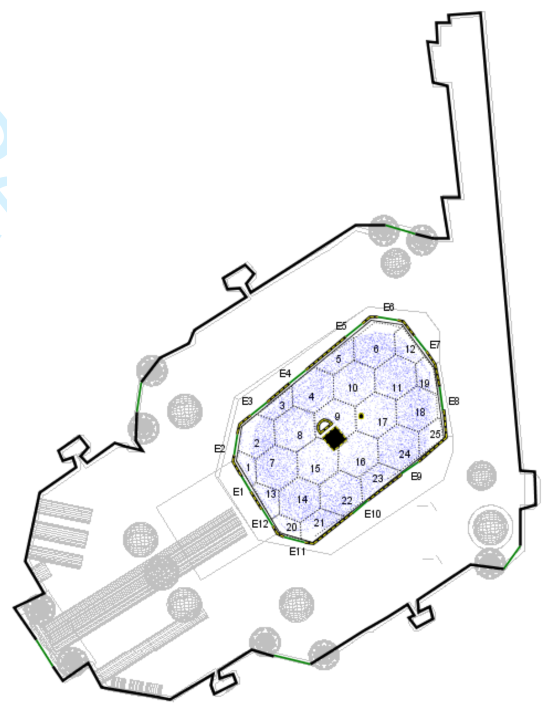

Fig. 12. Case Study 2: Pentagonal-based Evacuation Spatial Environment

VIII. Model Validation

In M&S, validation is an integral part in bug-free model development for precisely-accurate representation of the real system [25] [26]. The goal of the validation process is to produce a model that represents correct behaviour for decision-making purposes and to increase the models credibility to an acceptable level.

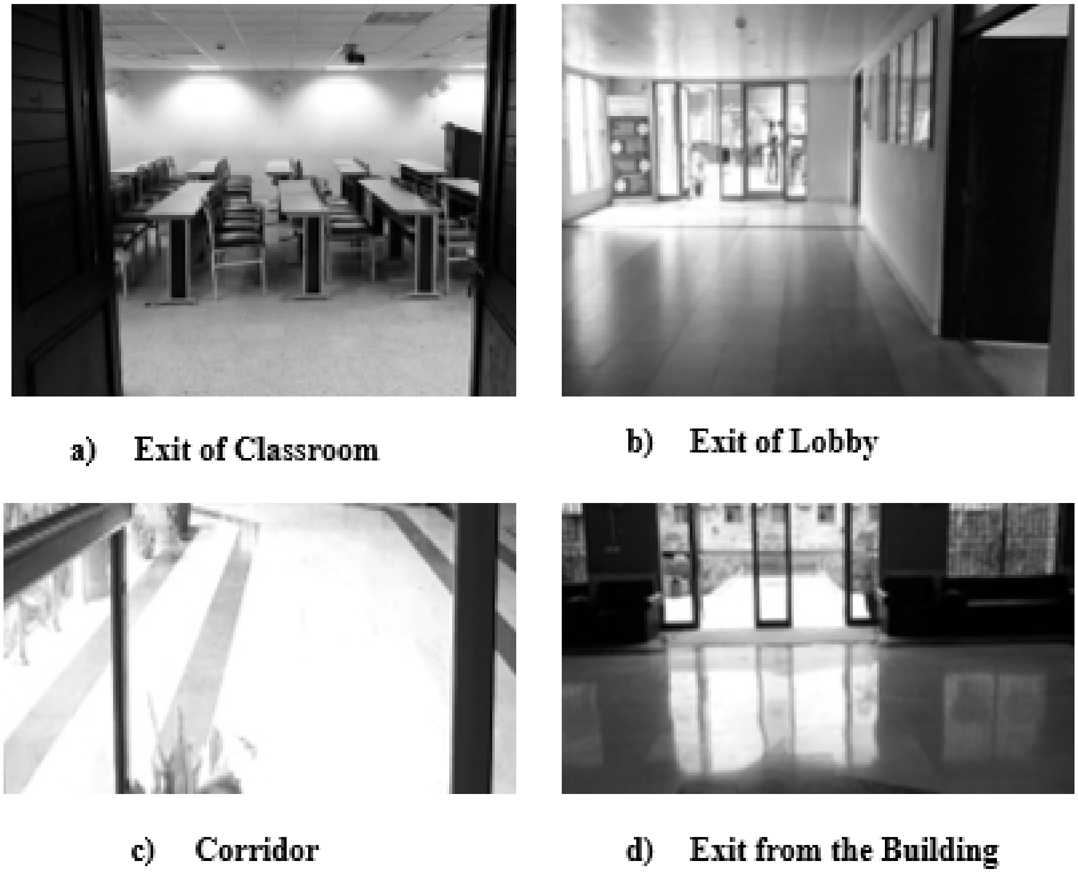

Fig. 13. Local Building Structure

TABLE II
MEAN EVACUATION TIME (MINUTES) FOR CASE STUDY 2

Crowd Evacuation Formations	Population	Evacuation Algorithms
		GA
Grid-based	5000	1.9
	10000	2.4
	15000	3.1
	20000	3.4
Sector-based	5000	1.7
	10000	2.1
	15000	2.5
	20000	3.0
Pentagonal-based	5000	1.7
	10000	2.0
	15000	2.3
	20000	2.8

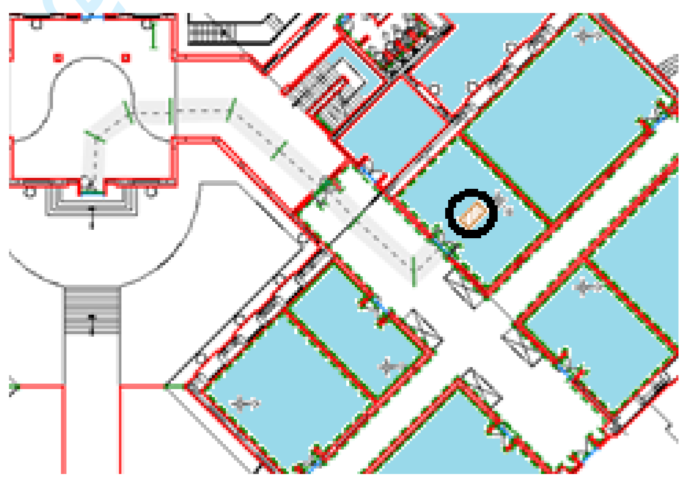

Fig. 14. Local Building Spatial Environment in Anylogic

The validation of simulation models following the real data is a tedious problem. In order to validate our simulation models, we have conducted an evacuation drill in a local building. The architecture of the local building is sketched in Anylogic to perform the simulation experiments. We examine the reliability of our model by comparing the results obtained by the evacuation drill with simulation results. Our building has different classrooms with multiple exits. The building structure is shown in Figure 13. The evacuation drill has been performed using twelve participants with the mean age of twenty four years. At first, the participants are seated in a classroom, as shown in Figure 13a. The evacuation starts with an emergency alarm. After that, all the participants start exiting from the classroom and move towards the lobby exit gate shown in Figure 13b. The time at the start of the alarm is noted for evaluation. From the lobby, the participants move towards the corridor, and finally, evacuate from the building through the main exit gate. At the moment of evacuation completion, the

total evacuation time is noted. This drill is repeated multiple times to record the mean evacuation time of participants.
We have also developed a simulation model based on the real dimensions the local building. Figure 14 shows the building design, which has been imported in Anylogic using CAD feature. Also, space markup elements are used to trace the walls and obstacles. In the simulation, the evacuation is being done from the same room which is used for the real experiment. The position of the participant, before the evacuation, is symbolized with a black circle in Figure 14. The walking speed (1.3 m/s), which is observed during the real-time evacuation, is used in the simulation model. Multiple simulation experiments are performed using this model with twelve participants to obtain the mean evacuation time. Finally, the evacuation time for each participant in both real and simulated experiments is plotted in Figure 15. The first participant takes 15 seconds to evacuate from the building in the real scenario. In contrast, the simulation result slightly deviates with a value of 16 seconds. However, the twelve participants evacuate in 31 seconds in real, whereas simulated evacuation is completed in 32.5 seconds. Therefore, from these observations, we can analyze and conclude that the results of both simulated and real experiments are almost similar with a minor error rate, which validates our model.

IX. Conclusion

Public safety and security is the primary concern of social security and event management companies to avoid unforeseen circumstances in overcrowded areas and mass gatherings. It requires quick evacuation of masses from a hazardous situation. M&S provides economical, risk-free, and harmless way to simulate and pre-plan evacuation of masses from hazardous situations. Crowd evacuation formation and optimization algorithms play a pivotal role in reducing the evacuation time and clogging at exit gates in such scenarios. In this work, we propose an agent-based crowd simulation and analysis framework to examine the effect of different crowd evacuation formations and algorithms on evacuation time.
We have scrutinized Grid-based, Sector-based and Pentagonal-based evacuation formations to study their influence on the evacuation plan using SD, ASD, and GA. Two case studies are presented for assessment of our proposed framework. Results show that the Pentagon-based evacuation formation with GA offers the best evacuation time for multiple scenarios. On average, a Pentagon-based evacuation formation using GA offers a 21.31% improvement as compared to the Grid-based evacuation using SD. Furthermore, we have validated our simulation model by performing real and simulation experiments on a local building to prove the real-time applicability and reliability of our model. Our proposed framework has the potential to be used at public safety and security organizations to benefit and develop safety plans for mass gatherings. Our future work aims to examine the impact of various evacuation formation on different environmental shapes such as circular, square and Pentagon. Also, we are extending our model to the micro level to incorporate the attributes and physical states of each pedestrian. It will help in providing detailed insights into human behaviour during evacuation.

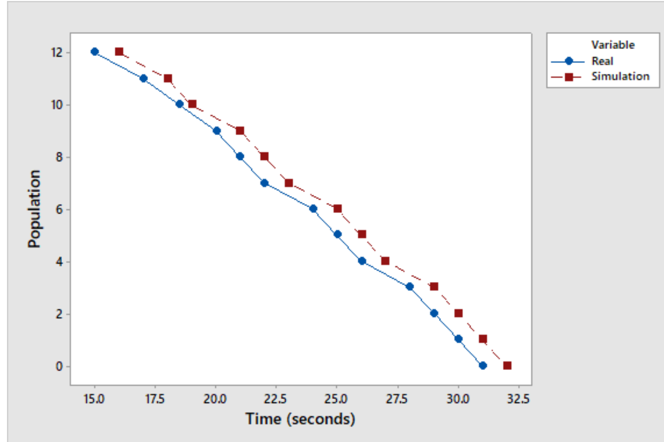

Fig. 15. Real and Simulated Evacuation Results for Multiple Agents

References

[1] K. McGrattan, S. Hostikka, R. McDermott, J. Floyd, C. Weinschenk, K. Overholt, Fire dynamics simulator users guide, NIST special publication 1019 (6) (2013) 1–85.
[2] M. J. Alghazzawi, G. Tashakor, F. Borges, R. Suppi, Crowd evacuation modeling and simulation using care hps, Procedia Computer Science 108 (2017) 2453 – 2457, international Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland. doi:https://doi.org/10.1016/j.procs.2017.05.029.
[3] Y.-P. Chiu, Y.-C. Shiau, Y.-H. Lai, Study on evacuation simulation under crowd-diversion condition, Advances in Mechanical Engineering 10 (7) (2018) 1687814018785092. doi:10.1177/1687814018785092.
[4] M. Kobes, I. Helsloot, B. De Vries, J. G. Post, Building safety and human behaviour in fire: A literature review, Fire Safety Journal 45 (1) (2010) 1–11.
[5] D. Helbing, I. J. Farkas, P. Molnar, T. Vicsek, Simulation of pedestrian crowds in normal and evacuation situations, Pedestrian and evacuation dynamics 21 (2) (2002) 21–58.
[6] J. Drury, C. Cocking, The mass psychology of disasters and emergency evacuations: A research report and implications for practice, University of Sussex Brighton, 2007.
[7] G. Baldini, O. Sallent, S. Subik, C. Wietfeld, The evolution of ict in public safety domains: Challenges and opportunities, International Journal of Disaster Recovery and Business, Continuity 2.
[8] S. P. Simonovic, Systems approach to management of disasters: methods and applications, John Wiley & Sons, 2011.
[9] J. Banks, J. S. Carson, B. L. Nelson, D. M. Nicol, Discrete-event system simulation: Pearson new international edition, Pearson Higher Ed, 2013.
[10] S. Zhou, D. Chen, W. Cai, L. Luo, M. Y. H. Low, F. Tian, V. S.-H. Tay, D. W. S. Ong, B. D. Hamilton, Crowd modeling and simulation technologies, ACM Transactions on Modeling and Computer Simulation (TOMACS) 20 (4) (2010) 20.
[11] I. Mahmood, M. Haris, H. Sarjoughian, 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 Simulation, ACM, 2017, pp. 231–240.
[12] A. Abdelghany, K. Abdelghany, H. Mahmassani, W. Alhalabi, Modeling framework for optimal evacuation of large-scale crowded pedestrian facilities, European Journal of Operational Research 237 (3) (2014) 1105–1118.
[13] J. Zhong, L. Luo, W. Cai, M. Lees, Ea-based evacuation planning using agent-based crowd simulation, in: Proceedings of the Winter Simulation Conference 2014, IEEE, 2014, pp. 395–406.
[14] J. Zhong, W. Cai, L. Luo, Crowd evacuation planning using cartesian genetic programming and agent-based crowd modeling, in: Proceedings of the 2015 Winter Simulation Conference, IEEE Press, 2015, pp. 127–138.
[15] K. Yang, A. H. Shekhar, F. U. Rehman, H. Lahza, S. Basalamah, S. Shekhar, I. Ahmed, A. Ghafoor, Intelligent shelter allotment for emergency evacuation planning: A case study of makkah, IEEE Intelligent Systems 30 (5) (2015) 66–76.

[16] K. Santhosh, P. Mohanty, Soft computation technique based fire evacuation system, Indian Journal of Science and Technology 8 (24) (2015) 1.

[17] Y. Han, H. Liu, Modified social force model based on information transmission toward crowd evacuation simulation, Physica A: Statistical Mechanics and its Applications 469 (2017) 499–509.

[18] M. Haris, I. Mahmood, M. Badar, M. S. Q. Alvi, Modeling safest and optimal emergency evacuation plan for large-scale pedestrians environments, in: 2018 Winter Simulation Conference (WSC), IEEE, 2018, pp. 917–928.

[19] M. Zhou, H. Dong, Y. Zhao, P. Ioannou, Optimization of crowd evacuation with leaders in urban rail transit stations, IEEE Transactions on Intelligent Transportation Systems PP (2019) 1–12. doi:10.1109/TITS.2018.2886415.

[20] Anylogic simulation platform, [Online] Available: https://help.anylogic.com/index.jsp (May 2020).

[21] Anylogic pedestrian library, [Online] Available: https://help.anylogic.com/index.jsp?topic=/com.anylogic.help/html/_Pedestrian/reference/PL.html (May 2020).

[22] D. Helbing, P. Molnár, Social force model for pedestrian dynamics, Phys. Rev. E 51 (1995) 4282–4286. doi:10.1103/PhysRevE.51.4282.

[23] Triangular distribution, [Online] Available: https://help.anylogic.com/index.jsp?topic=/com.anylogic.help/html/functions/triangular.html (May 2020).

[24] M. Mitchell, An introduction to genetic algorithms, MIT press, 1998.

[25] O. Balci, et al., Verification, validation and accreditation of simulation models, in: Winter Simulation Conference, Vol. 1997, 1997, pp. 135–141.

[26] R. G. Sargent, Verification and validation of simulation models, in: Proceedings of the 2010 Winter Simulation Conference, IEEE, 2010, pp. 166–183.