Using Heuristic Search with Pedestrian Simulation Statistics to Find Feasible Spatial Layout Design Elements

Fadratul Hafinaz Hassan[*1], Stephen Swift[2], Allan Tucker[3]

[*1]School of Computer Science, Universiti Sains Malaysia,

Penang, Malaysia

[2, 3]Department of Information System, Computing & Mathematics, Brunel University,

London, United Kingdom

[*1]fadratul@cs.usm.my; [2]stephen.swift@brunel.ac.uk; [3]allan.tucker@brunel.ac.uk

Abstract-A spatial layout design must consider not only ease of movement, but also safety in a panic situation, such as an emergency evacuation in a theatre, stadium or hospital. Using pedestrian simulation statistics, the activity of crowds can be used to study the consequences of different spatial layouts. A cellular automaton has been proposed to model pedestrian simulation so that pedestrian flows can be explored at a microscopic level. We examined four-way pedestrian flow statistics generated from feasible seating layout solutions using heuristic search techniques (hill climbing, simulated annealing and genetic algorithm style operators). The technique has shown promising results in identifying useful characteristics of spatial layout and sources of the pedestrian clogging phenomenon.

Keywords- Cellular Automata; Simulated Annealing; Hill Climbing; Genetic Algorithm; Pedestrian Simulation Statistics; Spatial Layout

I. INTRODUCTION

Pedestrian simulations on egress plans provide in-depth analysis of pedestrian movements during evacuation processes, but do not offer any consideration for the distribution of objects or obstacles inside the floor layout itself. The smooth flow of pedestrians is formed by a feasible allocation of obstacles in a layout. Thus, in architectural planning, it is important to carefully consider the position of obstacles in a layout as well as the location of exits. There are a number of projects that have used pedestrian simulation to test pedestrian behaviors for airport terminal floor plans, stadium plans, shopping malls and galleries [1-5]. However, many pedestrian simulation studies on evacuation plans are often focused on pedestrian behavior and do not assess the interaction with obstacles inside the floor plan. If we could analyze the relationship of different pedestrians with the obstacles, then it would allow us to configure a better distribution of obstacles in a layout. For example, pedestrian statistics generated from the simulations could be investigated further and show us where the collision points are in a full grid layout. In this way, we can identify 'bad' regions and predict feasible layouts. If we could also study different types of pedestrian flow at a microscopic level of the simulation, then it would give us greater control on different sets of pedestrians in and out of the layout. For instance, there are two different groups of pedestrian in the hospital evacuation; medical staff and patient.

Earlier works either produce an optimal pedestrian flow or an optimal spatial layout design, not automatically optimize both of these simultaneously. Furthermore, the generated spatial layout is not an automatically discovered novel layout, rather they are chosen from the existing floor plan database. Hence, this investigation is a useful reference for other practitioners and researchers interested in the application of heuristic search methods that automatically find a viable architectural layout design that optimizes pedestrian flow.

The work presented in this paper is a proof of concept study that explores a novel approach to learning optimized layouts of egress plans using simulated annealing (SA) and hill climbing (HC) with updates that involve simple genetic algorithm (GA) operators. Previously, we have used these algorithms to simulate a 10-by-10 grid size with two types of pedestrian moving inside the space; one moving to the left and one moving to the right [6, 7]. Experiments from our previous study have shown that promising results for simulated annealing and genetic algorithm operator algorithms in pedestrian simulation modeling. We have demonstrated that these algorithms were able to automatically find adequate solutions to spatial layout design problem when incorporated with the pedestrian simulator.

Here we use statistics generated from a cellular automata simulation of pedestrians to score the current layout. We carry out a substantial evaluation on a number of different real-world scenarios and enhance the pedestrian behavior by allowing each type of pedestrian to randomly change their direction throughout the simulation. In this study, we used these algorithms to simulate a 20-by-20 grid size with four types of pedestrian moving inside the space; one moving to the left, one moving to the right, one moving up-to-down and another moving down-to-up based upon the cellular automata model of Yue et al. [8]. This also involves static objects (walls), distributed on every side of the rooms, and non-static objects randomly distributed. We

have also explored the use of a heat map operator during our simulation in order to identify the "clogging" areas. It should be noted that the aim of the experiments was not necessarily to find the best layout, but rather to find feasible characteristics of good layouts and to identify 'bad' regions using statistics generated from cellular automata pedestrian simulations.

This paper is organized as follows. We discuss the research motivation and justify the cellular automata (CA) model of pedestrian simulation. Then we briefly introduce Hill Climbing (HC), Simulated Annealing (SA) and the extended SA using Genetic Algorithm-style Operator (SA-GAO), followed by the proposed move operator, fitness function and heat map operator. The next section presents experimental results on a classroom layout, a stadium and a theatre. Finally, we present our conclusions.

II. MOTIVATION

This study in this paper concerns empirical simulation with the aim of establishing those design elements of spatial layout that increase the efficiency and safety of pedestrian movement. A 'good' geometrical layout not only ensures safety but encourages pedestrians to move in their comfort zone whilst maximizing pedestrian flow. Spatial layout with 'bad' design elements reduce the pedestrian flow rate and risk increasing queuing patterns that lead to congestion, for example in mass events such as football matches and music concerts.

Bentley, Gero and Liggett [9-11] state that spatial layout design is practically important in architectural design because it is the basis of the development of most designs and is concerned with finding the optimal locations for a set of interrelated objects. Spatial layout deals with the design of two dimensional layouts and is one of the most interesting and complex of the formal architectural design problems, i.e. finding a satisfactory space arrangement with regards to objective requirements such as pedestrian flow. It is a complex process, with huge numbers of (often conflicting) objectives, many constraints and often thousands of parameters. But the most difficult aspects of these design problems are pedestrian, thus to find a feasible layout that will successfully generate a smooth flow of pedestrian and ensures safety is vital in this research.

Firstly, a brief summary of previous work in pedestrian simulations will be given before proceeding with the presented research. A number of work have implemented pedestrian simulation model to simulate pedestrian flows under different domains of special events like soccer games, concert or carnival event, shopping mall, passenger ship, high rise building, airport terminal, stadium and galleries [1-5]. They used this model to address congestion and safety issues in the carnival event; to describe pedestrian behavior in their route-choice, to evaluate pedestrian performance indicators such as the distribution of pedestrian across the shopping mall, the ease of navigation, etc. Others manage to establish guidelines on the safety-assessment of passenger ships or high rise building during emergency evacuation, to study the pedestrian's choice of action, such as moving ahead, stopping to wait, position exchange, lane switching, back stepping, etc. and to learn the mechanism of how to avoid collisions. Hence, based on the previous works, the model of pedestrian movement was identified as a useful method to help designers to understand the relationship between spatial layout and human behavior.

For this assessment, it is important to use a pedestrian simulation model that captures realistic observations. We will show that the CA model is capable of reproducing various important phenomena in pedestrian flow. By coupling this model with heuristic search techniques, we can use resultant pedestrian flow statistics to score a 'good' design layout characterized by a smooth flow of pedestrians. We will discuss empirical observations and present experimental results on three different layouts. It is, however, reasonable to start with a discussion of the CA model and a brief introduction to heuristic search techniques (HC, SA and SA-GAO).

III. CELLULAR AUTOMATA MODEL

Cellular Automata (CA) have been adopted widely and are regarded as effective tools in the study of pedestrian dynamics [12-19]. In this case the local movements of the pedestrian are modeled with a matrix of preferences which contains the probabilities for a move, related to the preferred walking direction and speed, to the adjacent directions [13]. Schadschneider [20] introduces the interesting concept of floor field to model the long-ranged forces. This field has its own dynamic (diffusion and decay), is modified by pedestrians and in turn modifies the matrix of preferences, simulating interactions between individuals and the geometry of the system. All the agent-based models are also microscopic models and are based on some elementary form of intelligence for each agent (attempts to provide a vision and/or cognition capabilities). Simple behavioral rules are implemented (turning directions, obstacle avoidance) in order to reproduce more complex collective phenomena [21].

The simple CA local rules can create an approximation of actual individual behavior; this is sufficient as a model for this microscopic pedestrian study. Although the microscopic pedestrian study does not replace the macroscopic one, it considers a more detailed analysis for design and pedestrian interaction. One of the most important factors concerning CA models is their suitability for large scale computer simulations [22]. For example, it is possible to simulate realistic traffic networks faster than real time. They are easily implemented on digital computers, and compared to the differential equation-based micro-simulation models, run exceedingly fast. The idea behind our approach is to use CA to simulate pedestrian movement in typical layouts of buildings and public spaces and use the resulting statistics to score a heuristic search for "good" spatial design. Fig. 1 below shows how we simplify the classroom layout [23] in our CA model. The simulation layout is slightly different from the original layout. Note that, the number of doors is increased and the position of the left wall is shifted to encourage flow

through the room. The walls are red and seats/obstacles are blue, the white cells can be occupied by a pedestrian. The arrow shapes are all pedestrians in which red arrows represent pedestrian moving to the right and up while the black arrows are the pedestrian moving to the left and down. The pedestrians share a same movement's goal that is to find the escape route of the room.
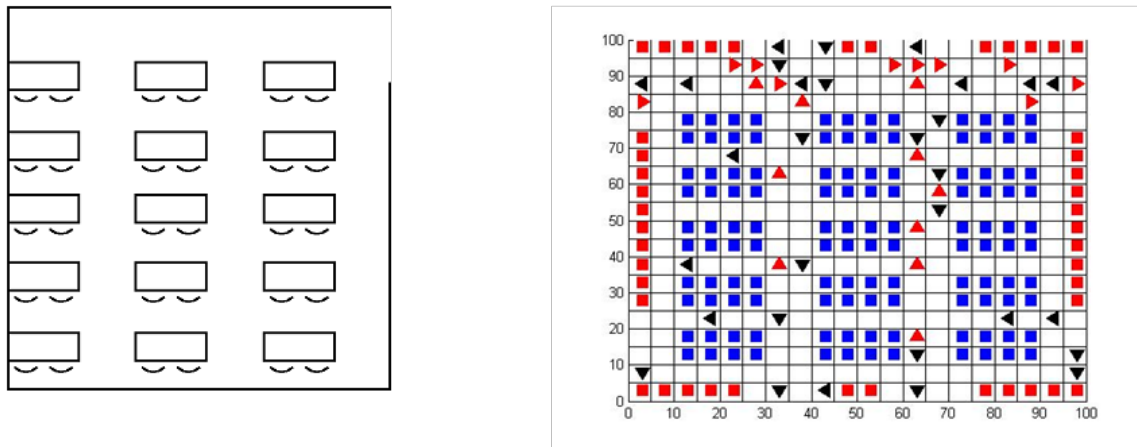

Fig. 1 Distribution of seats in a classroom. Transformation of the classroom into a grid of cells

The methods analyzed in this paper have been implemented using a pedestrian movement model developed by Yue et al. [8]. They proposed two pedestrian movement models on a square lattice for small systems based on cellular automata (CA), i.e. a two-way and four-way pedestrian flow. They introduced a technique to tactically simplify the process into the interaction of four dynamic parameters, which can reflect the pedestrian's judgment of the surrounding conditions and decide the pedestrian's choice of action, such as moving ahead, stopping to wait, position exchange, lane switching, back stepping, etc. In this paper, the two-way pedestrian flow system is simulated and studied using the Dynamic Parameters Model from Yue et al. [8] to consider direction split and pedestrians' walking preference.

In this model, pedestrians choose to wait or move according to the corresponding transition payoff based on four parameters:

- Direction-parameter indicating the cell's degree of approximation to the pedestrian destination;
- Empty-parameter indicating whether the cell is occupied or empty;
- Forward-parameter describing the proportion of empty cells in the field ahead of his or her target position;
- Category-parameter describing the proportion of the number of empty cells and pedestrians homogenous with the subject in his or her direction of destination in the field around his or her target position.

IV. METHODOLOGY

The experiments involved applying HC, SA and SA-GAO to solve the spatial layout problem. It was not feasible to have a full GA implementation due to the very complex fitness function involving several pedestrian simulations.

A. *Hill Climbing, Simulated Annealing and SA Genetic Algorithm Operator*

HC is a comparatively simple local search algorithm that works to improve a single candidate solution, starting from a randomly selected point. From that position, the neighboring search space is evaluated. If a fitter candidate solution is found, the search moves to that point. If no better solution is found in the vicinity, the algorithm terminates. The main disadvantage of using HC is that it often gets stuck in local maxima during the search [24].

SA is an extension of HC, reducing the chance of converging at local optima by allowing moves to inferior solutions under the control of a temperature function. This solution is followed if a better successive one can be found. Otherwise, it accepts a worse state with a certain probability that decreases with temperature. This is less extreme than taking randomized HC each time but still has the ability of escaping from the possible trap of local maximum/minimum or plateau [24].

The pseudo code for our implementation of this approach is listed below.

In order to set this as an HC, the starting temperature is set to zero. Note that the fitness here uses the statistics generated from 10 repeated runs of the CA pedestrian simulation. This is in order to ensure that one simulation does not result in a 'lucky' fitness score for one layout based upon the starting positions of pedestrians. A parallel update method is implemented in this simulation. It is an open boundary condition except where pedestrian blocked.

Finally, we extended our work by using GA-style operators [6, 7]. A full GA implementation was not feasible due to the very costly fitness function involving several pedestrian simulations. The initial 'parents' were selected from the best solutions generated (selections were made based on more consistent fitness values with a fitness higher than the original fitness) from a number of SA experiments. We experimented with different styles of combination for two 'parents' that more or less acted like a uniform crossover. The pseudo code for our implementation of this approach is listed below, Figs. 2 and 3.

Algorithm 1 Hill Climb and Simulated Annealing

Input:	Set number of iterations, *iteration*, random starting layout, *startrep,*
	starting temperature, *temperature*
	oldrep = startrep;
	Apply 10 pedestrian simulations to generate statistics, *stats*
	fit = fitness(*stats*)
	bestfit = fit
for	*loop* = 1:*iteration*
	rep=oldrep;
	Apply move operator to *rep*
	Apply 10 pedestrian simulations to *stats*
	newfit = fitness(*stats*);
	dscore = *newfit - fit*
	if ((*bestfit < newfit*) OR (rand(0,1) < e$^{(dscore/temperature)}$)
	bestfit = newfit
	oldrep = rep;
	else
	rep = oldrep;
	end if
	temperature = temperature * 0.9
end for	
Output:	*rep*

Fig. 2 Pseudo-code for the hill climb and simulated annealing

---

Algorithm 2 Genetic Algorithm Style Operator

Input:	Set selected starting layout with highest fitness, *rep*, and number of
	genes/objects, *n*
for	*loop* = 1:100
	Choose two parents randomly, *parent1*, *parent2* from list of *rep*
	Generate children with equal ratio of genes, *n* from both parents
	pars=[parent1;parent2]
	mapping=1+(rand(1,*n*)>0.5)
	child1 = [];
	child2 = [];
	for *i* = 1:*n*
	child1((*i**3-2):((*i**3)-2+2)) = *pars*(*mapping*(*i*),(*i**3-2):((*i**3)-2+2));
	child2((*i**3-2):((*i**3)-2+2)) = *pars*(((*mapping*(*i*)==1)+1),(*i**3-2):((*i**3)-2+2));
	end
end for	
Output:	*child1*, *child2*

Fig. 3 Pseudo-code for the genetic algorithm style operator

In our implementation of GA-style operator, the 'child' generated from randomly chosen 'parents' genes' with equal ratio (50% genes from 'parent1' and 50% genes from 'parent2'). Among the numerous crossover operators in practice, a uniform crossover is chosen in this proposed method for its advantages over the other forms. Uniform crossover is global and less biased when compared to that of standard and one point crossover. Uniform crossover does not select a set of crossover points. It simply considers each bit position of the two parents, and swaps the two bits with a probability of 50% [25-27].

B. Fitness Function

The fitness function is calculated based on the statistics that are generated using the pedestrian simulation. The statistics take the form of a 3x3 matrix, *leftstats*, representing the sum of decisions for left-moving pedestrians and a similar decision matrix for right-moving pedestrians; *rightstats*, up-moving pedestrians; *upstats*, and down-moving pedestrians; *downstats*. Therefore, the middle cell in each grid represents how many times the pedestrians decided to stay in the same cell as the last time. As we wanted to encourage free flow, we wanted to increase the fitness for layouts that resulted in many cases of left-moving pedestrians moving left, right-moving pedestrians moving right, up-moving pedestrians moving up and down-moving pedestrians moving down, whilst penalizing the fitness of any decisions where the left-moving pedestrians moved right, up-moving pedestrians moved down and vice versa. For example, consider the four *stats* matrices for left, right, up and down pedestrians in figure below. It is clear that the *leftstats* reflect a 'good' result as the pedestrians have generally moved in the desired direction more often whereas for *rightstats*, *upstats* and *downstats* this is not the case.

In general, we wish to maximise the first column in *leftstats*, the third column in *rightstats*, the first row in *upstats* and the third row in *downstats* whilst minimising the other statistics (shaded in the example, Fig. 4). Therefore, we use the following fitness functions:
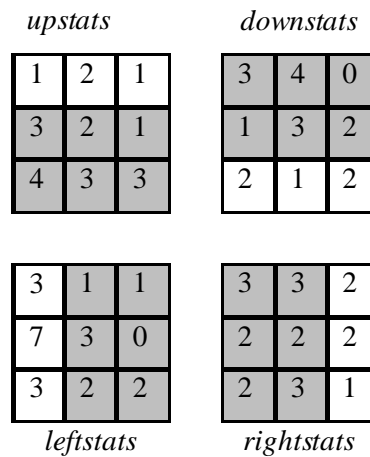

Fig. 4 Up, down, left and right statistics – described as 3x3 matrix

$$f_1 = \sum_{i=1}^{3}(a_{i3}) - \sum_{i=1}^{3}(a_{i1}) \tag{1}$$

$$f_2 = \sum_{i=1}^{3}(b_{i1}) - \sum_{i=1}^{3}(b_{i3}) \tag{2}$$

$$f_3 = \sum_{j=1}^{3}(c_{1j}) - \sum_{j=1}^{3}(c_{3j}) \tag{3}$$

$$f_4 = \sum_{j=1}^{3}(d_{3j}) - \sum_{j=1}^{3}(d_{1j}) \tag{4}$$

$$f(x) = \sum(f_1 + f_2 + f_3 + f_4) \tag{5}$$

Where:

f_1, f_2, f_3 and f_4 is the total fitness for right, left, up and down pedestrians;

f(x) is the fitness function;

a, *b*, *c* and *d* are *rightstats*, *leftstats*, *upstats* and *downstats*, respectively;

i is the number of row;

j is the number of column.

Higher fitnesses should reflect simulations whereby pedestrians have moved in the direction that they wish more often. For example, consider the four *stats* matrices in Fig. 2, the total fitness for right is (-2), left is 10, up is (-6) and down is (-2). It is clear that the left pedestrians reflect a 'good' result as the pedestrians have generally moved in the desired direction more often whereas for right, up and down pedestrians this is not the case.

C. Move Operator

The move operator is an enhancement from the previous study [6, 7] and takes into account the size of the simulation grid, randomly moving one object a fraction of this distance (determined by the parameter *changedegree*). The result of the move is then checked to see if the new coordinates are within the bounds of the grid and do not result in the object overlapping with others. In this update, we also checked that the result did not overlap with the static object (wall). The operator is defined below, where *unidrnd(min,max)* is a uniform discrete random number generator with the limits of *min* and *max*. The pseudo code for move operator is listed in the Fig. 5.

Algorithm 3 Move operator	
Input:	Set the size of simulation grid, *W*, size of object, *sizobj*, size of static object, *staticsizobj*
	Set the degree of change to make based upon fraction of the grid size:
	changedegree = W/2;
	oldx, oldy : current x-coordinate and y-coordinate of object
	changedegree : degree of change
	xchange, ychange : degree of change x-coordinate and y-coordinate
	newx, newy : new x-coordinate and y-coordinate of object
	Choose a random object in the grid, *i*
	[*oldx, oldy*] = current x and y coordinates of object *i*
	xchange = unidrnd(-changedegree/2, changedegree/2)
	ychange = unidrnd(-changedegree/2, changedegree/2)
if	((*oldy+ychange*) and (*oldx+xchange*) is within grid boundary AND new object position does not overlap another object taking into account *sizobj* AND does not overlap with static object taking into account *staticsizobj*)
	newx = oldx+xchange;
	newy = oldy+ychange;
end if	
Move object *i* to position [*newx, newy*]	
Output:	*newx newy*

Fig. 5 Pseudo-code for the move operator

D. Heat Map

The heat map is calculated based on how many times pedestrians could not move in their intended direction. We implement the same statistics generated from pedestrian simulation to calculate fitness function. Initially, the heat map is set up with all zeros inside and the same grid size as simulation grid, *W*. Then, whenever a pedestrian could not move in his/her intended direction, the position the pedestrian was in is increased by one. For example, consider the *leftstats* matrices for left pedestrians as in the Fig. 6. It is clear that the *leftstats* reflect a 'bad' result as the pedestrians have generally not moved in their desired direction.

0	1	3
1	3	7
0	2	3

leftstats

Fig. 6 Left statistics - described as 3x3 matrix for heat map calculation

In general, if the value in the third column of *leftstats* has a bigger value than the first column of *leftstats*, then the heat map matrix is increased to one in the same position with pedestrian matrix. In this way we will be counting all of the cells where pedestrians are blocked. The pseudo code for our implementation of this approach is listed in the Fig. 7.

Algorithm 4 Heat Map operator	
Input:	Set the size of simulation grid, *W,* number of iterations, *iteration*, *leftstats1* : first column of left pedestrian statistics *leftstats3* : third column of left pedestrian statistics *starthm* : starting heatmap
	starthm = zeros(W);
for	*loop* = 1:*iteration* *hm = starthm;* **if** (*leftstats3 > leftstats1*) *hm = starthm + 1;* **end if**
end for	
Output:	*hm*

Fig. 7 Pseudo-code for the heat map operator

V. RESULT

We present the experiment result obtained by the procedure described in the Methodology section. The first experiments involved running HC, SA and SA-GAO on the problem of trying to arrange 15 pre-defined objects in a 20-by-20 grid (classroom layout) with 5 'left' pedestrians, 5 'right' pedestrians, 5 'up' pedestrians and 5 'down' pedestrians. In this experiment, we increase the simulation size from 10-by-10 grid [6, 7] to 20-by-20 grid size. The length of simulation took approximately two hours for one simulation in a 20-by-20 grid size. SA-GAO was run with one hundred random combinations of mappings. Eight 'parents' were selected from the consistent resulting solutions of previous SA experiments where each of their fitness values is higher than the original fitness (43.434). Each type of pedestrian randomly changes his/her direction during simulation. The second experiments involved running SA on the problem of trying to arrange 12 pre-defined seats/objects in a 20-by-20 grid (theatre layout). Finally, the third experiments involved running SA on the problem of trying to arrange 13 pre-defined seats/objects in a 20-by-20 grid (stadium layout). Temperatures starting for SA were set at 0.5, 0.7 and 0.9, respectively. Each algorithm was run 10 times with different pedestrian starting points and the learning curves were inspected. The final fitnesses and quality of the layouts were then investigated. Finally, some inspection of sample simulations on the final layouts and their final heat maps was carried out in order to find any interesting characteristics.

A. Summary Statistics

In order to obtain statistical estimates on the significance of the results, we ran this process several (10) times for each method (HC, SA and SA-GAO). Thus, final fitnesses were obtained for each method. Then paired *t*-test was used to compare the relative performances of the differences approaches and measure whether the differences between their mean final fitnesses are statistically significant.

TABLE 1 STATISTICAL SIGNIFICANCE PERFORMANCE COMPARISON BETWEEN METHODS

Method 1	*Method 2*	*Paired t-test*	
		p-value	eta squared statistic
SA-GAO	SA09	0.000	0.790
SA-GAO	SA07	0.000	0.912
SA-GAO	SA05	0.000	0.880
SA-GAO	HC	0.000	0.854

In order to obtain a *p*-value indicating the statistical significance of these findings, a paired *t*-test was used, the results of which are shown in Table 1. The table also shows the effect size for paired-samples *t*-test through the eta squared values. Method 1 (SA-GAO) always outperform Method 2 (SA09, SA07, SA05 and HC) with $p<.0005$ and eta squared ($>.14$). In general, we found that SA-GAO treats combinations of two existing solutions as being "near", making the "children" share the properties of their parents, so that a child of two good solutions is probably better than a random solution as in HC and SA.
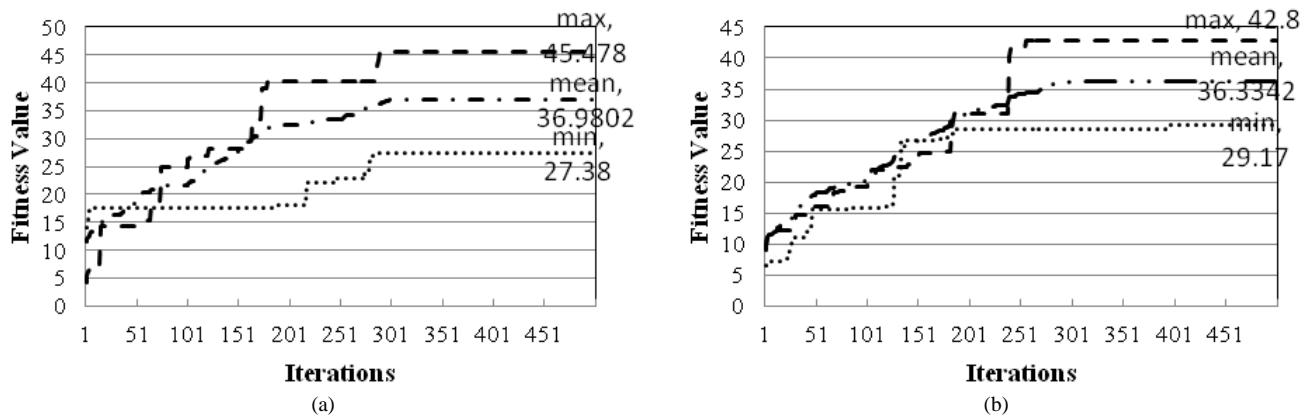
Fig. 8 Max, min and mean learning curves for (a) SA0.9 and (b) HC from classroom layout

The graphs in Figs. 8 (a) and 8(b) show the best, worst and mean learning curves for the SA0.9 and HC algorithm over 10 reruns. The total iteration for every learning curve is 500 iterations. These curves characterize the typical SA0.9 with a noisy search at high temperatures in the early phases followed by smoother learning in the later stages. It seems the SA0.9 result in more diverse final fitnesses. This could be due to the fact that the fitness landscape is very noisy and difficult to negotiate, resulting in many local optima. However, the HC curves in Fig. 8(b) show far less variations in final fitness in the first half of phases which is surprising. It is clear that the SA0.9 does indeed escape sub-optimal local peaks in the fitness on some occasions with the maximum and mean final fitness being higher than HC. Clearly this is likely to be due to HC techniques exploiting the best available solution for possible improvement but neglecting exploring a large portion of the search space.

B. Comparing Original Layout with Higher Fitness Layout

In this section, we compare the final layout of the original seating plans [23] in the classroom, theatre and stadium with improved layouts generated from the simulations. The improved layouts selected from higher fitness than the original layouts produced from experiments running on the HC, SA and SA-GAO. The final layout with the red square symbols are the permanent walls, and the blue squares in Figs. 9-22 represent the final positions of random desks. The black and red arrows represented left, right, up and down pedestrians moving in each way. For the color heat map, the darker tone of blue cells shows smaller values of congestion, the lighter tone of blue cells shows bigger values, the yellow and red cells show the biggest values of congestion.

1) Classroom Layout:

The layout of Fig. 9 is based on the original distribution of seats in classrooms [23]. This configuration produced clogging phenomena near the exit, sharing lane and T-shaped channel. It shows in the final layout where all pedestrians moved and clogged near the exit on the lower left of the room. Clogging effect can be seen clearly on the bottom left of the heat map, where there are red and yellow cells. Exits on the top and bottom walls are not far enough apart; an average exit throughput becomes significantly less due to a collective slow-down that emerges from pedestrians crossing each other's paths. This disruptive interference effect led the 'up' and 'down' pedestrians to share the same lane. Notice there are two clear 'paths' with a lighter tone of blue cells in the heat map layout, showing that a disruptive interference effect leading to pedestrians bumping and blocking each other. It is observed that a clogging also occurs at the junction of the T-shaped channel in the merging flow from the up and down pedestrians' route to the left and right pedestrians' route. Notice a small number of jamming patterns in yellow in the bottom part of the heat map layout, showing the clogging effect where different types of pedestrian merge at the T-shaped channel. However, the newly generated layout from the simulations is not a perfect layout. Although the improved layout as can be found in the experiment using classroom layout produced better fitnesses with fewer congestion spots, the overall configurations of desks inside the layout are not practical. Some layouts are unrealistic due to the open creative approach - we did not want to add too many restrictions.
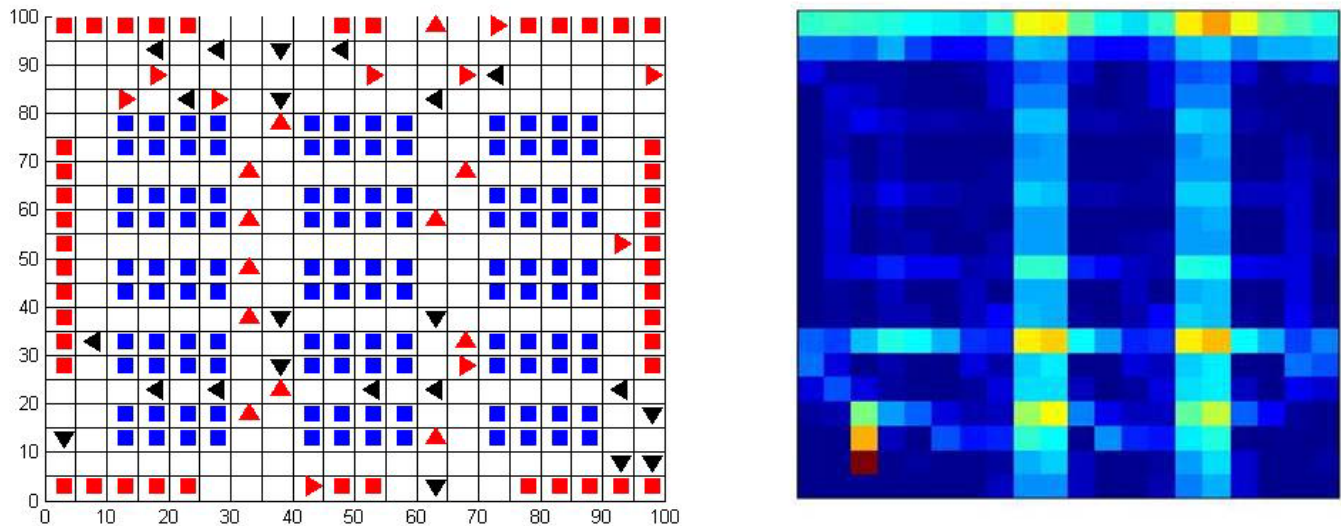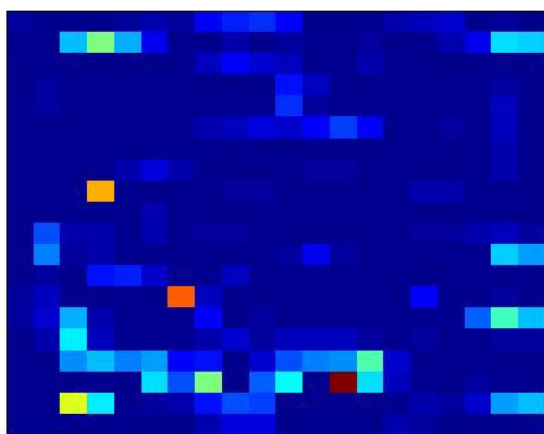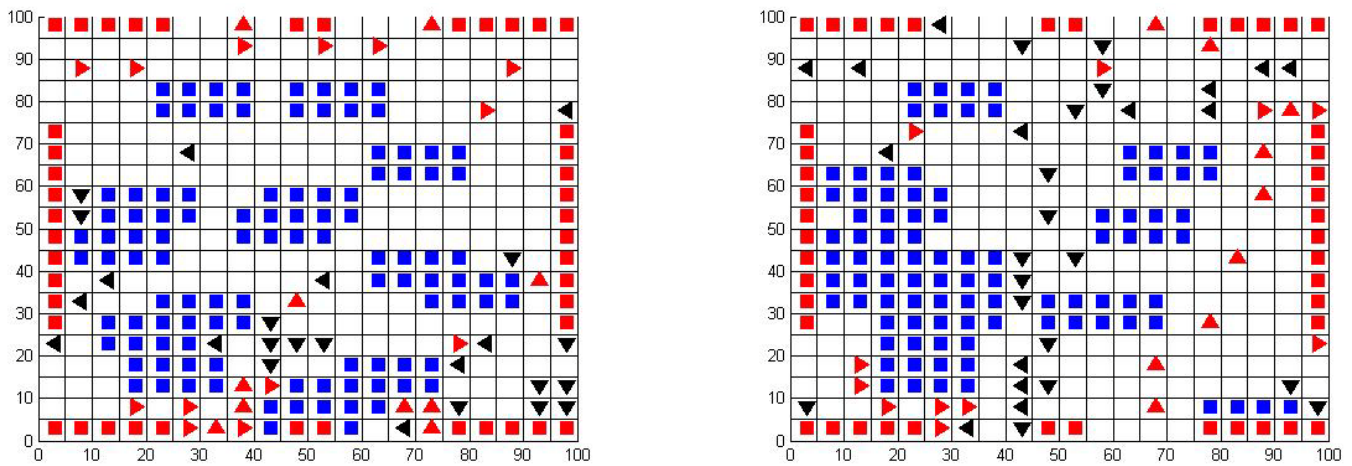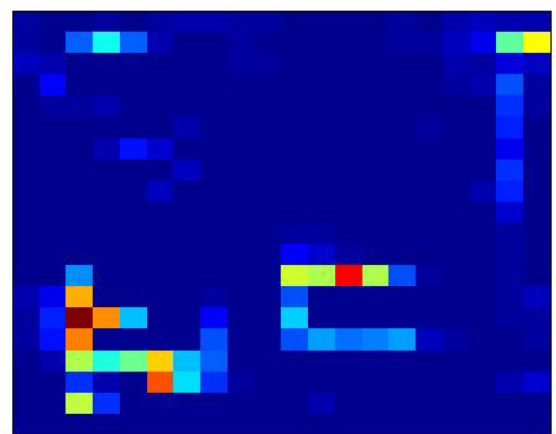
Fig. 9 Original classroom layout with final fitness of 43.434 and heat map


(a)	(b)

Fig. 10 Classroom final layouts using SA-GAO and their heat maps, (a) 'child2' with fitness of 56.876 and mapping of '112121121222111' represents the highest fitness of all children, and (b) the second highest fitness from 'chi1d1' with fitness of 53.750 and mapping of '221122211222112'

The higher the fitness, the better the layout produced, as in Fig. 10 (a) and (b). The layouts were generated from SA-GAO simulations. There is no further disruptive interference phenomenon and there are generally fewer jamming spots, as shown on the both heat maps. A group of objects in the centre of the final layouts may separate the two exit routes for 'up' and 'down' pedestrians far enough to be able to dismiss the disruptive interference effect. Most of the objects are shifted sideways to the

walls, creating bigger major lanes in the upper side of the layout and also creating free spaces near the exits. Although it has better fitness than the original layout, clogging regions still exist in this newly generated layout. Jamming phenomena can be identified at the largest densities of the layouts, where most objects were arranged. This is shown in the same region of its heat map where lots of red, orange and yellow cells exist. The jamming might also be because of instances of counter flows, where different groups of pedestrians mutually block each other. It is not possible to turn around and move back, especially when there is a large flow of people within a dense region. Although the newly generated layout from SA-GAO simulations is not completely an ideal layout, nonetheless some of the 'bad' characteristics in original layouts were avoided. Overall, the jamming spots are less in the new heat map layouts compared to the original one.
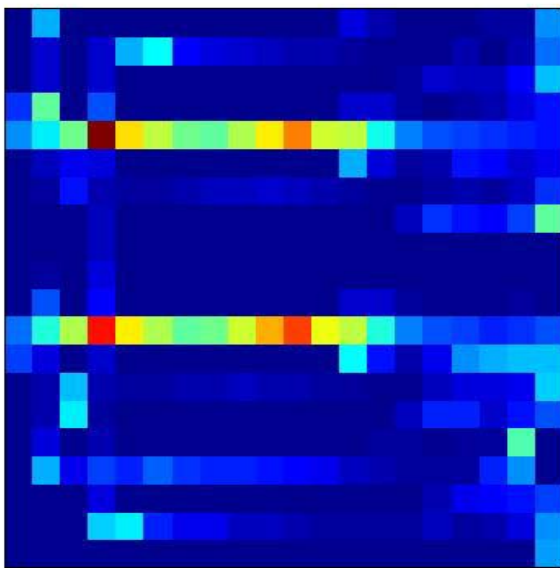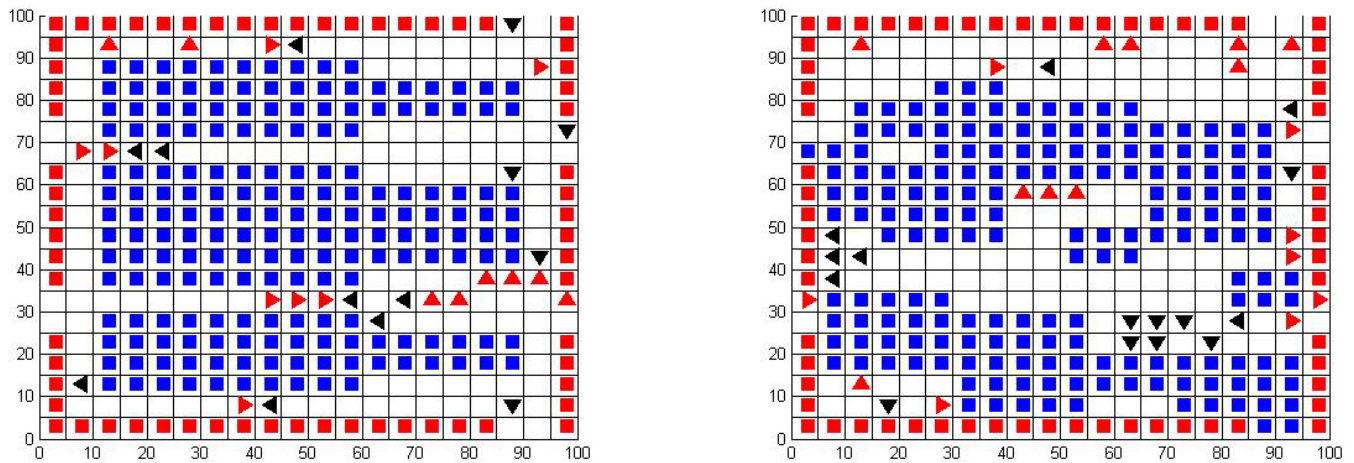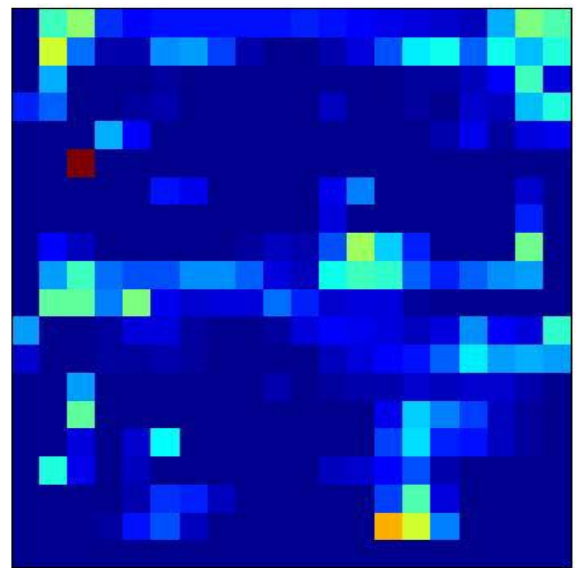
(a)   (b)

Fig. 11 Final layout and heat map of (a) original theatre layout with final fitness of 4.286 and (b) improved layout with the highest fitness from SA0.9 algorithm with final fitness of 12.838

2) Theatre Layout:

Fig. 11(a) is the final layout of the original theatre seating plan [23] with a final fitness of 4.286. The seating arrangements create a one cell size of paths that encircle every group of seats. Apparently, the 'up' and 'down' pedestrians have only two exit routes compared to the 'left' and 'right' pedestrians, who have twice as many exit routes. Surprisingly, there are less jamming spots in the 'up' and 'down' exit paths. However, there are clear jamming spots along the middle corridors of the 'left' and 'right' exit paths, as shown in its heat map. Notice that the two middle corridors, ranging from left to right, are the most used by pedestrians compared to other paths. This may be because these two paths are nearer to the exits, thus pedestrians display a tendency to use the shorter path. The clogging might also occur because the path is only one cell wide to reduce collisions between pedestrians. Notice also the jamming pattern worsens at the junction of the T-shaped channel in the merging flow from the branch channel to the main channel.

In the next layout, Fig. 11(b) is the final seating layout of the same theatre plan generated from the SA0.9 algorithms with the highest fitness of 12.838 from ten simulations. The final layout looks similar to the original layout with long corridors from left to right of the layout but with a wider and zig-zag shaped path. The wider and zig-zag path created along the corridor which may help to reduce collisions in panic situations. The newly generated layout cannot be totally guaranteed as being 'perfect' with a practical seating arrangement layout despite having the highest fitness. Notice that some of the exits in the new layout have been blocked. Nevertheless, the 'bad' element in the original layout – clogging along the corridors - was reduced in the new layout. Notice that, in its heat map in Fig. 11(b) there are fewer collision points with only one red cell near the exit.

(a)   (b)
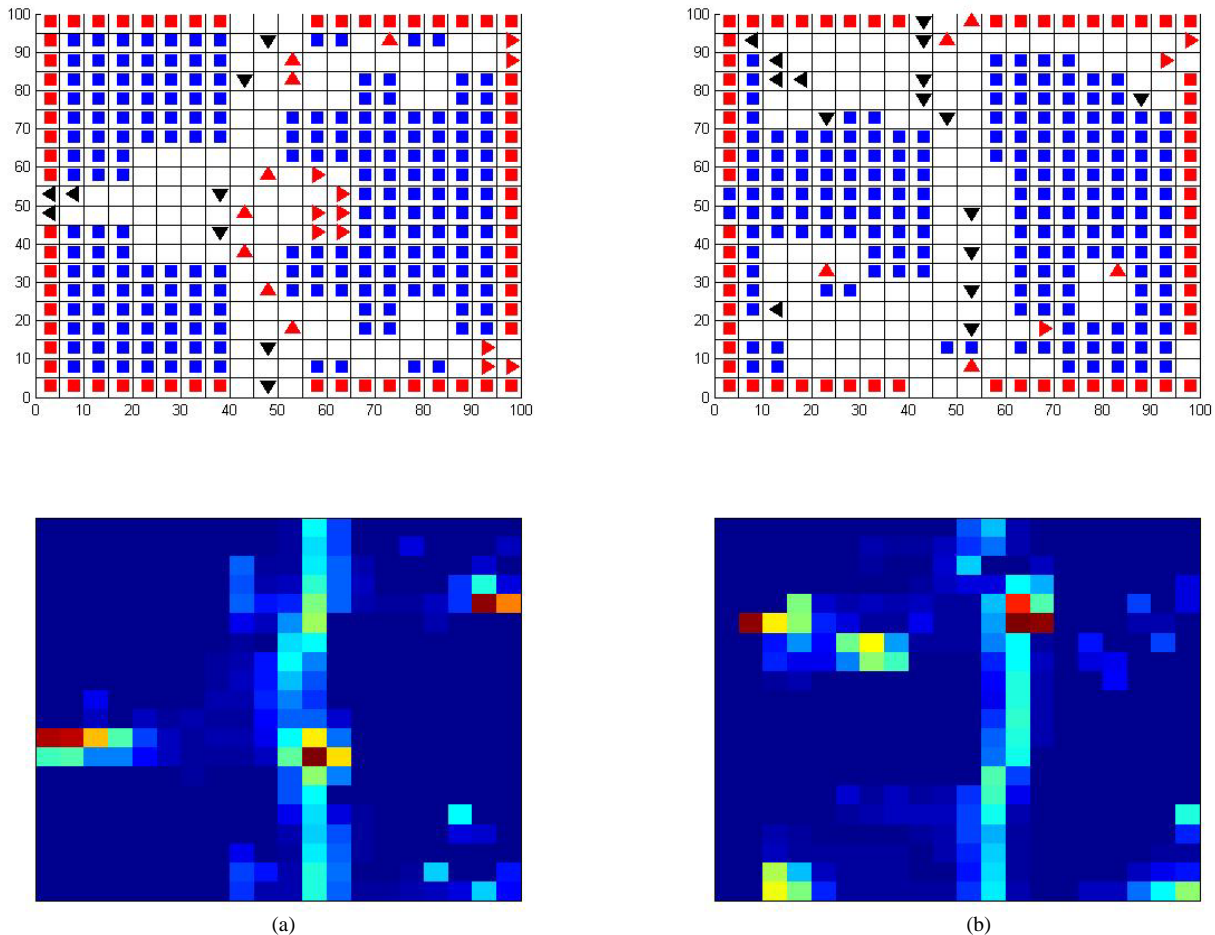
Fig. 12 Final layout and heat map of (a) original stadium layout with final fitness of 2.884, (b) improved layout with the highest fitness from SA0.9 algorithm with final fitness of 5.650

3) Stadium Layout:

The original stadium seating layout, as in Fig. 12(a), shows a clear path in the middle of the layout and zig-zag shaped paths near exits at the left and right walls. Three collision points appear near the left exit, top right exit and in the middle of the layout. Some collisions can also be seen along the corridor from top to bottom of the layout and worsen in the middle of the corridor. This may be because there is a T-shape junction and a cul-de-sac area in the middle of the layout. Notice there are three darker red cells in the heat map (Fig. 12(a)) that reflect the same area of the final layout. There are also light blue and yellow cells along the middle corridor in the heat map, which shows that jamming occurs on this long path.

Meanwhile, Fig. 12(b) shows the final layout discovered by SA with the highest fitness at 5.650. The seating arrangement appears to be similar to the improved layout with a long corridor from the top to bottom layout and a zig-zag shaped path near the top right exit. Although the two exits are blocked at the left exit and bottom right exit, the fitness is better than the original layout fitness. This may be because in a space with multiple exits, pedestrians usually use one exit and ignore other alternatives. This may not be a 'perfect' layout but some of the 'bad' characteristics found in the original layout through clogging points have been reduced to a minimum. Notice there are fewer red cells in the heat map of Fig. 12(b), no 'serious' jamming in the centre of the layout and no collision point near the top right exit.

C. Exploring Negative Final Layout Characteristics with Lower Fitness

Although a newly generated layout from the simulations sometimes produced poor layouts, certain 'bad' characteristics of the generated layout can be used as a list of guidelines on what to avoid in spatial layout design. Therefore, we are going to

further explore some of the 'bad' characteristics of the final layouts in the classroom, theatre and stadium discovered by the algorithms with lower fitness.

1) Sharing Lane:

1790	5071	2002
1103	10	1079
457	69	119

upstats

474	248	295
757	35	741
1904	5767	1979

downstats

1359	1557	538
4606	**216**	854
1254	1364	952

leftstats

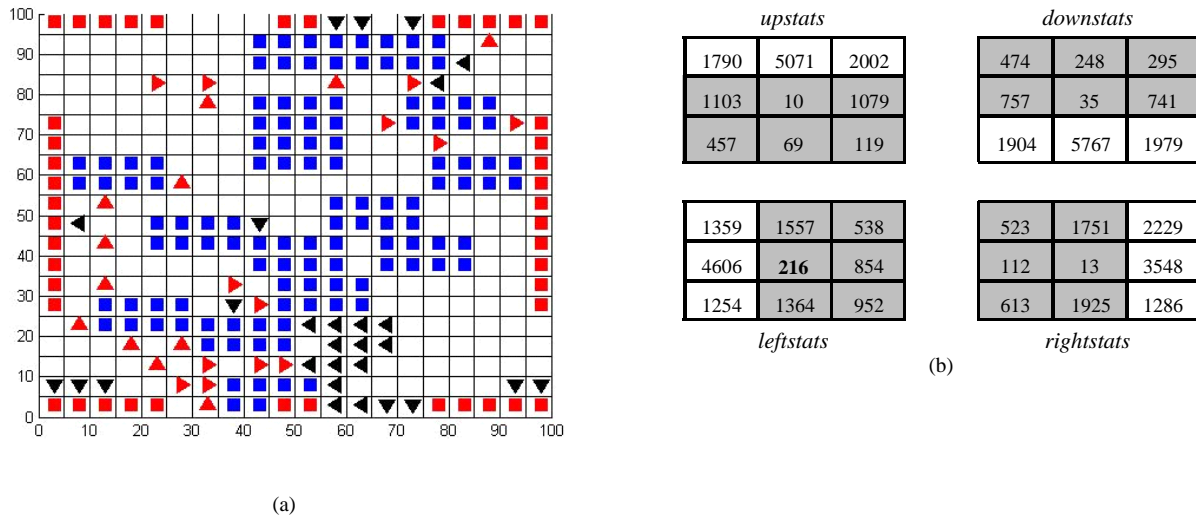|		
523	1751	2229
112	13	3548
613	1925	1286

rightstats

(b)

(a)

Fig. 13(a) Classroom final layout using HC with the lowest final fitness of 29.170, (b) final pedestrians statistics

From Fig. 13(a), we can see clearly that bad layouts are generated from the lower fitness of classroom layout. Fig. 13(a) is the 'bad' layout with the lowest fitness from 10 runs of HC. Notice that the 'left' and 'right' pedestrians bump into each other because of the narrow path that they have to share. Notice a jammed group of 'left' and 'right' pedestrians in the lower layout, where there is a high density location. Fig. 13(b) shows that 'left' and 'right' have the lowest final statistics (4606 and 3548) when compared to 'up' and 'down' (5071 and 5767). 'Left' pedestrians have the largest middle value of *leftstats*, at 216. In this scenario, the 'left' pedestrians might have to stop and wait to give priority to 'right' pedestrians to move in the sharing lane. The sharing lane phenomena also existed in the original stadium layout, as shown in Fig. 12(a). There is only one exit route spanning from the top to the bottom of the stadium, thus the 'up' and 'down' pedestrians are forced to share the same route. As a result, there is a clogging effect along the path, as shown in the heat map of Fig. 12(a).

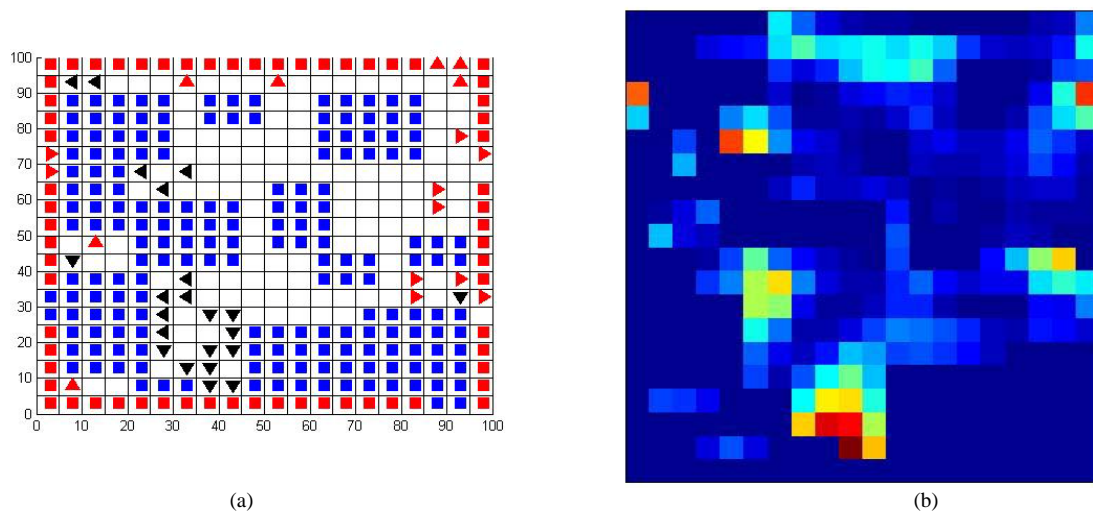(a)   (b)

Fig. 14 (a) Theatre final layout using SA0.9 with the lowest final fitness of -2.780, (b) its final heat map

2) Cul-de-sac:

The lower the fitness, the worse the layout generated, as in Fig. 14(a) which has a final fitness of -2.104 discovered by SA. This is the worst final fitness from 10 simulations and even the original theatre layout has better fitness of 4.286. Seating arrangements in this stadium layout create more than one cul-de-sac region. This resulted in numerous collision points scattered in the layouts, especially in the cul-de-sac area. Notice that there are a high number of red and yellow cells in the heat map, Fig. 14(b). There are no clear exit paths created and most of the exit areas are blocked by the seats, resulting in pedestrians being stuck on the dead-end route. The cul-de-sac regions also appear in the original stadium layout, Fig. 12(a). This 'bad' characteristic existed in the middle of the layout, thus the pedestrians get trapped inside.
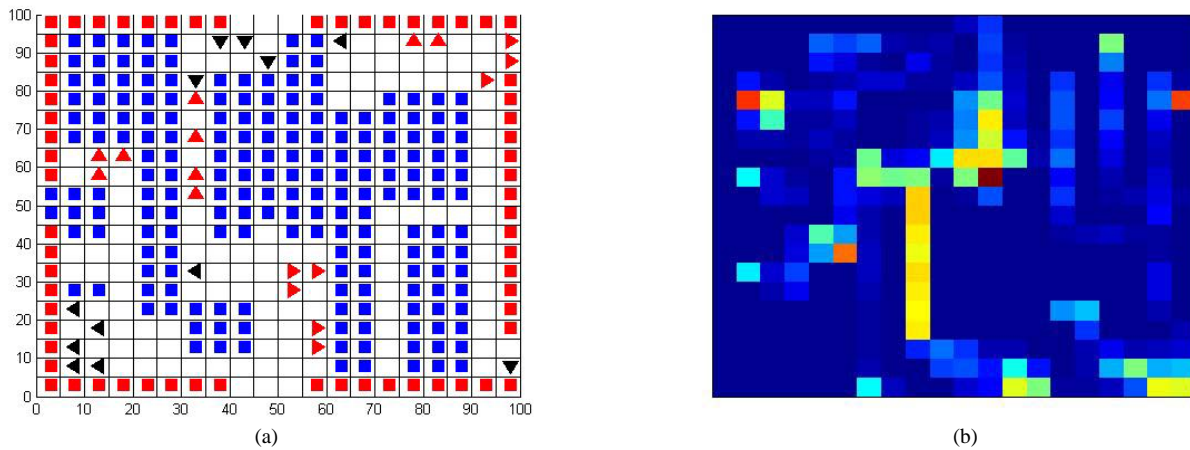
Fig. 15 (a) Stadium final layout using SA0.9 with final fitness of -1.31, and (b) its final heat map

3) One Cell Width of Path:

Fig. 15(a) shows poor stadium layout with final fitness only -1.31 using SA0.9 compared to the original stadium with final fitness of 2.884. There is no clear path created for 'left' and 'right' pedestrians and the seats are mostly gathered in the middle of the layout. A long path with a width of one cell exists in the centre of the layout. There is no alternative path and it can be seen in the final layout that 'up' and 'down' pedestrians have no other choice than this narrow path. They are forced to share this same one cell width path, thus producing jamming along that path. Notice there is a long yellow cell in the heat map (in Fig. 15(b)), showing a clogging effect along this one cell width route. The original theatre layout in Fig. 11(a) also has this same 'bad' characteristic. The layout has two paths with a width of one cell that spans from left to right of the layout. Both paths produce jamming effects along the corridor, as shown in the heat map of Fig. 11(a).

D. Pedestrian Flow Behavior

We now explore some of the characteristics of the flow of pedestrians rather than just their spatial layout characteristics. In this next experiment, we re-run selected layouts with different numbers of repeated simulations for each iteration (*pedsimsmax*) and find their average fitness for each type of pedestrians. Each selected layout was run 3 times with *pedsimsmax* values of 5, 10 and 15. The learning curves for up, down, left and right were inspected.
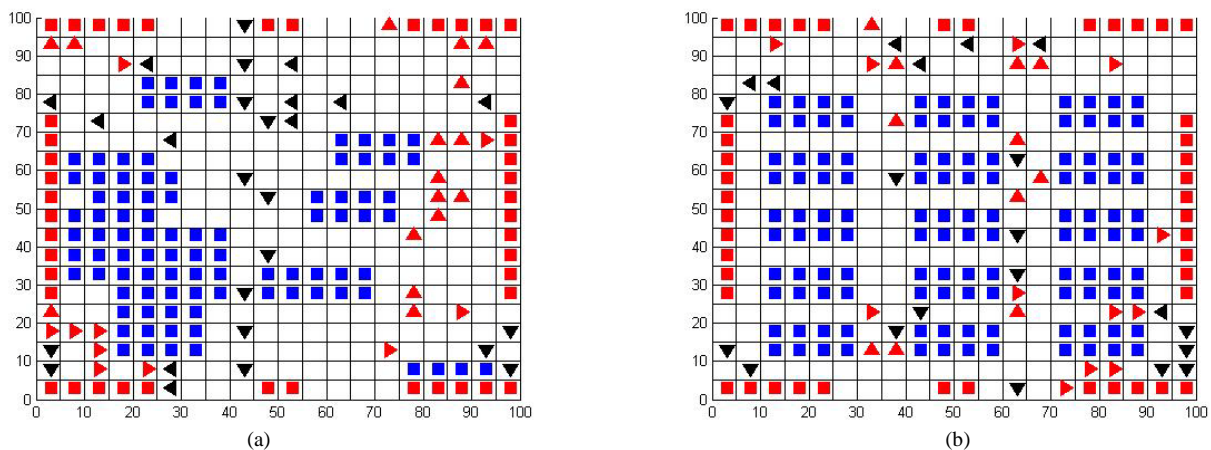

Fig. 16(a) 'child1' from SA-GAO classroom layout with fitness of 53.750, (b) original classroom layout with fitness of 43.434

1) Disruptive Interference Effect:

We observe the simulations of 'up', 'down', 'left' and 'right' pedestrians of classroom layout discovered by SA-GAO with a fitness of 53.750 and original classroom layout with a fitness of 43.434, as shown in Figs. 16(a) and (b). As discussed in the previous section, the disruptive interference effect clearly exists in the original classroom layout. In this section we are going to examine in detail the fitness of each pedestrian and then compare it with a SA-GAO layout that has better fitness. In the original layout, Fig. 16(b), the 'up' and 'down' pedestrians are forced to share the same lane because the exit routes are not far enough apart. This is due to a collective slow-down that emerges with the 'up' and 'down' pedestrians crossing each other's paths (the disruptive interference effect). Looking now at the simulation of pedestrians on a good layout, Fig. 16(a) that we found using SA-GAO, high fitness solutions generally involved creating separate routes that were far enough apart for 'up' and 'down' pedestrians not to share a path or to cross each other's paths. It is clear (in Fig. 16(a)) that the 'up' (red arrow) and 'down' pedestrians (black arrow) use these different paths to avoid collisions.

2) Chemotaxis or Virtual Trace:

It appears that the 'up', 'down' and 'left' pedestrians leave a virtual trace for the following pedestrians to travel, as shown in the SA-GAO final layout, Fig. 16(a). The implementation of the interactions between the pedestrians uses an idea similar to chemotaxis. In the process of chemotaxis, some insects create a chemical trace to guide other individuals to food locations. This is also the central idea of the pedestrian simulation models used for the simulation of trail formation. In this approach, the pedestrians also create a trace. In contrast to trail formation and chemotaxis, however, this trace (footprints) is only virtual although one could assume that it corresponds to some abstract representation of the path in the mind of the pedestrians. Its main purpose is to transform the effects of long-range interactions (e.g. following people walking some distance ahead) into a local interaction (with the "trace"). For example, in Fig. 16(a), the 'up' and 'down' pedestrians use a path followed by the 'virtual trace' created by the initial pedestrian. As a result, the 'up' pedestrian (red arrow) uses the right path and the 'down' pedestrian (black arrow) uses the left path.
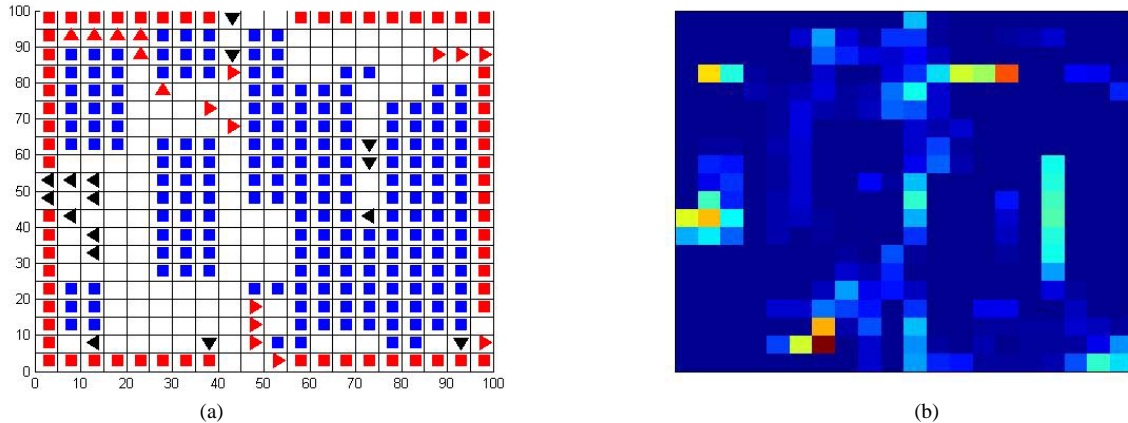

(a)   (b)

Fig. 17 (a) Stadium final layout with final fitness of 3.914, (b) its heat map

(a)   (b)
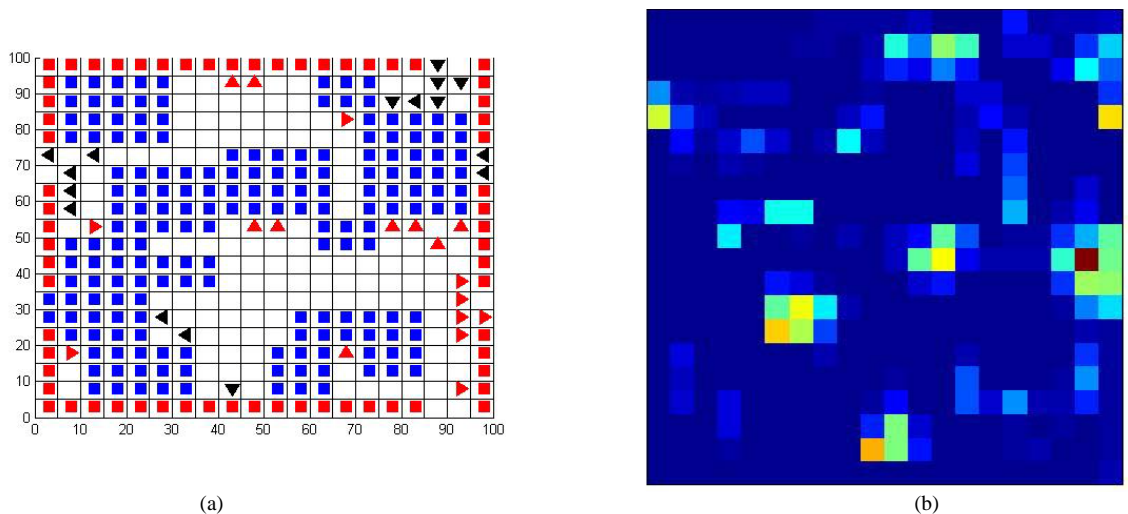
Fig. 18 (a) Theatre final layout with final fitness of -1.976, (b) its heat map

3) Arching Formation and Clogging at Exit:

Our simulation produced arching behavior near the exits, as shown in the figures above. The pedestrian arch formation occurred near the left exit of stadium layout in Fig. 17(a) and near the right and left exits of theatre layout in Fig. 18(a). Notice that the exit is only wide enough for two pedestrians, but everyone wants to be first. The pedestrians wedge into the area in front of the doorway is like a cork. More pedestrians result in an arch of gridlock as everyone seeks the shortest straight-line vector to the exit. The exit clogs, with intermittent bursts of escapees when force exceeds friction. This is dangerous, especially in panic situations, when nearly all pedestrians will try to squeeze past preceding pedestrians, resulting in the arching and clogging at exits. The heat maps above show the clogging points at the same areas of arching and jamming near the exits.

4) Fluctuation of Flow at Bottleneck:

The classroom layout above, Fig. 19 (a), has a narrow path with one cell width on the top left of the layout. We observe a deadlock situation within that narrowing corridor among 'up' and 'down' pedestrians. Notice the darker tone of red and orange in the same bottleneck area in the heat map (Fig. 19(b)). The bottleneck area reduced the capacity for pedestrians to pass through and a jam will occur when the incoming flow exceeds the capacity of the bottleneck. 'Down' pedestrians are able to

pass the bottleneck and the pedestrians with the same walking direction can easily follow. In this layout, we assume that 'down' pedestrians have priorities to pass the bottleneck while the 'up' pedestrians have to stop and wait.

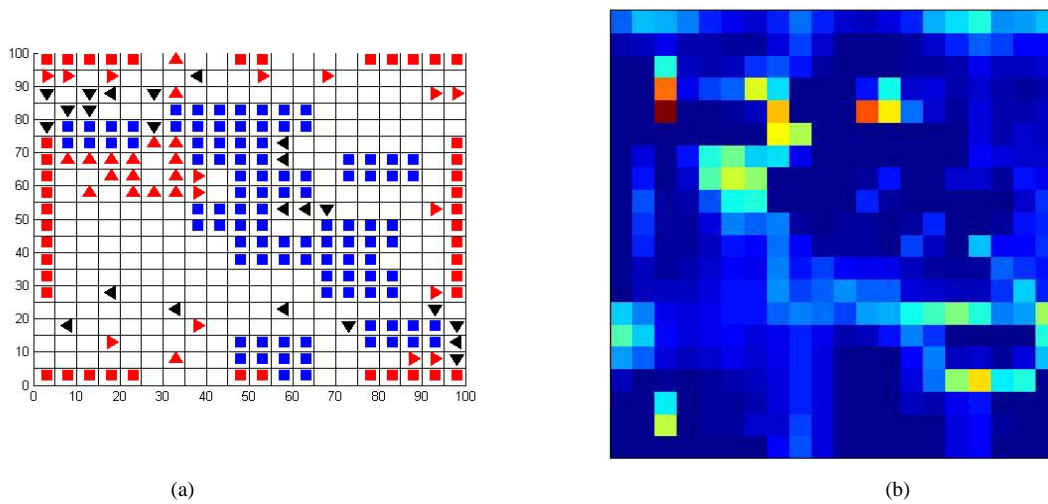(a)   (b)

Fig. 19 (a) Classroom final layout using SA-GAO with final fitness of 46.658 and (b) its heat map

E. Exploring Layout at Iteration of 500

We then ran the generated classroom, theatre and stadium layouts for 500 iterations using SA to see whether the layouts were able to evolve and produce another feasible seating arrangement. We observe that after 500 iterations, none of the layouts generated from SA-GAO changed their initial layout but some of the layouts generated from SA did have some minor adjustments to their seating arrangement.

Newly generated layouts show a positive outcome by eliminating not all but some of their bad characteristics, such as cul-de-sac areas and clogging near exits, as shown in the figures below.

(a)   (b)
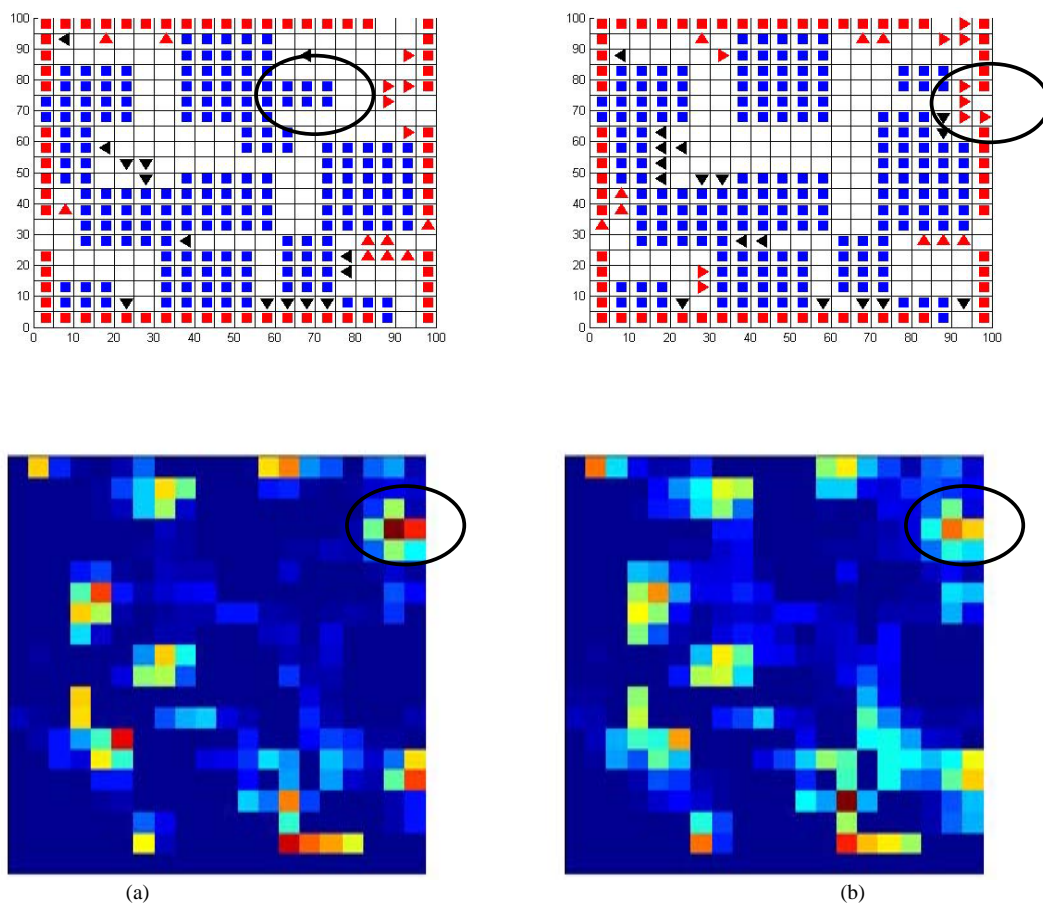
Fig. 20 Theatre final layout and heat map using SA0.9 with of fitness of -2.014 at iterations of (a) 100 and (b) 500

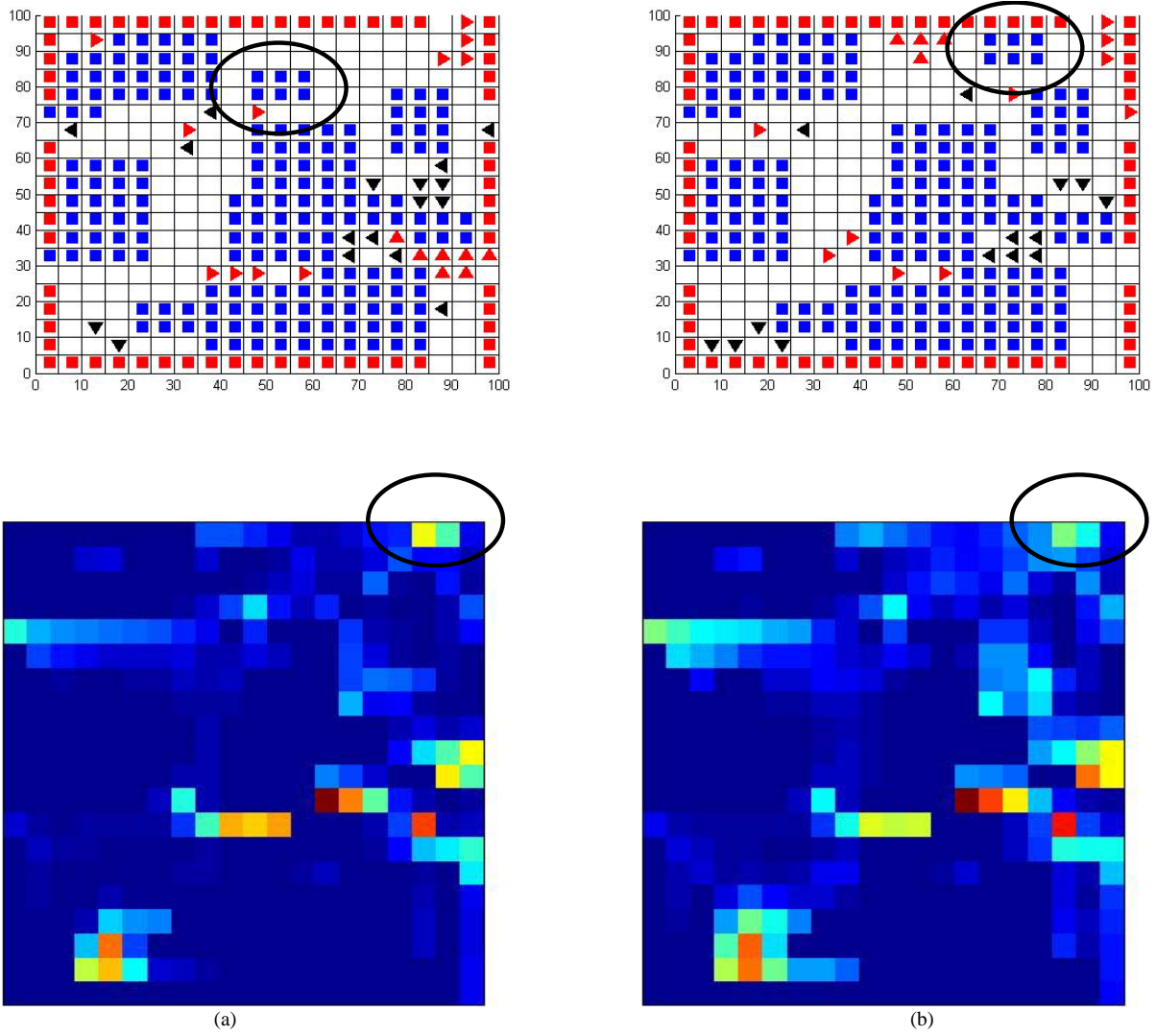Fig. 21 Theatre final layout and heat map using SA0.9 with final fitness of 9.372 at iterations of (a) 100 and (b) 500

(a)   (b)
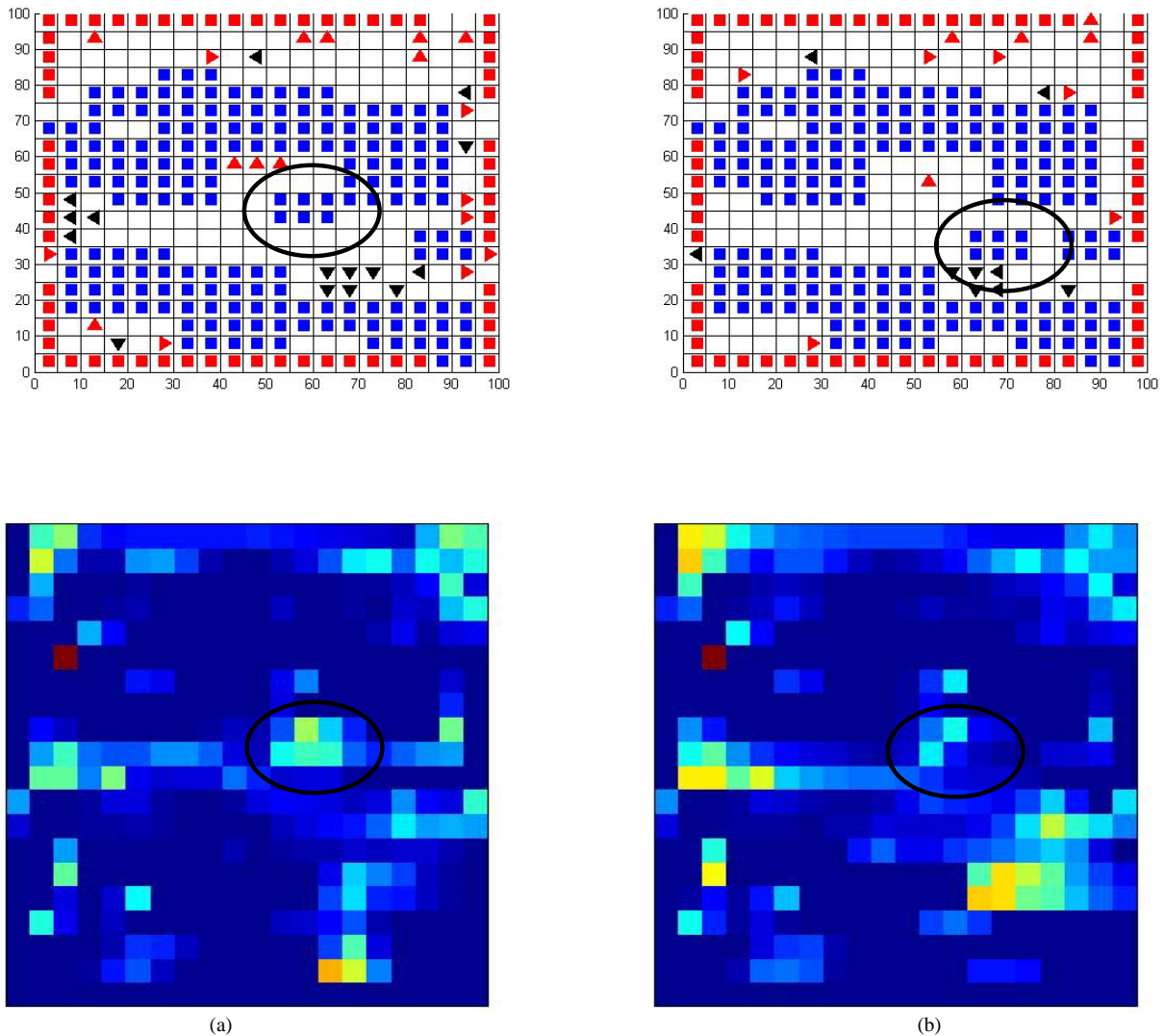
Fig. 22 Theatre final layout and heat map using SA0.9 with of fitness 12.838 at iterations of (a) 100 and (b) 500

Fig. 20(a) and Fig. 21(a) both show theatre final layouts at 100 iterations with fitness of -2.014 and 9.372. There is a clogging phenomenon in front of the top right exit in the layout of Fig. 20(a) at 100 iterations. After 500 iterations, the clogging is still spotted but this time with less intensity. This may be because the two 6-by-2 sized objects in Fig. 20(a) move to being in front of the exit after 500 loops, as shown in the heat map (Fig. 20(b)). Notice also that a 6-by-2 sized object in the top middle of the layout in Fig. 21(a) shifts to be near the top right exit after 500 iterations, as shown in Fig. 21(b). The jamming intensity near the exit reduced after 500 iterations as we compared the heat map in Figs. 21(a) and (b). From the above simulations, objects that present in front of or near exits might help to reduce any jamming effect at exits.

A 6-by-2 sized object position in the middle of the theatre layout above, Fig. 22(a), creates a cul-de-sac region at 100 loops. The dead-end region resulted in the pedestrians getting trapped inside and produced a congestion effect, as shown in the heat map, Fig. 22(a). After 500 loops, the 6-by-2 sized object that created the cul-de-sac region moves to open up with some spaces in the centre of the layout, allowing a smooth flow of pedestrians. Notice that there is less clogging intensity in the second heat map in Fig. 22(b).

VI. CONCLUSIONS

In this paper, we proposed pedestrian flow simulations combined with heuristic searches to assist in the automatic design of classroom layout. Using pedestrian simulation statistics and heat map agents, the activity of crowds can be used to study the consequences of different spatial layouts. Based on the results that have been observed in this paper, we have shown promising results in identifying a well designed characteristic of spatial layout and the sources of the pedestrian clogging phenomenon. We have identified several key results:

- SA-GAO, SA and HC are able to automatically find adequate solutions to this problem when incorporated with the pedestrian simulator. SA has more variation in final fitness when compared to HC. Whilst HC cannot 'escape' local

optima SA does sometimes manage to do this with better final solutions – in general, the distribution of final fitnesses is higher for SA though more adventurous solutions are explored.

- SA-GAO generated better solutions compared to SA and HC solutions: the SA-GAO children show higher fitnesses than their parents and less congestion spots. This implies that solutions with lower fitnesses may still offer useful information and when these are recombined in a constructive way, they generate better overall layouts than if no recombination is used.
- The highest fitnesses produced useful layouts and less clogging areas; passageways with increased width and a staggered layout created a zig-zag pattern path. A zig-zag shaped path can reduce the pressure in panicking crowds and ease pedestrian movement. A wider lane helps to avoid a long waiting lane and clogging effect. These demonstrably produce a smoother flow when running the simulations and exploring the statistics of movement.
- Clogging phenomena appear to be in front of the exits, cul-de-sac regions, narrowing corridors, and exit routes that are not far enough apart. The arching formation and clogging at exits, jamming inside dead-end paths, fluctuation of flow at bottlenecks, and disruptive interference effects demonstrably exist in these areas when exploring the heat maps and running the simulations.
- An obstacle a few feet in front of the exit can help break up an arch, splitting the jam into smoothly flowing streams. A group of objects in between the two exit routes can avoid different types of pedestrian sharing paths or to cross each other's paths. Through experimentation, we identified less jamming intensity when exploring the heat maps and a smoother flow of pedestrians.

Although the newly generated layout from the simulations is not a perfect layout, some 'good' qualities from the improved layout transpires as being feasible and offers a valuable contribution to the work of designers of pedestrian facilities in general - in particular, spatial layout architects. Based on the proposed model, one could suggest possible extensions that would be helpful in improving the safety aspects and performance-based design of buildings. Additional further work could involve extending the optimization techniques in a number of ways. As discussed in Section B, the optimal solutions presented in this research are near optimal and yet not realistic in terms of practicality. The configuration of the objects inside the layout doesn't consider the ergonomic position and the practical functionality of every object. Adding more specific rules to the optimization algorithms may produce better realistic results. Future work will be directed towards the gathering of real world data sets and complex systems in order to validate the model equations.

REFERENCES

[1] M. Batty, J. Desyllas, and E. Duxbury, "The discrete dynamics of small-scale spatial events: agent-based models of mobility in carnivals and street parades," *International Journal of Geographical Information Science,* vol. 17, no. 7, pp. 673-697, 2003.

[2] J. Dijkstra and H. Timmermans, "Towards a multi-agent model for visualizing simulated user behavior to support the assessment of design performance," *Automation in Construction*, vol. 11, no. 2, pp. 135-145, 2002.

[3] X. Pan, C. S. Han, K. Dauber, and K. H. Law, "Human and social behavior in computational modeling and analysis of egress," *Automation in Construction*, vol. 15, no. 4, pp. 448-461, 2006.

[4] G. Smedresman, "Crowd Simulations and Evolutionary Algorithms in Floor Plan Design," 2006.

[5] N. Zhu, J. Wang, and J. Shi, "Application of Pedestrian Simulation in Olympic Games," *Journal of Transportation Systems Engineering and Information Technology*, vol. 8, no. 6, pp. 85-90, 2008.

[6] F. H. Hassan and A. Tucker, "Using Cellular Automata Pedestrian Flow Statistics with Heuristic Search to Automatically Design Spatial Layout," *22nd IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, p. 32, 2010.

[7] F. H. Hassan and A. Tucker, "Using Uniform Crossover to Refine Simulated Annealing Solutions for Automatic Design of Spatial Layouts," *International Joint Conference on Evolutionary Computation*, p. 373, 2010.

[8] H. Yue, H. Hao, X. Chen, and C. Shao, "Simulation of pedestrian flow on square lattice based on cellular automata model," *Physica A: Statistical Mechanics and its Applications*, vol. 384, no. 2, pp. 567-588, 2007.

[9] P. Bentley, "Evolutionary design by computers," Morgan Kaufmann, 1999.

[10] J. S. Gero and V. A. Kazakov, "Evolving design genes in space layout planning problems," *Artificial Intelligence in Engineering*, vol. 12, no. 3, pp. 163-176, 1998.

[11] R. S. Liggett, "Automated facilities layout: past, present and future," *Automation in Construction*, vol. 9, no. 2, pp. 197-215, 2000.

[12] D. Bauer, S. Seer, and N. Brändle, "Macroscopic pedestrian flow simulation for designing crowd control measures in public transport after special events," *Proceedings of the 2007 Summer Computer Simulation Conference*, Society for Computer Simulation International San Diego, CA, USA, p. 1035, 2007.

[13] V. J. Blue and J. L. Adler, "Cellular automata microsimulation for modeling bi-directional pedestrian walkways," *Transportation Research Part B*, vol. 35, no. 3, pp. 293-312, 2001.

[14] J. Dijkstra, H. J. P. Timmermans, and A. Jessurun, "A multi-agent cellular automata system for visualising simulated pedestrian activity," *Theoretical and Practical Issues on Cellular Automata,* vol. 10, no. 4, p. 6, 2000.

[15] T. Meyer-König, H. Klüpfel, and M. Schreckenberg, "Assessment and analysis of evacuation processes on passenger ships by microscopic simulation," *Schreckenberg and Sharma*, vol. 2, pp. 297-302, 2001.

[16] K. Narimatsu, T. Shiraishi, and S. Morishita, "Acquisition of local neighbor rules in the simulation of pedestrian flow by cellular automata," *Lecture Notes in Computer Science*, vol. 3305, pp. 211-219, 2004.

[17] N. Pelechano and A. Malkawi, "Evacuation simulation models: Challenges in modeling high rise building evacuation with cellular automata approaches," *Automation in Construction*, vol. 17, no. 4, pp. 377-385, 2008.

[18] W. F. Fang, L. Z. Yang, and W. C. Fan "Simulation of bi-direction pedestrian movement using a cellular automata model," *Physica a-Statistical Mechanics and Its Applications*, vol. 321, no. 3-4, pp. 633-640, 2003.

[19] L. Yang, W. Fang, J. Li, R. Huang, and W. Fan, "Cellular automata pedestrian movement model considering human behavior," *Chinese Science Bulletin*, vol. 48, no. 16, pp. 1695-1699, 2003.

[20] A. Schadschneider, "Cellular automaton approach to pedestrian dynamics-theory," *Pedestrian and Evacuation Dynamics*, Springer, pp. 75-86, 2001.

[21] A. Penn and A. Turner, "Space Syntax Based Agent Simulation," *Pedestrian and Evacuation Dynamics*, pp. 99, 2001.

[22] R. Barlovic, T. Huisinga, A. Schadschneider, and M. Schreckenberg, "Open boundaries in a cellular automaton model for traffic flow with metastable states," *Physical Review E*, vol. 66, no. 4, pp. 6113-6123, 2002.

[23] D. Helbing, L. Buzna, A. Johansson, and T. Werner, "Self-organized pedestrian crowd dynamics: Experiments, simulations, and design solutions", *Transportation Science*, vol. 39, no. 1, p. 1, 2005.

[24] Z. Michalewicz and D. B. Fogel, *How to Solve It: Modern Heuristics*, Berlin; New York: Springer, 2000.

[25] R. L. Haupt, S. E. Haupt, and J. Wiley, "Practical genetic algorithms," Wiley Online Library, 2004.

[26] X. B. Hu and E. Di Paolo, "An efficient genetic algorithm with uniform crossover for air traffic control," *Computers & Operations Research*, vol. 36, no. 1, pp. 245-259, 2009.

[27] S. Narmadha, D. V. Selladurai, and G. Sathish, "Multi Product Inventory Optimization using Uniform Crossover Genetic Algorithm," *International Journal of Computer Science and Information Security*, vol. 7, no. 1, pp. 170-179, 2010.