Macroscopic effects of microscopic forces between agents in crowd models

Colin M. Henein and Tony White
Carleton University, Ottawa, Canada

ABSTRACT

Crowd scenarios have attracted attention from computer modellers, perhaps because of the impracticality of studying the phenomenon by traditional experimental methods. For example, Kirchner has proposed an agent-based crowd model inspired by fields of elementary particles [2], but chose not to incorporate crowd forces. We argue that crowd forces (and associated injuries) are an essential characteristic of crowds, and that their omission will negatively affect the model's ability to make predictions (e.g. time for a crowd to pass through an exit). To support this position we describe an evolution of Kirchner's model that includes a vector-based particle field to represent forces. We show qualitative and quantitative differences compared to Kirchner's model when force is included. The Swarm Force model demonstrates – by showing non-linear effects of force – the necessity of force in crowd models.

PACS CODES / KEYWORDS

PACS Codes: 45.70.Vn; 89.40.+k; 89.65.Lm; 05.65.+b
Keywords: Cellular automata; Non-equilibrium physics; Pedestrian dynamics

INTRODUCTION

Crowd situations are one of the few situations in which humans flock together. Like other aggregations of animals (e.g. cattle herds, fish schools) and material (e.g. fluid and particle flow) there has been an interest in determining the basis of crowd effects, for example, in cases where large groups of people must move through a doorway, narrow hall or other restricted environment.

Several modellers [e.g. 1] have been inspired by material models (e.g. flow of grain in a hopper) that have generally been represented by systems of differential equations that capture the interactions between passive particles. (A grain within the flow is passive in the sense that it will always react in the same way when encountering the same flow conditions.) Helbing's social forces model combines physically realistic estimations of mechanical forces (like body compression) with gravity-like "forces" acting upon agents at a distance to represent intentional collision avoidance behaviours between people in crowds. The model is successful at representing aggregate crowd effects; people are not grains, however, and Helbing's direct and invariant relationship between sensory perceptions of individuals (noticing a person ahead) and the responses engendered (repulsion inversely proportional to distance) denies the contributions of individual agents who make local decisions based on personal strategies. This abstracts away a very important attribute of crowds: the effect of non-homogeneity.

One model that has adopted a stochastic approach to the problem of crowd dynamics with a focus on the individual is Kirchner's field-based model [2]. This model represents two individual factors – desire to move toward an exit and desire to follow others – within a physical space laid

This article was published as *Physica A 373* pp. 694-712.

out in a grid pattern. The model takes the metaphor of particle fields that can be sampled locally by people within the crowd to carry the limited and local information available to individuals in a crowd, thus embodying swarm intelligence techniques like stigmergy (information sharing mediated through the environment) and decentralisation [c.f. 3]. This local focus makes the model more computationally efficient and physically more reasonable. A problem with the Kirchner model, however, is that it abstracts physical force out of the crowd model entirely.

In our view, physical force is characteristic of crowds, and cannot be abstracted out of the modelling of dense crowds without a dramatic alteration of the pattern of results. In the case of a crowd exiting through a doorway, we expect that significant changes to numbers of agents exiting, exit time and exit dynamics would be observed. In addition, without a concept of force, a model cannot predict the injuries which directly affect the injured, and indirectly affect other individuals trying to exit around them. While we appreciate Kirchner's individual focus, we believe that his model has gone too far in removing force from his model.

In addition to problems surrounding the presence of force, we also found that Kirchner's model makes some assumptions concerning undesirable exit rates and cell occupations that can be addressed by modifications to the model's movement rules.

It is these observations which motivate this paper.

In order to overcome the issues outlined above we propose a new model. The Swarm Force model retains Kirchner's individual focus, particle field metaphor and the spirit of his movement rules. Our contribution is to augment this model with an explicit particle-based time-evolving representation of force transmitted by and amongst agents. The new model also includes improvements directed toward the noted problems with exit rates and cell occupations.

In the next section we will consider the importance of force in crowds, and present our view as to the characteristic aspects of force that must be embedded within a crowd model. Following that, we will briefly outline Kirchner's model and explain how the characteristic force aspects have been integrated into it to create a cohesive model. Following this we will present results and analysis.

FORCE IN CROWDS

Force in models of crowds is a basic element that must be represented for three reasons. First, force has a direct effect on movement: people in a crowd can be pushed around. Second, force is a perceptible input to the cognitive system and a major source of information in an information-starved situation. Third, force carries the consequences of dangerous crowd scenarios: injuries. Fruin has summarized these force effects on crowds as follows:

> It is difficult to describe the psychological and physiological pressures within crowds at maximum density. When crowd density equals the plan area of the human body, individual control is lost, as one becomes an involuntary part of the mass... Shock waves can be propagated through the mass sufficient to lift people off of their feet and propel them distances of 3 m (10 ft) or more. People may be literally lifted out of their shoes, and have clothing torn off… Crowd forces can reach levels that are almost impossible to resist or control. Virtually all crowd deaths are due to compressive asphyxia and not the "trampling" reported by the news media. Evidence of bent steel railings after several fatal crowd incidents show that forces of more than 4500 N (1,000 lbs.)

occurred. Forces are due to pushing, and the domino effect of people leaning against each other. [4]

Given that forces are characteristic of crowds and crowd disasters, we might ask what level of commitment a model ought to have with respect to forces. We see three possible positions for treatment of force in a model.

The first position (taken implicitly by Kirchner) hypothesises that force is not an essential characteristic of a force model; the corollary to this position is that results can be accurate with or without force so there is little lost by abstracting it out of a model.

The second position is that force can be accounted for without an explicit simulation (perhaps as a linear scaling factor upon the output of a non-force model).

The third position, which we here espouse, is that, being characteristic of crowds, force has a complex effect on individual actions; force cannot be removed from a crowd model without totally changing the dynamics of the system.

We propose that an individual-centred crowd model should contain physical force as a factor within the model. We believe that Helbing's model lacks Kirchner's individualistic focus, while Kirchner's model lacks Helbing's account of force within the model[1]. The purpose of this study is to improve upon Kirchner's model by including force effects in the crowd simulation.

Modelling Forces in Crowds

Fruin's citation, above, certainly highlights the important role that force plays in any crowd disaster, but it is not only the obvious lack of injuries that results from the absence of force in a model. As people move within a crowd, force effects can cause them to be diverted from their individually planned movement pattern. This causes a change in exit time which cannot be accounted for without the presence of force. Injuries – an obvious consequence of force – themselves have effects on non-injured agents still trying to exit; these effects can be obstructive or facilitative depending on the spatial distribution of injuries and hence can affect the agents' exit times in a way that would not be possible for a force-free model to explain.

The purpose of a model is to reduce the complexity in a real-world system down to its essential aspects. In order to incorporate force into our crowd model we must determine those essential aspects of force that must be represented. We have adopted four general principles concerning the modelling of force:

1. Force is *directed*. It is applied by one agent to another in a particular direction. Forces in models must be implemented as vectors rather than scalars.

2. Force carries *consequences*. Force is not an invisible field that guides cognitive decision making; rather, once exerted it has measurable effects that are outside an agent's control.

[1] Kirchner has proposed a revision to his model [5] that includes a force-like *friction* parameter, but this parameter relates only to contention for space in the crowd and not to force applied between agents that can lead to forced movement (in the form of shockwaves) or injuries.

The two major force consequences we have chosen to model are injuries and loss of individual control. An agent who is being pushed may lose the ability to decide where to move within the model. Agents who experience excessive force become injured and can no longer move within the model.

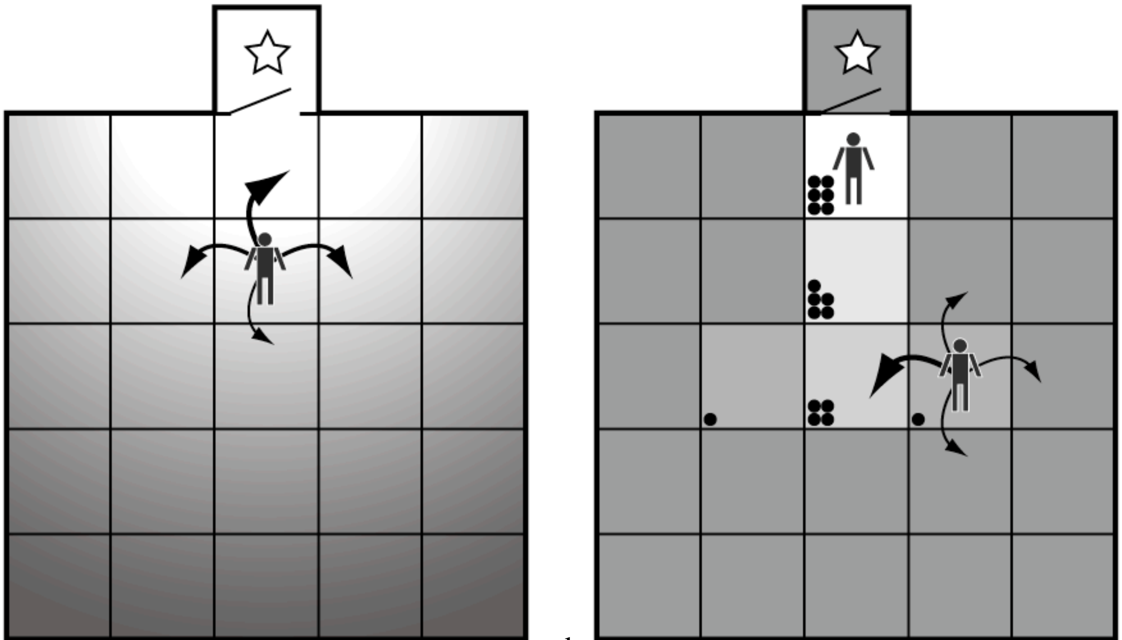
3. Force is *propagated*. Fruin [4] has told us that injuries within real crowds "are due to pushing and the domino effect of people leaning on each other." This means that force in a crowd model must be transmitted through the crowd in the direction of the force, and that it is additive (vector-wise) when encountering other forces. Moreover, being a physical effect, this occurs over a certain time period. Force applied at the rear of a crowd is not immediately felt at the front; instead it travels from person to person, and is experienced by all people as it is transmitted.

4. Force is generated *purposefully*. Force is not simply an emergent property of masses of people. People exert force in some circumstances and not in others. Our conception of this purposeful exertion is that agents push when blocked in their desired movement.

One other expected effect comes out of the last two rules: force peaks within the crowd (corresponding to locations of injuries) should not be distributed at random. This is because force *propagates* (and is additive) in the direction in which it is exerted, and *purposeful* pushing provides that most of the force will be exerted toward the door. Thus we should expect that injuries will not tend to occur near the back of the crowd, but rather will occur in specific areas where the force becomes concentrated. This effect has been noted in studies of real crowd disasters; some researchers have found injuries right at the front of the crowd, others finding injuries in a line near the front where the force from those pushing off the front wall meets the force from those pushing from the rear [4].

KIRCHNER'S FIELD MODEL

Kirchner's model consists of agents moving about in a rectangular grid (see Figure 1). Each cell within the grid represents a location for an agent to stand, with the constraint that under no circumstances may two agents occupy the same cell. Cell adjacency is defined by the four Cartesian neighbours of a cell.



a. b.

Figure 1: a) Kirchner Model Space with Static Field Agent (*pictured*) can move to neighbouring cells (*arrows*) or remain still. Cell selection is probabilistic based on agent's perceived desirability of cell (*arrow size*). Movement to exit cell (*starred*) results in disappearance of agent at end of the time step. Static field (*gradient overlay*) represents inherent cell desirability; field strength on own and neighbouring cells is accessible to the agent. **b) with Dynamic Field** Previous agents (*e.g. top agent moving up from centre*) have dropped dynamic bosons (*circles*). Bosons probabilistically decay (*middle group*) or diffuse to neighbouring cells (*lower group*). If k_S is 0, lower agent ignores static field, uses boson density like a trail (*gradient overlay*) to follow preceding agents.

Kirchner represents information available to agents by means of elementary particles (bosons) distributed throughout the space. Bosons are assumed to be placed directly onto cells, and no field gradient exists within the cell. By measuring the concentration of the boson field on their own cell (or adjacent cells) agents are able to extract localised information about their surroundings. The boson fields thus serve as spatial data structures, and agents do not have recourse to any information that is not encoded in this spatial data structure.

Two fields are provided for by Kirchner. The *static field* is a gradient with high values near desirable areas (i.e. exits, for present purposes) and low values elsewhere. An agent can hill-climb this gradient directly to these desirable areas. As expected from the name, the static field is initialized at the beginning of the model run, and does not change during the run. The value of the static field for a space with one door at co-ordinates (a, b) is given by equation (2). This equation is easily extended to spaces with multiple doors by taking (a, b) as the closest door in each case.

$$s_{ij} = [\ (a-i)^2 + (b-j)^2\ ]^{1/2} \qquad (1)$$

$$S_{ij} = \max_{ij}(s_{ij}) - s_{ij} \qquad (2)$$

The *dynamic field*, by contrast, is updated throughout the simulation. It counts the *dynamic bosons* dropped by agents as they move from one cell to another in the grid. (Initially the grid contains no bosons.) When an agent moves from a cell (i, j) to a neighbouring cell it drops a boson on the origin cell; this is represented in the dynamic field by $D_{ij} \rightarrow D_{ij} + 1$. The dynamic field is analogous to the pheromones released by ants moving through a landscape, as perceived by following ants. Just as the decay and diffusion of ant pheromone odours allow for ant trails to change and be optimised [3] the dynamic field bosons diffuse and decay probabilistically (allowing for agents to co-operate in path finding in the model). Each boson decays with probability δ in each time step. Those bosons which do not decay diffuse (move to a random neighbouring cell) with probability α.

To navigate through the model, agents determine a score associated with each neighbouring cell and its current cell. The agent then probabilistically selects a cell which it desires to occupy in the next time step. All agents deliberate in this manner at the same time. Movement is then simultaneous in the model, and an agent which finds its desired cell has become occupied is forced to remain stationary for that simulation step.

The score for each cell is a function (3) which is zero-valued for walls and occupied cells. Otherwise, its value is a mathematical combination of the field values present on that cell. Two sensitivity parameters, k_S and k_D, represents the agent's consideration of the static and dynamic fields respectively. If k_S is low, then the agent moves through the grid ignorant of which locations are more desirable than others. If k_S is high, then the agent is aware of which neighbour is more desirable. If k_D is low, then the agent is not concerned with the movements of others when planning its own movements. If k_D is high the agent will be disposed to follow other agents through the grid.

$$p_{ij} = N \exp(k_D D_{ij}) \exp(k_S S_{ij}) (1 - n_{ij}) \xi_{ij} \qquad (3)$$

In this function, p_{ij} represents the probability that an agent will select a neighbouring cell (or its own cell) with co-ordinates (i, j). D_{ij} and S_{ij} represent the value of the dynamic and static fields (respectively) at this location, n_{ij} is the occupation number (0 if unoccupied, 1 otherwise) and ξ_{ij} is the occupation number (0 for walls and occupied cells, 1 otherwise). N is the normalisation number equal to $(\Sigma\, p_{ij})^{-1}$.

For the purpose of modelling crowd exit behaviours, Kirchner designates cells in the walls at the points of interest to be exit cells. When agent steps onto an exit cell its next move will be to disappear from the model, having successfully exited the space.

Timestep progression

In each timestep the model must move the agents, determine exits and update the dynamic field. Model execution proceeds as follows, and is repeated until all agents have exited:

```
foreach dynamic-boson
        if random < δ then
                remove dynamic-boson from grid
        elseif random < α then
                move dynamic-boson to random neighbour cell
        endif
endfor

foreach agent
        determine the p_ij of neighbouring (and current) cells
```

```
        probabilistically select neighbouring (or current) cell
endfor

foreach agent (in different random order each time)
        if selected cell is unoccupied and not own cell
                move to selected cell, unless occupied
                update D on origin cell
        endif
endfor

foreach exit-cell
        if an agent is on the cell
                remove agent from the model
        endif
endfor
```

BUILDING THE SWARM FORCE MODEL

The swarm force model is inspired by Kirchner's force model. Our goal was to preserve aspects of Kirchner's model, wherever possible, and extend Kirchner's field metaphor to integrate force information into the system. We have been guided by the force principles discussed above in creating a model of crowd behaviour that incorporates force for increased realism. Our changes can be divided into three categories: addition of a dynamic force field, cell selection changes and addition of injuries.

Force Field Addition

The most important change to Kirchner's model is to add a dynamic force field. In keeping with Kirchner's physical model of agent interaction with elementary particles through static and dynamic fields, forces are represented in the model through force bosons. Force bosons are unit vectors, analogous to directional elementary particles. The force field has a discrete value on each cell, namely the aggregation (vector sum) of the force bosons on that cell. Thus the force field represents the direction and magnitude of the force experienced by an agent on a cell. The force on a cell is not perceptible by agents on neighbouring cells.

Force is generated (force bosons deposited) within the model by agent pushing. Agents push when they desire to move to a cell but cannot do so because it is occupied. Agents within the model have a variable capacity for pushing (ρ, the number of bosons dropped per push) that is selected at the start of the model from a normal distribution whose mean and standard deviation are input parameters to the model.

Each timestep, the force bosons on a cell are propagated to neighbouring cells according to Fruin's observation (discussed above) that crowd forces are repeatedly re-transmitted from person to person *through inter-personal contacts* within crowds. This re-transmission at the agent level carries the consequence that it is the aggregated force field vector (sum of force bosons) on a cell that is propagated, rather than the individual bosons themselves[2]. Force is

[2] The consequence of this re-transmission at the agent level is that the individual force bosons are not necessarily preserved. In a crowd an individual pushed with equal force from the left and right takes up both forces; the rightward force is not retransmitted through the individual to the other side.

dissipated (i.e. its underlying bosons disappear) if it would move onto an empty cell or a wall cell (or a cell containing an injured agent, see below).

The aggregated force field vector at a point $\boldsymbol{v}_{ij}$, provides both the magnitude (number of bosons deposited) and direction (direction of vector sum of force bosons) of force transmission. Because the coordinate system is discrete while $\boldsymbol{v}_{ij}$ is continuous, a method of quantizing the direction of $\boldsymbol{v}_{ij}$ to one of the four neighbouring cells is required. When $\boldsymbol{v}_{ij}$ points between two cells, the model – in considering each force boson to be deposited – assigns the boson to one of the two cells probabilistically. The probability of being deposited on one of the cells is inversely proportional to the deviation of $\boldsymbol{v}_{ij}$ from that cell and is given by equation 4 (in which p_a is the probability of selecting the neighbour with the lower angle, p_b is the probability of selecting the neighbour with the higher angle, and θ_{ij} represents the angle – taken modulo 90 – of $\boldsymbol{v}_{ij}$).

$$1 - p_a = \theta_{ij} / 90 = p_b \qquad (4)$$

Cell Selection Changes

Kirchner's cell selection mechanism has been changed in two ways. Most importantly, the model is changed so that normal cell selection is bypassed if the force experienced by an agent exceeds a threshold (we have used three times the force the agent can produce when pushing others). If this level of force is experienced, the agent is deemed to have selected the cell in the direction of the force (quantization of the direction is as described above for force boson propagation). No other changes are made to the movement routine or other model rules (in particular, the agent is not permitted to share a cell with another agent or a wall, movement is still simultaneous, the agent will push if the move cannot be made).

Two other changes were required in order to make the force rules effective. The force model depends on a realistic crowd in which agents cluster together and vie for empty spaces. This is because force cannot be transmitted through empty space. In Kirchner's model, agents will never desire occupied cells due to the $(1 - n_{ij})$ term of equation 3. This results in cells lying empty for at least one iteration after being left by an agent [6]. This artificially spaces the crowd out, and is not a desirable property of a crowd model. It is reasonable, however, that agents will *prefer* empty cells to occupied cells, a choice of an occupied cell in effect being a bet that the occupying agent will move on.

Accordingly we replace Kirchner's occupancy number n_{ij} with our vacancy factor ϕ_{ij} which is 0 if cell (i, j) is unoccupied and 0.5 otherwise. This term halves the cell score for occupied cells relative to unoccupied ones. The revised cell selection formula appears as equation 5.

$$p_{ij} = N \exp(k_D D_{ij}) \exp(k_S S_{ij}) (1 - \phi_{ij}) \, \xi_{ij} \qquad (5)$$

The five-cell neighbourhood used by Kirchner is both unnecessary and incompatible with the introduction of the ϕ_{ij} term. The five-cell neighbourhood was likely adopted by Kirchner in order to combat agents' tendency to make paradoxical retrograde progress when desired cells were occupied. (This retrograde action would occur when an agent would normally desire a cell, but the probability of its selection is set to 0 due to its occupancy number n_{ij}. The agent would be forced to select a less desirable cell; providing an option to 'stand still' by selecting the current cell avoided the inevitability of this retrograde progress.) With the replacement of the n_{ij} term by the ϕ_{ij} term there is a different mechanism for standing still (agent can now select the occupied cell, and may be forced to stand still if there is contention for the cell) so the ability to select the current cell is less relevant.

In addition to the superfluous nature of the five-cell neighbourhood given the new ϕ_{ij} term, the ability to select the current cell interacts poorly with its definition. This is because the option to select the self-occupied cell (with $\phi_{ij} = 0$ for this available cell) results in large p_{ij} for the current cell relative to occupied neighbour cells. This negates the purpose of the ϕ_{ij} term by encouraging standing still relative to the selection of occupied neighbouring cells.

For these reasons, the Swarm Force model uses the more common four neighbour Cartesian arrangement, only considering the four adjacent cells and not the agent's current cell. This allows the ϕ_{ij} term to carry out its function of allowing agents to bet on occupied cells, with the preservation of the 'stand still' option through the case of unsuccessful bets rather than through direct choice.

Addition of Injuries

One key characteristic of force (as summarized above) is that it carries consequences in the form of injuries. Injuries have several important impacts on agent exit behaviours: individual injured agents cannot exit, some nearby agents are impeded by injured agents, and injured agents can become force-breaks resulting in facilitation of some agents' progress. Without agent injuries the model cannot investigate individual safety in various scenarios.

The Swarm Force model considers that an agent becomes injured upon experiencing a force greater than a threshold, φ. Injured agents become totally inactive, no longer moving through the model, and are treated like wall cells by other agents and the force propagation algorithm.

Instead of using the vector force, agents use a scalar sum of the incoming forces on a cell when determining whether they become injured. This means that two equal and opposing forces that cancel each other out on a particular cell may still injure an agent on that cell who becomes squeezed between those forces.

Timestep Progression

In addition to the steps required by the Kirchner model, the Swarm Force model must propagate force, determine injuries and bypass cell selection. Model execution proceeds as follows, and is repeated until all agents have exited:

```
foreach dynamic-boson
        if random < δ then
                remove dynamic-boson from grid
        elseif random < α then
                move dynamic-boson to random neighbour cell
        endif
endfor

foreach agent
        if force on current cell > (3 * agent's pushing force) then
```

```
                select neighbouring cell in direction of the force
        else
                determine the p_ij of neighbouring cells
                probabilistically select neighbouring cell
        endif
endfor

foreach agent (in different random order each time)
        if selected cell is unoccupied then
                move to selected cell
                update D on origin cell
        else
                apply pushing force onto neighbouring cell
        endif
endfor

foreach exit-cell
        if an agent is on the cell
                remove agent from the model
        endif
endfor

foreach cell c_ij with more than one force boson
        if c_ij is unoccupied
                set force on c_ij to 0 magnitude
        endif
        foreach boson in vector sum of bosons at c_ij
                remove boson from c_ij
                quantize direction to a neighbour cell
                if neighbour cell is occupied and uninjured
                        move boson to neighbour cell
                endif
        endfor
endfor
```

RESULTS

The purpose of this investigation was not to repeat the analysis of Kirchner's own model, but rather to investigate the effects produced as we add force to the simulation. Accordingly we have been guided in our analysis by Kirchner's and present results in comparative form[3].

Kirchner's simulation is not completed until all agents have successfully exited the modelled space; he takes the number of iterations required for this complete exit as his measure. This combination of completion condition and measure is ill-suited to a model with injuries because an agent can become injured in front of the door in such a way as to prevent other agents from exiting; if agents are prevented from exiting then the number of iterations required for a complete exit is undefined. We prefer a measure that counts the number of successful exits in a fixed number of iterations.

---

[3] A brief note is in order regarding the data provided here under the label of Kirchner's model. As discussed, Kirchner's model uses a five-cell neighbourhood while the Swarm Force model employs a four cell neighbourhood and a revised cell selection formula. Because our interest is in the addition of force and not in the effects of neighbourhood and cell selection we present Kirchner's model in these results running with our four-cell neighbourhood and cell selection rules; in other words, the difference between the Kirchner and Force cases, below, is the difference between non-force and force-enabled models with otherwise identical neighbourhood and cell-selection rules.

Our analysis proceeds as follows. We first validate our model against Kirchner's using his measure (complete exit) and with our model's injuries disabled to ensure that a complete exit is possible. Then we show both models (again with injuries disabled) under our fixed-iteration measure to provide a reference between measures. Finally we enable injuries in our model using our fixed-iteration measure to test the importance of incorporating injuries directly within the model.

Model Parameters

We have adopted the fixed parameter values that Kirchner used wherever possible. Except as noted below, parameters were set as summarized in Table 1.

Table 1: Parameter table with default settings

Parameter	Value	Description
Grid size	63×63	Outer boundary of modelled space (includes wall cells)
Agent Pop.	1116	Starting agent density (30% of available cells)
α	0.3	Probability of dynamic boson diffusion
δ	0.3	Probability of dynamic boson decay
$\bar{\rho}, s_\rho$	5, 1	Normal distribution parameters of agent pushing forces
χ	3ρ	Threshold of forced cell selection (ρ = agent's pushing force)
φ	see below	Injury threshold (agents experiencing larger force are injured)

The model was implemented using the Repast framework for Java [7]. Graphical results indicate the average of 20 runs made using the parameter values indicated.

Comparison to Kirchner's model

The simplest possible comparison with Kirchner's model is to compare it with the results of our model with injuries disabled. The most important results presented by Kirchner are in his figure 5. These results show the time for all agents to exit under the influence of varying k_S (in figure 5a) and varying k_D (in figure 5b). In this comparison we have set the threshold for injuries in the Force model arbitrarily high, so that agents will not become injured but will otherwise experience the modelled force effects. We have chosen to present results for fixed k_S and varying k_D (Kirchner's figure 5b) because these are the most revealing results.

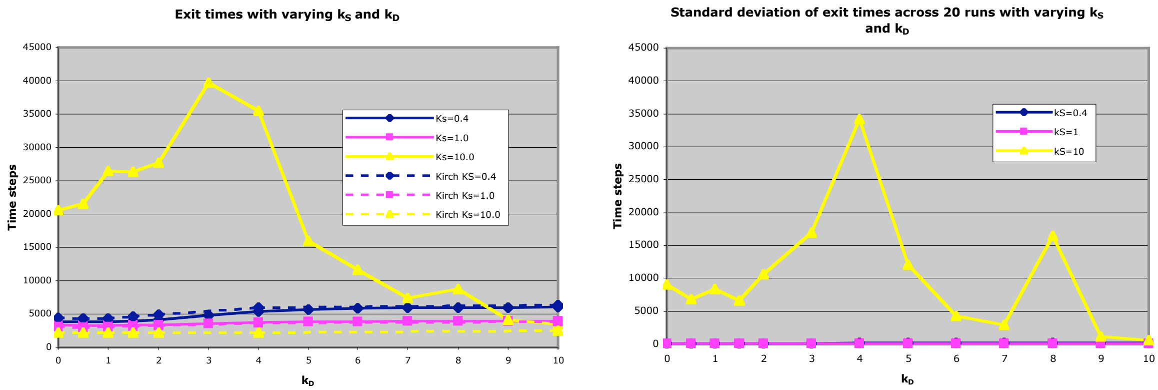


Figure 2: Exit Time Comparison of Kirchner and Swarm Force model. The exit time is the number of iterations required for all agents to exit the model. Twenty runs were made for each combination of parameter values, yielding a distribution of exit times across runs for each point. Each point in the left chart represents the mean of this distribution. Each point in the right chart represents the standard deviation of this distribution[4]. Kirchner's results are indicated with dotted lines, while our results are indicated with solid lines.

Kirchner's results indicate that faster exit times are obtained when values of k_S increase. When values of k_D increase, this results in a slightly longer time to exit. When k_S is low these patterns are mirrored by the Swarm Force model. When k_S is high, however, the Swarm Force model produces a dramatically different effect. The number of iterations required to exit range from 800% to 1600% of Kirchner's model when k_D is below 5. Where Kirchner's model indicated a clear benefit to high values of k_S, the Swarm Force model suggests high costs for this parameter setting.

To understand these results we must look at the effect of the k_S parameter on the exiting agents. This parameter controls individual agents' desire to move directly to the door. A global effect emerges from high k_S values is that the area around the door becomes highly congested. The entire population of agents moves toward the door, leaving few to no empty spaces within the crowd. We describe this quality independently of the size of the crowd in numbers or space by calling it a *high-density* crowd. Conversely, a *low-density* crowd has the quality of being spaced out with empty cells between individuals.

[4] The peak at k_S=10, k_D = 8 in both charts is due to a single outlier run in which agents required 80 000 steps to exit. Excluding this run, the mean exit time of 5044 and standard deviation of 2051 follows the graph trends.

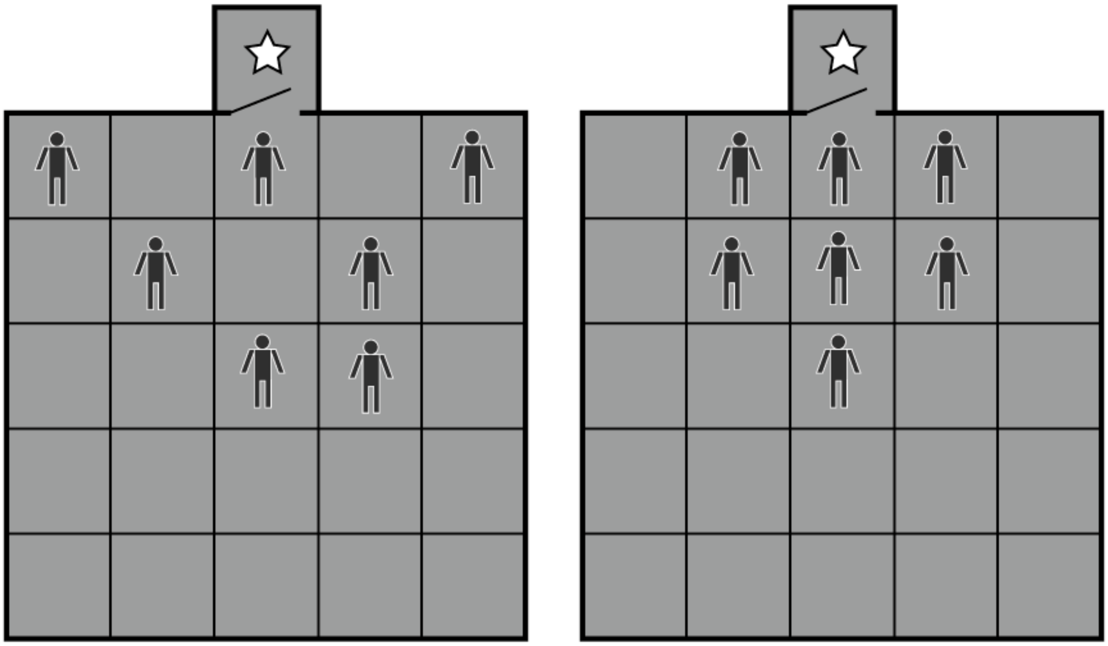


Figure 3: Low density crowd (left) and high density crowd (right)

The differences between Kirchner's model and the Force model are minimal when the influence of k_S is low. When k_S is low there is little pressure for agents to move toward the door and agents tend to be dispersed yielding a low density crowd. This low crowd density means lots of space around agents and therefore agents are infrequently frustrated in their desired movements by the presence of other agents; since agents must be blocked by other agents in order for force to be exerted, these low crowd densities translate into less force and therefore close agreement with the non-force model. In addition, the presence of empty cells means that when force is generated it is not easily transmitted throughout the crowd.

Although the Swarm Force model (with injuries prevented) predicts similar exit times to Kirchner's model when k_S is low, the model displays a dramatically different pattern when k_S is high. Times to exit increase significantly under these parameter values, particularly when k_D is low. Under these conditions agents move directly toward the door at the outset of the simulation, creating a high density crowd. As k_S causes agents to prefer cells leading to the door, and as there are few open spaces within the crowd, agents tend to select occupied cells. With the selection of occupied cells, agents begin to push and this changes the dynamics of the system significantly from Kirchner's results. This effect disappears as values of k_D increase because the importance of k_S is mathematically decreased. This decreased influence of k_S results in less co-ordinated movement toward the exit, lower density, more space around agents, less force applied and transmitted – hence fewer deviations from the non-force scenario.

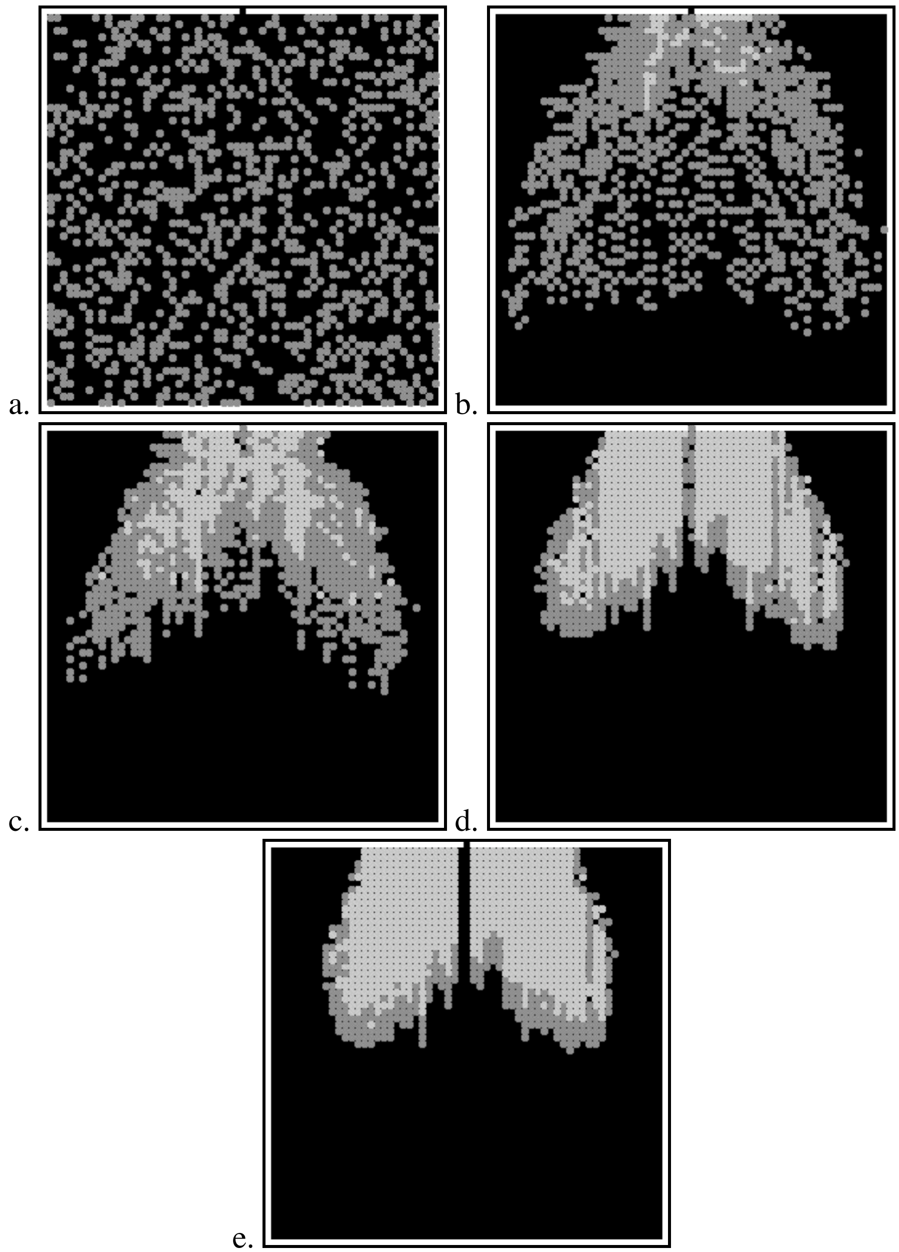


Figure 4: Sample run of slow exit due to force. k_S = 10; k_D = 3; injuries off (a) Initial model conditions at first time step. Agents *(dark grey circles)* distributed throughout space delineated by walls *(white border)* must exit through door *(black cell in wall)*. (b) Conditions at time step 18, note agents from lower corners take longer to reach the door area because diagonal movement is not possible. (c) Conditions at time step 35. Crowd around door now capable of efficiently transmitting force, and many agents *(light grey circles)* are now bypassing normal cell selection due to excessive force. (d) Conditions at time step 66. Note free movement of agents on centre aisle. (e) Conditions at time step 118. Aisle has exited, agents at edges of aisle and not at rear are light grey and cannot step sideways due to pressure from behind. This stable pattern can persist for many time steps.

The longest exit times obtained when k_S=10 and k_D < 5 show characteristic patterns as the model unfolds with force on but injuries prevented (see figure 4). The agents pack themselves around the door at the outset of the model. There are few empty spaces within the crowd. Cell-selection rules, under the influence of high k_S and low k_D, provide that the rear-most ranks of agents will push toward the door. Forward of these agents, agents experience the combined forces of the agents behind them and lose their ability to select their own cells; all agents forward of the last few ranks are forced to join the push toward the front. Large additive forces are brought to bear upon agents at the front. Our discussion now divides the agents into two groups: those in the area we call the *aisle* (a rectangular region, centred on the door, that extends front-to-back through the crowd), and those outside this region.

For agents not on the aisle, the fully-packed crowd pushes forward upon the front agents who are pinned against the wall. Although the rear agents could probabilistically select a cell that is not toward the door, the high k_S value provides that this is unlikely; forward of these agents there is no opportunity for movement as agent cell selection is bypassed due to the high force. In short, non-aisle agents are effectively jammed in position.

For agents on the aisle, the crowd pushes forward, and an agent at the front is forced to exit. This opens a space in the crowd which is immediately filled by an agent stepping forward. As time progresses, agents in the aisle are able to exit and the aisle gradually clears.

Agents not in the aisle or at the rear are not able to step sideways into the empty aisle because they are being forced to select the cell in front of them by the combined force of the agents standing behind them. This leads to a situation in which only the rear-most agents in the model are free to move. At this point, progress depends on one of the rear-most agents probabilistically moving laterally into the aisle. This process is laboriously repeated until enough agents have exited to alter the force characteristics such that agents close enough to the door (who prefer a lateral step to a forward step) are no longer pinned in place.

It should be noted that this unrealistic exit scenario is obtained only when the agents have an infinite capacity to experience force without becoming injured. When agents become injured they provide force breaks within the crowd which significantly alter exit dynamics (see below).

The distribution of results (increasing exit times) as k_D is increased from 0 to 3, is explained by differences in the initial packing of agents around the door. Higher values of k_D create a pack near the door that is shorter along the front wall and longer front-to-back. This results in a longer exit time because the exit takes place by slowly peeling off the ranks from the rear, and there are more ranks when k_D is higher. The value of k_D does not affect the formation of or behaviour of the aisle. As k_D exceeds 3, the decreasing exit times are due to the k_S parameter's reduction in influence as k_D becomes more important in cell selection.

The results of this analysis support our claim that force effects are extremely important to studying crowd behaviours and hence for crowd models. The introduction of a simple force model of pushing without injuries results in dramatically different exit behaviour in dense crowd conditions. This supports our thesis that crowd models must not abstract away issues of force.

Comparison of measures

As discussed above, it is necessary when analysing injuries to use a measure that does not require all the agents to exit due to the potential for the door to become blocked by an injury. We ran both the Kirchner and the Swarm Force model through 350 iterations and measured the number of agents exiting within that time. The value 350 was selected because it allows for the initial stages of model exit to be completed, while precluding a complete exit for the fastest runs (which would lead to a floor effect obscuring differences between fast runs).

In inspecting graphs of the present results with respect to the ones just described, bear in mind the reversal in sense of the Y axis: *more* agents exiting is analogous to *fewer* agents remaining after a fixed number of iterations.

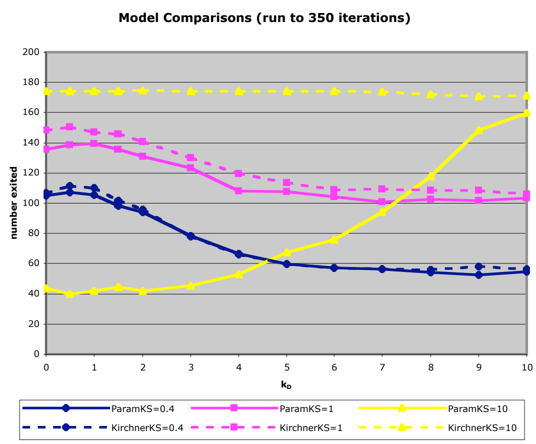


Figure 5: Number of exits comparison of Kirchner and Swarm Force Model.

There are broad similarities in the pattern of results between the new run-to-350-iterations measure and Kirchner's measure. As before, when k_S is 0.4 or 1.0 we see an improvement as k_D increases. In addition, these values of k_S produce agreement between the Swarm Force model and the Kirchner model. As before, Kirchner predicts maximal exit rates when k_S is 10, while the Force model disagrees, showing low agent exits within 350 iterations.

This demonstrates two points; first, the outcome of the complete exit is visible from the crowd's progress in the early stages of the model's execution. Second, the new technique measures similar model effects to the old technique, and will allow us to relate our exploration of modelled injuries (below) with Kirchner's data despite the difference in measures.

One noticeable difference between the two measures concerns the Force model's results when $k_D < 5$ and $k_S = 10$. In the run-to-350-iterations measure the curve is smooth and does not experience a peak around k_S=3, as was apparent in Figure 2. This is a reflection of the timing of the exits in the Swarm Force model. While the run-to-350 measure tends to explore the creation of the aisle region, the run-to-completion measure also explores the slow exit phase of the non-aisle agents.

Effects of injuries

With runs to 350 iterations, we can enable the injury parameters of the model in order to assess the effects of injuries on exit behaviours of agents.

In order to cause agent injuries it is necessary to determine another threshold value in the agent's injury response. When an agent experiences force greater than the threshold, φ, they are deemed to be injured. The φ parameter represents hardiness; the larger the φ parameter, the more force the agents can experience before becoming injured.

Like Kirchner, we have yet to determine the correspondence between model units of time, space and force and the physical units experienced by real crowds. Rather than selecting a threshold arbitrarily, we have elected to characterize this particular variable across a range of values. (The performance of a particular individual is hypothesised, then, to lie at some point along this axis.)

In gathering model data, we have selected representative values of k_S used by Kirchner in his paper for explorations of other parameters. We have coupled these values with low, medium and high values of k_D. The injury threshold was varied between 15 and 225 bosons.

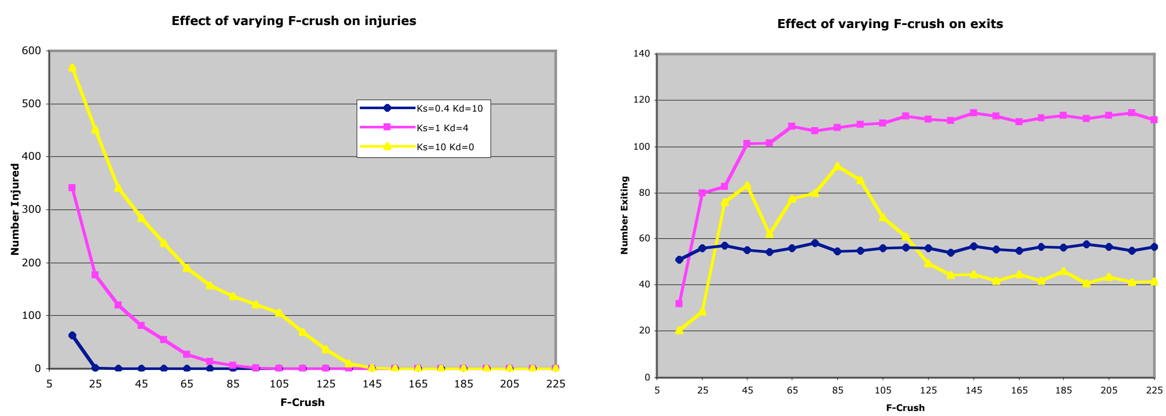


Figure 6: Effect of varying φ on number of injuries and exits

For each parameter set, we explored the injury space up to a point where the threshold was too high for any agent to become injured. Unsurprisingly, as the increasing threshold made it require more force to become injured, injuries decreased (see figure 6). In terms of numbers of injuries, as k_S increases (for a fixed injury threshold) more agents become injured.

The parameter settings that were the best for preventing injuries were different from the parameter settings that enabled more agents to exit. The best exit rate was produced by the moderate value of k_S (as visible in Figure 5 as well). A lower exit rate was obtained from the lower value of k_S. A non-monotonic relationship was found for high values of k_S: at high and low injury thresholds the number of agents exiting was low; for injury thresholds between 35 and 95, however, exit rates were substantially higher.

Now let us consider the mechanisms that underlie these results. In general, the pressure to move to the door – and hence the crowd density – has a dominating effect on both injury production and numbers of agent exits. Higher crowd densities when $k_S >> k_D$ produce more injuries, while modest crowd densities when k_S and k_D are both moderate produce the best numbers of exits. Low crowd densities brought on by $k_S << k_D$ result in few injuries but also few exits.

When k_S=0.4 and k_D=10 the lack of injuries is a consequence of the lack of force being exerted in this scenario (due to low crowd densities just discussed). The two other conditions are more revealing.

When k_S=1 and k_D=4 there are three separate phases displayed. The first phase (occurring when $\varphi < 50$) sees a substantial drop-off in injuries accompanied by a substantial increase in the number of agents exiting. The second phase (occurring when $50 < \varphi < 100$) sees a continuing decline in injuries until no agents are injured, but only a modest increase in number of agents exiting. The third phase ($\varphi > 100$) involves a consistent lack of injuries and no change to the exit rate. This indicates that a moderate number of injuries does not necessarily have a large effect on the number of agents exiting when k_S=1 and k_D=4. Larger numbers of injured agents do have an impact on number of exits.

When k_S=10 and k_D = 0 the cell selection policy provides a strong pressure to move toward the door, and consequently high density near the door. The φ parameter seems to have a simple relationship with the number of agents injured: we see a decline in numbers of injuries as φ increases. Numbers of exits, however, display a complex relationship against φ. There are four phases: (1) when $\varphi < 45$, increasing force threshold results in an increase in exits, which then (2) roughly plateaus while $45 < \varphi < 90$. (3) When $\varphi > 90$, number of exits decrease until (4) there are no more injuries and the exits levels off.

The interpretation of this four-phased relationship when k_S=10 and k_D = 0 is made simpler by the well-known fact in crowd research that obstacles can actually facilitate exits in emergencies. This is because obstacles disrupt symmetrical patterns of force and allows shelter for exiting agents from the forces behind them [8]. This pattern is evident in our model when injured agents act as force breaks within the crowd (figure 7). The swell in agent exits when $45 < \varphi < 90$ is attributable to the protective effect of these injured agents on the others. Practically speaking, agents just forward of the injured agents are free to make their own cell selections while others are pinned in place; this increased flexibility means that agent exits are facilitated, and avoids the kind of stasis evident in figure 4e, where density was high but we artificially prevented injuries. When $\varphi > 90$ stasis reappears as injured agents are too sparsely distributed for injuries to have a protective effect. When $\varphi < 45$ many agents are injured and this affects both the number of agents capable of moving, and the probability of injured agents blocking the door.

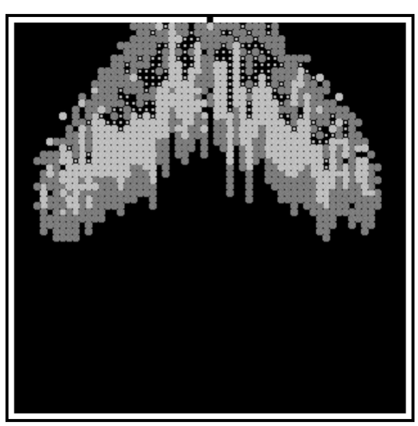


Figure 7: Effect of injured agents *(small grey dots)* as force break. (k_S = 10, k_D = 0, φ = 55, timestep 53) Agents in front of the force break are making their own cell selections *(dark grey)* as opposed to agents behind that are not making their own cell selections *(light grey)*. Note that injuries are not randomly distributed throughout the crowd, but rather are concentrated in the centre of the crowd, closer to the door.

GENERAL DISCUSSION

Having explored the characteristics of the Swarm Force model, we now turn to the implications of this exploration on the two central issues of this paper: the predictions made by Kirchner's force-free model and the centrality of force to models of crowds.

First we consider the overall relationship between the injuries-enabled force model and Kirchner's non-force model. Table 2 summarizes the number of agents exiting for the values of k_S and k_D investigated.

Table 2: Comparison of Kirchner and Swarm Force model with force and injuries on.
Swarm Force model data is given both as mean[5] and range across all sampled values of φ.

Parameters (k_S, k_D)	Kirchner Model (agents exiting)	Swarm Force Model (mean agents exiting)	Swarm Force Model (range agents exiting)
(0.4, 10)	59	55.6	51.0-58.1
(1, 4)	120	104.3	31.65-114.8
(10, 0)	170	55.1	20.4-91.6

It is clear from Table 2 that force has a noticeable impact on the number of agents able to exit in the first 350 timesteps of the simulation. In all cases the range of agents exiting is below the number of agents exiting in Kirchner's simulation. This effect ranges from quite subtle (when k_S=0.4 and k_D=10 and the principal difference between the models is simple pushing rather than injuries) to a 46-88% range in exit reduction (depending on values of φ, when k_S=10 and k_D=0 and the Swarm Force model predicts injuries in large numbers).

On the question of facilitating agent exits, Table 2 demonstrates that when force is taken into account, our model makes a very different prediction from Kirchner's model. Specifically, our model suggests that conditions that reduce crowd density to moderate levels (namely moderate values of k_S and k_D) will result in many more agent exits than will high crowd densities (as Kirchner predicted).

Ability to replicate crowd paradoxes

Helbing has observed that crowds exhibit paradoxical collective behaviours when speeds and commitment to courses of action increase. These behaviours are characterized by simple paradoxical phrases like *faster is slower* and *freezing by heating*. The Swarm Force model demonstrates these two paradoxes.

[5] We present the mean data to provide the reader with an overall sense of the effects of injuries caused at each parameter level. From an analytic perspective, however, an average across values of φ may not be warranted as we hypothesize that there is a single true value of φ for each individual. We prefer to analyze the range data as this provides a best-case and worst-case prediction of the effects of injuries on the exiting crowd.

The *faster is slower* paradox [1] refers to the fact that crowds passing through a bottleneck may make slower progress when individual's velocities are higher due to the enhancement of clogging at the exit and interpersonal force effects. Our model directly demonstrates this principle as moderate speeds toward the door (brought on by moderate parameter values) yield the best exit rates. Faster speeds (brought on by $k_S >> k_D$) yield fewer agent exits in a fixed interval because of clogging and also force effects.

The *freezing by heating* paradox [9] concerns particle energies: High particle energy can result in stasis where lower energies permit free movement. Our model directly demonstrates this principle: High particle energy (in the case of pushy dense crowds) demonstrates the stasis effect of figure 4e that is only broken when the energies are dissipated through force breaks – a demonstration of freezing by heating.

Force is an essential aspect of crowd models

This brings us to a matter raised in the introduction where we suggested that there were three possible positions that a model could adopt on the question of crowd forces.

The first position suggested that modelling force was non-essential and that results could be accurate with or without force. Based on our investigation of the same model both with and without a force implementation we believe that this position cannot be supported. The addition of force to the model, with or without injuries, dramatically alters the exit behaviours of modelled agents.

The second position suggested in the introduction is that force can be treated post-facto by applying some correction factor to the data. The complex relationships of figure 6, and our understanding of this relationship by reference to the protective effects of injuries, suggest that the effects of force cannot be modelled after the fact by applying a simple "force penalty" measure.

Instead, our results support the third conclusion, that force is characteristic of crowds and cannot be abstracted from a crowd model without significantly changing the behaviour of the model.

CONCLUSION

This paper has considered two basic questions. The first being how force should be treated in an individually-based model of crowds. The second being an exploration of the force-related dynamics of the Swarm Force model over and above the basic functionality of Kirchner's model.

As regards the first question, by demonstrating force-related non-linear changes in numbers of agents exiting we have shown that force must be treated as an essential aspect of crowd models. Important effects like the facilitative aspect of injuries are lost if force is not considered, or if it is treated only as an add-on parameter that scales the output of a model that otherwise does not consider force effects.

As regards the second question, we have shown a much more complex pattern of exit times as compared with Kirchner's model – both with his own measures, and also using our own measure that enables investigation of injuries that can block the door. Repeating elements of Kirchner's exploration of his parameter space, our addition of force interacts with his parameters to produce differences, sometimes non-linear, in numbers of injuries and exits. The Swarm Force model also reproduces the known fact that force breaks within the crowd (here in the form of injuries) can facilitate exits in dense crowds motivated to exit quickly. These effects cannot be replicated by models that do not model forces as force is key to their action.

Further work remains to be done on the Swarm Force model before it can be considered to be predictive of real scenarios. In particular, the model needs to be related to real-world parameters like distance and time, and including human capabilities relating to perception, force thresholds, and strength.

REFERENCES

1. Helbing, D., Farkas, I. & Vicsek, T. (2000a) Simulating dynamical features of escape panic. *Nature, 407,* 487-490.
2. Kirchner, A., Schadschneider, A. (2002) Simulation of evacuation processes using a bionics-inspired cellular automaton model for pedestrian dynamics. *Physica A, 312*, 260-276.
3. Bonabeau, E., Dorigo, M. & Theraulaz, G. (1991) *Swarm intelligence: From natural to artificial systems*. New York: Oxford University Press.
4. Fruin, J. (1993) The causes and prevention of crowd disasters. In: Smith, R.A., Dickie, J.F. (eds): *Engineering for crowd safety*. Elsevier: New York.
5. Kirchner, A., Nishinari, K. & Schadschneider, A. (2003) Friction effects and clogging in a cellular automaton model for pedestrian dynamics. *Physical Review E, 67, 056122*.
6. Henein, C.M. & White, T. (2005) Agent-Based Modelling of Forces in Crowds. In: Davidsson, P., Logan, B. & Takadama, K. (eds.) *Multi-agent and multi-agent-based simulation* (Lecture Notes in Computer Science v. 3415). New York: Springer, 173-184.
7. Collier, N. (2003) *RePast: An Extensible Framework for Agent Simulation.* Available online: http://repast.sourceforge.net/
8. Helbing, D., Farkas, I.J., Molnár, P., Vicsek, T. (2002) Simulation of pedestrian crowds in normal and evacuation situations. In: Schreckenberg, M., Sharma, S.D. (eds.): *Pedestrian and evacuation dynamics*. Springer-Verlag, New York, 21-58.
9. Helbing, D., Farkas, I. & Vicsek, T. (2000b) Freezing by heating in a driven mesoscopic system. *Physical Review Letters, 84,* 1240-1243.