**EVALUATING FORCE-BASED SIMULATION MODELS BY PEDESTRIAN TYPE**

**A DEMAND FOR NOVEL DATA ACQUISITION APPROACH**

**Summary**

Force based simulation models have been used widely to simulate crowd’s microscopic level. Measuring the effect of model’s parameters is important to gain a better understanding on observations including escape rate, and desired speed satisfaction. This study uses Nomad and social force models to investigate crowd observations when varying crowd population by different pedestrian ages (young, adult, and elderly people) on their parameter’s distribution in two scenarios including unidirectional and bidirectional flows of bottlenecks. Through our experiment, we find that the observations depend on the percentage of pedestrian types and their placement in the crowd. It aims to raise the need of a novel data acquisition approach that can distinguish pedestrian type’s parameter distribution when simulating crowds at difference people type-oriented or mixed venues rather than using the same parameter distribution for interchangeable pedestrians detected by camera-based capability. A discussion for further research is conducted then.

1. **Introduction**

Crowd simulation plays an important role in quantitative crowd dynamics understanding and layout design assessment especially in crowd disaster (**Helbing, 2014)**. In microscopic level, the motion of each individual *p* is defined by Langevin equation:

|  |  |
| --- | --- |
|  | (1) |

where integrating the forces acting on *p* and captures random influence and uncertainty. The forces mainly comprise subject’s desired acceleration force and repulsive forces being constituted by neighbour interaction and obstacle repulsion at time *t*. Nomad model **(Hoogendoorn, 2003)** and social force model **(Helbing, 1995)** recently attract more studies when they are efficient to simulate motion base cases and self-organization phenomena as mentioned by the latest survey in the field **(Duive and Hoorgendon, 2013)**. Each of above two models has possessed a long-life modification period in order for simulating the additional factors affecting individual’s acceleration or being easier towards calibration process. Therefore a huge number of variants increase by the time by their original authors. Main chronological variants of each model are summarized briefly below:

|  |  |  |  |
| --- | --- | --- | --- |
| **Year** | **Author** | **Model-based** | **Main contribution** |
| 2003 | Hoogendoorn, Bovy | Nomad | Original version of Nomad Model |
| 2009 | Campenella, Hoogendoorn, Daamen | Nomad | Add an extra repulsive force created by opposite flows |
| 2012 | Daamen, Hoogendoorn | Nomad | Omit obstacle force for calibration process |
| 1995 | Helbing, Peter Molnar | Social-force | Original version of Social Force |
| 2000 | Helbing, Farkas, Vicsek | Social-force | Add Friction and Velocity dependence |
| 2005 | Helbing, Buzna, Johansson, Werner | Social-force | Add angular component on interaction force |
| 2008 | Johansson, Helbing, Shukla | Social-force | Add angular component on interaction force |
| 2010 | Moussaid, Helbing, Theraulaz | Social-force | Add Group behaviour effects |

**Table 1**. Nomad and Social Force-based variants

In this study, we only select simple variants including **(Daamen and Hoogendoorn, 2012)** and **(Johansson and Helbing, 2005)** for our study’s purpose since they have sufficiently model’s parameter values. Details of variants used in this work are represented as follows:

* 1. **Nomad Model**

Nomad model **(Daamen and Hoogendoorn, 2012)** is an agent-based model that predicts walking acceleration of a pedestrian as a function of the free velocity , the current speed , the position , and distance between pedestrians *p* and *q* as follows:

|  |  |
| --- | --- |
|  | (2) |
| = | (3) |

here is the set of pedestrians who are standing in front of pedestrian *p* (checked by the constraint ) and where

|  |  |
| --- | --- |
|  | (4) |

and

|  |  |
| --- | --- |
|  | (5) |

and

|  |  |
| --- | --- |
|  | (6) |

where stands for desired walking direction pointing from the current position of pedestrian *p* to the target (exit door). The factor has the length equal to the free speed of pedestrian *p* . The model has four pedestrian-specific parameters that need to be set in simulation environment as in Table 2:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Component** | **Description** |
|  | Desired Acceleration | free speed of pedestrian *p* |
|  | Desired Acceleration | acceleration time |
|  | Interaction Force | interaction strength constant |
|  | Interaction Force | interaction range |

**Table 2**. Nomad model’s initial parameters

* 1. **Social Force Model**

Social force model **(Helbing, 2000)** and **(Johansson and Helbing, 2005)** represents that a pedestrian *p* at time *t* is trying to move with a certain desired speed in a desired direction pointing from pedestrian *p*’s current position to his target position. Therefore, pedestrian *p* tends to correspondingly adapt his actual velocity with a certain acceleration time . The acceleration time represents pedestrian *p* changes its current velocity and return to its desired velocity. Pedestrian *p*’s acceleration at time *t* also depends on repulsive forces coming from surrounding pedestrians and obstacles. The repulsive force’s directions are represented in Figure 1. The model’s formula is represented in equations (7-16).



**Fig 1**. Repulsive forces and on pedestrian *p* created by pedestrian *q* and wall γ

|  |  |
| --- | --- |
|  | (7) |
| = | (8) |

where is desired speed of pedestrian *p* and varies over time, given by:

|  |  |
| --- | --- |
|  | (9) |
|  | (10) |

where and are the initial desired speed and the maximum desired speed of pedestrian *p*, respectively. In social force model is constrained by constant value ***c >* 1**.

In equation (9), is considered as panic parameter model of pedestrian *p*. It illustrates how strongly pedestrian *p* aligns his preferred velocity with the motion of crowd surrounding him, given by equation (11) as suggested by **(Helbing, 2005) and (Andreasen, 2010)**:

|  |  |
| --- | --- |
|  | (11) |

where is computed by average actual speed in the desired direction. Equation (9) is transformed into equation (12) for the condition at time *t*=0

|  |  |
| --- | --- |
|  | (12) |

When is going down in equation (11) as pedestrian *p* is in high density place (e.g bottle neck scenario), implies → 1 which implies → as in equation (9).

When is going up, it implies → 0, which implies → . Since > by ***c >* 1**, it means that when average velocity is going up, the desired force going down, and vice versa. When is higher than desired force has negative direction to decelerate pedestrian *p*’s actual speed.

Interaction force created by neighbour pedestrian *q* is given by equation (13)

|  |  |
| --- | --- |
|  | (13) |

where is the angle between pedestrian’s *p* velocity direction and the vector pointing from *p* to *q*, and is the distance between pedestrians *p* and *q*. is an extra weight component to emphasize whether pedestrian *p* pays attention to other pedestrians behind him, the component is given by equation (14)

|  |  |
| --- | --- |
|  | (14) |

where is angular component of the model, and set , and , implies that 1. When, it means pedestrian *p* doesn’t pay attention to other people behind him. When 1, the interaction force is modified by the angular component.

The interaction force is given by equation (15)

|  |  |
| --- | --- |
|  | (15) |

where factors A and B are model parameters, represent the strength of interaction force and how fast the force decreases based on the distance between pedestrians *p* and *q*. Factors and are radii of respective pedestrians *p* and *q*. Factor is the unit vector pointing from pedestrian *q* to pedestrian *p* to illustrate the force direction making pedestrian p avoid pedestrian *q*.

The obstacle force between pedestrian *p* and wall γ in equation (8) is represented in equation (16)

|  |  |
| --- | --- |
|  | (16) |

here U is a model parameter to represent the strength of obstacle force, and is the unit vector pointing from wall γ to pedestrian *p* to make the agent avoid the wall. Because of exponent component in equation (16), the obstacle force always satisfies the condition.

To summary, the social force model comprises parameters that need to be set at initial simulation time as in Table 3:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Component** | **Description** |
|  | Desired Acceleration | Initial desired velocity |
|  | Desired Acceleration | Acceleration time to reach desired speed |
|  | Desired Acceleration | Constant to find maximum velocity |
|  | Interaction Force | Angular component |
| *A* | Interaction Force | Interaction strength |
| *B* | Interaction Force | Interaction range based on distance between *p*, *q* |
| U | Obstacle Force | Obstacle interaction strength |
|  | Simulation | Radii of pedestrian *p* in simulation environment |

**Table 3**. Social Force model’s initial parameters

**2. The model’s parameters acquisition through calibration process**

This section presents calibration process from above author’s studies for finding parameter’s values.

**2.1. Nomad model’s parameter acquisition**

The study of **(Daamen and Hoogendoorn, 2012)** considered model parameters in Table 2 different between pedestrian types including children (to age 14), adults, and elders (age 65 and older). Finding parameters for these types was performed in lab environment at emergency exit door scenario with different hat colour worn by pedestrian types. Camera-based approach was used to record individual’s trajectories and calibrate for finding Nomad parameters based on maximum likelihood estimation. Table 4 shows corresponding parameter values for the model. The difference in free speed between pedestrian types is small whereas other parameters varies considerably as equations (17-19)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Parameters** | **Children** | | **Adult** | | | **Elderly** | |
| Avg. | St. Dev. | Avg. | St. Dev. | | Avg. | St. Dev. |
| (m/s) | 1.04 | 0.05 | 1.00 | | 0.06 | 0.98 | 0.03 |
| (s) | 2.44 | 0.60 | 2.31 | | 0.46 | 2.08 | 0.27 |
| (m/s2) | 0.90 | 2.20 | 0.63 | | 1.23 | 0.52 | 0.13 |
| (m) | 0.48 | 0.12 | 0.52 | | 0.10 | 0.49 | 0.06 |

**Table 4**. Nomad parameter’s values by pedestrian types (estimated by average and standard deviation)

|  |  |
| --- | --- |
|  | (17) |
|  | (18) |
|  | (19) |
|  | (20) |

The table give useful information on how pedestrian types behave differently in emergency situation. Elderly people are more aggressively to walk with their desired speed than children do. For the interaction strength parameter, standard deviation was not reliable because the author’s calibration experiment was performed in various scenarios to get average values. Therefore, we only use the mean values for our simulation without considering standard deviation values. However, the strength between children is strongest comparing to those values between adults and elderly in a population with a large heterogeneity. In the last parameter, children also have the lowest value; it implies that the interaction force affecting children can be easier changed by distance than it does on elders and adults.

**2.2 Social force model’s parameter acquisition**

**2.2.1 Desired acceleration component**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Reference** |
| (m/s) | avg. = 1.34, st. dev. = 0.26 | (Helbing, 1995) |
| avg. = 1.3, st. dev. = 0.3 | (Helbing, 2005) |
| (s) | 0.5 | (Helbing, 1995) |
| 1.0 | (Helbing, 2000), (Helbing, 2005) |
|  | 1.3 | (Helbing, 1995), (Helbing, 2005) |

**Table 5**. Social force model’s parameters in desired acceleration component in simulation environment

Table 5 presents parameters in simulation environment from social force model’s authors. They are efficient when producing successfully crowd phenomena such as lane formation, stop and go waves, faster-slower effect. However, to find parameter values from actual pedestrians recorded by camera approach, authors **(Johansson and Helbing, 2008)** and (**Zeng, 2014**) had to assume that desired speed of a pedestrian is his maximum speed and does not change over the time. This is different from its original definition in equation (9). Reaction time was computed statistically as the duration letting people catch up their free speed.

**2.2.2 Interaction Force component**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Reference** |
| *A* (m/s2) | 3.0 | (Helbing, 2005) |
| *B* (m) | 0.2 | (Helbing, 2005) |
|  | 0.75 | (Helbing, 2005) |

**Table 6**. Social force model’s parameters in interaction force component in simulation environment

Extracting interaction force component’s parameters in Table 6 from actual pedestrians was performed by **(Johansson and Helbing, 2008)** in calibration process by considering in equation (8) constant over the time and equal to the maximum velocity of each experimental pedestrian. Each video in experiment also generated a broad range of combinations for above parameters caused by its detected pedestrians. Evolutionary optimization techniques was then applied to find the best combination (*A* = 0.42 ± 0.26, B = 1.65 ± 1.01, 𝜆 = 0.12± 0.07) since the fitness function was the distance error between real trajectory tracking and the new position predicted by the model. The best combination was then applied for all pedestrians in simulation environment by its distribution.

**2.2.3 Obstacle Force component**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Reference** |
| U(m2/s2) | 10.0 | (Helbing, 1995) |

**Table 7**. Social force model’s parameters in obstacle force component in simulation environment

To our best knowledge, calibrating obstacle force hasn’t yet performed by the model’s authors by either camera or other data acquisition approaches.

1. **Simulation scenarios for Nomad and Social force models by pedestrian types**

In general, measuring the effect of agent’s parameters in force-models has been considered as future studies of **(Weijerman, 2013)** and **(Sun, 2014)**. In this study, we focus on the impact of pedestrian type’s parameter distribution by creating two population prototypes. The first one *(Pdifferent)* uses different distributions on each parameter according pedestrian types. The second one *(Paverage)* is generated from normal distribution of the first prototype on each parameter. Situations representing actual place’s population percentages are constructed as in Table 8:

|  |  |  |
| --- | --- | --- |
| **Situation** | **Pedestrian Percentages** | **Situation simulation** |
| *S1N*, *S1S* | 0% children, 80% adults, 20% elders | working place |
| *S2N*, *S2S* | 60% children, 30% adults, 10% elders | school place |
| *S3N*, *S3S* | 10% children, 20% adults, 70% elders | retirement home |
| *S4N*, *S4S* | 30% children, 60% adults, 10% elders | shopping center |

**Table 8**. Situations *SiN* and *SiS* are used to simulate Nomad and Social Force models, respectively

In social force model, keeping the same differential trend in equations (17-20), we use following values for pedestrian types based on the mean and standard deviation values found in the model’s calibration process. *Maximum velocity is 5m/s*, with c = 1.3, ***we choose average desired speed of children= 2m/s, averaged acceleration time to reach desired speed = 1.25(s), A=1.25 force increasing unit, R=0.1 force change by distance. Std of all parameters= 10% mean***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Social Force Parameters** | **Pedestrian type’s parameters** | | | | | | **Sharing parameters** |
| **Children** | | **Adults** | | **Elders** | |
| **Avg.** | **Std.** | **Avg.** | **Std.** | **Avg.** | **Std.** |
| (m/s) | 1.8 | 0.18 | 1.3 | 0.13 | 0.8 | 0.08 | =1.3  = 0.75  U=10.0  meanradii=0.3  sdradii=0.05 |
| (s) | 1.25 | 0.125 | 1.0 | 0.1 | 0.5 | 0.05 |
| ***A***(m/s2) | 4.0 | 0.4 | 3.0 | 0.3 | 2.0 | 0.2 |
| ***B***(m) | 0.1 | 0.01 | 0.3 | 0.03 | 0.2 | 0.02 |

**Table 9**. Social force’s parameters for pedestrian types

To make the experiment simple, we assume other information including parameters c, , *U*, and radii are the same distribution between pedestrian types. Standard deviation values of parameters *A*, *B* are set to 0.

**3.1 Implementation Techniques**

Our simulation is developed with following configuration. Nomad and social force models are implemented on C library for performance purpose. The source code can be found at **(\*)**.

* Python version 3.4.1
* Numpy library version 1.8.1 is used to generate Gauss distribution for pedestrian’s parameter values with an error rate < 0.01.
* Matplotlib library version 1.3.1 is used to plot our measuring results.
* Pygame engine version 1.9 to visualize obstacles and update pedestrian’s position with a frame rate of 100 fps.

The simulation allows pedestrians start at a specific area and move to reach the predefined target. We use Euler’s method to update new velocity and position of each pedestrian as equation (20-21).

|  |  |
| --- | --- |
|  | (20) |
| V( | (21) |

where *p* is the position, *V* is the velocity, *a* is the total combinatorial acceleration given by Nomad model in equation (3) or total force given by force model in equation (7). is the time step and set 0.01second to perform real-time crowd modelling.

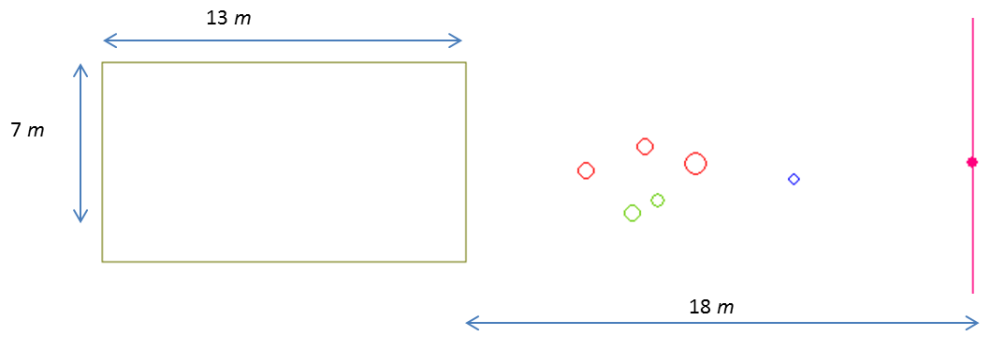
Cartesian coordinator system is applied on Pygame’s screen with a pixel factor to simulate the pixel number per meter. *O*(0,0) root coordinator is aligned at the centre of simulation screen.

**3.2 Simulation Scenario**

Our experiment is performed by two scenarios of motion base cases including unidirectional and bidirectional flows.

**3.2.1 Simulation scenarios for Nomad model**

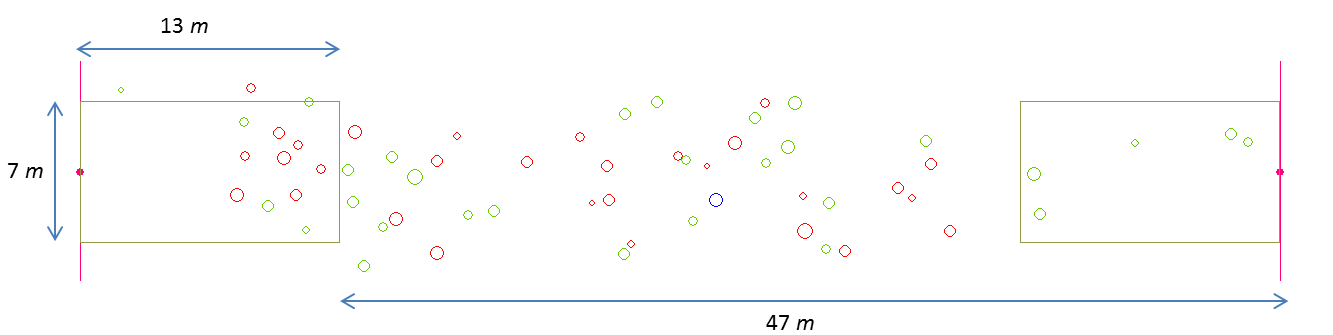
Unidirectional flow is designed for all pedestrians move from a start area to reach a target line. A start area is 48*m2*, and the distance from the closet area’s edge to the target line is 13*m*. Population are randomly placed in the start area. Pedestrian types use the same radii distribution with meanradii =0.3 and sdradii = 0.05. Colour configuration for our simulation is described in Table 10.



**Fig 2**. Unidirectional flow simulation for Nomad model

|  |  |
| --- | --- |
| **Colour** | **Description** |
| *Green* | Children |
| *Red* | Adults |
| *Blue* | Elders |
| *Yellow* | Start area |
| *Pink* | Target line |

**Table 10**. Colour Configuration

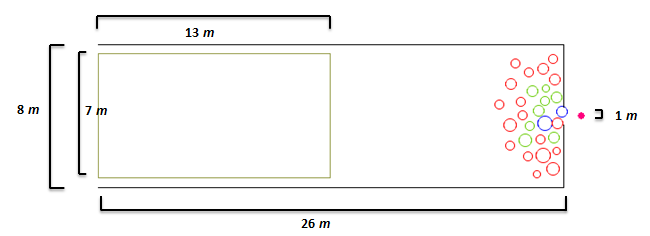


**Fig 3**. Bidirectional flow simulation for Nomad model

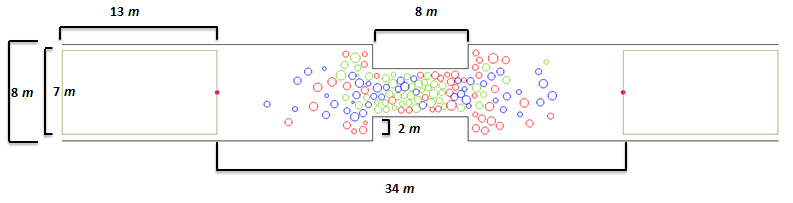
Bidirectional flow is designed for two symmetric start areas. They have opposite target lines correspondingly. Distance information is performed on Fig. 3.

**3.2.2 Simulation scenarios for Social force model**

In unidirectional and bidirectional flow scenario, we design obstacle walls for bottlenecks with following information in Figures 4 and 5.



**Fig 4**. Unidirectional flow simulation for social force model



**Fig 5**. Bidirectional flow simulation for social force model

To verify our simulation implementation suit to crowd phenomena capabilities of social-force model, we reproduced efficiently faster-is-slower effect in unidirectional flow when escaping a bottleneck from **(Helbing, 2000)**, and phenomena including lane formation, and freeze-by-heating effect in bidirectional flow from **(Helbing, 2005)**.

A yellow-start area is designed sufficiently to simulate the maximum population number up to 70 pedestrians (with pedestrian’s *meanradii*=0.3 and *stdradii=0.05*).

**3.2.3 Pseudo code Implementation**

* 1. **Crowd Observations**

In Nomad, we only measure first 50s, several simulations doesn’t run well because, people stand in front, more sensitive, make other people move backward.

Each population type is simulated *30* times at each population number to get average values on following crowd observations. *Unidirectional scenario duration measure = 100 seconds*, *bio = 200 seconds. Don’t get more, since more pedestrians move very slowly affect to measure escape rate*

**Plot distribution here**

Vary population number and percentages of pedestrian types

* + 1. **Average escape rate**

Maximum for uni-directional scenario = **70** pedestrians

Maximum for bi-directional scenario = **60** pedestrians

***Different 0.2 is significant comparing to partial obstruction change flow rate (0.2-0.6) Shiwakoti***

***Optimal pedestrian type percentages for maximize total escape rate***

***More lane creation, or more people same type together closer than other?***

**3.3.2 Desired speed satisfaction efficiency (**

**3.3.3 Pedestrian type placement in start area**

Which pedestrian escape first, percentages

**3.4 Effect of distribution impact on crowd evacuation**

**3.4.1 Change distribution of parameters**

**3.4.2 Vary crucial parameters and keep other**

1. **Further research discussion**

This section presents the impacts of data acquisition approach and the impact of pedestrian type in evacuation plan studies.

**4.2.1** ***What are optimal parameter values for each pedestrian type or a specific person by actual collected data?***

This question is useful for simulating accurately crowds at different venues and understanding fully parameter-awareness of a specific person on different scenarios.

Answering the question also involves technical implementation aspects about how to collect and manage data in the case of large-scale of participants.

We also found that desired force plays an important role when dealing with who will crush other people. This information heavily depends on panic information (computed by average speed and initial desired speed); however current start of the art in this area hasn’t yet found it. In fact, they assume that desired speed is maximum speed. ***(Plot analysis about initial speed)***

**(Jeroen and Hoorgendon, 2015)** use Bluetooth to detect pedestrian congestion in train station

**(Helbing, 2014)** uses GPS to monitor urban population on paramaters extracted from bottleneck

**4.2.2** ***Given a n-pedestrian capacity of a start area in the corridor leading to exit gate, how many percentage of each pedestrian type should be constituted in order to maximize flow rate and desired speed satisfaction, or minimize total escape time?***