UNIVERSITY OF WEST LONDON
The Career University

UWL REPOSITORY

repository.uwl.ac.uk

A motion planning method for simulating a virtual crowd

Xiong, M. and Tang, Shanyu ORCID logoORCID: https://orcid.org/0000-0002-2447-8135 (2014) A motion planning method for simulating a virtual crowd. Journal of Simulation, 8 (1). pp. 37-49. ISSN 1747-7778

http://dx.doi.org/10.1057/jos.2013.11

This is the Accepted Version of the final output.

UWL repository link: https://repository.uwl.ac.uk/id/eprint/3954/

Alternative formats: If you require this document in an alternative format, please contact: open.research@uwl.ac.uk

Copyright:

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy: If you believe that this document breaches copyright, please contact us at open.research@uwl.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

Rights Retention Statement:

A Motion Planning Model for Simulating a Virtual Crowd

Muzhou Xiong, Shanyu Tang

School of Computer Science,

China University of Geosciences, Wuhan 430074

email: mzxiong@gmail.com

Abstract

A model of motion planning for agent-based crowd simulation is one of the key techniques for simulating how an agent selects its velocity to move towards a given goal in each simulation time step. If there is no on-coming collision with other agents or obstacles around, the agent moves towards the designated goal directly with the desired speed and direction. However, the desired velocity may lead the agent to collide with other agents or obstacles, especially in a crowded scenario. In this case, the agent needs to adjust its velocity to avoid potential collisions, which is the main issue that a motion planning model needs to consider. This paper proposes a method for modelling how an agent conducts motion planning to generate velocity for agent-based crowd simulation, including collision detection, valid velocity set determination, velocity sampling, and velocity evaluation. In addition, the proposed method allows the agent to really collide with other agents.

Hence, a rule-based model is applied to simulate how the agent makes a response and recovers from the collision. Simulation results from the case study indicate that the proposed motion planning method can be adapted to different what-if simulation scenarios and to different types of pedestrians. The performance of the model has been proven to be efficient.-

Keywords: Motion planning, collision avoidance, collision response, agent-based simulation, crowd simulation

Introduction

Crowd simulation (Thalmann and Musse, 2007) has become an efficient tool for psychologists, sociologists, computer scientists, and even for architecture designers and urban planners to study the crowd behavior and its movement under diverse what-if scenarios in recent decades. Agent-based model (Nguyen et al., 2005; Pelechano et al., 2005), as a main and important method for simulating how an agent behaves and moves in a diverse environment, emphasizes on the issues of individual aspects, including pedestrian's psychological and social behaviours, communication amongst pedestrians, and individual decision making process.

A typical agent-based model usually holds two layered modules: a upper layer module for decision making and a lower module to deal with motion planning. In each simulation time step, the decision making part is executed first to generate a desired position. THE Motion planning part is then triggered to execute to produce velocity for agent to arrive at the desired position.

Agent will move along the direction until the end of the simulation step, if no potential collision is detected; otherwise the velocity will be adjusted based on the predefined trade-off between the preferred direction and the collision-free directions.

There is a large amount of existing work on collision avoidance for motion planning of multi-agent systems. Some of them Fox et al. (1997); Gayle et al. (2002); Jaillet and Simeon (2004) are aimed at controlling the motion of robotics in dynamic environments, but without considering the motion of the other agent involved in the possible collision. Similar to those methods for the motion planning of robotics, literatures Sud et al. (2007); Li and Gupta (2007) focused on the motion of multi-agent also regards the other agents as passively moving obstacle. If both of the agents use such methods, oscillation may happen, as described in (van den Berg et al., 2008) where *Velocity Obstacle* (VO) (Fiorini and Shiller, 1998) is used.This paper focuses on how to model the process of motion planning, and the modelling process of path-planning is beyond the scope of this paper. A process framework of motion planning is proposed to determine valid velocity set, sample candidate velocity, evaluate velocities and select a velocity to avoid potential collisions along the pre-determined moving path by path planning module. As for the velocity evaluation and selection, the paper proposed a improved RVO method for agent to avoid potential collision with other agent. *Reciprocal Velocity Obstacles* (RVO) (van den Berg et al., 2008) considers the mutual effect on the agents involved in the collision and provides a cost function for velocity selection. However, the meaning of a key parameter, namely *safety factor*, as well as its value selection, is not clear.

Regarding this, this paper proposed a method for selecting the value of the parameter for the RVO method, by considering some psychological issues and principles of crowd movement. The improved method does not guarantee generating collision-free movement of agent, which is actually true in some specific scenarios in real life. Hence, the paper also proposed a method for collision response to model how agent recover from a collision.

The potential collision will not always be avoided, and collision may really happen between agents. The possible reason could be: 1) one or both of the two pedestrians involved in the collision do not detect the collision; 2) the velocity adjustment is not enough to avoid it; and 3) one of the two pedestrians is aggressive and expects the other one to respond to the coming collision such that the velocity adjustment from the other pedestrian is not enough to avoid the collision. Once collision does occurs, collision response is needed for both of the agents in the subsequent simulation step(s). The process of response is dependent on the type of collision: a slight collision may allow the agents to continue their movement only with a subtle change of original velocity; a heavy collision, in the contrary, will make them stop. After the agent recovers from the collision, it continues to move along its desired trajectory and avoid the potential collisions.

The rest of the paper is organized as follows. A brief overview of prior work is given in the *Related Work* section. The overview of the motion plan ning model is described in *Model Overview* section. The methods for collision avoidance is introduced in the *Collision Avoidance* section and *Analysis of Safety Factor* section. Section *Collision Response* gives the rules for agent response to collision. The experiment results of applying the proposed mo-

tion planning mechanism into the crowd simulation is shown in *Experiments Results* section. Finally, the paper is concluded in the section of *Conclusion*.

Related Work

The motion planning is originally from the research field of robotics, where it used for controlling robots in the environments, and avoiding the potential collision amongst other robots or moving obstacle, in the mean time. However, most of the existing work (Gayle et al., 2002; Jaillet and Simeon, 2004) do not take into account the fact that the other robot may be affected by the presence and the possible motion of this robot. When the similar method is applied to crowd simulation for collision avoidance (Lamarche and Donikian, 2004; Koh and Zhou, 2007), the trajectory from the simulation result may be deviated from that of the real pedestrian. A typical example is that oscillation happens when agent avoids the collision with the other one (van den Berg et al., 2008) . Another type of motion planning mechanism for crowd simulation is based on social norms. These methods aim to unveil the principles inside the crowd movement and mimic the behavior of real human in dynamic environments, in which some social norms and psychological factors are combined. It can make the result more realistic to those observed in real life. Generally, there are two kinds of implementation to this: one, like the methods in (Stephen, 2004; Treuille et al., 2006), concentrates the principles and features of crowd movement in a macroscopic level, losing the personality of each pedestrian; the other one, like the method proposed by (Shao 113 and Terzopoulos, 2005), focuses on modeling the individual behaviors at the

microscopic level, lacking the consideration of the movement of the whole crowd.

Different from those mentioned methods for collision avoidance by selecting collision-free velocity, another type of method aims to avoid potential collisions by selecting proper path along agent's moving trajectory. Model proposed by Ahn et al. (2011) presents a trajectory variant shift method by re-using and shifting real trajectories captured from video data to avoid collisions. Ahn et al. (2012) gives the collision avoidance model by reducing short term collision avoidance through long term anticipation of pedestrian trajectories. One of the main objectives of these methods is to reduce the occurrence of collisions. However, it lacks the ability to handle the exceptional and abrupt scenarios which are not covered by the trajectories captured from real videos. On the contrary, this paper proposes a motion planning model to simulate how agent selects velocity to avoid near potential collisions, with the result of generating a velocity for the action of movement.

Apart from those, there are few studies on collision response. (Pelechano et al., 2007) uses social force to simulate such response. However, it lacks the consideration of the collision type and parameters of agent. There has been no method proposed for collision response after it really happens. In this paper, a method is proposed to navigate agents in dynamic environments among the other agents or obstacles, which considers the effect made by the other agent in the presence of collision. Moreover, it does not guarantee that the generated motions are collision-free, and a response will be performed once the collision happens.

Model Overview

The main issue of motion planning of crowd simulation is to select velocity for agent to move towards the given destination. As mentioned above, how to select the given destination is beyond the scope of this paper. During its movement around the simulation environment, it may encounter potential collision with other agents or obstacle, *collision avoidance* model is then needed for simulating agent's movement. Collision may really occur during crowd movement, especially in crowded area. Hence, the motion planning model also needs to handle how agent makes a response to a collision, which is modelled by *collision response* module.

The overall modelling process of motion planning can be described as follows. The simulation time is divided into desecrate time slots with equal interval. At the beginning of each simulation step, the destination for the simulation time step should be given from some higher level model (for example, a behaviour model). The given destination also determines the preferred moving direction. Some environment information, like agents and obstacles around should also be sensed based on the current location of the agent. With this input information, the model helps agent to detect any potential collisions. If there is no potential collision, the agent will choose preferred velocity (preferred direction as well as the given preferred speed) to move in the current simulation time step. Otherwise, agent should deviate from its preferred velocity and choose a proper velocity in order to avoid the on-coming collision. With this velocity, if the agent really collides with another agent or obstacle, the motion planning model also needs to treat how agent recover

from the collision, this refers to the collision response part. The whole process of the motion planning model is illustrated in Figure 1. Detailed process of collision avoidance and response is introduced in the following sections. Notations used in the following discussion are listed in Table 1.

Table 1: Notations of Parameters Describing Movement of Agent A

Notation	Meaning
$\mathbf{v}_A$	the current velocity
$\mathbf{v}'_{A,i}$	a candidate velocity
$\mathbf{v}_A^{prefer}$	the preferred velocity
v_A^{max}	the maximum speed, such that $\forall \mathbf{v}'_{A,i}, \parallel \mathbf{v}'_{A,i} \parallel \leq v_A^{max}$
r_A	the radius of the agent
P_A	current location of the agent
$panic_A$	panic level of the agent
$density_A$	density around the agent
v_A^{cap}	Capable speed of agent in the environment
v_A^B	relative velocity of agent A with respective to B
RVO_B^A	Velocity Set of Agent A causing collision with Agent B RVO

Collision Avoidance

The process of collision avoidance could be divided into four stages:

1. detecting whether there exists an on-coming potential collision with other agent or obstacles; if there is no such collision, just choose the preferred velocity and complete the collision avoidance process; otherwise, go to step 2);

2. establishing the validate set for all candidate velocities based on kinematic and dynamic principles;

3. sampling velocities from the validate velocity;

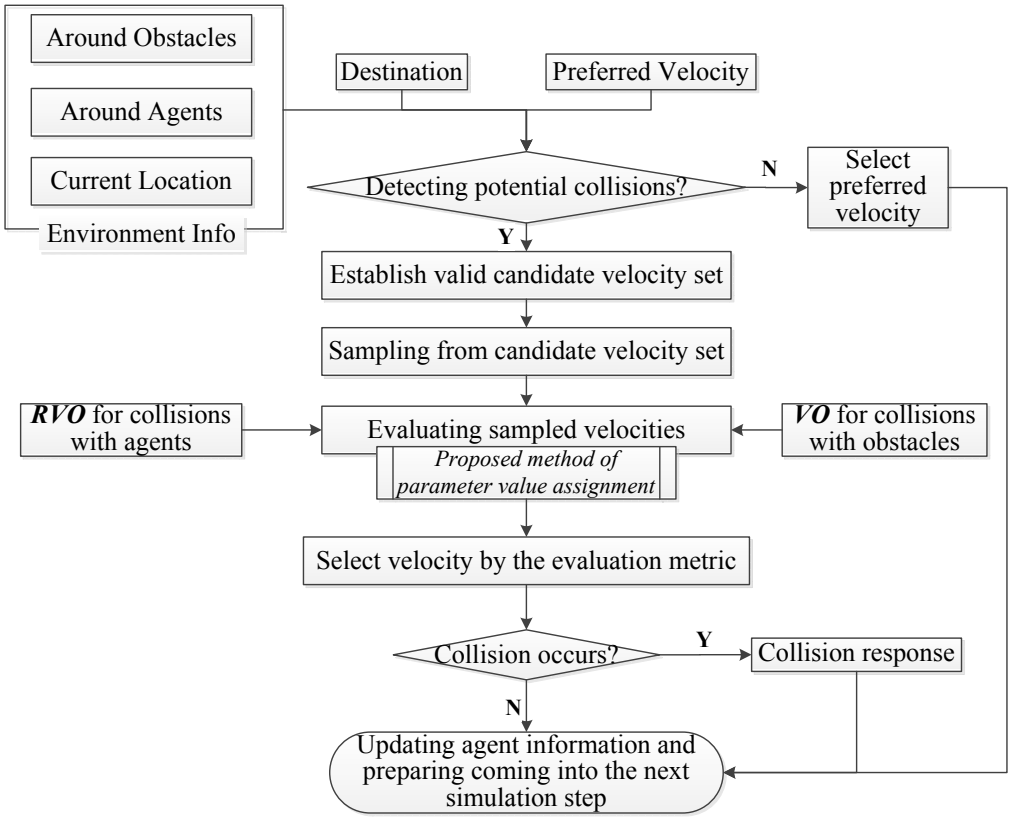


Figure 1: Overview of the Motion Planning Model

4. evaluating each sampled velocity by the established evaluation method and choosing the best one as the moving velocity for the current simulation time step.

In the following discussion, the shape of agent is assumed as a circle with a specified centre point and radius. A collision of agent with another one is defined as the two circles representing agents intersects. Similarly, an agent collide with obstacle can be also described as the circle for the agent intersects with the obstacle.

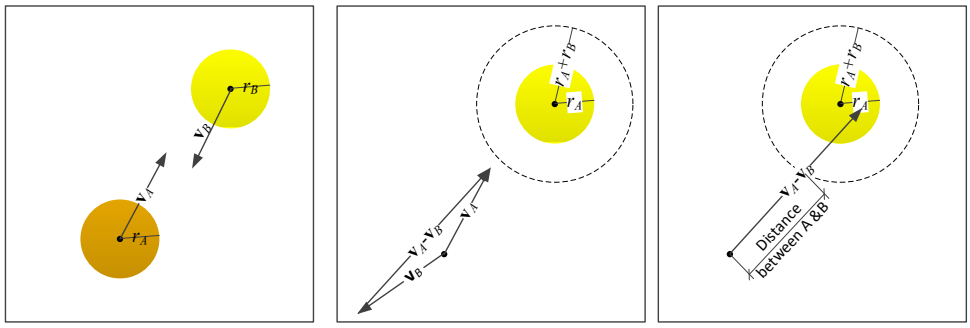
On-coming Collision Detection

The problem of collision detection can be converted into whether moving circles intersect or not. Taking the detection of agent colliding with another one as example, the collision detection can be executed as the following steps (illustrated in Figure 2). The initial status (location,shape and location) of the two agents are shown in Figure 2a. Based on these information, the model detect a potential collision as follows.

1. The circle of agent A can be overlaid on agent B, i.e., agent A can be reduced to a point and the circle radius of circle come into $r_A + r_B$ (as illustrated in Figure 2b) ; the relative velocity of A respective to B can also be calculated by $\mathbf{v}_A - \mathbf{v}_B$. In this case, agent B can be considered as a static object with the radius $r_A + r_B$, and agent$A$ is a moving point.

2. In order to detect the potential collision, it is then needed to figure out the intersection point of the new circle of agent B and the ray for the current location of agent A with the direction of the relative velocity $\mathbf{v}_A - \mathbf{v}_B$. If there exists a intersection point, the distance between A's current location and the intersection point can be calculated. The *time-to-collision*, which means how soon agent A will collide with agent B, can be also figured out (if there exists two intersection points, select the smaller one as the time-to-collision). Otherwise, if there is no such a point, the distance can be considered as infinite, and the time-to-collision can also be considered as infinite, which means there is no on-coming collision with agent A and B. This modelling process is

illustrated in Figure 2c.



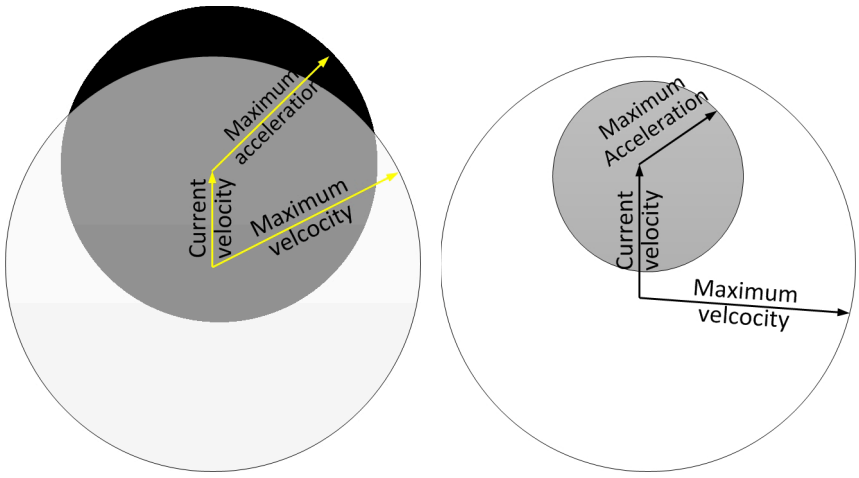
(a) Original Status (b) Relative Velocity (c) Distance between A&B

Figure 2: Process of Calculating Time to collision

Validate Velocity Set and Velocity Sampling

If an agent detects an on-coming collision and desires to avoid the collision, it will deviate from its preferred velocity and choose another one to achieve it. A validate velocity set should be first established to determine all possible candidate velocities. The validate velocity set is built based on the agent's kinematic and dynamic principles. The validate velocity set is based on agent's maximum speed and maximum acceleration. In Figure 3 the white circle represents all the velocities with the speed less than the maximum speed. The black circle is the velocity set for is all the velocities to whichcan be accelerated or decelerated from the current selected velocity. Theintersection part of the two set (presented in gray colour) is then the validate velocity set, inside which each velocity can been considered as candidate velocity. However, it is not possible to evaluate every velocity belongs to the validate velocity set, for that the number of velocity is infinite. Hence, typical

velocities needs to be sampled from the set, which refers to the process of velocity sampling.



(a) Validate velocity set type I (b) Validate velocity set type II

Figure 3: Validate Velocity Set Determination

The main idea of velocity sampling from the validate velocity set is to choose typical velocity to represent all the velocities belonged to the set. Regarding this, the sampling velocity should be uniformly distributed inside the intersection of the two circles (i.e., the validate velocity set). The shape of the validate velocity set falls into two types: 1) the shape illustrated in Figure 3a, in which some velocity satisfied the acceleration requirement is outside the maximum speed set; and 2) the shape of validate set is whole velocity set satisfying the requirements of acceleration (shown in Figure 3b).

For both type of interaction part, a uniformly distributed function is built by Equation 1).

$$P(x, y) = uniformPoint(r_{speed}, r_{acc}, centre_{speed}, centre_{acc}) \quad (1)$$

where $P(x, y)$ is a point inside the intersection part; x and y represents the two velocity components along x and y coordinate, respectively; r_{speed} and r_{acc} represents the maximum speed and value of acceleration holding by the agent; $centre_{speed}$ is the centre point of the velocity set for those satisfying the maximum speed requirement; accordingly, $center_{acc}$ is the centre point of the velocity set for those satisfying the maximum value of acceleration requirement. With a given number of velocity sampling and executing the equation with the same number of time, the results (i.e., a serial of points), uniformly distributed inside the intersection part.

Velocity Evaluation

A velocity evaluation method is proposed in (van den Berg et al., 2008) to evaluate whether the candidate velocity is proper for current situation:

$$penalty_{A,B}(\mathbf{v}'_i) = w_i \frac{1}{tc_i(\mathbf{v}'_i)} + \parallel \mathbf{v}_A^{pref} - \mathbf{v}'_i \parallel, \tag{2}$$

where $\mathbf{v}'_i$ is a candidate velocity of agent A, $tc_i(\mathbf{v}'_i)$ is the time to collision with agent B by holding the current candidate velocity $\mathbf{v}'_i$, $\mathbf{v}_A^{pref}$ is the preferred velocity, and w_i is the safety factor.

This formula indicates that a trade-off is made for the velocity selection between maintaining current velocity to ignore the collision and deviating from the preferred velocity to avoid the collision: the first item of the formula ($w_i \frac{1}{tc_i(\mathbf{v}'_i)}$) reflects how long the collision will occurs with the candidate velocity, $\mathbf{v}'_i$; and the second part ($\parallel \mathbf{v}_A^{pref} - \mathbf{v}'_i \parallel$) is about how much the candidate velocity deviates from the preferred velocity. The velocity selection

for an agent, A, is subjected to the following process: 1) given a candidate velocity, the penalty values for all its neighbors are calculated by Formula 2 and the minimum value is selected; 2) as a result, every candidate velocity holds such a value; 3) the velocity, with the minimum penalty among all the candidate velocities, is selected. This process can be formulated as

$$\mathbf{v} = \min_{\mathbf{v}'_i \in CV} \min_{j \in N} penalty_{A,B}(\mathbf{v}'_i), \tag{3}$$

where CV is the set of all the candidate velocities of the agent i, and N is the set of all the detected neighbors of the agent.

In order to apply Formula 2 to evaluate velocity, the time-to-collision is needed to be figured out. In *On-coming Collision Detection* section, a method is mentioned to calculate time-to-collision for detecting potential collisions. In this method, the core problem is to determine how to calculate the relative velocity between two agents or agent and obstacle. Actually, the existing methods of VO and RVO are proposed for calculating the relative velocity. VO and RVO are two velocity sets in which any velocity will generate collision. These two velocity sets hold different assumptions: VO assumes that one think the other will not response to his collision avoidance; on the contrary, RVO assumes that both of the two agents will respond to the other's motion of collision avoidance and make the same contribution to the collision avoidance. Details about VO and RVO can be found in (van den Berg et al., 2008) and (Fiorini and Shiller, 1998) respectively. In the following discussion of collision avoidance, VO/RVO can be referred to either the velocity set or the conception with the corresponding assumption, which can

be distinguished in the context.

When RVO is applied, it implies that the agents contribute the same effort to avoid each other[1]. From this, it can be determined that B should choose $\mathbf{v}'_B = \mathbf{v}_B - \Delta\mathbf{v}_A$ ($\mathbf{v}_B$ is the current velocity of agent B) as its new velocity, if A chooses $\mathbf{v}'_A$ as its new velocity ($\Delta\mathbf{v}_A = \mathbf{v}'_A - \mathbf{v}_A$ and $\mathbf{v}_A = \mathbf{v}'_A + \Delta\mathbf{v}_A$). The relative velocity of A with respect to B then can be calculated as:

$$\begin{aligned} v_A^B &= \mathbf{v}'_A - \mathbf{v}'_B \\ &= \mathbf{v}'_A - (\mathbf{v}_B - \Delta\mathbf{v}_A) \\ &= \mathbf{v}'_A - (\mathbf{v}_B - (\mathbf{v}'_A - \mathbf{v}_A)) \\ &= 2\mathbf{v}'_A - \mathbf{v}_A - \mathbf{v}_B. \end{aligned} \tag{4}$$

Then, the time-to-collision can be figured out by applying Equation 4 to calculate the relative velocity between two agents.

Similarly, we can also apply the formula to select velocity based on the assumption of VO. The only difference is the calculation of the relative velocity of A with respect to B, which can be get by the following equation.

$$v_A^B = \mathbf{v}'_A - \mathbf{v}_B. \tag{5}$$

Analysis of Safety Factor

In order to apply Equation 2 to evaluate candidate velocities and obtain the velocity with the minimum value of *penalty*, the value as well as its meaning

[1]This is what Lemma 7 in the literature of (van den Berg et al., 2008) concerns.

of parameter w_i should be determined, which is not mentioned clearly in the paper van den Berg et al. (2008). The discussion in this section can also be applied when using VO to calculate the relative velocity.

Generally, a pedestrian will follow her/his preferred velocity until it encounters a collision; once a potential collision is detected, the pedestrian may adjust the velocity with a minimum deviation. Given all the parameters of two agents, A and B, the velocity set of A, RVO_B^A can then be generated. If $\mathbf{v}_A^{prefer} \notin RVO_B^A$, it implies that there will be no collision and $\mathbf{v}^{prefer}$ will be selected. When $\mathbf{v}_A^{prefer} \in RVO_B^A$, a collision will be caused if selecting the preferred velocity. In this case, another candidate velocity will replace the preferred velocity. If a candidate velocity, $\mathbf{v}'_A$, satisfies the condition $\mathbf{v}'_A \notin RVO_B^A$, the value of penalty, calculated by Formula 2, only depends on the velocity deviation due to the infinity value of time-to-collision. Otherwise ($\forall \mathbf{v}'_A \in RVO_B^A$), the value penalty will also be contributed by the non-infinite value of time-to-collision on the basis of the velocity deviation. As an example, Figure 4 illustrates the penalties for all the candidate velocities of agent, A, when it come towards the other agent, B, with the following conditions:

$$\begin{aligned} \mathbf{v}_A &= (-1.4m/s, 0), \mathbf{v}_B = (+1.4m/s, 0), \\ \mathbf{v}_A^{prefer} &= (-1.4m/s, 0), \mathbf{v}_B^{prefer} = (-1.4m/s, 0), \\ P_A &= (15m, 0), P_B = (0, 0), \\ r_A &= r_B = 2m, \\ v_A^{max} &= v_B^{max} = 1.4m/s, \\ w_i &= 5. \end{aligned} \tag{6}$$

If all the parameters are given, how much the penalty raised due to collision is determined by the *safety factor*, w_i. If $w_i = 0$, the penalty only depends on the velocity deviation from Formula 2 and the agent will always choose the preferred velocity, which causes the agent not to consider the coming collision. With the increase of w_i, the penalty will be raised accordingly.

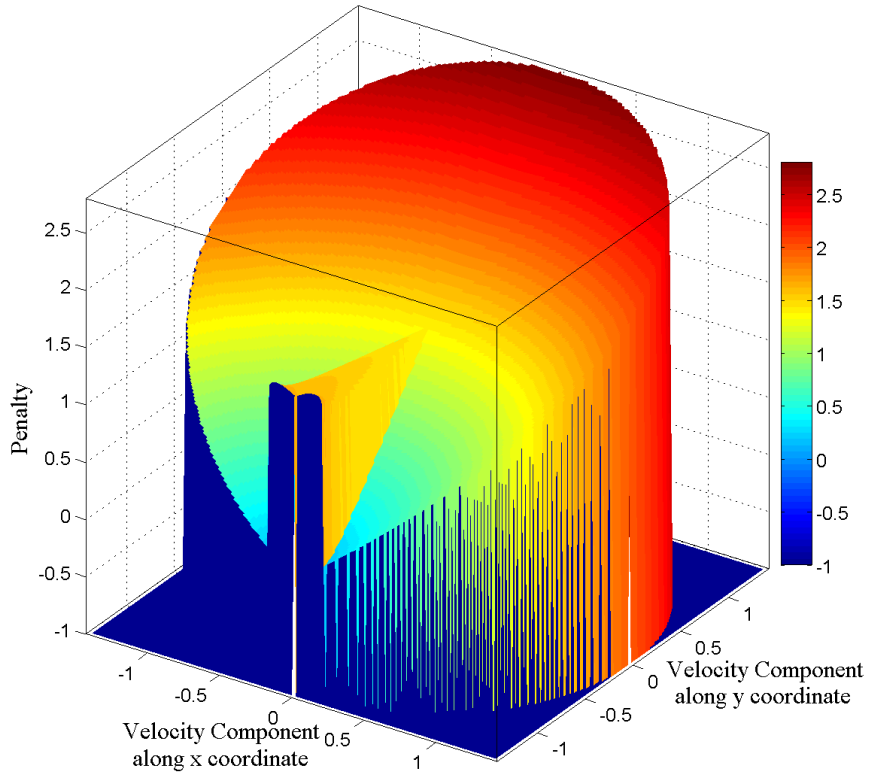


Figure 4: An Example of Penalty Value Distribution

Actually, the velocity selection is the process to minimize the penalty, which can be achieved at either the edge of RVO (avoiding the collision) or a point inside RVO (ignoring the collision). If the preferred velocity does not locate at the center of RVO, there exists a deviation, labeled as dv, between the preferred velocity and the one inside the RVO and holding the minimum penalty. Since the distance between them is much larger than the radii of

the agents, we can assume $dv \simeq 0$. If the minimum penalty is achieved by a velocity outside the RVO, it should locate at the edge of the RVO with a minimum velocity deviation, which is labeled as v_s. If the collision-free motion is needed, the condition:

$$\parallel v_A^{prefer} - v_s \parallel < \frac{w_i}{t_1} + dv \simeq \frac{w_i}{t_2} \tag{7}$$

must be satisfied, where t_1 is the time to collision with the velocity achieving the minimum penalty inside the RVO, and t_2 is the time to collision with the preferred velocity. Here the first item in the inequality is for the velocity on the edge of RVO; the second item is for the velocity at a specified point in RVO; and the last item is for the velocity along the preferred velocity. Again, because the distance between them is much larger than the radii of the agents, we can assume $t_1 \simeq t_2$, due to the low velocity of pedestrian. Actually the angles among the three velocities are very small. Then we have,

$$w_i > t_2 * \parallel v_A^{prefer} - v_s \parallel, \tag{8}$$

where $t_2 * \parallel v_A^{prefer} - v_s \parallel$ denotes the threshold: if w_i is larger than it, the agent prefers to avoid the collision; otherwise, it will move along its preferred direction and ignore the collision. This can be explained by the fact that the agent ignores the coming collision when the distance between the two agents is far; and the agent avoids the collision once they get closer.

However, with a normal sensor range, such distance is much larger than the given sensor range. Figure 5 illustrates the relationship between the

distance agents beginning to avoid each other and the safety factor. The parameters of the two agents are set the same to those in Figure 4, with the only exception of the $200m$ of the distance between them. The distance can be very large,when $w_i >= 1.7$ $(24.67m)$. In case of this, the selection of safety factor will be nonsense when the normal sensor range is much smaller than the corresponding distance. The reason for this is that the distance between the two agents is much larger than the radius of the agent.

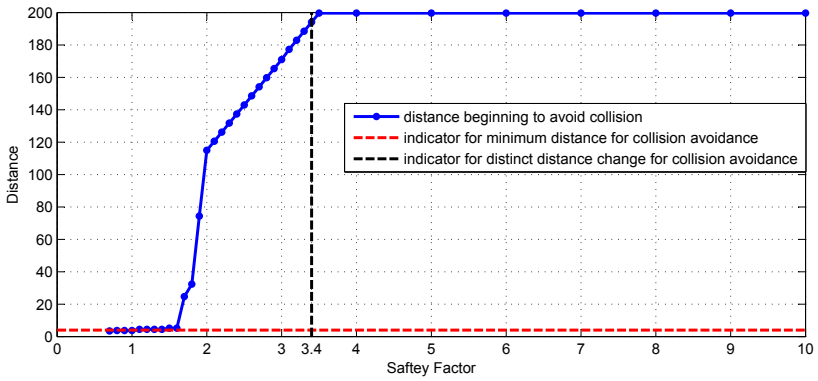


Figure 5: Relationship between Safety Factor and Minimum Distance for Collision Avoidance

The meaning of the safety factor when an agent meets several agents differs from that when it encounters only one. Here, the agent, namely i, still needs to make a trade-off, but among all the other agents around. This depends on the value of the penalty with a safety factor. Consider two agents, j and k around, the agent i needs to select a velocity which make a minimum penalty from Formula 2. There exit two velocities, v_j and v_k, which achieve the minimum penalty for agent j and k, respectively. Based on the previous analysis, we can claim that both of the two velocities will make a collision-free motion for the corresponding agent. Now, it needs

to determine which one is smaller. For velocity v_1, we have the penalty $p_j = w_i/t_j + dv_j$, where t_j is the time to collision with agent j, and dv_j is the velocity deviation between the preferred velocity and v_j . Similar, for velocityv_2, the corresponding penalty is $pk = w_i/t_k + dv_k$. In case of $dv_j < dv_k$ and $t_j > t_k$, p_j should be smaller than p_k, which makes v_j as the selected velocity. However, if the two velocities are subject to the conditions of $dv_j > dv_k$ and $t_j > t_k$, the safety factor, which scales the first item by w_i times, makes an impact on the selection of minimum penalty. Let's consider the following two extreme conditions for the two velocities: $t_j \to \infty$ and $dv_k = 0$, which indicates that v_j will make a collision-

free motion with agent j, v_k is along the direction of the preferred velocity of agent i, and $1/t_k \to 0$. Remember v_j will make a collision-free motion with agent

j, then the maximum of the velocity

derivation is $\|\mathbf{v}_i^{prefer}\|$. It is achieved when $\|\mathbf{v}_j\| = 0$. In such conditions, we need to compare which of the two items is smaller: w_i/t_k or $dv_j = \|\mathbf{v}_i^{prefer}\|$. The threshold of safety factor can be formulated as $tk * \|\mathbf{v}_i^{prefer}\|$. The other extreme assumption is that both i and k move towards each other with the speed $\|\mathbf{v}_i^{prefer}\|$, and $\|\mathbf{v}_k^{prefer}\|$ ($\|\mathbf{v}_i^{prefer}\| = \|\mathbf{v}_k^{prefer}\|$), with the distance D_{ik}, t_k then can be calculated as $D_{ik}/2 * \|\mathbf{v}_k^{prefer}\|$. Then the threshold of the safety factor is $D_{ik}/2$. Then the meaning of safety factor for navigation agent around several others nearby is: if the value of safety factor is set larger an half of the minimum distance to all its neighbors, the agent prefers to avoid the instant collision; otherwise, it just ignore the collision and moves along the direction of its preferred velocity. We name this threshold for safety factor is the *strict* one. Associated with the previous analysis, the maximum distance between two agents should be the sensor range. Then the *loose*

threshold for safety factor should be half of the sensor range.

Selection of Safety Factor

From the analysis in last subsection, the safety factor, w_i, represents how far an agent begins to avoid the collision. The following issue is about how to select the value of w_i in different scenarios.

The movement of a pedestrian in a crowd is constrained by the features of the crowd. The fact is that the pedestrian's speed is related to the pedestrian density around her/him: if the density is high, the walking speed should be slow; otherwise, the pedestrian can walk freely with a high speed. (Hughes, 2002) proposes a relationship between pedestrian's walking speed and density, formulated as follows:

$$f(\rho) = \begin{cases} A, & \rho \leq \rho_{trans} \\ A(\rho_{trans}/\rho)^{1/2}, & \rho_{trans} < \rho \leq \rho_{crit} \\ A(\frac{\rho_{trans}\rho_{crit}}{\rho_{max}-\rho_{crit}})^{1/2}\frac{(\rho_{max}-\rho)^{1/2}}{\rho}, & \rho_{crit} < \rho \leq \rho_{max} \end{cases} \tag{9}$$

where ρ_{trans}, ρ_{crit}, and ρ_{max} have typical values of 0.8, 2.8 and $5.0m^{-2}$, and the typical value for A is $1.4m/s$. Adopting this formula, a speed of an agent, A, can be calculated with the given density, which is called capable speed, v_A^{cap}.

On the other hand, the pedestrian may hold her/his preferred speed, $\| \mathbf{v}_A^{prefer} \|$, which indicates how fast she/he desires to walk and related to her/his panic level. Here we simply divided the panic level into three categories: normal, hurry, and emergency. The corresponding preferred veloci-

ties, are set with the values of 1.4, 2.0, and $3.5m/s$, respectively. The maximum acceleration is set to 1, 2, and $2.5m/s^2$ for the three types of agents. This setting corresponds to the human step frequency reported in (Mazarakis and Avaritsiotis, 2005), which varies from 0.9Hz to 3.5Hz, with a mean of 2Hz. Then, we set the safety factor, w_i, is determined by the follows:

$$w_i = collision_d_{standard} \frac{v^{cap}}{\| \mathbf{v}_A^{prefer} \|}. \tag{10}$$

we adopt the *field of vision* proposed in (Feurtey, 2000), which is composed by a sector, centering at the position of the agent, with the angle of 160^o and radius of $10m$. Then *collision* $d_{standard}$ is set to $5m$, which is the half of sensor range. Associated with Formula 8, that the agent will either avoid or just ignore the coming collision can be determined.

The concept of RVO requires the assumption that both of the agents will make the same contribution for the collision avoidance. If both of the agents detect the same coming collision, RVO is applied in Formula 2. However, if one of them does not catch sight of the other one (e.g. one following the other one), RVO should not be applied. If this case, VO is applied in Formula 2 for the one detecting the collision. The awareness of the other agent can be determined by checking whether the other agent is in the vision field of the agent. In case an agent encounters a moving/static obstacle, VO is also applied in Formula 2 to generate the motions for collision avoidance.

Collision Response

If a collision does occur during the simulation, *collision response* is needed for the agent. We assume that each agent is represented by a circle. The collision between two agents can then be detected when the circles of the two agents overlap. In this paper, a rule-based collision response model is applied, which is proposed in our previous paper Xiong et al. (2010).

Collisions can be distinguished by the collision type and the speeds with 373 which the two agents collide. Different types of collision can cause an agent to respond differently. First of all, a collision can be categorized into two types: *side collision* and *head-on collision*. An example of the two types of collisions. An head-on collision for Agent A with Agent B occurs when: 1) the circles representing A and B overlap; and 2) the ray starting from the center of the circle representing A and along the direction of A's velocity, intersects the circle representing B. All the other collisions are considered as side collision. A side collision, which we assume to be less severe, may only slow down the involved agents slightly. The head-on collision, on the contrary, reflects a significant collision, in which an agent must stop for a while before it continues its movement. Secondly, the agent's current speed is categorized into three categories, i.e., less than 1.7m/s, between 1.7 and 2.8m/s, and larger than 2.8m/s. The rules of the collision response made by the agent are defined in Tables 2 and 3 (where v_A and v_B represents the current speed of the Agent A and B, respectively), involving the following actions:

i) the agent reduces its speed (here we assume the agent drops to 80% of

its current speed);

ii) the velocity direction will be turned away from the collision with an angle φ (here $\varphi = \pi/9$);

iii) the agent stops for the time of t_{stop} (it is assumed that the $t_{stop} = 0.5$s for side collision and $t_{stop} = 2$s for head-on collision); and

iv) the agent falls down, where the agent also needs time of t_{fall} before it continue to move (here t_{fall} is set as 3s).

$i + ii$ means the agent will execute the two responses simultaneously. Note, for actions *iii* and iv a random positional deviation is applied to the agents to avoid repeated collisions. While semi-random, this adjustment is performed according to the resulting collision force between the two agents. This adjustment only aims to separate an agent colliding with another one, but not to consider any on-coming collisions. As a result, the agent may continuously *re-collide* with other agents around, which can further increase the overall number of collisions.

Table 2: Collision Response of Agent A Colliding with Agent B (Side Collision)

Agent A \ Agent B	$v_B \leq 1.7$	$1.7 < v_B \leq 2.8$	$v_B > 2.8$
$v_A \leq 1.7$	$i + ii$	iii	iii
$1.7 < v_A \leq 2.8$	i	$i + ii$	iii
$v_A > 2.8$	i	i	$i + ii$

Table 3: Collision Response of Agent A Colliding with Agent B (Head-on Collision)

Agent A \ Agent B	$v_B \leq 1.7$	$1.7 < v_B \leq 2.8$	$v_B > 2.8$
$v_A \leq 1.7$	*iii*	*iii*	*iv*
$1.7 < v_A \leq 2.8$	*iii*	*iv*	*iv*
$v_A > 2.8$	*iii*	*iii*	*iv*

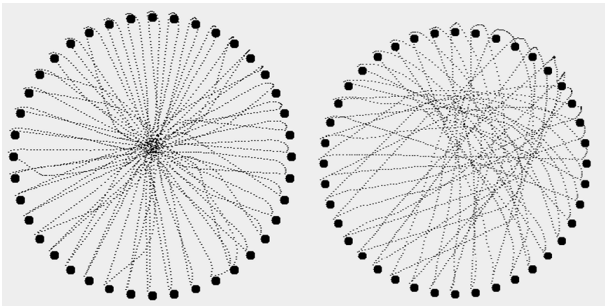
Case Study

In this section, the experiment results are showed to validate the proposed motion planning model. The experiments are divided into three parts: different values of safety factor for agent navigation, collision response for different collision type, and a concrete scenario for agent moving around the simulation environment.

Evaluation of Safety Factor

The first group of experiment is designed to examine the impact of the safety factor on the collision avoidance when multiple agents are around. In the experiment, all the agents in the experiment, uniformly locating at a circle, desire to achieve the position at the other side of the circle. The value of safety factor is fixed set to 0.1 and 15 respectively, and no collision response is considered. Figure 6 illustrates the simulation results of this scenario, where the solid circles represent the initial positions of agents; dark dots represent the trajectories of those moving agents. Each agent desires to move to the other side of the circle in the simulation environment. When the safety factor is set to 0.1, it is shown in Figure 6a that all the agents continues to move and

get very closed before they begin to avoid the coming collision; compared with this, the agents begin to avoid the others with 15 for the safety factor, when they are far away from each other, (shown in Figure 6b). This validates the fact that: the agent prefers to avoid each other when safety factor is larger; and they become sluggish and pretend not to detect the coming collision, once the safety factor is smaller. Finally, they avoid the other agents around until the distances among them drop to a threshold.



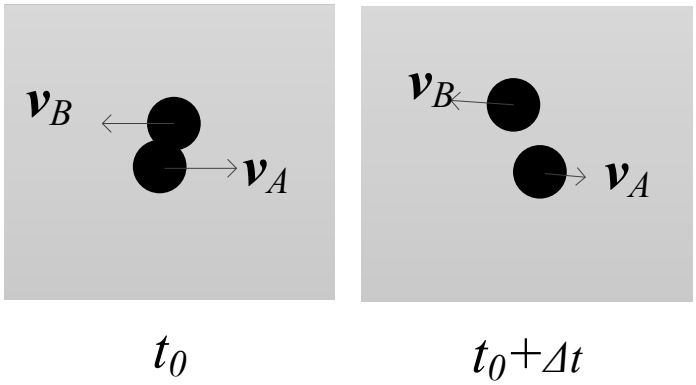
(a) $w_i = 0.1$ (b) $w_i = 15$

Figure 6: Agents Changing Their Positions in Circle with Different Value of Safety Factor

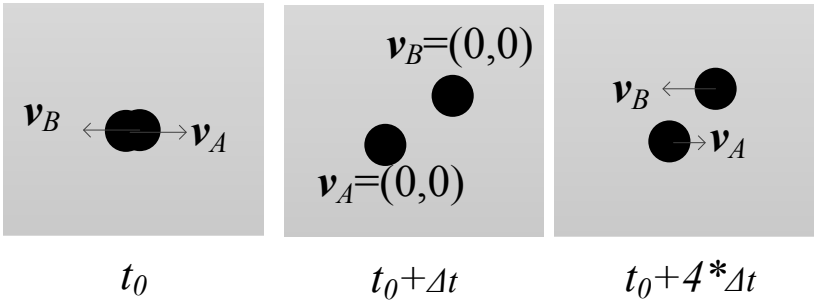
Collision Response

Different types of collision responses, i.e., side and head-on collision are demonstrated in Figure 7, where Δt represents the simulation time step and is set to $0.5s$ here. There are two agents involved in the collision response process. Both agents hold the speed of $1.4ms$.

For side collision (demonstrated in Figure 7a), the two agent drops its speed to 80% of its original speed and turn away from its original direction with an angle $\pi/9$. As mentioned in Table 2, it needs no time for collision



(a) side collision response



(b) head-on collision response

Figure 7: Response for Side and Head-on Collisions

recovery (speed of both agent less than $1.7ms$), two agents involved in side collision is able to continue their movement immediately after the occurrence of side collision, only by adjusting their speed and moving direction. Detailed position and velocity information of two agents before and after the collision is shown in Table 4.

Head-on collision, as mentioned Table 3, makes agents involved in the collision stopping/falling and waiting for collision response and recovery. After collision occurs, the two agents are separated by collision force. Since both agents hold speed less than $1.7ms$, they need $2s$ stop time for collision response, as demonstrated in Figure 7b. Detailed position and velocity

Table 4: Positions and Velocities of Agents in Process of Collision Recovery (Side Collision)

Notation \ Time	t_0	$t_0 + \Delta t$
$\mathbf{v}_A$	$(1.0, 0)$	$(0.94, -0.34)$
$\mathbf{v}_B$	$(-1.0, 0)$	$(-0.94, 0.34)$
P_A	$(10.0, 10.0)$	$(10.47, 9, 83)$
P_B	$(10.4, 10.4)$	$(9.93, 10.57)$

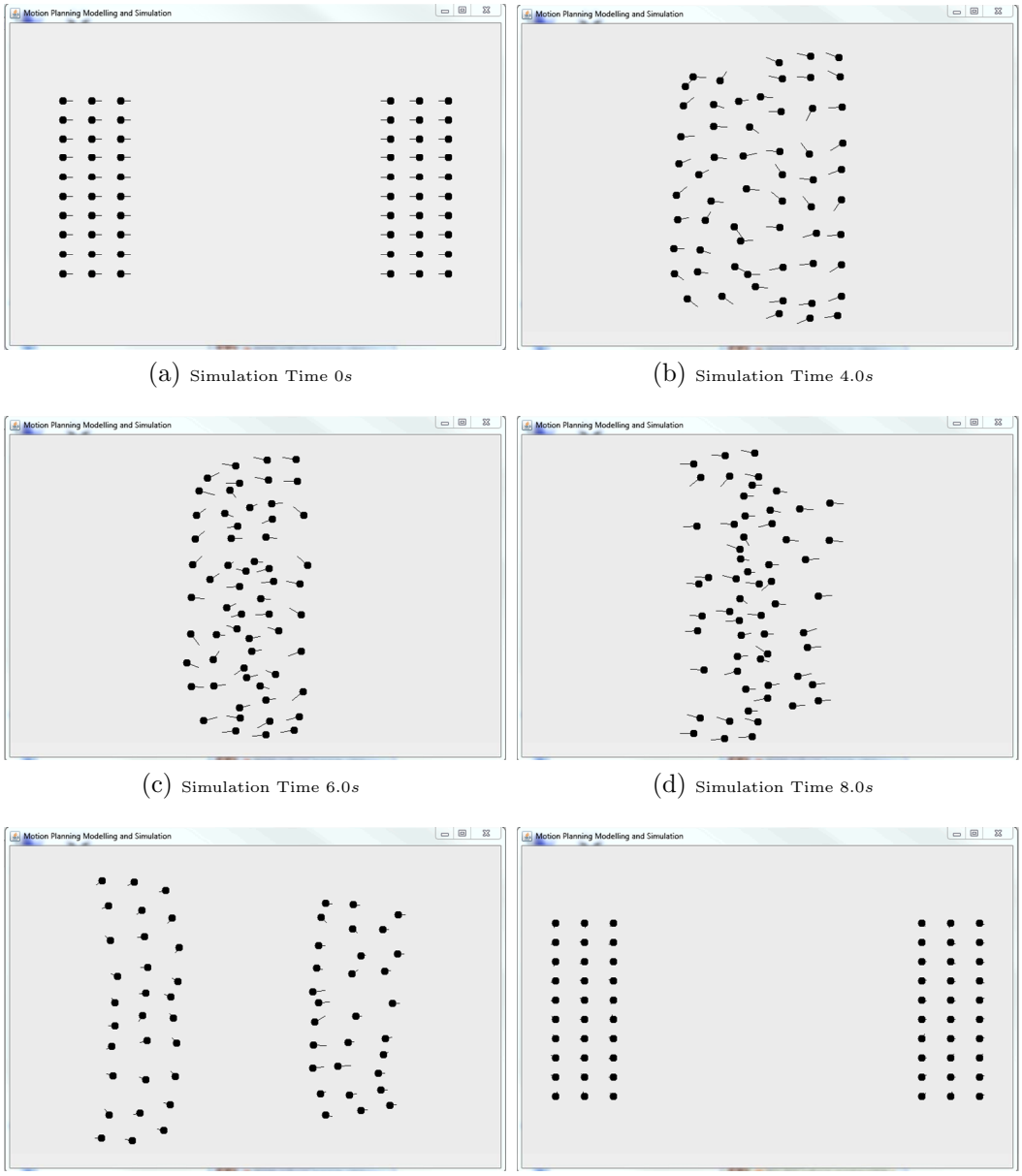
information of two agents before and after the collision is shown in Table 5.

Table 5: Positions and Velocities of Agents in Process of Collision Recovery (Entire Collision)

Notation \ Time	t_0	$t_0 + \Delta t$	$t_0 + 4\Delta t$
$\mathbf{v}_A$	$(1.0, 0)$	$(0, 0)$	$(1, 0, 0)$
$\mathbf{v}_B$	$(-1.0, 0)$	$(0, 0)$	$(-1.0, 0)$
P_A	$(10.0, 10.0)$	$(9.3, 9.3)$	$(9.8, 9.3)$
P_B	$(10.3, 10.0)$	$(11.0, 10.7)$	$(10.5, 10.7)$

Scenario with Different Panic Levels

The following experiments apply the proposed motion planning model into a concrete scenario. There are two groups of agents in this scenario, including three columns agents in each group. Initially, the two groups locate at left and right edge of the environment, with the distance of $16m$. Then the agents in both of the two groups begin to move towards each other to exchange their positions between the two group to exchange their positions between the two groups, which means the goal of each agent is $16m$ away from its original position. For example, an agent locating at the point with the coordinate $(5, 20)$ desires to achieve the point $(21, 20)$.



(a) Simulation Time 0s (b) Simulation Time 4.0s

(c) Simulation Time 6.0s (d) Simulation Time 8.0s

(e) Simulation Time 12.0s (f) Simulation Time 19.0s

Figure 8: Two Groups of Agents Exchanging Their Positions with Normal Panic Level

Two experiments are conducted for the scenario, differentiating in the composition of agent panic level in each groups. Agents in the first experiment are all in normal panic level, with the preferred speed of $1.4m/s$. Figure 8 shows the snapshot of the simulation process, where agents are dotted with black color and the associated short line indicates the agent's velocity, both in direction and speed. From the experiment result, it can be observed that the shape of agent columns does not change much. For an agent with normal panic level, agent executes the collision avoidance motion when they are relatively far away from each other. This makes the motion of collision avoidance deviates lightly from its preferred velocity. This causes agents in each group moves in order even when they meet and come across each other. On the contrary, agents composed with different panic level behaves much differently for the second experiments. In this experiment, each group agents are composed of three types of panic levels, coloured with black, green and red for normal, hurry and emergence panic level, respectively (as shown in Figure 9). Since agent emergence and hurry panic level can tolerate the on coming collision with a shorter distance that the normal panic level agent, agents can moves much closer until they conducts the motion of collision avoidance. This makes collision occurs with a higher probability, which makes the agent to conduct the motion of collision response. Agent during the process of collision response is represented as circle in Figure 9. Additionally, it can also be observed that the agents with emergence panic level holds the highest speed which makes the distance between other agents becoming larger.

(a) Simulation Time 0s (b) Simulation Time 2.0s

(c) Simulation Time 5.0s (d) Simulation Time 7.0s

(e) Simulation Time 10.0s (f) Simulation Time 22.0s

Figure 9: Two Groups of Agents Exchanging Their Positions with Three Panic Levels

Performance Evaluation

The performance evaluation for the proposed motion planning model is based on the scenario used in the previous Section. Two experiments are compared for performance evaluation, for agents with normal and three types of panic levels. The movement for the scenario can be divided into three stages: 1) agents start to move at the beginning of the simulation; 2) two groups of agents get closer and begin to conduct the motion of collision avoidance; and 3) after two groups of agents come across each other, they continue their movement to achieve the desired destination.

The simulation result is shown in Figure 10, and the three stages of agent movement is different with the simulation cost. At the first stage, it can be observed that the simulation cost is relative low. When it comes into the second stage, it goes higher for both experiments. This is due to the increase of the velocity sampling number. At the initial stage, agents can apply their preferred speed since the time to the potential collision is long. Noticed that the valid velocity set is intersection part of the maximum velocity set and maximum acceleration set. As the on-taking speed decreases, the intersection part will goes larger. Due to the sampling resolution keeping the same during the simulation process, larger intersection part means more velocity will be sampled inside the valid velocity set. This makes the simulation cost growsup in the second stage. In stage 3, the simulation cost drops again because agents is able to choose the preferred velocity as the moving velocity. At the end of stage 3, the simulation cost decreases fast due to some agents arriving at the destination.

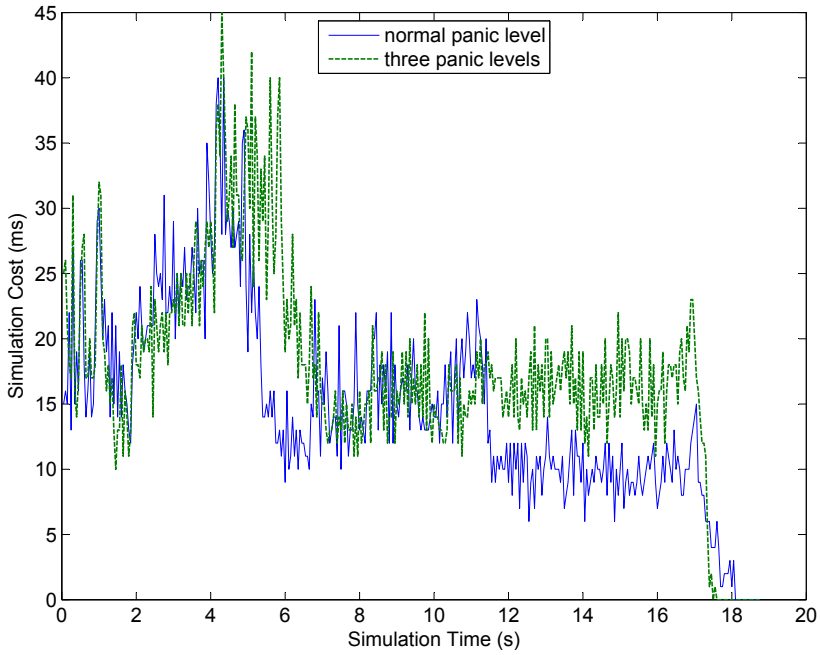


Figure 10: Performance of Motion Planning Model Compared with Two Scenarios

It can also be observed that the experiment with different panic levels holds a higher simulation cost than the one with normal agent. This is because experiment with different panic levels may cause more collisions. Agents involved in collisions needs to conduct the motion of collision response, which takes extra simulation cost for this.

Acknowledgement

The authors thank editors and anonymous referees for helpful comments of this paper. This work was supported in part by National Natural Science Foundation of China (grant No.61103145), and the Fundamental Research Funds for the Central Universities (China University of Geosciences (Wuhan),

No.CUG120409).

Conclusion

In this paper, an adaptive motion planning mechanism is proposed for crowd simulation, including motion planning and collision response. Motion plan ning is divided into two separately parts, collision avoidance and collision response. When an agent detects an on-coming collision and decides to avoid it, it selects a velocity deviating from its preferred velocity for collision avoidance. The model allows agent to collide with other agents or obstacles in some special scenarios, which causes the execution of the rule-based collision response method. From the experiment results, it shows that the proposed motion mechanism can generate different motions in dynamic environments. The agent with different panic levels may behave differently. In the future work, such differences should be considered for the collision avoidance, which makes the agent with lower panic level pays more contribution in the collision avoidance than the one with higher panic level does.

References

Ahn, J., Gobron, S., Silvestre, Q., Ben Shitrit, H., Raca, M., Pettré, J., Thalmann, D., Fua, P., and Boulic, R. (2011). Long term real trajectory reuse through region goal satisfaction. In *Proceedings of the 4th international conference on Motion in Games*, MIG'11, pages 412–423, Berlin, Heidelberg. Springer-Verlag.

Ahn, J., Wang, N., Thalmann, D., and Boulic, R. (2012). Within-crowd immersive evaluation of collision avoidance behaviors. In *Proceedings of the 11th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry*, VRCAI '12, pages 231–238, New York, NY, USA. ACM.

Feurtey, F. (2000). Simulating the collision avoidance behavior of pedestrians. Master's thesis, the Univ. of Tokyo.

Fiorini, P. and Shiller, Z. (1998). Motion planning in dynamic environments using velocity obstacles. *Int. Journal of Robotics Research*, 17(7):23–33.

Fox, D., Burgard, W., and Thrun, S. (1997). The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, 4(1):23–33.

Gayle, R., Sud, A., Lin, M., and Manocha, D. (2002). Reactive deforming roadmaps: Motion planning of multiple robots in dynamic environments. *Int. Journal of Robotics Research*, 21(3):233–255.

Hughes, R. (2002). A continuum theory for the flow of pedestrians. *Trasnsportation research PartB*, 36(6):507–535.

Jaillet, L. and Simeon, T. (2004). A prm-based motion planner for dynamically changing environments. In *in Proceedings of IEEE/RSJ Int. Conference on Intellegent Robots and Systems*, pages 1606–1611.

Koh, W. L. and Zhou, S. (2007). An extensible collision avoidance model for realistic self-driven autonomous agents. In *Proceedings of the 11th*

IEEE International Symposium on Distributed Simulation and Real-Time Applications, pages 7–14. IEEE Computer Society.

Lamarche, F. and Donikian, S. (2004). Crowd of virtual humans: a new approach for real time navigation in complex and structured environements. *Computer Graphics Forum*, 23(3):509–518.

Li, Y. and Gupta, K. (2007). Motion planning of multiple agents in virtual environments on parallel architectures. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 1009–1014.

Mazarakis, G. P. and Avaritsiotis, J. N. (2005). A prototype sensor node for footstep detection. In *Wireless Sensor Networks, 2005. Proceedings of the Second European Workshop on*, pages 415–418.

Nguyen, Q. H., McKenzie, F. D., and Petty, M. D. (2005). Crowd behavior cognitive model architecture design. In *Proceedings of the 2005 Behavior Representation in Modeling and Simulation (BRIMS) Conference*, pages 55–64, Universal City, CA.

Pelechano, N., Allbeck, J., and Badler, N. (2007). Controlling individual agents in high-density crowd simulation. In *ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA'07)*, pages 99–108.

Pelechano, N., O'Brien, K., Silverman, B., and Badler, N. (2005). Crowd simulation incorporating agent psychological models, roles and communication. In *Proceedings of the First International Workshop on Crowd Simulation*, EPFL, Lausanne-Switzerland.

Shao, W. and Terzopoulos, D. (2005). Autonomous pedestrians. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 19–28.

Stephen, C. (2004). Flow tiles. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer Animation*, pages 233–242, San Diego, California, USA.

Sud, A., Andersen, E., Curtis, S. andLin, M., and Manocha, D. (2007). Real-time path planning for virtual agents in dynamic environments. In *Proceedings of IEEE Virtual Reality*, pages 91–98.

Thalmann, D. and Musse, S. (2007). *Crowd Simulation*. Springer.

Treuille, A., Cooper, S., and Popović, Z. (2006). Continuum crowds. In *Proceedings of ACM SIGGRAPH*, pages 1160–1168.

van den Berg, J., Lin, M., and Manocha, D. (2008). Reciprocal velocity obstacles for real-time multi-agent navigation. In *Proceedings of IEEE International conference on robotics and Automation*, pages 1928–1935.

Xiong, M., Lees, M., Cai, W., Zhou, S., and Low, M. Y.-H. (2010). Analysis of an efficient rule-based motion planning system for simulating human crowds. *The Visual Computer*, 26(5):367–383.