Simulating Human Collision Avoidance Using a Velocity-Based Approach

Ioannis Karamouzas and Mark Overmars

Center for Advanced Gaming and Simulation, Utrecht University, The Netherlands

Abstract
We present a velocity-based model for realistic collision avoidance among virtual characters. Our approach is elaborated from existing experimental data and is based on the simple hypothesis that an individual tries to resolve collisions long in advance by slightly adapting its motion. We have evaluated our model by testing it against a wide range of challenging scenarios. In all of our simulations, the characters exhibit smooth and visually convincing motions, avoiding all collisions with minimal effort. The method reproduces emergent behaviour, like lane formation, that have been observed in real crowds. It is relatively easy to implement and and is fast, allowing the simulation of crowds of thousands of characters in real time.

Categories and Subject Descriptors (according to ACM CCS): I.2.9 [Robotics]: Kinematics and dynamics I.2.1 [Applications and Expert Systems]: Games

1. Introduction

An important aspect in realistic crowd simulation is the way that virtual characters interact and avoid collisions with each other. The problem is very challenging since people are accustomed to real-world situations and thus, they can easily detect inconsistencies and artifacts in the simulations.

Over the past few years, several models have been proposed for solving interactions between virtual humans. State-of-the-art techniques for local collision avoidance are based on the seminal work of Reynolds [Rey99], as well as the concept of Velocity Obstacle [FS98] that was introduced in robotics. In these approaches, the trajectories of the characters are linearly extrapolated and used to predict and resolve collisions in the near future. Although such approaches exhibit robust avoidance behaviour, the resulting simulations are far from realistic. The characters seem to lack anticipation, whereas the generated motions are often characterized by oscillatory behaviour. These problems tend to be exacerbated in dense and congested environments, leading to unconvincing crowd flows.

Contributions. In this paper, we address the problem of visually compelling and natural looking avoidance behaviour between interacting virtual characters. We first exploit existing motion capture data to gain a better understanding into how humans solve interactions in real-life. Based on our analysis and some known facts about human locomotion, we propose a model for realistic collision avoidance. In our approach, each character anticipates future collisions and tries to resolve them in advance by slightly adapting its orientation and/or speed. Consequently, the characters avoid all collisions as early as possible and with minimal effort which results in a smooth and optimal flow.

We demonstrate the potential of our approach in several challenging scenarios, as can be seen in Figure 1. Experiments show that our model leads to energy-efficient motions and considerably less curved paths, compared to existing approaches. The generated motions are oscillation-free, allowing us to directly infer the orientation of the characters from their underlying velocities without the need of smoothing out abrupt changes in their steering directions. The technique is easy to implement and can be used to simulate crowds of thousands of characters at real-time frame rates.

Organization. The remainder of the paper is organized as follows. Section 2 provides an overview of prior work related to our research. In Section 3, we analyze experimental interaction data and present some key concepts regarding how pedestrians interact with each other in real life. Our model for local collision avoidance is described in Section 4. Experiments to evaluate our approach are presented in
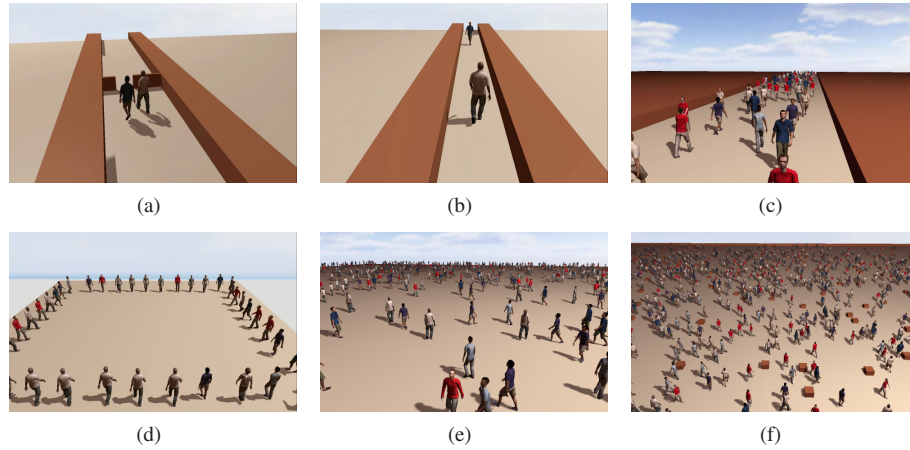
Figure 1: *Test-case scenarios: (a)-(b) Interactions in confined environments. (c) Agents crossing paths in a hallway. (d) Square scenario. (e) 1000 agents in an obstacle-free environment. (f) 2000 agents navigating through an obstacle-filled environment.*

Section 5. Finally, some conclusions and plans for further research are discussed in Section 6.

2. Related Work

Numerous models have been proposed to simulate individuals, groups and crowds of interacting characters. These include continuous methods that unify global and local navigation into a single framework (e.g. [Hug03, TCP06]), as well as approaches that decompose global planning from local collision avoidance (e.g. [LD04, GO07, SGA*07]). Since our current research focuses on local interactions between virtual characters, this section briefly overviews existing work in collision avoidance.

Reactive steering. In reactive steering, the character adapts its previously computed motion to the dynamic and static obstacles found along its path. Reactive navigation techniques originate from the robotics community and are based on variants of potential-field approaches [Kha86]. In the animation community, the concept of reactive planning was introduced by the seminal work of Reynolds on boids [Rey87]. Reynolds used simple local rules to create visually plausible flocking behaviour. In the civil and traffic engineering community, Helbing used physical forces to describe the social interactions between virtual pedestrians [HM95]. Since his original work, a number of models have been proposed to simulate crowds of pedestrians under normal and emergency situations (e.g. [HBJW05, PAB07]). Nevertheless, in all these approaches, due to lack of anticipation and prediction, the characters interact when they get sufficiently close. Consequently, the resulting motions tend to look unnatural and contain undesirable oscillations.

Collision prediction. An alternative way for solving interactions between virtual humans is based on collision prediction; assuming that each character maintains a constant ve-

locity, the future motions of the characters are predicted by linearly extrapolating their current velocities and then used to detect and avoid collisions in the near future. Based on this concept, Reynolds has introduced the *unaligned collision avoidance behavior* [Rey99], whereas Feurtey [Feu00] devised an elegant collision detection algorithm that predicts potential collisions within time and resolves them by adapting the speed and/or the trajectories of the agents. Inspired by Feurtey's work, Paris *et al.* [PPD07] presented an anticipative model to steer virtual pedestrians without colliding with each other. More recently, Karamouzas *et al.* [KHBO09] tackled the collision prediction problem using social forces, whereas Pettré *et al.* [POO*09] proposed an egocentric model for solving pair-interactions between virtual pedestrians based on a detailed experimental study. In their model, collisions are resolved using a combination of velocity and orientation adaptations depending on the role that each pedestrian has during the interaction (passing first or giving way).

Closely related to the aforementioned techniques is the notion of *Velocity Obstacle* (VO) introduced in [FS98]. For a given entity, the VO represents the set of all velocities that would result in collision at some moment in time with another entity moving at a given velocity. Using the VO formulation, any number of obstacles can be avoided by considering the union of their VOs and selecting a velocity outside the combined VO. Recently, van den Berg *et al.* [vdBLM08] extended the VO formulation to guarantee oscillation-free behaviour between two interacting characters and introduced the concept of *Reciprocal Velocity Obstacle* (RVO). RVO randomly samples the velocity space and heuristically searches for a velocity that leads to a collision-free motion, minimizing at the same time the deviation from the character's desired velocity. Based on the RVO formulation, Guy *et al.* [GCK*09] proposed a highly-parallel al-

gorithm for collision avoidance. Their proposed *Fast Velocity Obstacle* (FVO) provides avoidance behaviour similar to the RVO, but is considerably faster since it only guarantees collision avoidance for a discrete time interval and uses a discrete optimization method to compute the final motion of each character. Similarly, the *Optimal Reciprocal Collision Avoidance* (ORCA) formulation allows each agent to select a collision-free velocity by solving a low dimensional linear program [vdBGLM09].

Example-based approaches. Example-based techniques have also been explored for simulating interacting virtual characters. These approaches use example behaviours from video or motion capture data to drive the simulation of virtual characters. In [LCL07], a database of human trajectories is learnt from real pedestrian crowds. During the simulation, each virtual character searches and retrieves the most similar example from the database. A similar approach has been proposed by Lee *et al.* [LCHL07] aiming at reproducing human group behaviours in simulated environments. Although such methods can realistically model crowds of virtual humans, their applicability is limited by the size of the example databases. In addition, these approaches are too computationally expensive for real-time interactive applications and are typically used for offline simulations.

Comparison to previous work. Our approach is somewhat similar in nature to the VO methods mentioned above. We also use the velocity space to plan the avoidance maneuvers of each virtual character. However, we significantly reduce the set of admissible velocities by taking into account how imminent potential collisions are. Note that approaches like FVO also reduce the admissible velocity domain. Our model, though, is based on observed behaviour of real people ensuring convincing avoidance behaviours.

To determine an optimal solution velocity among the admissible ones, we use a cost function similar to the ones proposed in [Feu00, PPD07, vdBLM08]. Nevertheless, we also take into account the energy that is required to perform a certain avoidance maneuver. This ensures smooth motions, allowing us to directly infer the character's orientation from the direction of its velocity vector, even for dense and congested examples. Note that in most velocity-based approaches, the orientation may oscillate over adjacent keyframes and thus, some sort of smoothing is required, which may lead to undesirable effects.

Our approach also bears some resemblance to the model proposed by Pettré *et al.*, since we use their experimental study to elaborate our collision avoidance algorithm. Our analysis, though, focuses on the *predicted time to collision* between interacting participants and the deviation from their desired velocities, whereas they studied the effect that the *minimum predicted distance* has on the participants' accelerations. Additionally, their approach is mainly suited for pair-interactions. Although it has been extended to simultaneously solve multiple interactions, in the resulting simulations the characters seem to abruptly stop and change their orientation. In contrast, our approach can be used to resolve collisions among multiple characters in real-time, yielding to visually convincing simulations.

3. Human Locomotion and Collision Avoidance

In this section, we present some key concepts about human locomotion and human-human interactions. We subsequently analyze existing motion capture data to gain more insight into how individuals solve interactions and avoid collision with each-other in real-life. In the next section, we use our observations to formulate a model of realistic collision avoidance.

Notion of personal space. An important concept in social interactions is the *personal space* that surrounds an individual. More formally, the personal space is the portable territory around an individual that others should not invade. It regulates the psychophysical distance that the individual needs to maintain in order to feel comfortable. The size and the shape of the personal space are constantly changing depending on the crowd density, as well as the travel speed of the walker. According to Goffman [Gof71], the personal space can be represented as an oval, narrow to the sides of the individual and long in front of him/her. Gérin-Lajoie *et al.* [GLRM05] have experimentally confirmed Goffman's observations and found that this area can be defined by an elliptic curve.

Locomotion and energy expenditure. An extensive amount of work has been conducted on human and bipedal locomotion. Based on studies related to the energy expenditure in human walking, it is well accepted that individuals favour energy efficient motions. It has been experimentally confirmed that humans adapt their motions so that the energy required to perform a step becomes minimal (see e.g. [Inm66]); the energy is transferred back and forth from kinetic to potential, minimizing the total energy expenditure. The criterion of minimum energy has been widely used in recent approaches in the field of robotics and computer animation that are based on optimization theory. Such approaches include the total energy consumption in their objective functions aiming to obtain believable and natural-looking motions (see e.g. [WP03, SH07]).

Interactions and the principle of least effort. The *principle of least effort* originates from the field of psychology stating that given different possibilities of actions people select the one that requires the least effort [Zip49]. Based on the least effort theory, many systems for crowd simulation have been proposed. Crowd Dynamics [Sti00], for example, uses an agent-based framework in which agents follow the paths of least resistance. More recently, Guy *et al.* [GCC*10] introduced a novel algorithm for simulating large-scale crowds by computing a biomechanically energy-efficient trajectory for each individual agent. However, all

these approaches aim to control the macroscopic (global) behaviour of the virtual humans, whereas our focus is on the local interactions of the individuals. Therefore, based on the principle of least effort, we hypothesize that an individual, upon interacting with another individual, tries to resolve potential collisions as early as possible by slightly adapting his motion.

The individual will adjust his trajectory in advance, trying to reduce the interactions with the other walker. By slightly changing his orientation or speed, he makes his intentions known to the other. Eventually, some sort of coordination takes place and the collision is resolved with minimal effort. In the following, we test the validity of our hypothesis by analyzing publicly available experimental data.

3.1. Experimental Analysis

The experimental dataset was provided by Pettré and his colleagues in order to gain more insight into interactions between pairs of walkers. 30 subjects participated in the experiment and in total 429 trials were recorded. Each trial consists of two participants crossing paths orthogonally while walking toward the opposite corners of a square area. We refer the reader to [POO*09] for a detailed explanation of the experimental protocol.

Our analysis focuses on the effect that the time to collision between two interacting participants has on the deviation from the desired motion of each participant. In each trial, similar to [POO*09], the *interaction* starts at time t_s when the two participants can see each other and ends at time t_f when the distance between the participants is minimal, that is

$$t_f = \underset{t \geq t_s}{\operatorname{argmin}} \|\mathbf{x}_2(t) - \mathbf{x}_1(t)\|, \tag{1}$$

where $\mathbf{x}$ defines the position of the participant's trunk [ALHB06], approximated by interpolating the two shoulder markers of the participant. During the interaction period, $t_s < t < t_f$, we estimate the future motions of the two participants by linearly extrapolating their current trajectories:

$$\mathbf{x}_i' = \mathbf{x}_i + t\mathbf{v}_i, \ i \in \{1, 2\}, \tag{2}$$

where the participant's velocity $\mathbf{v}_i$ is calculated using backward finite difference. Then, we can determine whether and when the participant P_1 will collide with P_2 as follows:

$$\|\mathbf{x}_2' - \mathbf{x}_1'\| = r_1 + r_2, \tag{3}$$

where r denotes the radius of a participant derived from the distance between the two shoulder markers. Solving the above equation, we can predict the time to collision $tc(t)$ between P_1 and P_2 at time t. To be able to run comparisons over all trials, the predicted time to collision for any time
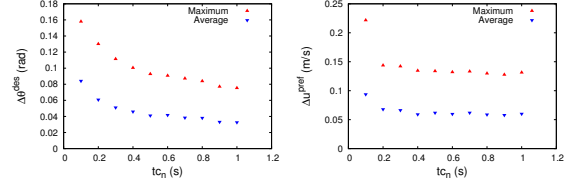

Figure 2: *left: maximum and average deviation from the desired orientation as a function of the normalized time to collision, right: maximum and average deviation from the preferred speed as a function of the normalized time to collision.*

$t_s < t < t_f$ is normalized as follows:

$$tc_n(t) = \frac{tc(t)}{\underset{t \in [t_s, t_f]}{\max} \{tc(t)\}} \tag{4}$$

$tc_n(t)$ ranges from 0 to 1, where 0 indicates that the two participants are already colliding and 1 corresponds to the maximum time to collision between the two participants for the current trial (note that the maximum collision time averaged over all trials is $4.2s$).

Time to collision and desired orientation. For any time $t_s < t < t_f$, we define the participant's deviation from his/her desired orientation as follows:

$$\Delta\theta_i^{des}(t) = \arccos\left(\frac{\mathbf{v}_i(t)}{\|\mathbf{v}_i(t)\|} \cdot \mathbf{n}_{\theta_i^{des}}\right), \tag{5}$$

where $\mathbf{n}_{\theta^{des}}$ is the unit vector pointing towards the goal position of the participant. Figure 2 (left) plots the participants' deviations from their desired orientations as a function of the normalized time to collision. We grouped the tc_n into 10 clusters of equal length and then determined the maximum and average deviation angle per cluster. As can be inferred from the figure, the maximum deviation angle is quite small for predicted collisions that will take place in the far future, $tc_n \geq 0.9$. After a small increase, the max deviation angle remains more or less constant for a long period, $0.45 \leq tc_n \leq 0.8$. As soon as potential collisions starts to become imminent ($tc_n < 0.4$), the deviation angle increases reaching to a peak when the tc_n tends to 0. Similar trend is also observed when looking at the evolution of the average deviation angle with respect to tc_n.

Time to collision and preferred speed. During the interaction period, we define the preferred speed u^{pref} of a participant by taking the average speed over this period. Thus, for any time $t_s < t < t_f$, the participant's deviation from his/her preferred speed can be described by:

$$\Delta u_i^{pref}(t) = \left| \|\mathbf{v}_i(t)\| - u_i^{pref} \right| \tag{6}$$

Figure 2 (right) shows the effect that the normalized time to collision has on the maximum and average speed deviation of the participants. Compared to the orientation deviation, the speed deviation remains constant for a longer period of
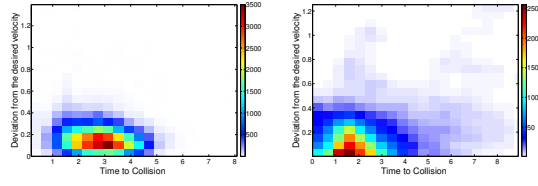
Figure 3: *Density of the deviation from the desired velocity as a function of the predicted time to collision.* **left:** *participants that cross paths orthogonally,* **right:** *participants that have to avoid head-on collisions.*

time, $tc_n \geq 0.2$. Note also the abrupt increase of the maximum deviation at $tc_n < 0.2$, which allows the participants to successfully resolve threatening collisions by refining their speed.

Discussion. In the aforementioned, we studied the influence that the normalized time to collision has on the speed and the orientation of the participants. To obtain a clear overview of how participants solve interactions, we also cumulated all pairs of predicted collision times and corresponding deviation velocities $\Delta \mathbf{v}^{des}$ for all trials. For any time $t_s < t < t_f$, the participant's deviation from his/her desired velocity is defined as follows:

$$\Delta \mathbf{v}_i^{des}(t) = \|\mathbf{v}_i(t) - \mathbf{v}_i^{des}\|, \tag{7}$$

where the desired velocity is determined by the participant's preferred speed and orientation, that is $\mathbf{v}_i^{des} = u_i^{pref} \mathbf{n}_{\theta_i^{des}}$. Figure 3 (left) shows the density plot of the corresponding bivariate data. As can be seen, in most of the trials the majority of the participants prefer to solve interactions in advance, that is when $2.0 \leq tc \leq 4.0$, favouring small changes in their velocities. Note also that very rarely participants have to adapt their motions at imminent collision times, that is when tc is close to 0, which shows the ability of people to efficiently predict and avoid collisions.

In conclusion, our analysis confirms our hypothesis that individuals resolve collisions long in advance by slightly adjusting their motion. The current study focuses on participants that have perpendicular trajectories. Another challenging case is when interacting participants have to avoid head-on collisions. For that reason, we have recently conducted an experimental study similar to the one proposed by Pettré *et al.* Preliminary analysis of the corresponding interactions data seems to support our main hypothesis (see Figure 3, right). Note, though, that due to the fact that the participants have exactly opposite desired directions, some miscommunication on how they pass each other might be detected. Thus, as can be observed in the figure, there are cases that collisions are resolved at the last moment.

4. Collision Avoidance

In this section, we present our velocity-based algorithm for local collision avoidance. Our approach is derived from our

experimental observations and can be integrated into existing approaches for crowd simulation.

Problem formulation. In our problem setting, we are given a virtual environment in which n heterogeneous agents $A_1, ..., A_n$ have to navigate without colliding with the environment and with each other. For simplicity we assume that each agent moves on the 2D plane and is modeled as a disc with radius r_i. At a fixed time t, the agent A_i is at position $\mathbf{x}_i$, defined by the center of the disc, and moves with velocity $\mathbf{v}_i$. The motion of the agent is limited by a maximum speed $u_i^{\max}$. Furthermore, at every time step of the simulation, the agent has a desired velocity $\mathbf{v}_i^{des}$ that is directed towards the agent's goal position g_i and has magnitude equal to the agent's preferred speed u_i^{pref}.

4.1. Agent Collision Avoidance

An agent A_i solves interaction with the other agents in three successive steps:

Step 1 - Retrieve the set of colliding agents. In the first step of our algorithm, we compute the set CA_i of first N agents that are on collision course with the agent A_i. We first extrapolate the future position of A_i based on its desired velocity $\mathbf{v}_i^{des}$. Similarly, we predict the future motions of all the nearest agents that A_i can see by linearly extrapolating their current velocities (A_i can only estimate the actual velocities of the other agents and not their desired ones). The viewing area of A_i is modeled as a cone defined by A_i's desired direction of motion and an effective angle of sight of $200°$ corresponding to the field of view of a typical human.

Based on the predicted trajectories, we can now determine whether the agent A_i will collide with another agent A_j. We assume that a collision occurs when A_j lies inside or touches the personal space of A_i, resulting in the following equation:

$$\|(\mathbf{x}_j + \mathbf{v}_j t) - (\mathbf{x}_i + \mathbf{v}_i^{des} t)\| \leq r_j + (r_i + \mu_i), \tag{8}$$

where $r_i + \mu_i$ denotes the size of A_i's personal space (default value of the minimum security distance μ is set to $\mu = 0.5m$). Note that in our simulations, for efficiency reasons, the personal area has a disc shape. This led to realistic behaviour and hence, the use of an elliptical personal space was not necessary. Solving the above equation for t, we can deduct the possible collision time tc_{ij} between A_i and A_j. If $tc_{ij} \geq 0$, the agent A_j is inserted into the set of the agents that are on collision course with A_i. Consequently, the set CA_i is defined as:

$$CA_i = \bigcup_{j \neq i, 0 \leq tc_{ij}} \{A_j, tc_{ij}\} \tag{9}$$

We sort this set in order of increasing collision time and keep the first N agents. Experiments have indicated that this number can be kept small (default value is $N = 5$). This not only reduces the running time of our algorithm, but also reflects natural human behaviour. In real-life, an individual
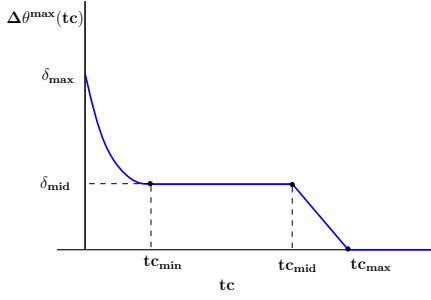
Figure 4: *Maximum orientation deviation for an agent, as a function of the predicted time to collision.*

takes into account a limited number of other walkers, usually those that are on collision course with him/her in the coming short time.

Step 2 - Determine the set of admissible orientations and speeds. In the second step of the algorithm, we retrieve the set of candidate orientations $O_i \in \mathbb{R}^2$ and speeds $U_i \in \mathbb{R}$ that the agent A_i can select in order to resolve the collisions with the agents that belong to the set CA_i. First, we determine the collision time tc with the most threatening agent, that is the agent in the set CA_i with which A_i will collide first:

$$tc = \min_{\substack{tc_{ij} \in CA_i}}^{j \neq i} \{tc_{ij}\} \qquad (10)$$

Then, based on our analysis in Section 3.1, we compute the maximum angle that the agent A_i can deviate from its desired velocity. The maximum deviation angle $\Delta\theta_i^{\max}$ is approximated by a piecewise function (see Figure 4) as follows:

$$\Delta\theta_i^{\max}(tc) = \begin{cases} \frac{\delta_{\max} - \delta_{mid}}{e^{tc}} + \delta_{mid}, & \text{if } 0 \leq tc < tc_{\min} \\ \delta_{mid}, & \text{if } tc_{\min} \leq tc < tc_{mid} \\ \delta_{mid} \frac{tc_{mid} - tc}{tc_{\max} - tc_{mid}} + \delta_{mid}, & \text{if } tc_{mid} \leq tc \leq tc_{\max} \\ 0, & \text{if } tc_{\max} < tc \end{cases} \qquad (11)$$

The $tc_{\max}$ defines the maximum time that A_i anticipates a collision. Note that, in the experimental analysis, the maximum collision time averaged over all experiments is $4.2s$. However, in our simulations, each agent has to simultaneously solve interactions with multiple agents. Thus, a higher anticipation time is used to ensure smooth avoidance behaviour (default value is $tc_{\max} = 8s$). The threshold tc_{mid} regulates the start of the constant part of the function, whereas the $tc_{\min}$ defines collisions that are imminent to A_i leading to higher deviation angles (default values are $tc_{mid} = 6s$, $tc_{\min} = 2.5s$). The parameter $\delta_{\max}$ determines the maximum admissible angle that the A_i can deviate from its desired direction of motion. In our simulations, we assume that virtual characters cannot backtrack and thus, $\delta_{\max} = \pi/2$. Finally, the parameter δ_{mid} defines the deviation angle during the constant interval of the function (default value is $\delta_{mid} = \pi/6$).

Having retrieved the maximum deviation angle $\Delta\theta_i^{\max}$, we determine the agent's admissible orientation domain O_i as follows:

$$O_i = \{ \mathbf{n}_\theta \mid \theta \in [\theta^{des} - \Delta\theta_i^{\max}, \theta^{des} + \Delta\theta_i^{\max}] \}, \qquad (12)$$

where $\mathbf{n}_\theta = [\cos\theta, \sin\theta]^T$ is the unit vector pointing in direction θ and θ^{des} is the orientation derived from A_i's desired velocity.

A similar approach is also used to determine the admissible speed domain U_i of the agent A_i. Based on the speed deviation plot, shown in Figure 2, the U_i is approximated as follows:

$$U_i = \begin{cases} u \mid u \in [0, u^{\max}], & \text{if } 0 \leq tc \leq tc_{\min} \\ u \mid u \in [u^{pref}, u^{pref} \pm \Delta u_i^{\max}], & \text{if } tc_{\min} < tc \leq tc_{\max} \\ u^{pref}, & \text{if } tc_{\max} < tc \end{cases} \qquad (13)$$

where for the maximum speed deviation $\Delta u^{\max}$ holds that $\Delta u_i^{\max} \leq \min(u^{\max} - u^{pref}, u^{pref})$. In our simulations, we set the default value of $\Delta u_i^{\max}$ to $\Delta u_i^{\max} = 0.4m/s$, whereas the default value for $u^{\max}$ is set to $u^{\max} = 2.4m/s$. Note also that the admissible speed domain takes into account the maximum speed constraint on A_i.

Step 3 - Select an optimal solution velocity. In the final step of the algorithm, we compute an optimal solution velocity for the agent A_i. First, we deduce the set of feasible avoidance velocities FAV_i from the agent's admissible orientation and speed domains as follows:

$$FAV_i = \{u \, \mathbf{n}_\theta \mid u \in U_i \wedge \mathbf{n}_\theta \in O_i\} \qquad (14)$$

In practice, an infinite number of feasible velocities exist. Thus, we restrict the O_i domain to a discrete set of orientation samples (default size of the discretization step is set to 0.078 radians). Similarly, we discretize the U_i domain into a set of adjacent speed samples (default distance between adjacent samples is set to 0.1).

Next, we select the agent's new velocity $\mathbf{v}_i^{new}$ from the set of feasible velocities. Among the candidate velocities $\mathbf{v}^{cand}$, we retain the solution minimizing the energy needed to adapt the motion of the agent, the risk of collisions with the other agents and the deviation from the agent's desired velocity:

$$\mathbf{v}_i^{new} = \arg\min_{\mathbf{v}^{cand} \in FAV_i} \{ \; \overbrace{\alpha(1 - \frac{cos(\Delta\phi)}{2}) + \beta \frac{\left| \|\mathbf{v}^{cand}\| - \|\mathbf{v}_i\| \right|}{u^{\max}}}^{\text{Energy}} \\ + \underbrace{\gamma \frac{\|\mathbf{v}^{cand} - \mathbf{v}_i^{des}\|}{2u^{\max}}}_{\text{Deviation}} + \underbrace{\delta \frac{tc_{\max} - tc}{tc_{\max}}}_{\text{Collisions}} \} \qquad (15)$$

where $\Delta\phi$ defines the angle between the agent's current velocity vector and $\mathbf{v}^{cand}$. Consequently, the energy expenditure in our cost function is approximated by taking into account changes both in the speed and the direction of the

agent. Regarding the collision cost, tc denotes the minimum predicted collision time between the agent A_i and the agents in the set CA_i, assuming that A_i selects a velocity $\mathbf{v}^{cand}$; note that tc is upper bounded by tc_{max}. The constants $\alpha, \beta, \gamma, \delta$ define the weights of the specific cost terms and can vary among the agents to simulate a wide variety of avoidance behaviours. We set the default values to: $\alpha = 1$, $\beta = 0.05$, $\gamma = 1$, $\delta = 1$. In all of our simulations, these parameters generated the best agreement with our experimental observations, leading at the same time to visually convincing simulations.

Having retrieved the new velocity $\mathbf{v}_i^{new}$, the agent advances to its new position $\mathbf{x}^{new}$ as follows:

$$\mathbf{x}_i^{new} = \mathbf{x}_i + \mathbf{v}_i^{new} \Delta t, \qquad (16)$$

where Δt is the time step of the simulation. Note that if the agent is also subject to dynamic constraints (e.g. maximum acceleration), we can either take them intrinsically into account upon constructing the set of feasible velocities, or restrict the newly derived velocity based on the given constraints. During each simulation cycle, we also update the orientation of the agent. The energy term in our cost function does not allow the agent to abruptly change its direction, ensuring smooth avoidance motions. Thus, the new orientation θ_i^{new} of the agent is directly inferred from the solution velocity, that is:

$$\theta_i^{new} = \arctan(\mathbf{v}_i^{new}) \qquad (17)$$

4.2. Resolve Threatening Collisions

In most cases, the agent A_i solves interactions in advance, smoothly evading the other agents. It is possible, though, for the agent A_i to collide with another agent A_j, especially in very crowded environments. A collision occurs when A_j invades the personal space of A_i, which does not necessarily mean that A_j penetrates A_i. As a result, the agent A_i must select a new velocity that will resolve the collision as fast as possible. For that reason, we drop the energy cost term from our optimization function and modify the Equation (15) as follows:

$$\mathbf{v}_i^{new} = \operatorname*{arg\,min}_{\mathbf{v}^{cand} \in FAV_i} \{\gamma \frac{\|\mathbf{v}^{cand}\|}{u^{max}} + \delta \frac{tc}{tc_{max}}\}, \qquad (18)$$

where now tc denotes the time that is needed to escape the collision, that is the time that A_i needs to exit from the disc $B(\mathbf{x}_j, r_j + \mu_i + r_i)$ centered at agent A_j and having radius equals to the sum of radius of A_j and the personal space of A_i. Note also that the desired velocity in the deviation cost term is set to zero.

4.3. Avoiding Static Obstacles

An agent A_i has also to avoid colliding with the static obstacles of the environment. In our simulations, such obstacles

are modeled as axis aligned boxes. Then, collisions are resolved by following an approach similar to the one described above.

We first retrieve the nearest obstacle neighbours of agent A_i. We assume that the agent has a global knowledge about the environment and thus, we take all nearest obstacles into account and not only the ones that are inside the agent's visual field. In the second step of the algorithm, we define the set of admissible speeds and orientations for the agent, based on the minimal time to collision between the agent and its obstacle neighbours. These sets are then merged to the admissible speed and orientation domains defined in the second step of Section 4.1. In our simulations, we assume that the time to collision has the same effect on agent-agent interactions and on static obstacle avoidance. Thus, the admissible speed and orientation domains of A_i are defined by the agent's most threatening neighbour, whether this is an obstacle or an agent. Finally, a new velocity $\mathbf{v}_i^{new}$ is computed as described in the third step of Section 4.1.

We note that our algorithm allows the agent to avoid collisions with the obstacles but not to move around them. Therefore, some higher level path planning approach should be used to ensure that the agent cannot get stuck in local minima and can intelligently plan its path around the obstacles (see e.g. [LD04, GO07]).

5. Experimetal Results

We have implemented our collision avoidance algorithm to test its applicability in real-time applications and validate the quality of the generated motions.

5.1. Quality Evaluation

We evaluated our approach against a wide range of test-case scenarios, as proposed in the *Steerbench* framework of Singh *et al.* [SKFR09]. These scenarios range from simple interactions between pairs of agents to more challenging and large test cases (see Figures 1 and 5):

- Overtake: An agent moves down a hallway and encounters a slower agent in front.
- Squeeze: Two agents have to avoid a head-on collision while walking through a narrow corridor (Figure 1(b)).
- Doorway: Two agents travelling in the same direction have to pass through a narrow door to reach the other side of a hallway (Figure 1(a)).
- Confusion-obstacle: Four agents travelling in opposite directions cross paths, avoiding at the same time collisions with a static obstacle.
- Oncoming-groups: A group of agents encounters another group travelling in opposite direction.
- Hallway: Many agents cross paths in a hallway while walking from opposite directions (Figure 1(c)).
- Crossing: Two groups of agents having perpendicular trajectories meet at the intersection of a crossing (Figure 5).
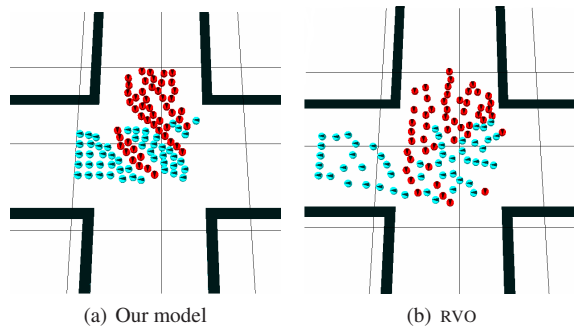
(a) Our model   (b) RVO

Figure 5: *Interactions at a crossing. Using our approach, stripe formations emerge that result in an efficient and smooth flow compared to* RVO.

- Group-swap: Two groups of 50 agents travel in opposite directions and have to exchange their positions.
- Square: 40 agents are placed along the perimeter of a square and have to walk toward their antipodal positions (Figure 1(d)).
- Random: 1000 agents with random goals navigating through an environment void of obstacles (Figure 1(e)).
- Forest: 2000 agents wandering through an environment filled with small-sized obstacles (Figure 1(f)).

We refer the reader to the video accompanying this paper for the results we obtained in simulating several of the aforementioned scenarios. Artifacts in the video, such as foot sliding, are due to our simplified animation technique and not because of our steering algorithm. The virtual characters were animated with a simple motion capture database consisting of walking and idle animations. The speed of the animation was scaled to the speed of the characters, whereas animation blending was used to smoothly transition from one animation to another.

Realism. In all of our simulations, the agents smoothly evade collisions with other agents and static obstacles. As can be observed in the supplementary video, the motions of the agents are oscillation-free. In addition, our approach exhibits emergent phenomena that have been observed in real crowds, such as the dynamic formation of lanes (see e.g. hallway scenario), queuing behaviour (see e.g. oncoming-groups and square scenarios), formation of diagonal stripes in crossing flows (see e.g. intersection scenario), as well as the emergence of slowing down and stopping behaviour to efficiently resolve imminent collisions (e.g. random and forest scenarios). Natural, human-like, behaviour is also captured by our system. In the doorway scenario, for example, one of the two agents slows down and gently allows the other one to pass through the door first. Note, though, that after the collision has been resolved, the two agents will keep moving behind each other instead of reforming to walk side-by-side. To alleviate this, we have recently extended our method to take into account group formations (see [KO10] for details).
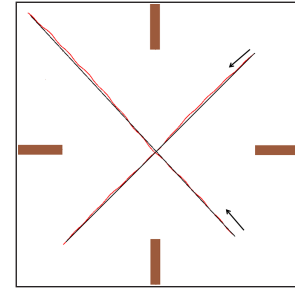

Figure 6: *Comparison between real (red lines) and simulated (black lines) trajectories.*

Comparisons. Using our model, we reproduced several interaction examples from the experimental dataset of Pettré *et al.* As can be inferred from the companion video, the simulated agents were able to follow the trajectories of real humans very closely. Figure 6 shows a comparison between a real and a simulated interaction. In this example, at any time step, the simulated trajectories deviate from the real ones by an average of only $0.21m$.

We have also run comparisons with van den Berg's Reciprocal Velocity Obstacle method (using the RVO library [vdBPS*08]). We chose the RVO because of its increased popularity among the methods that are based on the VO formulation, and its many existing variants. The main difference between our approach and VO methods is that in the latter, at every simulation step, each agent tries to find an optimal collision-free velocity. Consequently, in rather confined and crowded environments, the agent may not be able to find such a velocity and thus, the only solution would be to abruptly adjust its direction or change its speed and stop. In contrast, our approach (by reducing the set of admissible velocities) favours small changes in the velocity of each agent, even though such changes may lead to a collision in the (far) future. Assuming that the other agents will also slightly adapt their motions, collisions are resolved in advance with minimal effort resulting in a smoother, more optimal flow.

We also quantitatively compared our approach with RVO. For that reason, a number of quantitative quality metrics have been devised to objectively evaluate the steering behaviours of the virtual agents. In particular, we computed the total time that an agent needs in order to reach its goal. Furthermore, we used the sum total of squared curvature samples along an agent's path as a measure of the smoothness of such a path [KH97]. Since we strive for effort efficient movements, we also computed the total acceleration of each agent [SKFR09], approximated as the sum of its instantaneous accelerations. To explicitly indicate the amount of turning effort spent by the agent, we also measured the total degrees that an agent had to turn while advancing toward its destination.

Scenarios		Time		Smoothness		Total Accel		Degrees Turned	
		Mean	Stdev	Mean	Stdev	Mean	Stdev	Mean	Stdev
Crossing	Our Model	39.41	1.06	0.06	0.16	135.71	61.05	82.25	64.59
	RVO Model	48.28	6.25	3.56	2.3	464.56	130.65	311.84	142.59
Group-swap	Our Model	39.87	0.84	0.08	0.11	190.64	46.86	44.90	16.10
	RVO Model	47.81	5.06	4.07	2.62	528.55	68.86	348.24	104.12

Table 1: *Statistics for the crossing and group-swap scenarios. The reported time, smoothness, total acceleration and total degrees turned are measured in sec, $(rad/m)^2$, m/sec^2 and deg respectively.*

Table 1 reports the corresponding statistics for the crossing (Figure 5) and the group-swap scenarios. A student t-test was performed afterwards to determine how significant the results were. The analysis has shown that our approach leads to time efficient motions, allowing the agents to follow much more smoother paths than the agents in RVO (the differences in time and smoothness are statistically significant, $p < 0.01$). In addition, using our approach, the agents have to spend less effort to avoid potential collisions and thus, the total acceleration is significantly lower compared to RVO ($p < 0.01$). The same applies to the turning effort of the agents.

In the near future, we also plan to quantitatively compare our technique with the new release of the RVO library that is based on the optimal reciprocal collision avoidance (ORCA) formulation [vdBGLM09], as well as with the recent approach of Ondřej *et al.* [OPOD10] that formulates the collision avoidance behaviour of virtual humans using a combination of visual stimuli and motor response laws.

5.2. Performance Analysis

Besides the quality, we are also interested in the performance of our proposed approach. To test its usability in real-time applications, we selected a varying number of agents and placed them randomly across the forest environment. Each agent had to advance toward a random goal position avoiding collisions with the obstacles and the other moving agents; when it had reached its destination, a new goal was chosen.

Figure 7 highlights the performance of our approach for up to $4,000$ agents on a 2.4 GHz Core 2 Duo CPU. Note that our current implementation is unoptimized and uses only one CPU thread for computing the avoidance maneuvers of the agents. As the figure indicates, even for 3000 agents, it takes $129\,ms$ per simulation step to compute the three steps of our collision avoidance algorithm. Since, in our system, the velocities of the agents were updated at $5\,fps$, it is clear that our approach can simulate thousands of virtual characters at interactive rates.

6. Conclusions and Future Work

In this paper, we present a velocity-based method for realistic collision avoidance among virtual characters. The intu-
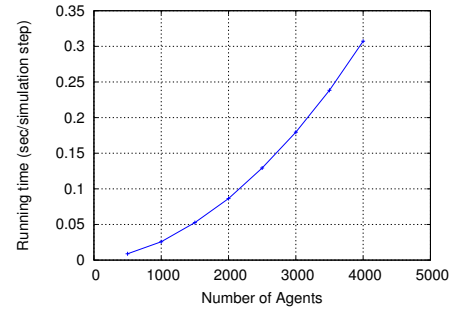

Figure 7: *The performance of our algorithm on the forest scenario as a function of the number of agents. The reported numbers are simulation only.*

ition behind our approach is based on observed behaviour of real people. The main hypothesis of our model is that an individual tries to resolve collisions in advance by slightly adapting his preferred direction and speed. Consequently, in our simulations, the virtual characters exhibit smooth and convincing motions, avoiding all collisions with minimal effort.

We have experimentally confirmed our hypothesis on pairs of interacting participants by studying the effect that the time to collision has on the desired velocities of the participants. Currently, we are developing a tool to semi-automatically track pedestrians in outdoor environments. This will allow us to gain more insight into the avoidance behaviour of individuals, couples and small groups of pedestrians, so that we can extend our model accordingly and capture a wide variety of crowd characteristics.

We also plan to conduct an experimental study in order to investigate the behaviour of individuals when they have to avoid collisions with static obstacles. In our model, the virtual characters avoid in a similar way both static obstacles and other characters. However, we believe that in real life people react to obstacles much earlier and resolve the potential collisions by mainly adapting their orientations. Another important extension for future work is to determine the parameters that affect an individual's desired velocity. In our current work, we assume that the desired speed of a virtual human remains constant during the entire simulation. Nevertheless, parameters such as the crowd density can have a significant effect on the velocity leading to different avoidance behaviours. Finally, we also plan to combine our approach with a more sophisticated animation technique [vBPE10] to obtain higher quality animations.

References

[ALHB06] ARECHAVALETA G., LAUMOND J.-P., HICHEUR H., BERTHOZ A.: The nonholonomic nature of human locomotion: a modeling study. *International Conference on Biomedical Robotics and Biomechatronics* (2006), 158–163. 4

[Feu00] FEURTEY F.: *Simulating the collision avoidance of pedestrians*. Master's thesis, University of Tokyo, 2000. 2, 3

[FS98] FIORINI P., SHILLER Z.: Motion planning in dynamic environments using velocity obstacles. *International Journal of Robotics Research 17* (1998), 760–772. 1, 2

[GCC*10] GUY S., CHHUGANI J., CURTIS S., PRADEEP D., LIN P., MANOCHA D.: Pledestrians: A least-effort approach to crowd simulation. In *SCA '10: ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2010). 3

[GCK*09] GUY S. J., CHHUGANI J., KIM C., SATISH N., LIN M. C., MANOCHA D., DUBEY P.: Clearpath: highly parallel collision avoidance for multi-agent simulation. In *SCA '09: ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2009), pp. 177–187. 2

[GLRM05] GERIN-LAJOIE M., RICHARDS C., MCFADYEN B.: The negotiation of stationary and moving obstructions during walking: anticipatory locomotor adaptations and preservation of personal space. *Motor control 9*, 3 (2005), 242. 3

[GO07] GERAERTS R., OVERMARS M.: The corridor map method: A general framework for real-time high-quality path planning. *Computer Animation and Virtual Worlds 18* (2007), 107–119. 2, 7

[Gof71] GOFFMAN E.: *Relations in public : microstudies of the public order*. New York : Basic books, 1971. 3

[HBJW05] HELBING D., BUZNA L., JOHANSSON A., WERNER T.: Self-organized pedestrian crowd dynamics: Experiments, simulations, and design solutions. *Transportation Science 39*, 1 (2005). 2

[HM95] HELBING D., MOLNÁR P.: Social force model for pedestrian dynamics. *Physical Review E 51* (1995), 4282–4286. 2

[Hug03] HUGHES R.: The flow of human crowds. *Annu. Rev. Fluid Mech 35* (2003), 169–182. 2

[Inm66] INMAN V.: Human locomotion. *Canadian Medical Association Journal 94*, 20 (1966), 1047. 3

[KH97] KANAYAMA Y., HARTMAN B.: Smooth local-path planning for autonomous vehicles. *The International Journal of Robotics Research 16*, 3 (1997), 263. 8

[Kha86] KHATIB O.: Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research 5* (1986), 90–98. 2

[KHBO09] KARAMOUZAS I., HEIL P., BEEK P., OVERMARS M. H.: A predictive collision avoidance model for pedestrian simulation. In *Proc. of Motion in Games* (2009), pp. 41–52. 2

[KO10] KARAMOUZAS I., OVERMARS M.: Simulating the local behaviour of small pedestrian groups. In *VRST '10: Proceedings of the 17th ACM Symposium on Virtual Reality Software and Technology (to appear)* (2010). 8

[LCHL07] LEE K., CHOI M., HONG Q., LEE J.: Group behavior from video: a data-driven approach to crowd simulation. In *SCA '07: ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2007), pp. 109–118. 3

[LCL07] LERNER A., CHRYSANTHOU Y., LISCHINSKI D.: Crowds by example. *Computer Graphics Forum 26* (2007), 655–664. 3

[LD04] LAMARCHE F., DONIKIAN S.: Crowd of virtual humans: a new approach for real time navigation in complex and structured environments. *Computer Graphics Forum 23* (2004), 509–518. 2, 7

[OPOD10] ONDŘEJ J., PETTRÉ J., OLIVIER A.-H., DONIKIAN S.: A synthetic-vision based steering approach for crowd simulation. *ACM Trans. Graph. 29*, 4 (2010), 1–9. 9

[PAB07] PELECHANO N., ALLBECK J. M., BADLER N. I.: Controlling individual agents in high-density crowd simulation. In *SCA '07: ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2007), pp. 99–108. 2

[POO*09] PETTRÉ J., ONDREJ J., OLIVIER A.-H., CRÉTUAL A., DONIKIAN S.: Experiment-based modeling, simulation and validation of interactions between virtual walkers. In *SCA '09: ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2009), pp. 189–198. 2, 4

[PPD07] PARIS S., PETTRÉ J., DONIKIAN S.: Pedestrian reactive navigation for crowd simulation: a predictive approach. *Computer Graphics Forum 26*, 3 (2007), 665–674. 2, 3

[Rey87] REYNOLDS C. W.: Flocks, herds, and schools: A distributed behavioral model. *Computer Graphics 21*, 4 (1987), 24–34. 2

[Rey99] REYNOLDS C. W.: Steering behaviors for autonomous characters. In *the proceedings of Game Developers Conference* (1999), pp. 763–782. 1, 2

[SGA*07] SUD A., GAYLE R., ANDERSEN E., GUY S., LIN M., MANOCHA D.: Real-time navigation of independent agents using adaptive roadmaps. In *VRST '07: Proceedings of the 2007 ACM symposium on Virtual reality software and technology* (2007), pp. 99–106. 2

[SH07] SAFONOVA A., HODGINS J.: Construction and optimal search of interpolated motion graphs. In *ACM SIGGRAPH 2007 papers* (2007), p. 106. 3

[SKFR09] SINGH S., KAPADIA M., FALOUTSOS P., REINMAN G.: Steerbench: a benchmark suite for evaluating steering behaviors. *Computer Animation and Virtual Worlds 20*, 5-6 (2009), 533–548. 7, 8

[Sti00] STILL G.: *Crowd Dynamics*. PhD thesis, University of Warwick, 2000. 3

[TCP06] TREUILLE A., COOPER S., POPOVIĆ Z.: Continuum crowds. *ACM Transactions on Graphics 25*, 3 (2006), 1160–1168. 2

[vBPE10] VAN BASTEN B., PEETERS P., EGGES A.: The step space: example-based footprint-driven motion synthesis. *Computer Animation and Virtual Worlds 21*, 3–4 (2010), 433–441. 9

[vdBGLM09] VAN DEN BERG J., GUY S. J., LIN M. C., MANOCHA D.: Reciprocal n-body collision avoidance. In *International Symposium of Robotics Research* (2009). 3, 9

[vdBLM08] VAN DEN BERG J. P., LIN M., MANOCHA D.: Reciprocal velocity obstacles for real-time multi-agent navigation. In *Proc. of IEEE Conference on Robotics and Automation* (2008), pp. 1928–1935. 2, 3

[vdBPS*08] VAN DEN BERG J., PATIL S., SEWALL J., MANOCHA D., LIN M.: Interactive navigation of multiple agents in crowded environments. In *I3D '08: Proceedings of the 2008 symposium on Interactive 3D graphics and games* (2008), pp. 139–147. 8

[WP03] WU J., POPOVIĆ Z.: Realistic modeling of bird flight animations. *ACM Transactions on Graphics (TOG) 22*, 3 (2003), 895. 3

[Zip49] ZIPF G. K.: *Human Behavior and the Principle of Least Effort*. Addison-Wesley, 1949. 3