


Intuitive Method for Pedestrians in Virtual Environments

View metadata, citation and similar papers at core.ac.uk

brought to you by  CORE

provided by Toulouse Capitole Publications

IRIT, Vortex team

University of Toulouse, 118 route de Narbonne, 31062 Toulouse, France

boes@irit.fr, sanza@irit.fr, sanchez@irit.fr

Abstract. Recent works about pedestrian simulation can actually be sorted in two categories. The first ones focusing on large crowd simulation aim to solve performance and scalability issues at the expense of behavioral realism of each simulated individual. The second ones aim at individual behavioral realism but the computational cost is too expensive to simulate crowds. In this paper, we propose an alternate approach combining a light reactive behavior with cognitive strategies issued from real life videos. This approach aims at the real time simulation of small crowds of pedestrians (one to two hundred individuals) but with concerns for visual realism regarding heterogeneous behaviors, trajectories and positioning on sidewalks.

Keywords: crowd, pedestrians, heterogeneity, small groups, and anticipation.

1 Introduction

At first sight, a real crowd seems to be chaotic and unpredictable. Nevertheless, the local interactions between pedestrians generate auto-organization and emergent structures [4]. For example, parallel lines are formed when many people walk in two opposite directions in a corridor. Moreover, there is a relation between density and average speed of a crowd [15], giving concrete data for computer animations. However, the problem of simulation of a virtual crowd remains quite complex, even if there are many approaches to pedestrian simulation.

Ennis et. al have led a study to find out which criteria are determinant to make a realistic simulation [3]. Most of these criteria are already taken in account by existing simulations, like obstacle avoidance and walking in appropriate areas, but one is not: walking in small groups. Despite it is not mentioned by Ennis, having pedestrians with heterogeneous appearances and behaviors is an obvious key to realism, but not often present in existing models.

In this paper, we present an intuitive approach based on real life observations of pedestrians. The proposed method combines a reactive algorithm of collision avoidance and behavioral strategies. Our goal is to improve visual realism by simulating heterogeneous behaviors and by maintaining small groups of pedestrians.

The second section presents previous works about crowd simulation. The third section presents the model of pedestrian. The third section shows the results of our approach. The last section concludes this paper and gives further works.

2 Related Work

Discrete crowds (also called agent-based simulations) focus on individuals. Local behavioral rules are given to each agent and a realistic global behavior is expected to emerge. The Hidac model [10] uses a combination of psychological and geometrical rules with a social and physical forces model to simulate high-density crowds in normal or panic situations. The behaviors are computed at two levels: the high level behavior (for navigation, learning, communication and decision making) and the low-level motion (for perception, motion and locomotion). A huge number of behaviors are simulated like stopping, queuing, pushing, propagating panic and falling (figure 1a). In [11], Shao simulates a virtual train station. The characters can see all the mobile objects and a limited number of closest mobile objects. They have reactive, motivational and cognitive routines that include low and high level actions like making a purchase or taking a seat. The actions are triggered according to a set of current goals and internal physiological, psychological, or social needs (figure 1b).

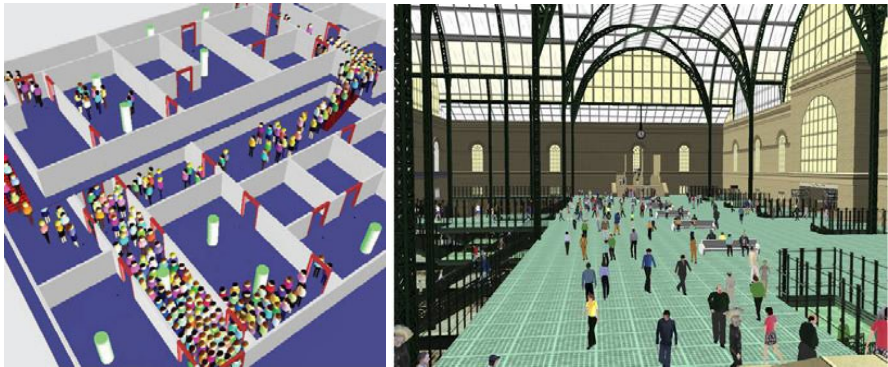


Fig 1 A crowd in a complex building using Hidac [10] (left). The waiting room of the Pennsylvania station [11] (right).

Since each agent makes its own decision, discrete crowds allow a great diversity among pedestrians and provide very realistic behaviors, but the computational cost is expensive, which limits the size of the crowd it can handle. Another drawback of discrete crowds is that the local rules are difficult to create. For example, several rules are needed to achieve simple tasks like obstacle avoidance. Since agents have to perceive the world they populate, discrete crowds are dependent of the type of the environment (indoor or outdoor) and of the way it is constructed.

Continuous crowds have a global point of view. Crowd motion is computed with a potential field used by every pedestrian [6]. In [12], Treuille's works are based on three hypothesis: people have a goal, people try to go as fast as possible, people try to avoid areas of discomfort. Thus, the characters move by trying to minimize these three parameters (length of the path, time and discomfort). He uses potential fields to update people's positions from a combination of different grids (density, goals, boundaries...). The model is used in different simulations like evacuation in urban environment (figure 2a).

Recently, aggregate dynamics have combined discrete and continuous models to reach a large number of pedestrians and to handle very dense crowds [8]. The algorithm consists in computing the preferred velocity for each agent and interpolating it with the continuum velocity of the local flow. Finally, he performs collision resolution. The system can manage very dense crowds, until 100 000 characters at 2 frames per second (figure 2b).

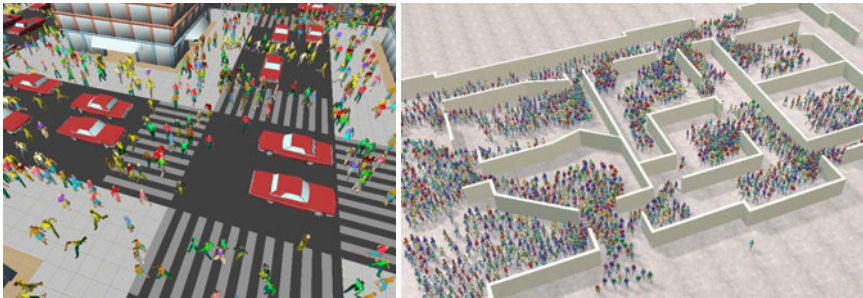


Fig. 2 Evacuation of a building by [12] (left).and [8] (right)

Continuous and aggregate crowds can deal with large dense crowds but are less realistic than agent-based simulations, especially when our eye is caught by one particular character in the simulation. These three approaches are the most popular, but not the only ones existing. The crowd patches method puts together patches of precomputed trajectories [14], it allows to populate infinite worlds



Fig. 3 Animation of crowd with patches [14] (left). Animation of crowd by examples [7] (right).

but virtual humans are not autonomous and the simulation lacks of interactivity (figure 3a).

The crowd by example method constructs a database of situations from the tracking of videos of real crowds (figure 3b). Virtual pedestrians search the database to copy the appropriate trajectory [7]. This method shows very realistic behaviors, but for a small number of pedestrians.

Tables 1 and 2 recapitulate main characteristics and differences amongst the previously described related works.

Table 1 Interactivity means possibilities of manual control or/and on the fly editing of the simulation

	Max number of pedestrians	Real Time Simulation	Interactivity
Shao & Terzopoulos	1200 (without 3D rendering)	Yes (without 3D rendering)	Average
HiDAC	600	Yes	Good
Continuous crowd	10000	No	Low
Aggregate crowds	100000	0 to 10k agents	Low
Crowd patches	3000	Almost	None
Crowds by example	40	No	None

Table 2 Heterogeneity indicates both heterogeneity in characteristics of pedestrians and behaviors. Singularities indicate if the method allows behavioral singularities such as unexpected stops or a pedestrian that do not respect “the rules”. Realism means plausibility of simulated crowds

	Heterogeneity	Singularities	Realism	Groups
Shao & Terzopoulos	Average	No	Good	No
HiDAC	Average	Yes	Good	No
Continuous crowd	None	No	Low	No
Aggregate crowds	None	No	Low	No
Crowd patches	Average	No	Low	No
Crowds by example	Low	Yes	Good	No

According to this table we will try to maintain real-time simulations while handling heterogeneous interactive pedestrians walking in small groups. Although scalability is still one of our concerns, the maximum number of simulated pedestrians is not our main priority in this paper.

3 Approach

We focused on three main goals. First, simplicity and genericity: we wanted our method to be easily implemented in any environment. We also wanted it to be able to allow heterogeneous behaviors and small groups of people. To help us to reach a great degree of realism we shot videos of pedestrians walking the downtown

streets of Toulouse, France. We extracted precious data from these videos. They are described later. The next section explains how we performed the classic tasks of collision avoidance and retention in walkable areas. The last sections present how we introduced heterogeneity and small groups.

3.1 *The Environment*

The virtual town project (figure 4) consists of a virtual model of an urban environment in which evolve virtual cars and virtual pedestrians. This environment has been seen in [1] focusing on automatic learning and handling of crosswalks.

Virtual pedestrians are animated using motion captures. Motion realism is not our main concern but behavioral realism and heterogeneity amongst agents are. So each virtual pedestrian is imbued with its own internal parameters such as maximum velocity, weight, height, and perception capacities.

About perception, the pedestrians are able to perceive vehicles, other pedestrians and objects (traffic lights for example) that are in front of them, in their field of view. The field of view is an angular sector which depth and width are set according to the capacities of the pedestrian.

Bullet Physics (<http://bulletphysics.org>) handles collision detection and their effects on the pedestrians and the objects. The use of physics engine avoids interpenetration in case of collision of two objects.

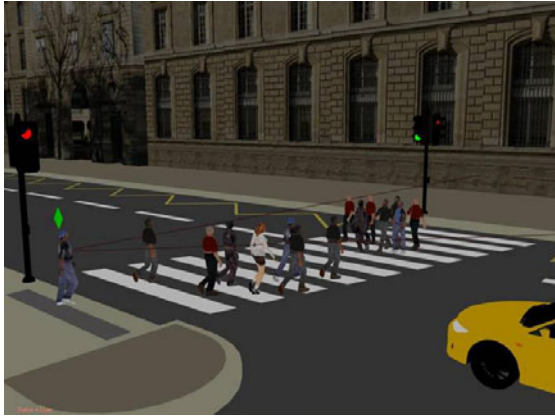


Fig. 4 The virtual town project

3.2 *Collision Avoidance and Walkable Areas*

3.2.1 **Obstacle Avoidance**

We observed on the videos that pedestrians seem to follow a free space created by those who precede them: they favor *free directions*. We represent these *free*

directions in a simple table, called the *direction table*. Each agent has its own table and each cell corresponds to a direction it can take. At any time of the simulation, the value of the cell is the distance the agent can walk, following the corresponding direction, without running into an obstacle (figure 5). The table is initialized with a value that is the maximal distance for which obstacles are taken in account. When an obstacle is perceived, the distance to this obstacle is inserted in the appropriate cell only if it is inferior to the current value of the cell.

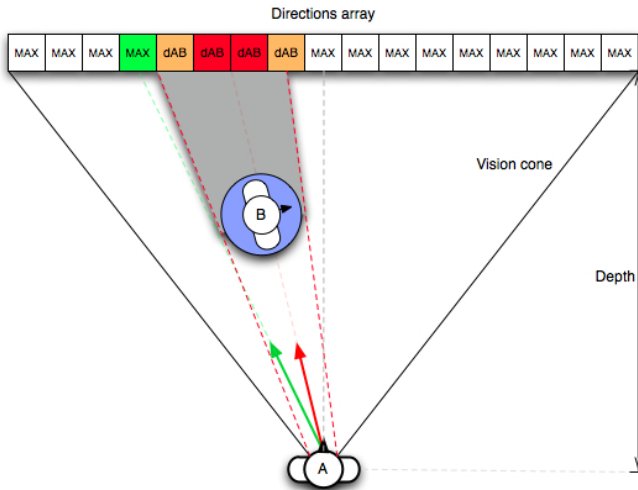


Fig. 5 Agent A perceives agent B on the left. Some cells of the first half of the table are filled with the distance between A and B, the others are empty, they contain the maximal distance. Right arrow represents the aimed direction of the agent A, the left one is the computed direction to avoid agent B.

Each agent computes its *desired direction*. It is the direction that it needs to adopt in order to reach its target. Once its *direction table* is updated according to its perceptions, an agent checks the table if its *desired direction* is free. If not, the agent will have to look for the *closest free direction*. It is the nearest cell containing the maximal distance. The final direction that the pedestrian takes is a weighted average of his *desired direction* and the *closest free direction*, with a greater weight for the latter.

If two cells can pretend to be the closest free direction, the cell with the greater index is chosen. This simulates the natural tendency of people to avoid an obstacle by the right rather than the left when the two solutions are equivalent. In order to obtain smoother trajectories, an agent can adjust its direction even if its *desired direction* is free. This happens when an adjacent cell of the one corresponding to the desired direction contains a small distance. This means that an obstacle is near the trajectory, the agent will then shift its orientation from a cell on the other side in order not to get too close of the obstacle.

The number of cells depends on the angle pitch between each cell. If the pitch is too small, agents don't modify their trajectory strongly enough, if it is too high agents shake and have unnatural trajectories. A pitch of five degrees proved to be the best compromise.

3.2.2 Anticipation

Most of the collisions are easily avoided with this technique (especially with static obstacles), but some still occasionally occur with moving objects. To prevent such collisions, agents don't only perceive size and position of other objects. They also perceive speed and orientation. Therefore they are able to extrapolate the trajectory of other agents. The anticipated position (and not the current position) of perceived obstacles is used to update the *direction table* (figure 6). The amount of time n over which the agent anticipates depends on several criteria: its speed (more he is fast, less he anticipates over a long time), the distance to the other agent (almost no anticipation for very close obstacles) and the angle between the two trajectories (maximum anticipation for perpendicular trajectories, almost no anticipation for parallel trajectories).

Static and moving obstacles are avoided thanks to the same technique, using the same direction table. This allows our method to be easily implemented on any

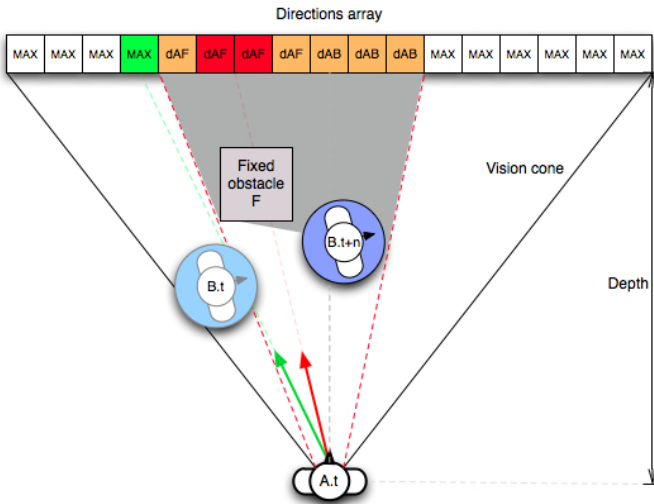


Fig. 6 Agent A extrapolates agent B position. A will avoid B by the left. Without anticipation, A would have turn right and a collision would have occurred for B is moving in this way. The top-left arrow shows the closest free direction cell, the top-right arrow is the desired direction cell.

environment: the only requirement is the perception of distance, position, size, speed and orientation, which is basic. Moreover this technique sticks to reality: if an obstacle stands in our way, we adjust our trajectory just enough to avoid it.

In crowded environment, the method favors the choice of the less crowded space in front of the agent (figure 7).

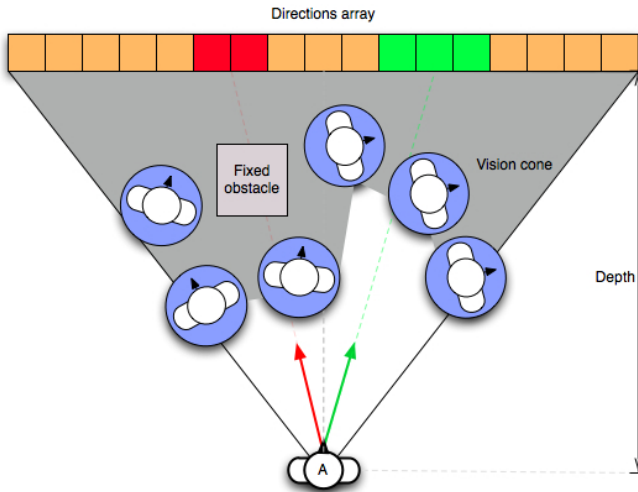


Fig. 7 Example of direction array in crowded environment. The left arrow indicates the desired direction. The right arrow is the computed one.

At each simulation step, the computation of the direction array is performed once for each agent. Performing only one pass ensures a performance gain but can generate some collisions that are easily handled by the physics engine.

3.2.3 Walkable Areas

To ensure that agents stay on the pedestrian network (sidewalks and crosswalks), we tagged borders with *border cells* (figure 8).

They are perceived by agents and treated as obstacles by the direction table. Agents tend to avoid borders, and stay in safe zones. Border cells are not physical obstacles, if an agent is pushed through a border (it happens when sidewalks are crowded), he will cross it and walk on the road. The direction table allows agents to slightly adjust their trajectory but not to make brutal changes, therefore if a pedestrian walks quickly perpendicularly to a border (it happens if his target is on the road), he will cross it. Of course the treatment of border cells is deactivated for pedestrians who walk on the road.

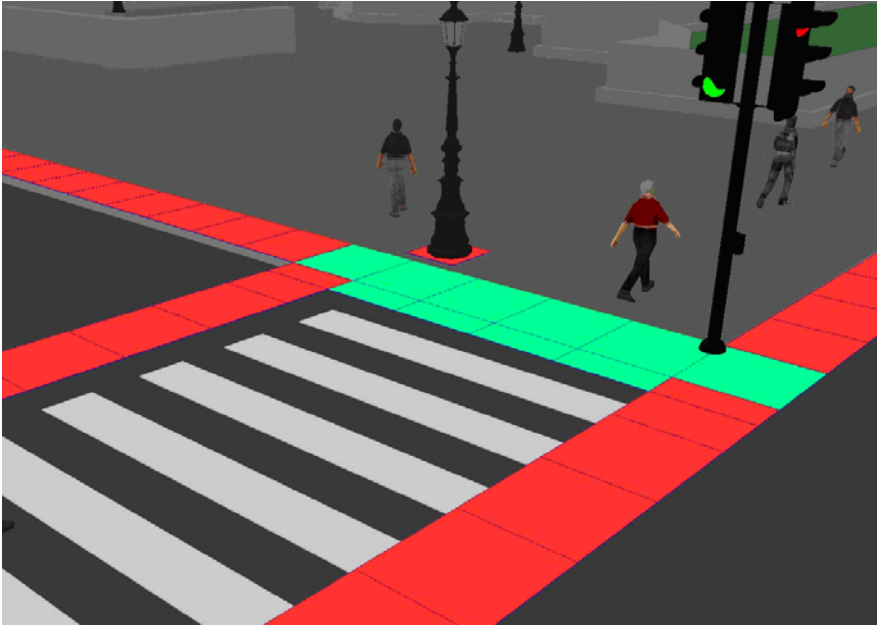


Fig. 8 Border cells (red) prevent pedestrians to massively walk outside the appropriate areas

3.2.4 Slowing Down and Stopping

Each agent computes an *obstruction rate*, depending on the filling of the *direction table* (number of non-empty cells and average distance). A rate equal to zero means an empty table (no obstacle). Agents slow down if the rate becomes too high, but it will never cause them to stop. The number of non-empty cells has to be taken in account in order to avoid very close but small objects.

$$\tau = \sum_{i=0}^N (d_{\max} - d_i) \cdot \frac{N^p}{N}$$

Where N is the total number of cells, N^p the number of non-empty cells and d_{\max} the maximal distance that it was initialized with.

An agent also slows down when someone walks too close in front of him, approximately at same speed and with the same orientation. Thus, they maintain a personal free space (figure 9).

Agents are able to perceive traffic lights. If it is red for pedestrians, agents willing to cross the street will stop when they arrive at the border of the sidewalk or when they get too close to someone else waiting for the light to turn green.



Fig. 9 Agents waiting at a crosswalk

3.3 *Heterogeneity*

In real life, crowds are very heterogeneous, both in terms of behavior and of appearance. This diversity is difficult to simulate but is a key to realism. We focused more on the behavior than on the visual aspect. From our observations, we identified three movement strategies: slow strategy, classical strategy and fast strategy.

Slow strategy: People walking slowly are either older persons or people going for a stroll. As they are about 50% slower than classical pedestrians (6 km/h), they do not care about distant obstacles. They only give attention to what is close to them. Their direction table is initialized with a small maximal distance.

Classical strategy: The majority of pedestrians follow this strategy. Classical pedestrians present an average behavior: they stay in appropriate areas, they slow down when too many people are in front of them, but they overtake if someone is too slow.

Fast strategy: Pedestrians that are rushing try to always walk at their maximum speed. They move about 50% faster than classical agents. Their obstruction threshold is higher. Therefore, they slow down less often than classical pedestrians. They are reckless: they don't give attention to border cells so they easily walk on the road if it allows them to overtake a pedestrian or to take a shorter path (figure 10).



Fig. 10 A fast pedestrian overtakes slower agents and walks recklessly on the road

The repartition of these strategies is important in order to get a realistic simulation: a majority of pedestrians must follow the classical strategy. A crowd composed of 80% classical, 10% slow and 10% fast pedestrians gave good results. These strategies bring heterogeneity and singular behaviors to the simulation. Fast agents do not respect the usual rules, like some people in real life.

3.4 Small Groups

In real life, we observe that more than half the people walk in small groups of two to six pedestrians. We counted on our videos 726 pedestrians, 43% of them walk alone, while 32% walk in pairs, 18% in groups of three people, 7% in groups of four people and the last 2% in groups of five or six people (figure 11).

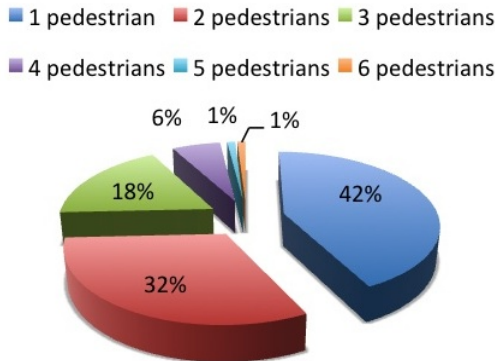


Fig 11 Repartition of pedestrians computed from video observations

In our model, groups consist of a leader and followers. The leader decides of the speed and the direction of the group, followers copy their behavior on him (figure 12).

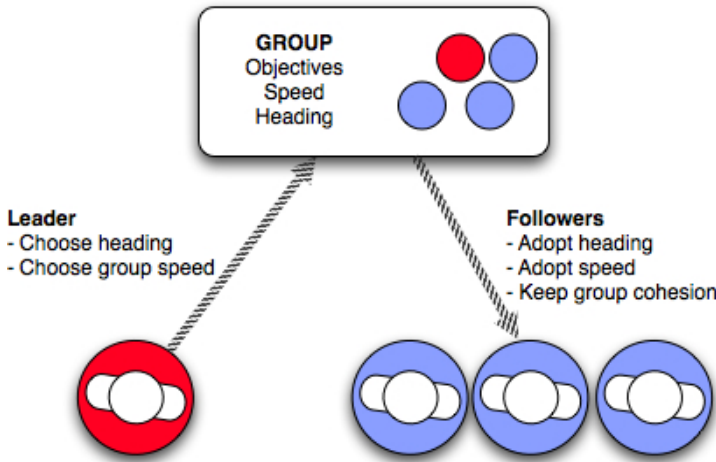


Fig. 12 Interaction between group structure and agents

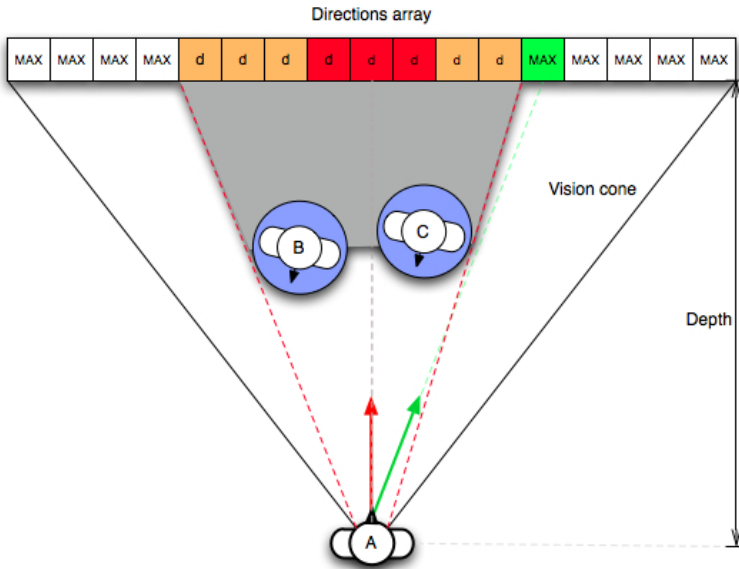


Fig. 13 Agents B and C are part of a group, agent A sees them as a single obstacle, he will not walk between them

They all share the same objectives. How to combine obstacle avoidance and group cohesion is an open question. For now, a group does not perform obstacle avoidance with moving objects. Alone pedestrians perceive groups as a single obstacle. They try to not cut through it (figure 13 and 14).

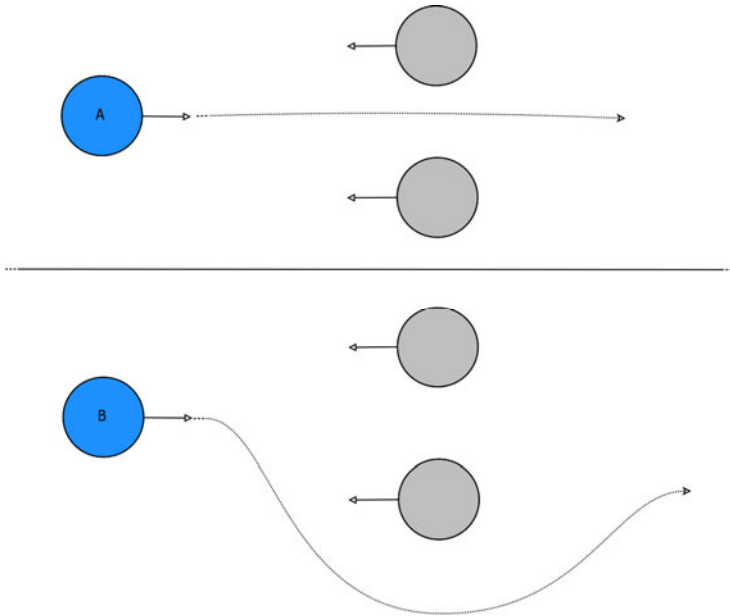


Fig. 14 Up, grey agents are not part of a group. Agent A cuts between them. Down, grey agents are part of a group. Agent B avoids them.

Each agent stores in its memory a list of other agents he knows. If an agent who walks alone (or is the leader of a group) meets one of them during simulation, they both will stop, stand a few seconds face to face and finally form a group. The leader of the new group is chosen arbitrarily. Fast pedestrians never stop when they meet a friend and do not form groups.

4 Results

4.1 Experimental Protocol

We have run several experiences. In each one of them we put several pedestrians in our virtual town. Each agent has an individual navigation path along the sidewalks. They must stay as much as possible on the sidewalks and cross the streets when they are allowed to.

In order to evaluate the pertinence of our choices, we ran series of tests. Each test was based on the same principle: two simulations were launched initialized the same way, but one of them had a deactivated feature or different parameters.

In the following screenshots, the textures of pedestrians indicate their behavior and characteristics as described in table 3.

Table 3 Characteristics and strategies according to textures (when activated in simulations)

Texture	Maximum speed	Strategy
White shirt, white mane	Slow	Slow
Black shirt, gray pants	Medium	Classical
Military outfit	Fast	Classical
Red jacket, black pants	Fast	Fast
Black jacket, blue cap	Very Fast	Fast

4.2 Collision Avoidance

The first test concerned collision avoidance. We ran two simulations with fifty pedestrians initialized the same way (same positions, same strategies repartition, no small group), but agents of one of them did not perform collision avoidance (figure 15).



Fig. 15 Left, collision avoidance is deactivated. Right, collision avoidance is activated.

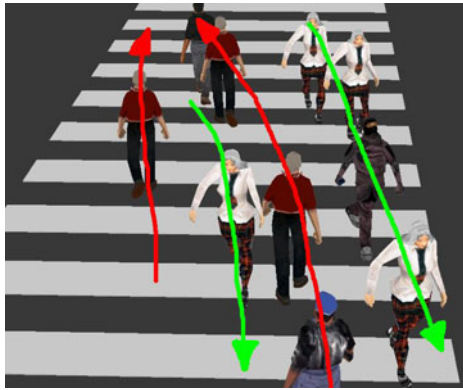
It was visually obvious that having no obstacle avoidance ruins the realism. We also followed five agents on each simulation, during one minute, and counted how many times a collision occurred with one of them. Twenty-five collisions occurred when avoidance is off. Only two occurred when it is activated (table 4).

Table 4 Collisions per pedestrian, one-minute simulation, 40 agents

Pedestrian	A	B	C	D	E	Total
With avoidance	0	0	1	0	1	2
Without avoidance	4	4	6	6	5	25

A usual observed (and realistic) feature of crowd simulation, and more specifically, of opposite flows of walkers, is the formation of lanes [2,5,16] due to collision avoidance. The pedestrian lanes consist of pedestrians that share the same intended direction and approximately the same velocity [13].

Those lanes are also presents in our simulation (figure 16).

**Fig. 16** Lanes formation in opposite flows

4.3 Walkable Areas

We tested the visual impact of retaining pedestrians in appropriate areas with two simulations. Both were initialized with the same pedestrians at the same positions, one of the simulations had *border cells* but the other had not. The result is shown by figure 17: it is obvious that realism is enhanced when agents walk where they are supposed.

4.4 Small Groups

The impact of the presence of small groups was evaluated by the comparison between a simulation where every pedestrian is alone and another where some of them walk together, in small groups. The simulation with small groups seemed more natural (figure 18).

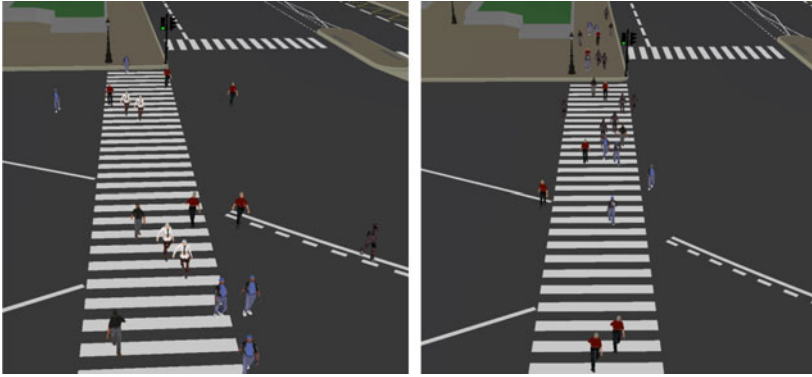


Fig. 17 Left, with no border cell, agents don't stay on sidewalks and crosswalks. Right, agents mainly stay on safe areas. It is obviously more realistic.

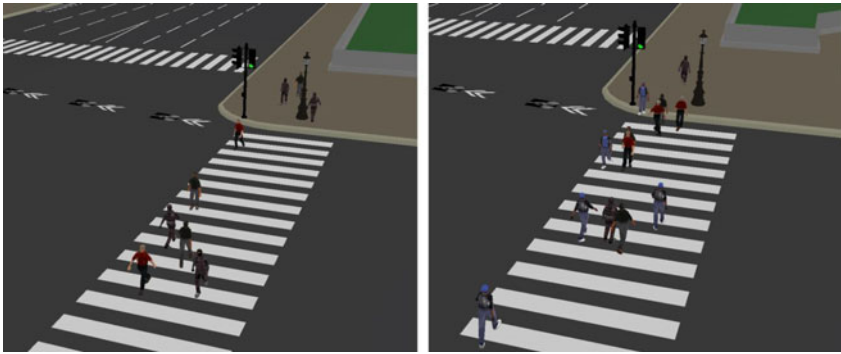


Fig. 18 Left, a simulation where everybody walks alone. Right, some agents walk in pairs or in groups of three persons.

4.5 Heterogeneous Pedestrians and Strategies

To measure the effect of strategies heterogeneity, we ran a simulation where every pedestrian follows the classical strategy and another where the 80-10-10 repartition was respected. The first one gave the impression of a “clone army” since everyone was moving at the same pace (figure 19). To enhance visualization, agents have been colored in accordance with their initial position. Left, a series of screen shots shows the evolution of a simulation where every agent follows the classical strategy. The crowd seems “frozen”. Right, 80% of the agents follow the classical strategy, 10% the slow strategy and 10% the fast strategy. There is more “mixing” between pedestrians, as fast agents overtake slow agents.

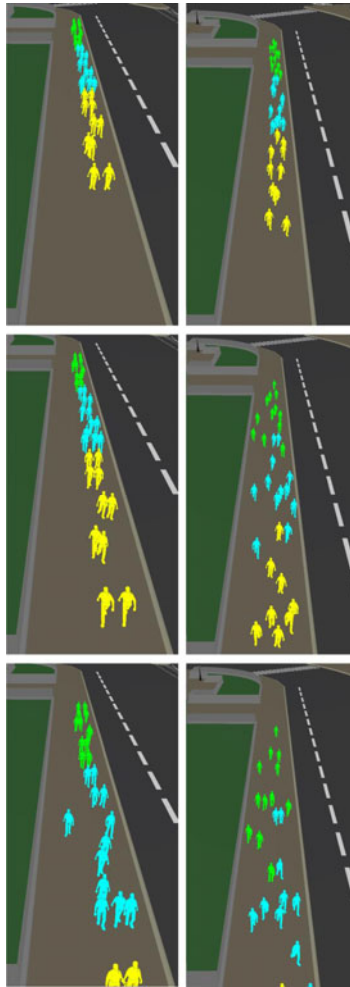


Fig. 19 Left, only classical strategy. Right, heterogeneous strategies

Heterogeneity in agents and strategies enriches the simulation, bringing singular behavior (very slow pedestrians, agents walking on the road, etc.).

4.6 Plausibility

According to our results, pedestrians smoothly avoid static and moving obstacles. They stay in appropriate areas. They have different behaviors and are able to form small groups.

Moreover, the different strategies allow singular perturbations like pedestrians crossing the street when and where they should not or walking out of sidewalks (figure 20).



Fig. 20 Left, a fast pedestrian walks out of sidewalk. Right, agents walks out of crosswalk

Thus, the main Ennis criteria to bring great realism to a simulation seem fulfilled.

To give a general idea of final results, figure 21 compares a real scene with a simulated one.



Fig. 21 Left, a real life photograph from downtown Toulouse. Right, a simulated scene in similar conditions

4.7 Implementation and Performance Considerations

The *direction table* technique is light and intuitive. It can easily be implemented in any model allowing obstacle detection, with no need for any sophisticated environment. Besides, each pedestrian being driven by an individual behavioral engine (to execute its strategy), one can change its behavior or manual control it at any time in the simulation.

On a computer with an Intel Core 2 Duo 2,4GHz and 2 GB RAM our simulation can handle up to fifty agents without any lag at 30 frames per second (including 3D rendering). With 75 agents the framerate is down to 20 fps, we manage to keep an interactive rate (10 fps) with 200 agents.

4.8 Synthesis

The tables 5 and 6 indicate our contribution relatively to discussed related work in section 2.

Table 5 Interactivity means possibilities of manual control or/and on the fly editing of the simulation

	Max number of pedestrians	Real Time Simulation	Interactivity
Shao & Terzopoulos	1200 (without 3D rendering)	Yes (without 3D rendering)	Average
HiDAC	600	Yes	Good
Continuous crowd	10000	No	Low
Aggregate crowds	100000	0 to 10k agents	Low
Crowd patches	3000	Almost	None
Crowds by example	40	No	None
Our work	200	Yes	Good

Table 6 Heterogeneity indicates both heterogeneity in characteristics of pedestrians and behaviors. Singularities indicate if the method allows behavioral singularities such as unexpected stops or a pedestrian that do not respect “the rules”. Realism means plausibility of simulated crowds

	Heterogeneity	Singularities	Realism	Groups
Shao & Terzopoulos	Average	No	Good	No
HiDAC	Average	Yes	Good	No
Continuous crowd	None	No	Low	No
Aggregate crowds	None	No	Low	No
Crowd patches	Average	No	Low	No
Crowds by example	Low	Yes	Good	No
Our work	Good	Yes	Good	Yes

5 Conclusion

In this paper, we have presented our works based on an intuitive approach for pedestrians. The main characteristics of our system are to manage heterogeneous behaviors and cohesion of small groups. In existing methods, these two principles are few used although they substantially increase the realism of the simulations.

Nevertheless, the system can be improved in several ways. The small groups could manage children with own behavior (running, returning back to the parents). The strategies could be continuous, from slow to fast, in order to get more heterogeneous behaviors. The optimization of the direction table is possible by sharing information from close agents moving in the same direction.

Future works focus on the number of simulated characters by using level of details at two levels: graphical and especially behavioral. In our group, we develop a system enabling to manage the LOD of the interactions with the objects [9,17]. This system could help us to simulate more characters and several complex actions.

References

- [1] Abdul Karim, A., Sanza, C.: Learning by implicit imitation in virtual worlds. In: CASA 2010, Workshop on Crowd Simulation, Saint-Malo (2010)
- [2] Daamen, W., Hoogendoorn, S.P.: Experimental research of pedestrian walking behavior. In: Transportation Research Board Annual Meeting, pp. 1–16 (2003)
- [3] Ennis, C., Gerdelan, A., O’Sullivan, C.: Plausible Methods For Populating Virtual Scenes. In: CASA (2010)
- [4] Helbing, D., Molnar, P., Farkas, I.J., Bolay, K.: Self-organizing Pedestrian Movement. Environment and Planning B: Planning and Design (2001)
- [5] Helbing, D., Buzna, L., Johansson, A., Werner, T.: Self-organized pedestrian crowd dynamics: Experiments, simulations, and design solutions. *Transportation Science* 39(1), 1–24 (2005)
- [6] Hughes, R.L.: The Flow of Human Crowds. *Annual Review of Fluid Mechanics* 35 (2003)
- [7] Lerner, A., Chrysanthou, Y., Lischinsky, D.: Crowds by Example. In: Eurographics (2007)
- [8] Narain, R., Golas, A., Curtis, S., Lin, M.C.: Aggregate Dynamics for Dense Crowd Simulation. In: Siggraph Asia (2009)
- [9] Panzoli, D., Peters, C., Dunwell, I., Sanchez, S., Petridis, P., Protopsaltis, A., Scesa, V., de Freitas, S.: A Level of Interaction Framework for Exploratory Learning with Characters in Virtual Environments. In: Plemenos, D., Miaoulis, G. (eds.) *Intelligent Computer Graphics 2010*. SCI, vol. 321, pp. 123–143. Springer, Heidelberg (2010)
- [10] Pelechano, N., Allbeck, J.M., Badler, N.I.: Controlling Individual Agents in High-Density Crowd Simulation. In: ACM Siggraph/Eurographics Symposium on Computer Animation (2007)
- [11] Shao, W., Terzopoulos, D.: Autonomous Pedestrians. ACM Siggraph/Eurographics Symposium on Computer Animation (2005)
- [12] Treuille, A., Cooper, S., Popovic, Z.: Continuum Crowds. In: Siggraph (2006)
- [13] Usher, J.M., Strawderman, L.: Simulation of Emergent Crowd Behavior Using Microsimulation of Individual Pedestrians. *Computers & Industrial Engineering* 59, 736–747 (2010)

- [14] Yersin, B., Maïm, J., Pettré, J., Thalmann, D.: Crowd Patches: Populating Large-Scale Virtual Environments for Real-Time Applications. In: Proceedings of Symposium on Interactive 3D Graphics and Games (2009)
- [15] Weidmann, U.: Transporttechnik der Fußgänger. IVT (90) (1993)
- [16] Weng, W.G., Shen, S.F., Yuan, H.Y., Fan, W.C.: A behavior-based model for pedestrian counter flow. *Physica A: Statistical and Theoretical Physics* 375 2, 668–678 (2007)
- [17] Zertal, S., Djedi, N., Sanza, C., Sanchez, S., Duthen, Y.: Exploitation des niveaux de détails dans la simulation du comportement d’humains virtuels. In: 1st International Conference on Information Systems and Technologies (2011)