

# Real-time Simulation of Crowds Using Voronoi Diagrams

J. Champagne and W. Tang

School of Computing, University of Teesside, United Kingdom

---

## Abstract:

*In this paper, we present a novel approach for real-time simulation of crowds. Our method is to compute generalised 2D Voronoi diagrams on environment maps for the locations of agents in the crowds. The Voronoi diagrams are generated efficiently with graphics hardware by calculating the closest Voronoi site and the distance to that site using polygon scan conversion and the z-buffer depth comparison. Because Voronoi diagrams have unique features of spatial tessellations which give optimised partitions of space for locating the agents especially groups of agents in a virtual environment, agents in the same group are placed within the Voronoi region which encloses the nearest locations to the geometric centre of the group. Each group of agents within its own Voronoi region follows the geometric centre of the region that is moving on the paths of the maps. During the simulation, agents in groups move closely together and avoid collisions with other groups on the way. With carefully designed rules for collision response the algorithm can generate natural-looking group behaviors of large crowds in real-time. Each agent within a group only detects collisions with other agents of the group and with static obstacles in the environment. Efficiency of the simulation is also gained through such multi-level behavioral simulation approach.*

Categories and Subject Descriptors (according to ACM CCS): I.3.1, I.3.7[Computer Graphics] Graphics Processors, Virtual Reality

---

## 1. Introduction

Realistic simulations of human crowds can be used in urban planning, architecture design, visualisations of emergency evacuation, and in digital productions for films with 3D animated large crowds and in computer games with real-time crowd simulation capabilities. Generating such simulations is a challenging topic in computer graphics. The problems are normally two-fold. Simulating large crowds that often consist of a large number of agents is a computationally expensive process. Even with today's compute power, a simulation of a few hundred 3D animated characters would be difficult in real-time and off-line rendering of a large crowd scene is also a time consuming process. Behaviours of the crowds must be realistically modelled to reflect the autonomous nature of agents in crowds. Not only realistic behaviour modelling is a task involving multidisciplinary studies on human behaviours, but also computing the behaviours can become the bottle neck of a simulation process to achieve an acceptable frame rate for large crowds.

There are two groups of previous research work in the area of crowd simulations, one is mainly based on modelling the motion dynamics of crowds using particle systems [HB03] in which the crowd of humans is considered as a large single entity, and the other is more emphasised on the modelling of behaviours of single agents to produce the phenomenon of a crowd as the result of the collective behaviour of the individuals. Bouvier and Cohen [BC95] have implemented a simulation of pedestrians based on Newtonian mechanics in which the individual behaviour of pedestrians remain very simple and sometimes the overall crowd behaviour is unrealistic. Reynolds extended his early work on three steering behaviours of individual agents named as boids (collision separation, velocity cohesion, and directional alignment) [Rey87] to include mathematical computations with path planning, pursuing a moving goal, and obstacle avoidance for individual behaviour or group behaviour simulations [Rey99]. The method proposed by Reynolds is only partially applicable for crowd simulation of people due to the absent of human behaviour modelling. Based on real life observations of

human crowds, Feutrey [Feu00] used his recoded data about crowd behaviours in simulation to predict the trajectories of the pedestrians for collision avoidances. Musse and Thalmann [MT97, TMK00 ] simulated groups of crowds by defining common goals for each group and using a multi-resolution collision detection algorithm with sociological rules to obtain human-like reactions in their crowd simulation system. In terms of improving the simulation efficiencies, Tecchia and Chrysanthou [TC00] proposed a technique for pedestrian simulation with a 2D grid system in which the positions of the pedestrians are managed with square cells of the 2D grid, only one agent is allowed to occupy one cell at a time and collision avoidances are quite simple. No group behaviours are taken into account in the simulation even with their later extended work [LMM03].

In this paper, we present a novel approach for efficient simulation of large crowds of people. We use graphics hardware to process 2D Voronoi diagrams with Voronoi regions representing the geometry locations of the agents especially groups of agent on the map of environments. The Voronoi diagrams offer unique features of space partitions and optimised local distributions of the agent groups, where agents belonging to the same group are located in the Voronoi region of the group so that all the points in the region are the closest points to its Voronoi site i.e. the geometric centre of the group. The Voronoi diagrams are computed by using polygon scan conversion and the z-buffer depth comparison. Our method is capable of dealing with 3D simulations of large crowds in real-time. Collision detections are processed at multi-levels between the groups in the crowds and the agents within a group. Each agent only detects collisions with other agents of its own group and collisions with static obstacles placed in the scene. Hence, the algorithm reduces the order of computation on collision detections. Reactive behaviours between the agents and the groups are modelled by various collision response rules based on collision distances and the density of the crowds.

In the next section, the use of Voronoi diagrams for crowd simulation and the implementation of graphics hardware for constructing the diagrams are described. Section 3 gives a detailed account on our behaviour modelling and collision detection algorithm. Test results are also shown this section. Finally, section 4 of the paper concludes our current research and points out the directions of future work.

## 2. Voronoi Diagrams for Crowd Simulation

The concept of Voronoi diagram has been aware for four centuries since 1664 when Descartes used Voronoi-like diagrams to demonstrate the positional matter in the solar system and its environment. Later the concept was mathematically formulated by Dirichlet in 1850 and Voronoi in 1908 [OBS00]. Voronoi diagram is a method to be used to decompose the space  $S$  into regions around each point  $P_i$  named as *Voronoi site*, such that all the points in the region around point  $P_i$  are closer to  $P_i$  than any other point in the space  $S$  [PS85]. Given a set of Voronoi sites, a Voronoi diagram divides the space into a region around each site point such that the borders of the regions are equidistance from the two nearest sites [OBS00]. The concept can be demonstrated in Figure 1.

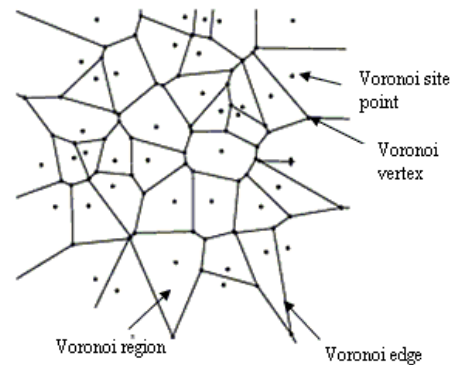
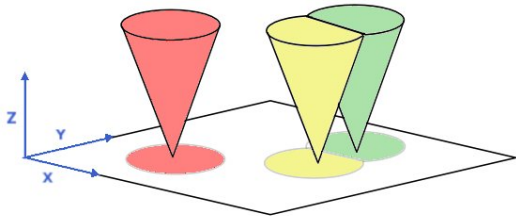


Figure 1: An example of Voronoi diagram: site points and Voronoi regions

From our day to day observations, when groups of people walking together in a public space, they normally seek out locations that allow them to stay close together as a group while trying to maintain their individual private spaces within the group. As Voronoi diagrams have unique features of spatial tessellations and point distributions with all the locations that fall within a Voronoi region are closer to the Voronoi site point than those outside the region, we use Voronoi regions to represent the locations of groups of agents by locating each agent in the region belonging to its group. We allow the agents to only move inside the Voronoi region of the group. In this case, the Voronoi site point is the geometric centre of the group and moves along paths towards the goal of the group, whilst the group as a whole is avoiding collisions with other groups during the navigation.

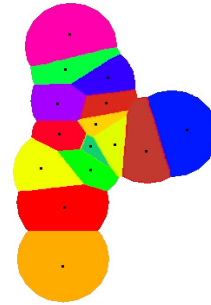
Recently the use of Voronoi diagrams has become popular in computer graphics but is mainly concentrated on non-photorealistic renderings based on the region-based Lloyd method [Llo82] for decorative mosaics and stipple drawing effects [DHvO00, Hau01, HHD03]. A fast method for calculating Voronoi diagrams is introduced by Hoff et al. [HCK\*99], in which graphics hardware is used to compute discrete Voronoi diagrams based on the Fortune's corn sweeping algorithm [O'Ro98]. Readers are referred to [O'Ro98] for a detailed treatment of the algorithm. Our implementation of Voronoi diagrams is similar to that described in [HCK\*99]. We compute generalised Voronoi diagrams using OpenGL functions for graphics hardware acceleration and the diagrams are generated by drawing cones aligned with the z-axis with its apex on the X-Y plane as shown in Figure 2. The Voronoi regions of agent groups are computed by placing the cones at the position of the group with a unique colour assigned to each cone and the cones are drawn with a fan of twenty triangles. The closest Voronoi site and the distance to that site are computed by using polygon scan conversion and the z-buffer depth comparison. The process of computing the projections of the cones is carried out at each simulation step.



**Figure 2 :** *Generating Voronoi diagrams by drawing cones aligned with z-axis*

Figure 3 shows Voronoi diagrams of a crowd consisting of 15 groups with geometric centres marked as black dots. The geometric centre of each Voronoi region is the centre towards which agents in the group are aligned. The centre is found by scanning through the Voronoi diagram buffer to compute the average position of the locations in the region according to the unique colour identification of each group. Due to the way of accessing the hardware buffer for querying the data, it is worth noting that the calculation of geometric centres is identified as the bottle neck of the simulation especially when the size of the environment map is large. The problem could be offset by using a lower

resolution map than the original one to speed up computation. Instead of using the Lloyd's Method [Llo82] to create centroid Voronoi diagrams (CVD), we only use the diagrams without relaxation iterations. This gives the maximum speed for computing the diagrams.

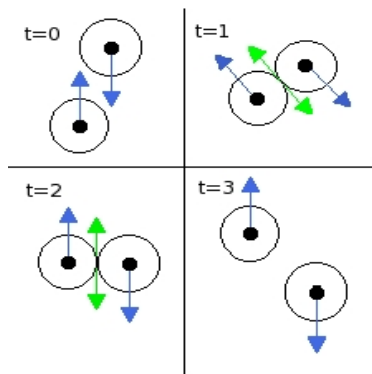


**Figure 3:** *Voronoi diagrams of a crowd consisting of 15 groups with geometric centres*

### 3. Group Collision Detection and Avoidance

The positions of agent groups are calculated with the direction of each group pointing towards its goal. At its geometric centre, a cone shaped field of view of 120 degree is attached to each group and the collision detections are carried out according to three types of distance check: close (less than 4 meters), near (less than 8 meters), and far (less than 15 meters). During the simulation there are two types of possible collisions occurring: front collisions when groups are walking facing each other or following collisions when groups are walking after each other. In an event of possible collisions, each group may react to one of the following behavioural responses based on distance between the groups and the density of the crowds: continue walking but avoiding collisions, waiting for other groups to pass, or taking the same speed as the other groups when there is a possible following collision. The speed of each group may also change according to the density conditions of the crowd. For example, in a densely populated area such as junctions of cross roads, a group would react to a close front collision by waiting for other groups to walk pass in order to avoid colliding with each other. The group may also change direction by swinging away by a small angle to its original trajectory to avoid collisions with other groups. To stay clear of a front collision, the new direction is calculated to be parallel to the tangent at the

intersection point of the bounding circles as shown in Figure 4. This makes the group moving smoothly around other groups and keeps enough space for the agents to move in the group's region. If several front collisions may arise, the average position of all conflicting groups is used to compute the tangent vector. For a following collision the speed of the group may change to the same speed as the other groups or slow down to avoid collisions. The density of the crowd also affects the acceleration attribute of the group, for example, in the case of a high density area where there is a very high potential of collisions, the acceleration of the group will be decreased. If the density is too high, the group may stop and wait for the other groups to pass.



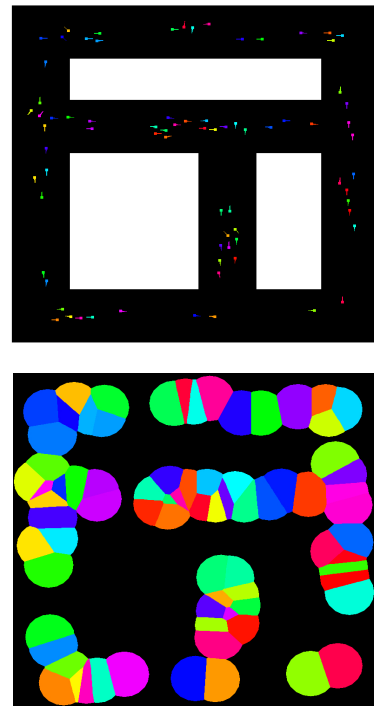
**Figure 4:** *Group frontal collision response at successive steps*

In the simulation, each agent in a group remains inside the group's region and maintains its velocity and acceleration. The agent is attached with a bounding circle for checking possible collisions with the agents of the same group and it also detects collisions with static obstacles placed on the path that the group is currently walking on. The distance between the positions of the agents and the obstacles are estimated and if a collision is detected, the direction of the agent may change to avoid the collision. The distance between the agent and the Voronoi site is also checked to prevent it going off the group's region.

#### 4. Results and Discussion

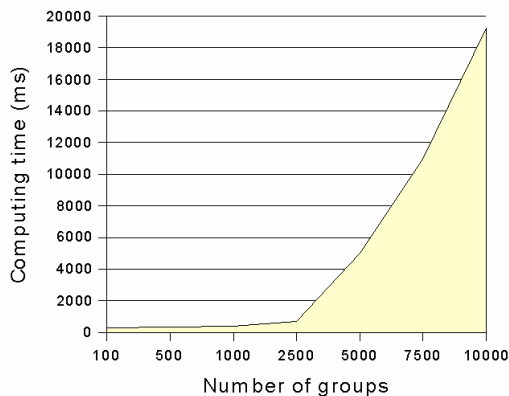
We implemented our algorithm into a crowd simulation system that works on both Windows and Linux. Simulation tests are carried out on a PC Athlon 1.4 GHz processor with an OpenGL compatible video card NVIDIA GeForce 2MX. The efficiency of the

simulation is tested on the same environment map with both 2D and 3D rendering. A simple user interface is designed to allow users of the system to select the number of groups wish to be simulated and for each group of the simulation the number of agents in group is randomly set between 1~5. Figure 5 shows a 2D rendering of a simulation of 80 groups with arrows indicating the walking directions of the groups. The rendered corresponding Voronoi diagrams are also shown in Figure 5 with shaded regions indicating the Voronoi regions of the groups. The most costly computation of the simulations are the processes of copying the content of z-buffer after the generation of the diagrams and computing the geometric centres of the Voronoi regions. As the processes are repeated at every update of the simulation, the size of the environment map is one of the factors that influence the speed of the simulation. The larger is the size of the map, the slower the simulation would be. The speed of the simulation process decreases proportionally as the size of the map increases.



**Figure 5:** *A square map with 80 groups of agents and walking directions (top image) and the corresponding Voronoi diagram rendering (bottom image)*

To test the effectiveness of the algorithm for simulating large crowds, we have measured the number of groups on a 2000 pixel by 2000 pixel square map with the time taken for each simulation update. As shown in Figure 6, without any rendering process taking place, the system can simulate at least 2500 groups of agents which consist of between 5000 to 10000 agents without rapidly consuming compute resources. The real-time simulation results are also due to the graphics hardware acceleration process which is an integrated part of our simulation algorithm.



**Figure 6:** Time taken to compute different number of groups on a fixed map size without any rendering process



**Figure 7:** Agents in groups walking closely together to avoid collisions with other agents and groups

However, in 3D rendering with animated characters the system can handle up to 200 groups in real-time

since 3D graphics process takes up a very high percentage of compute resources. In terms of behaviour modelling, the simulation algorithm produces realistic crowd movements, as shown in Figure 8, agents in groups move closely together avoiding other agents or groups in quite convincing ways.

It is worth noting that the simulation system works well with group behaviour simulations but also highly adaptable with simulating individual agents who do not belong to any groups. However, because one of the big advantages of using Voronoi diagrams is the spatial tessellations for local regions around the site points, the algorithm is particularly effective for simulating group behaviours by taking the full advantages of local space partitions and point distributions as well as the graphics hardware acceleration process of the system implementation. Figure 8 shows a screen capture of a 3D rendering of 80 groups. To improve the behaviour realism of the simulation, the algorithm can be extended to include complex collision responses by adding more behavioural rules that are based on real life observations and psychology studies [Goff71, Wol73].

## 5. Conclusions

We have presented a novel approach for efficient simulation of large crowds of people in real-time based on generating Voronoi diagrams for 2D space partitions for the locations of the agents in the crowds. The diagrams are constructed by using graphics hardware so that the method can efficiently attain the locations of the agent groups on 2D maps. In order to achieve realistic behaviour simulations a set of rules are designed for collision responses based on collision distance and density of the crowds. In addition to the behavioural rules of a multi-level collision detection scheme is also implemented to gain more computational efficiency. The system is capable of simulating 10,000 agents with realistic behaviours.

In computer graphics many recent non-photorealistic renderings make the use of Voronoi diagrams to generate sample point distributions for simulating tipping and decorative mosaics. As far as we know, our approach is the first to use Voronoi diagrams for crowd simulations. Instead of using the Lloyd's Method [Llo82] to create centroid Voronoi diagrams (CVD), we only use the diagrams without relaxation iterations. This gives the maximum speed for computing the diagrams but the space partitions may be less optimised than Lloyd's method would have done. We think the



**Figure 9:** A screen capture of a simulation involving 80 groups with 160 – 400 animated characters

advantage of not using CVD has outweighed the disadvantages of the less optimised distribution in terms of system performance for crowd simulations. The drawback is that without the iterations some groups may be located in very small Voronoi regions when they are in highly dense areas such as at junctions of cross roads. The problem can be solved by increasing the size of the Voronoi region using a larger cone in the situations where the density of the crowds is too high. Our method is not only applicable to group behaviour simulations but also useful in more general crowd simulations in which a mixture of groups and single agents are desirable. In such simulations while using hardware computations for groups of agents, software implementation can be used for single agents in order to maximize the efficiency of the simulation. Again the current system would benefit from more accurate modelling of behaviour and collision responses and to be extended to include more complex and realistic behavioural rules for agents to accomplish a higher level of behavioural realism.

#### References:

- [BC95] BOUVIER, E., COHEN, E.: Simulation of human flow with particle systems. *Proceeding of Simulators International XII*, Phoenix, AZ, (April 9-13 1995).
- [DHvO00] DEUSSEN, O., HILLER, S., van OVERVELD, K.: Floating points: a method for computing stipple drawings. *Proceedings of EUROGRAPHICS 2000 Computer Graphics Forum*, 19(4):40-51.
- [Feu02] FEUTREY, F.: Simulating the collision avoidance of pedestrians, Masters thesis, University of Tokyo, February, 2000
- [Gof71] GOFFMAN, E.: Relations in Public. Microstudies of the Public Order. *London: Allen Lane*. (1971)
- [Hau01] HAUSNER, A.: Simulating decorative mosaics. *Proceedings of SIGGRAPH 2001*, PP. 573-578
- [HB03] HOOGENDOOM, S, BOVY, P. H.L.: Simulation of pedestrian flows by optimal control

- and differential games *Optimal Control Applications and Methods*, vol. 24 (3) ,( Published Online: 13 Jun 2003) pp. 153 – 172.
- [HCK\*99] HOFF, K., CULVER, T., KEYSER, J., KIN, M., MANOCHA, D.,: Fast computation of generalized voronoi diagrams using graphics hardware. *Proceedings of SIGGRAPH 1999*.
- [HHD03] HILLER, S., HELLWING, H., DEUSSEN, O.,: Beyond stippling- methods for distributing objects on the plane. *Proceedings of EUROGRAPHICS 2003 Computer Graphics Forum*, 22(3):515-522
- [Llo82] Lloyd, S.,: Least square quantization in PCM. *IEEE Transactions on Information Theory* (1982), vol. 28 pp.129-137
- [LMM03] LOSCOS, M., MARCHAL, D., MEYER, A.,: Intutive crowd behaviour in dense urban environments using local laws, *Proceedings of Theory and Practice of Computer Graphics (June 2003)* Birmingham, UK. pp.122-129
- [MT97] MUSSE, S. R., THALMANN, D. : A model of human crowd behavior: Group inter-relation and collision detection analysis. *Proceedings of workshop of Computer Animation and Simulation (1997)*, Eurographics workshop, Budapest, Hungary, Springer Verlag, Wien, (1997), pp.39-51.
- [OBS00] Okabe, A.; Boots, B.; Sugihara, K. : *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, 2nd ed. New York: Wiley, 2000.
- [O’Ro98] Joseph O’Rourke, J.: *Computational Geometry in C* . 2nd ed. New York 1998
- [PS85] Preparata, F.R. and Shamos, M.I.: *Computational geometry: an introduction*. New York: Springer-Verlag, 1985.
- [Rey87] REYNOLDS C. W.: Flocks, herds and schools: A distributed behavioral model. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques* (1987), ACM Press, pp. 25–34.
- [Rey99] REYNOLDS C. W.: Steering behaviors for autonomous characters. In *Game Developers Conference* (1999).
- [TMK00] THALMANN D., MUSSE S. R., KALLMANN M.: From individual human agents to crowds. In *Informatik/ Informatique* (2000).
- [TC00] TECCHIA, F., CHRYSANTHOU, Y.: Real-time visualisation of density populated urban environments: a simple and fast algorithm for collision detection, *Eurographics UK Chapter Conference*, April 2000.
- [Wol73] WOLFF, M.: Notes on the behaviour of pedestrians’. In A. Birenbaum and E. Sagarin (eds.) *People and Places: The Sociology of the Familiar*. New York: , (1973), 35-48.