2
OVERVIEW OF CROWD SIMULATION IN COMPUTER GRAPHICS

Mohamed `Adi Mohamed Azahar
Daut Daman

INTRODUCTION

High-powered technology use computer graphics in education, entertainment, games, simulation, and virtual heritage applications has led it to become an important area of research. In simulation, according to Tecchia et al. (2002), it is important to create an interactive, complex, and realistic virtual world so that the user can have an immersive experience during navigation through the world. As the size and complexity of the environments in the virtual world increased, it becomes more necessary to populate them with peoples, and this is the reason why rendering the crowd in real-time is very crucial.

Generally, crowd simulation consists of three important areas. They are realism of behavioural (Thompson and Marchant 1995), high-quality visualization (Dobbyn et al. 2005) and convergence of both areas. Realism of behavioural is mainly used for simple 2D visualizations because most of the attentions are concentrated on simulating the behaviours of the group. High-quality visualization is regularly used for movie productions and computer games. It gives intention on producing more convincing visual rather than realism of behaviours. The convergences of both areas are mainly used for application like training systems. In order to make the training system more effective, the element of valid replication of the behaviours and high-quality visualization is added.

OVERVIEW OF CROWD SIMULATION

Real-time crowd simulation is a process of simulating the movement of a large numbers of animated characters or agents in the real-time virtual environment. Crowd movement in certain cases requires the agents to coordinate among themselves, follow after one another, walking in line or dispersing using different directions. All of these actions will contribute to the final collective behaviours of the crowd that must be achieved in real-time. Unlike non-real-time simulations which are able to know the full run of the simulated scenario, real-time simulations have to react to the situation as it unfolds in the moment. As stated by Thalmann and Musse (2007), real-time rendering of a large number of 3D characters is also a challenge, because it can exhaust the system resources quickly even for a powerful system.

Figure 2.1 Previous works in crowd simulation

Figure 2.1 shows a timeline of the previous works that have been done in crowd simulation fields. A behavioral animation of human crowd was based on foundations of group simulations of much more simple entities, notably flocks of birds (Reynolds 1987) and schools of fish (Tu and Terzopoulos 1994). Earliest procedural animation of flocks of virtual birds called Eurhythmy was developed from concept that was presented at The Electronic Theater at SIGGRAPH in 1985 and the final version was presented at Ars Electronica in 1989 (Amkraut et al. 1985). The flock motion was achieved by a global vector force field guiding a flow of flocks. In his early work, Reynolds describes a distributed behavioral model for simulating aggregate motion of a flock of birds. The idea was that a complex behavior of a group of actors can be obtained by simple local rules for members of the group. The flock was simulated as a complex particle system, using the simulated birds, called boids, as the particles. Reynolds, later extended his work by including various steering behaviors as goal seeking, obstacle avoidance, path following, or fleeing (Reynolds 1999), and introduced a simple finite-state machines behaviors controller and spatial queries optimizations for real-time interaction with groups of characters (Reynolds 2000).

More recent work was studies on group modeling based on hierarchies. Niederberger & Gross (2003) proposed architecture of hierarchical and heterogeneous agents for real-time applications. Behaviors were defined through specialization of existing behaviors types and depend on multiple inheritances to create new types. An approach that has become more common now was geometry baking. By taking snapshots of vertex positions and normals, complete mesh descriptions were stored for each frame of animation as in the work of Ulicny et al. (2004). Another approach was through dynamic meshes, which was using systems of caches to reuse skeletal updates. A hybrid of baked meshes and dynamics meshes was found in Yersin et al. (2005) where the graphics programming unit (GPU) was used to its fullest.

A real-time crowd model based on continuum dynamics has been proposed by Treuille et al. (2006). In their model, a dynamic potential field integrates global navigation with moving obstacles, efficiently solving for the motion of large crowd without the need for explicit collision avoidance. In addition, Mao et al. presented an effective and ready to use framework for real-time visualization of large-scale virtual crowd. Script that describes motion state and position information was used as an input. It provides convenient interface and makes the framework universal to almost all applications of crowd simulations (Mao et al. 2007).

CROWD RENDERING

The complicated part when dealing with thousands of characters is the quantity of information that needs to be processed. Simple approaches, where virtual humans are processed one after another without specific order will produce high computational cost for both the central processing unit (CPU) and graphics processing unit (GPU). This is the reason why data that flows through the same path need to be grouped for an efficient use of the available computing power. Therefore, for the best simulation result, characters capable of facial and hand animation are simulated in the area near to the camera to improve believability, while for farther area, less expensive representations are used. Concerning efficiency of storage and data management, database must be used to store all the virtual human-related data.

Crowd Rendering Issues

Figure 2.2 shows certain problems that arise when rendering crowd. For instance, collision avoidance problems for a group of peoples in the same place required different strategies in comparison with the methods used to avoid collision between individuals. Moreover, motion planning used in a group that walks together requires more information compared to individual motion planning. The trajectories computed for agents in the same group that walk together with similar speeds must be different even when they share the same environment and goals. In addition, other levels of behaviours can exist when treating crowd in this hierarchical structure. The group behaviours can be used to specify the way a group moves, behaves, and acts in order to fit different group structures (flocking, following, repulsion, attraction, etc). Individual abilities are also required in order to improve the autonomy and intelligence of crowd.

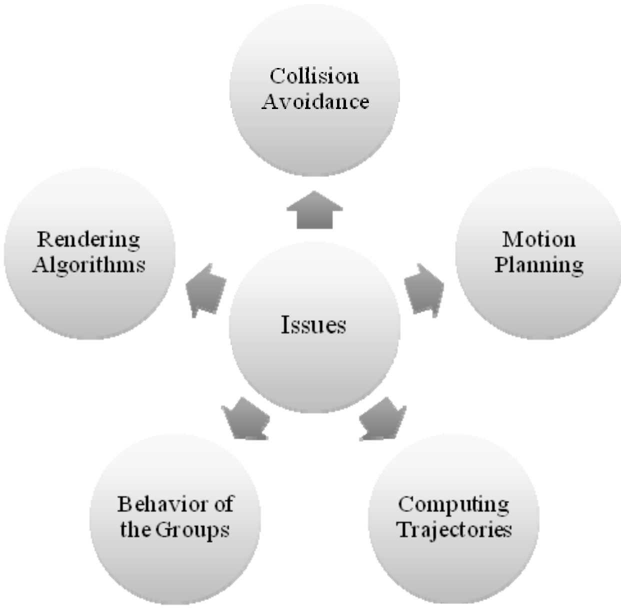

Figure 2.2 Crowd rendering issues

However, in order to render thousands of individuals, these complex behaviours cannot be provided individually due to the hardware constraints and computational time rates. Another problem relates on how to improve the intelligence and provide autonomy to scalable crowd, in real-time systems. Furthermore, the simulation of large crowd in real-time requires many instances of similar characters. That is why an algorithm to allow for each individual in the crowd to be unique is needed.

There are several techniques used to speed up rendering process in crowd simulation. Billboarded impostors are one of the methods used to speed up crowd rendering. Impostors are partially transparent textured polygons that contain a snapshot of a full 3D character and are always facing the camera. Aubel et al. (2000) proposed dynamically generated impostors to render animated virtual humans. A different possibility for a fast crowd display is to use point-based rendering techniques. Wand and Strasser (2002) presented a multiresolution rendering approach which unifies image based and polygonal rendering. An approach that has been getting new life is the geometry baking. By taking snapshots of vertex positions and normal, complete mesh descriptions are stored for each frame of animation as in the work of Ulicny et al. (2004). Another approach to crowd rendering using geometry is through dynamic meshes as presented in the work of de Heras et al. (2005), where dynamic meshes use systems of caches to reuse skeletal updates which are typically expensive.

Types of Crowd Rendering Methods

In recent years, researchers have applied several approaches, either separately or in combination, to develop crowd simulation in various graphics applications. In this section, five of the crowd simulation approaches will be reviewed as shown in the Figure 2.3.

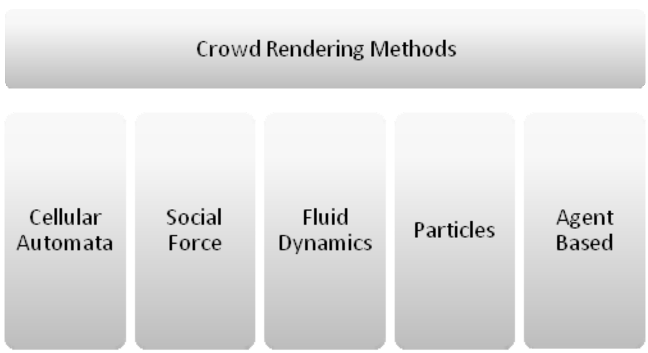

Figure 2.3 Crowd rendering methods

Cellular Automata

Cellular automata were proposed by Beuchat and Haenni (2000) and Georgoudas et al. (2007). Cellular automata are discrete dynamic systems consisting of a regular grid of cells. Cellular automata evolve at each discrete time step, with the value of the variable at one cell determined by the values of variables at the neighboring cells. The variables at each cell are simultaneously updated based on the values of the variables in their neighborhood at the previous time step and according to a set of local rules (Wolfram 1983). At present, cellular automata have been successfully applied to various complex systems, including traffic models and biological fields. In recent years, cellular automata models have been developed to study crowd evacuation under various situations. These models can be classified into two categories. The first one is based on the interactions between environments and pedestrians. The other is based on the interaction among pedestrians.

Social Force

Helbing and Molnar (1995) introduced a 'Social force model for pedestrian dynamics'. They suggested the motion of pedestrians can be described as if they are subject to social forces which are a measure of the internal motivation of individuals to perform certain actions or movements. They described three essential forces:

Acceleration - the velocity of the pedestrian varies over time, as it attempts to reach its optimum speed and as it avoids obstacles.

Repulsion - there is a repulsive force from other pedestrians and from obstacles and edges.

Attraction - pedestrians are sometimes attracted by other people or by other objects.

By putting these three forces together, Helbing and Molnar (1995) have produced an equation for a pedestrian's 'total motivation' and combining this with a term to allow for fluctuations in behaviour, produces the 'social force model'. They described computer models based on the equations, which have been successful in demonstrating various observed phenomena, for example lane formation. In Helbing et al. (2000), the social force model is applied to the simulation of building escape panic, with impressive results.

Fluid Dynamics

Helbing et al. (2002) have described that at medium and high densities, the motion of pedestrian crowds shows some striking analogies with the motion of fluids. For example, the footprints of pedestrians in snow look similar to streamlines of fluids or, again, the streams of pedestrians through standing crowds are analogous to riverbeds. Fluid-dynamic models describe how density and velocity change over time with the use of partial differential equations. Colombo and Rosini (2005) presented a continuum model for pedestrian flow to describe typical features of this kind of flow such as some effects of panic. In particular, this model describes the possible over-compressions in a crowd and the fall in the outflow through a door of a panicking crowd jam. They considered the situation where a group of people needs to leave a corridor through a door. If the maximal outflow allowed by the door is low, then the transition to panic in the crowd approaching the door may likely cause a dramatic reduction in the actual outflow, decreasing the outflow even more.

Particles

The majority of pedestrian simulations take this particulate approach, sometimes called the atomic approach. Early influential work was that of Craig Reynolds (1987) who worked on simulations of flocks of birds, herds of land animals and schools of fish. Each particle, or boid, was implemented as an individual actor which navigates according to its own perception of the environment, the simulated laws of physics, and a simple set of behavioural patterns. Reynolds (1999) has extended the concepts to the general idea of 'autonomous characters', with an emphasis on animation and games applications. Bouvier et al. (1997) described a generic particle model and applied it to both the problem of pedestrian simulation and to the apparently distinct problem of airbag deployment. They presented software allowing the statistical simulation of the dynamic behaviour of a generic particle system. In their system, the particle system was defined in terms of the:

i. Particle types - mass, lifetime, diffusion properties, charge, drag, interactions with surfaces, visualisation parameters.
ii. Particle sources or generators - size, geometry, rate and direction of emission.
iii. 3D geometry, including obstacles.
iv. Evolution of particles within the system.

Agent Based

Agent based are computational models (Goldstone and Janssen 2005) that build social structures from the ''bottom-up'', by simulating individuals with virtual agents, and creating emergent organizations out of the operation of rules that govern interactions among agents. Bonabeau (2002) supported the following point of view: in agent terms, collective panic behavior is an emergent phenomenon that results from relatively complex individual-level behavior and interactions among individuals; the agent based seems ideally suited to provide valuable insights into the mechanisms and preconditions for panic and jamming by in-coordination. In the last few years, the agent based technique has been used to study crowd evacuation in various situations. Agents based are generally more computationally expensive than cellular automata, social force, fluid-dynamic or particles models. However, their ability to allow each pedestrian to have unique behaviors makes it much easier to model heterogeneous humans, which are groups that contain individual with difference characteristic.

CONCLUSION

Crowd simulation brings great challenges into virtual reality application system either involving a small number of interacting characters or non-real-time system. In order to have a real and immersive application, virtual humans composing a crowd should simulate real crowd as similar as possible. In this chapter, we have discussed the overview of crowd simulation, crowd rendering issues, and a few types of common crowd rendering methods. The main goals of this chapter are to give general ideas of what is crowd simulation and methods involved in it.

REFERENCES

Amkraut S., Girard M. and Karl G., 1985, "Motion studies for a work in progress entitled Eurnythmy", SIGGRAPH Video Review, 21. (second item, time code 3:58 to 7:35)

Aubel A., Boulic R. and Thalmann D., 2000, "Real-time display of virtual humans: Levels of detail and impostors." IEEE Transactions on Circuits and Systems for Video Technology, 10 (2), pp. 207–217

Beuchat J. L. and Haenni. J. O., 2000, "Von Neumann's 29-State Cellular Automaton: A Hardware Implementation." In: IEEE Transactions on Education, Vol. 43, No. 3, AUGUST

Bonabeau E.,2002, "Agent-based modeling: methods and techniques for simulating human systems." In: Proceedings of the National Academy of Sciences of the USA(PNAS) 99(Suppl. 3): pp. 7280–7287

Bouvier E., Cohen E. and Najman L., 1997, “From crowd simulation to airbag deployment: Particle systems, a new paradigm of simulation.” Journal of Electrical Imaging, 6 (1), pp. 94–107

Colombo R.M. and Rosini M.D., 2005, “Pedestrian flows and non-classical shocks.” In: Mathematical Methods in the Applied Sciences 28: pp.1553–1567

De Heras P., Schertenleib S., Ma¨ım J. and Thalmann D., 2005, “Realtime shader rendering for crowd in virtual heritage.” In Proc. 6th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST 05)

Dobbyn S., Hamill J., O’Conor K. and O’Sullivan C., 2005, “Geopostors: A real-time geometry/impostor crowd rendering system.” In SI3D ’05: Proceedings of the 2005 Symposium on Interactive 3D Graphics and Games (New York, NY, USA), ACM Press, pp. 95–102

Georgoudas I.G., Sirakoulis G.Ch., and Andreadis. I.Th., 2007, “An Intelligent Cellular Automaton Model for Crowd Evacuation in Fire Spreading Conditions.” In: 19th IEEE International Conference on Tools with Artificial Intelligence.

Goldstone RL and Janssen MA., 2005, “Computational models of collective behavior.” In: Trends in Cognitive Sciences 9(9): pp. 424–430

Helbing D, Farkas I and Vicsek T., 2000, “Simulating dynamical features of escape panic.” In: Nature 407: pp. 487–490

Helbing D, Farkas IJ, Molnar P. and Vicsek T., 2002, “Simulation of pedestrian crowds in normal and evacuation situations.” In: Schreckenberg M, Sharma SD, editors. Pedestrian and evacuation dynamics. Berlin: Springer; pp. 21–58

Helbing D. and Molnar P., 1995, “Social force model for pedestrian dynamics.” In: Physical Review E 51(5):4282–6

Mao, T., Shu, B., Xu, W., Xia, S. and Wang, Z., 2007, “CrowdViewer: from simple script to large-scale virtual crowd.” In Proc. of the 2007 ACM Symposium on Virtual Reality Software and Technology, pp. 113-116.

Niederberger C. and Gross M., 2003, “Hierarchical and heterogeneous reactive agents for real-time applications.” Computer Graphics Forum 22, 3 (Proc. Eurographics’03)

Reynolds C. W., 1987, “Flocks, herds, and schools: A distributed behavioural model.” In Computer Graphics (ACM SIGGRAPH ’87 Conference Proceedings) (Anaheim, CA, USA), Vol. 21, ACM, pp. 25–34

Reynolds C. W., 1999, “Steering behaviours for autonomous characters.” In Proceedings of Game Developers Conference 1999, pp. 763–782

Reynolds C. W., 2000, “Interaction with groups of autonomous characters.” In Proc. Game Developers Conference ’00, pp. 449–460

Tecchia, F., Loscos, C. and Chrysanthou, Y.,2002, “Visualizing Crowd in Real-Time.” Computer Graphics Forums.

Thalmann, D. and Musse, S. R., 2007, “Crowd Simulation.” Springer-Verlag, London

Thompson P.and Marchant E., 1995, “Testing and application of the computer model *simulex*.” Fire Safety Journal 24, 2, pp. 149–166

Treuille A., Cooper S. and Popovic Z., 2006, “Continuum crowd.” ACM Transactions on Graphics 25 (3), pp.1160–1168

Tu X. and Terzopoulos D., 1994, “Artificial fishes: Physics, locomotion, perception, behaviour.” In Computer Graphics (ACM SIGGRAPH ’94 Conference Proceedings) (Orlando, FL, USA), Vol. 28, ACM, pp. 43–50

Ulicny B., de Heras Ciechomski P. and Thalmann D., 2004, “Crowdbrush: Interactive authoring of real-time crowd scenes.” In Proc. ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA’04), pp. 243–252.

Von Neumann, J. and A. W. Burks. Theory of self-reproducing automata. Urbana,, University of Illinois Press (1966)

Wand M. and Strasser W., 2002, “Multi-resolution rendering of complex animated scenes.” Computer Graphics Forum 21, 3 (Proc. Eurographics’02)

Wolfram S., 1983, "Statistical mechanics of cellular automata." In: Reviews of Modern Physics 55: pp. 601–644

Yersin B., Ma¨ım J., de Heras Ciechomski P., Schertenleib S. and Thalmann D., 2005, "Steering a virtual crowd based on a semantically augmented navigation graph." In First International Workshop on Crowd Simulation (VCRO*WD'05).*