Crowd Simulation

based on emergent behaviours

Carlos Pérez López

August, 2013

Master of Science,

Computer Animation and Visual Effects

Bournemouth University

Contents

Abstract

Abstract is the last thing you write, it should be concise. Spend time to write it correctly, it has to be perfect. Google how to write an abstract, it is not just summary!

Acknowledgements

To my parents, who paid the bills.

Chapter 1

Introduction

Currently, the presence of crowds in visual pieces, let they be films or video games, has acquired a lot of relevance. Scenes with a crowded train station, big streets full of pedestrians, tons of animals in a flock, troops of robots, hordes of zombies or armies of warriors can be found very often in modern films and games.

In the past, not many films could afford to employ this kind of resources in their scenes, since the only way of having more characters was actually including more characters in the cast. The use of extras highly increases the cost of a production, besides the requirement of a strict coordination and proper training in some situations. One of the reasons of the boom of big masses in films was the arrival of new technologies in computer graphics and artificial intelligence. Alongside with this, the generation and simulation of virtual crowds became a reality, saving money and time and making the use of crowds affordable to big and small studios.

Other of the reasons of the popularity of big groups of individuals is the strength they give to a shot. It is not only the matter of fact that crowds are visually stunning and their magnificence captures the attention of the audience, but also they are powerful tools to tell and enrich a story. At this point of time, it could be hard if not impossible imagine the saga of the Lord of the Rings without those epic battles of tens of thousands of warriors, or imagine the breathtaking apocalyptic scenes without those milliards of zombies wandering around.

Closely tied to the idea of crowd simulation, the concept of behaviour appears. This is what gives live to the crowd and make the spectator perceive the group nature of it. No matter the shape of the individuals, an army of brave warriors in a battlefield will find enemies and will show aggressive movements; on the other hand, on a ballroom gentlemen and ladies will dance graciously.

(a) *Crowd in a train station*

(b) *Troops of droids (Star Wars)*

(c) *Horde of zombies (World War Z)*

(d) *A Battlefield (Lord of the Rings))*

(e) *A Ballroom*

(f) *One vs Many (Matrix)*

Figure 1.1: *Crowd scenes*

The aim of this project is to propose an approach which gives the flexibility to simulate any sort of crowd needed for a scene. Taking base on how the real world works, this method is based on the principle than the group behaviour is determined by the specific behaviours of every individual. And here is where one of the main ideas for this project appears: Emergent Behaviour.

Leonard Kleinrock, professor of computer science in UCLA, states that emergent behaviour is unanticipated behaviour shown by a system (Kleinrock, 2011). Once a system is designed and defined by certain rules or mathematical equations, it may configure itself in a way that could not be anticipated. The interaction of a large number of simple individual things is very hard to predict; the complexity does not reside in the individuals, but on the way they are interconnected and they interact to each other. Professor Kleinrock proposes this example: we might know how a bunch of children behave when they are alone, but once you put them together in a group, you will observe behaviours that will surprise you.

Subsequently, how a real crowd behaves is something hard to predict, and how realistic it is depends directly on how realistic each individual is.

This thesis is structured as follows:

- **Chapter 2: Related Work.** It explains the previous approaches in this field, the similarities and differences to the proposed method, as well as the advantages and drawbacks they present.
- **Chapter 3: Technical Background.** Mathematical and physical concepts, algorithms and optimized routines, and object orientation features for the build of the crowd engine or how scripting may enhance the flexibility and scalability of the system.
- **Chapter 4: Agent Based Model.** This is where the current approach is explained into details. The main aims of the method are discussed, how all of this was faced, designed, implemented and tested, and a pipeline where this methodology might fit in a real production situation is presented.
- **Chapter 5: Crowd Engine.**
- **Chapter 6: Applications and results.** Some results of different tests will be shown in this chapter. A set of individual behaviours will be presented and the emergent behaviour observed will be explained and discussed.

- **Chapter 7: Application design and implementation.**
- **Chapter 8: Conclusion.** A final concluding chapter will summarize the whole approach, mentioning the main advantages and drawbacks, as well as presenting possible lines for future work.

Chapter 2

Related work

Generating crowds is a problem that has frequently been faced in the field of computer graphics and artificial intelligence. Numerous solutions have been proposed, following different approaches and applying different ideas and concepts. Crowd motions can be created by planning or simulation, and even some researches propose hybrid approaches.

2.1 Motion Planning for Crowd

Opposite to the philosophy of this thesis, a substantial sum of research establishes that a crowd is not only a group of individuals and involves problems that should be handled at the group level, making use of pre-planned techniques. Frequently, strategies such as virtual force fields, navigation fields, motion planning, navigation graphs, etc. are employed to drive the movement of the whole mass, forgetting by all means the individual nature of the crowd. S. Musse et al. state that a motion planning for a group walking together requires more information than an individual motion planning (Musse & Thalmann, 2001). In addition, this research claims for the need of model behaviours at the level of groups and crowds in order to acquire the beauty of synchronization, homogeneity and unity.

Figure 2.1: *Motion Planning for Crowd using Navigation Fields (Patil et al., 2011)*

Extending the control vertically, in (Musse et al., 2005) a hierarchical model for real time simulation of virtual human crowds is proposed allowing different control features at levels of crowd, groups or individuals. This is an example of hybrid approach where it is possible to increase the complexity of crowd-group-individual behaviours according to the problem to be simulated.

By means of these techniques, it was visually proven that very convincing and realistic results can be produced. Nevertheless, the identity and the decision capacity of each of the individuals is partially if not completely lost.

2.2 Crowd Motion Simulation

Purely simulation strategies, which is the case of this thesis, discard any pre-planned decision and the final result is entirely based on the global consequences of local interactions of members of the population. This is known as Agent-Based Model or Individual-Based Model. This will be detailed in Chapter **??** and for further information check (Reynolds, 1999).

The main idea this thesis is settled on is the simple principle stated by C. Reynolds, the pioneer of flocking behaviours, which claimed that very simple rules can arise emergent behaviours without involving any central coordination. Notice again the concept of emerge.

One of the earliest works in group behaviours was Craig Reynolds' flocking algorithm which was a distributed behaviour model for flocks, herds and schools (Reynolds, 1987). The individuals of his flocking system are called "Boids" and are subjected to three simple rules: cohesion, alignment and separation.

This classic research not only presents a flocking algorithm, but also marks a turning point proposing an individual virtual force model as the way to affect how each agent move. The approach presented in this thesis has adopted that model.

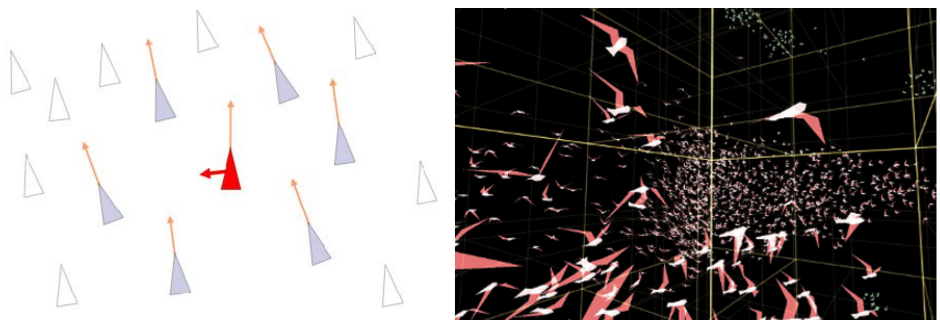


Figure 2.2: *Flocking System following the Virtual Force Model proposed by C. W. Reynolds (Reynolds, 1987)*

Starting from this robust and solid idea, tons of research paths can be taken in order to acquire and compare different approaches for crowd simulation. For instance, C. Wang & T. Li suggest an evolving crowd motion simulation (Wang & Li, 2006). In that research, the use of genetic algorithms is proposed to generate optimal virtual forces according to the given environment and desired movement behaviour.

Although Craig Reynolds presents a base-approach which models very accurately the way crowds behave in the real world, the main disadvantage is that configuring the forces in order to generate desired motion behaviours remains empirical. And again, we are witnesses of a characteristic inherent to emergent behaviours.

2.3 MASSIVE Software

MASSIVE stands for Multiple Agent Simulation System in Virtual Environment and is a software developed by Stephen Regelous in Weta Digital, as a request from

Peter Jackson to recreate those epic battle scenes that Tolkien described in the books of the Lord of the Rings. Massive has contributed to the creation of many awarding visual effects, particularly the sequences; and due to this, it has been developed into a complete product and has been licensed by many other visual effects houses.

Figure 2.3: *Award winning battle scene from 'The Lord of the Rings: Return of the King'*

Massive introduces a very interesting model approach which consists in treating each agent as a combination of a body and a brain. The body defines the physical characteristic of the agent and the brain is a fuzzy logic network which controls the actions of the agent such as following an arbitrary terrain, avoiding obstacles or interacting with other agents. This thesis has adopted this natural design combined with the flexibility that scripting languages provide.

Each action is associated to a pre-recorded animation, rather obtained from motion captured session or hand-animated, and will be blended between them in order to achieve the movement of the character. Apart from Artificial Intelligence features, it includes other abilities such as Rigid Body Dynamics (RBD), cloth simulation or GPU rendering.

Chapter 3

Technical Background

The background needed for doing research on engineering or any technical discipline is extremely important, and frequently is what marks the difference between a solid, consistent and robust study from a weak one. Many mathematical and physical concepts (specially physical) are critical to understand analytically the ideas, reasoning and logical models presented in this survey.

This chapter intends to describe formally the main ground concepts where this thesis is settled on. Although this may only be a brief glimpse of all the ideas applied directly or indirectly, and a much finer conceptual background might be developed, this should be enough for following the approach.

The bibliography resource consulted to write this section was the chapter A Maths and Physics Primer, in the book Programming Game AI by Example (Buckland, 2005).

3.1 Physics

In any research that involves Artificial Intelligence (AI) and modelling the real world, physical rules will acquire a fundamental role, particularly the ones concerned with motion. Next, the key concepts to understand the virtual force model will be presented.

- **Time.** This is a concept that everybody has in mind. Physically speaking, time is a dimension in which events can be ordered from the past through the present into the future. It is a continuous scalar quantity with no direction measured in seconds. Time in computer simulations and computer games might be measured in seconds, like in the real world, or in *virtual seconds* or *ticks*. This will be discussed more in depth in Section **??**: Application Design and Implementation.

- **Mass.** It is a scalar quantity measured in kilograms, and it is the measure of an amount of something. This property is directly linked to how fast bodies change of state. For example, if we imagine two people with the same properties except mass, the one with higher mass will require more time to change from standing to running.

- **Strength.** It is the scalar quantity which defines the physical power a person or animal has. Notice that this property is inherent to an individual and does not have direction, if it had, we would be talking about *force*, concept that will be introduced later. And again, this is related with how fast bodies change of state. The bigger strength, the faster change.

- **Position.** These are the location coordinates of a specific point related to an origin. This is not as simple as it might seem, because bodies have a volume, so which the exact position is, is a controversial discussion. Normally it is used the centre of mass, but other points can be used depending on the approach. In order to calculate the movement, we need to know the rate of change of the position, both the magnitude and the direction.

- **Velocity.** This is the vector which defines the rate of change of distance over time. Its standard unit of measurement is m/s. Mathematically it can be expressed as follows:

$$v = \frac{\Delta x}{\Delta t} \tag{3.1}$$

- **Acceleration.** It is a vector which defines the rate of change of velocity over time. Acceleration is written as m/s^2, and expressed by:

$$a = \frac{\Delta v}{\Delta t} \tag{3.2}$$

- **Force.** This is the main element of the physical model followed by this thesis. According to Isaac Newton: "An impressed force is an action exerted upon a body in order to change its state, either of rest, or of uniform motion in a right line". Therefore, a force is a quality that can alter an object's speed or line of motion. It is measured in Newtons and represented as a vector, with both magnitude and direction

We know that to change the position of an object (to move it), it is needed to make its velocity greater than 0, and in order to achieve that, there has to exist an acceleration. Basically, what the object is experimenting is a change of state due to an acceleration produced by forces.

Newton's second law states the relatioship between an object's mass m, its acceleration a, and the applied force F by the equation:

$$F = ma \tag{3.3}$$

Therefore, this gives us the key to perform a physically based simulation. According to all mentioned before, the motion an object experiments in a physically based virtual world can be calculated after synthesizing all the forces applied over it:

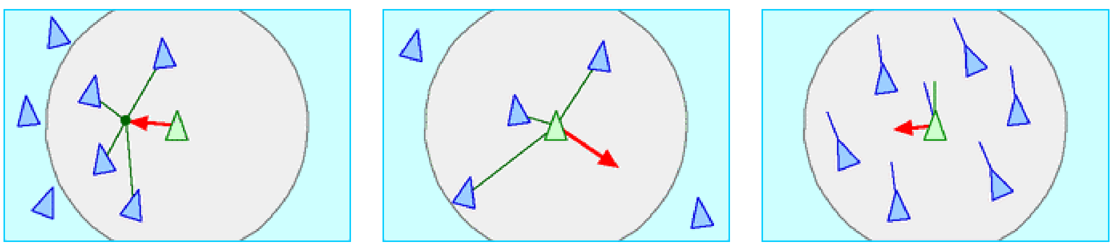
1. $a = \frac{F_{total}}{m}$

2. $v_{t+1} = v_t + a$

3. $p_{t+1} = p_t + v_{t+1}$

3.2 Reynold's Flocking Algorithm

Craig W. Reynolds proposed in 1987 a model to simulate natural group behaviours such as herds, flocks or schools (Reynolds, 1987). This is a very powerful mechanism to use in crowds, besides sharing the principles this thesis is based on. The method works by applying three simple rules to each individual of the flock, which Reynolds called *boids*, that make them move as a unit.



(a) *Cohesion Rule* **(b)** *Separation Rule* **(c)** *Alignment Rule*

Figure 3.1: *Reynolds' Model Rules*

Each boid knows a set of neighbours in the flock which influence its movement. Only the ones that are within a certain distance are considered neighbours, and the rest are ignored; thus, boids has local conscience of the flock. According to the virtual force model, those simple rules produce these three steering forces:

- **Cohesion.** This force makes the boid move towards the centre of mass of the neighbourhood so that they remain close to each other.

  Let p be the position and v the velocity of the current boid, p_i the position of the neighbour i and n the number of neighbours:

$$centreOfMass = \frac{\sum_{i=1}^{n} p_i}{n} \tag{3.4}$$

$$cohesionForce = normalize(centreOfMass - p)maxSpeed - v \tag{3.5}$$

- **Separation.** This force makes the boid move away from its neighbours, to avoid remaining too close.

  Let p be the position of the current boid, p_i the position of the neighbour i, d_i the distance to the neighbour i and n the number of neighbours:

$$separationForce = \sum_{i=1}^{n} \frac{normalize(p - p_i)}{d_i} \tag{3.6}$$

- **Alignment.** This attemps to keep the boid aligned with their neighbours.

  Let p be the position and v the velocity of the current boid, p_i the position and v_i the velocity of the neighbour i and n the number of neighbours:

$$averageHeading = \frac{\sum_{i=1}^{n} normalize(v_i)}{n} \tag{3.7}$$

$$alignmentForce = averageHeading - normalize(v) \tag{3.8}$$

3.3 Finite State Machine (FSM)

Finite State Machines, or FSM, have been the main instrument of choice to imbue an agent the illusion on intelligence. Some of the reasons are these:

- Quick and simple to code
- Easy to debug
- Little computational overhead
- They are intuitive
- They are flexible

Historically, a FSM is a rigidly formalized device used by mathematicians to solve problems, whose precursor might be considered the Turing Machine. The idea is to decompose an object's behaviour into easily manageable "chunks" or states. For instance, a light switch is a very simple FSM where *off* and *on* are the states. Transitions are made by the input of the fingers. By clicking the switch up it triggers the transition from off to on, and by clicking the switch down it triggers the transition from on to off. There is no action associated with the off state, but when it is on, the electricity is allowed to flow and light up the room.

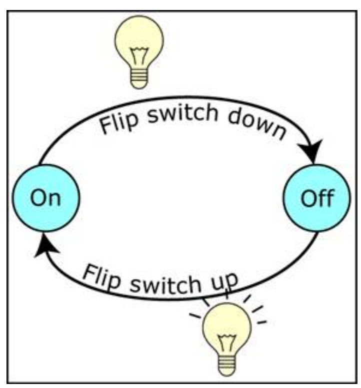


Figure 3.2: *Light Finite State Machine*

As mentioned above, one of the main advantages of FSM is that they are very intuitive. It's human nature to think about things as being in one state or another. Humans do not really work like FSM but sometimes it is useful to think our behaviour in this way. It is fairly easy to break down an agent's behaviour into a number of states and to create rules to transit among them actions associated to each state.

MESSAGE CAPABILITIES!!

Chapter 4

Agent-Based Model

4.1 Agent Body

4.1.1 Definition

4.1.2 Physical Properties

4.2 Agent Brain

4.2.1 Capacity of Decision

4.2.2 The Illusion of Intelligence

4.3 Interaction among agents. Message passing

4.4 Overview of Possible Behaviours

Chapter 5

Crowd Engine

5.1 Modelling the World

5.2 Handling Large Amounts of Agents

5.2.1 Cell Partition

5.3 Virtual Force Model

5.3.1 Prime Mover Force

5.3.2 Gravity Force

5.3.3 Friction Force

5.4 Handling Messages

5.5 Collision Detection

5.5.1 Bounding Cylinder

5.5.2 Bounding Sphere

Chapter 6

Technical Background

6.1 Physics

6.2 Reynold's Flocking Algorithm

6.3 Finite State Machine (FSM)

Chapter 7

Technical Background

7.1 Physics

7.2 Reynold's Flocking Algorithm

7.3 Finite State Machine (FSM)

Chapter 8

Conclusion

Your conclusion should be structured like this. An introductory sentence then:

8.1 Summary

A summary of what has been achieved.

- Bullet points are good to clarify
- Bullet points are fun
- I ran out of ideas

8.2 Known bugs and issues

This is optional, some people put it in future work, I think it is better to have a separate section for it. Issues and bugs are different. Bugs are unexpected behaviour in the program, something not running as it should. Issues are more due to algorithm limitations. If the algorithm only works for meshes of less than 100k polygons, it is a limitation, not a bug. A program crashing if you move around in a particular order is a bug. A render becoming ugly just on that particular spot in space, is a bug too.

8.3 Future work

What should be done in the future, what you would like to do.

- Become an astronaut
- Go out
- Make it work for real

Bibliography

M. Buckland (2005). *Programming Game AI by Example*. Wordware game developer's library. Wordware Pub.

L. Kleinrock (2011). 'What is emergent behaviour?'. Available from: http://curiosity.discovery.com/question/emergent-behavior [Accessed 10.08.2013].

S. R. Musse & D. Thalmann (2001). 'Hierarchical Model for Real Time Simulation of Virtual Human Crowds'. *IEEE Transactions on Visualization and Computer Graphics* **7**(2):152–164.

S. R. Musse, et al. (2005). 'Groups and crowd simulation'. In *ACM SIGGRAPH 2005 Courses*, SIGGRAPH '05, New York, NY, USA. ACM.

S. Patil, et al. (2011). 'Directing Crowd Simulations Using Navigation Fields.'. *IEEE Trans. Vis. Comput. Graph.* **17**(2):244–254.

C. W. Reynolds (1987). 'Flocks, herds and schools: A distributed behavioral model'. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '87, pp. 25–34, New York, NY, USA. ACM.

C. W. Reynolds (1999). 'Individual-Based Models'. Available from: http://www.red3d.com/cwr/ibm.html [Accessed 10.08.2013].

C.-C. Wang & T.-Y. Li (2006). 'Evolving Crowd Motion Simulation with Genetic Algorithm'. In *Proc. 3rd Intl. Conf. on Autonomous Robots and Agents - ICARA'2006, Palmerston North, New Zealand*, pp. 443–448.

You can have different appendices (A. B. etc...) and sub sections too.