Aerial Combat Simulation Using Massive

Ed Hicks[1]

March 2008

[1]email: cheezopath@gmail.com

Abstract

This project is an attempt to tackle the problem of simulating realistic AI in 3D flight and aerial combat scenarios, using the massive crowd simulation software production environment. The deliverable material takes the form of instructional tutorial videos and supporting files to better aid understanding and the following of the tutorials. They begin as a basic introduction to massive and its use, as the active massive user community is relatively small. The pace of the tutorials then changes to demonstrate the problems encountered and the solutions developed to overcome these problems. By the end a solid AI framework is established which can be built on to produce satisfying results for a number of in flight scenarios.

Contents

Chapter 1

Presentation Script

1.1 Introduction

Welcome to complex agent creation in massive. For this project I used massive to develop an agent designed to simulate an aerial combat scenario.

1.2 Aims

1. To achieve realistic behaviour in the context of a dogfighting scenario
2. To make use of fuzzy logic to solve behavioural problems
3. To test the limits of the massive system with an unconventional project
4. To extend my knowledge of 2D tracking and avoidance solutions into 3D
5. To solve problems introduced by the system
6. To solve problems introduced by the subject matter

Initially I set out to produce a simulation that could be used in a film effects pipeline to simulate large scale air operations, to build on my previous experience with massive and to push the limits of my AI design, and the limits of the system. After I began work and research it soon became clear that building an all aspect general purpose AI for fighter combat would be too big a project for the scope of the innovations unit, and so my aim changed to accommodate. The project was then more about laying down the groundwork to achieve the former aim, by testing and solving problems that would be encountered in development, things that had to be in place for behaviour of this type to be possible. Another thing I realised while working on massive is that it isn't widely used or known, being a specialised and relatively new piece of software. This made explaining my process and methods difficult, so I decided to make my submission format a set of video tutorials. The advantage of this approach was twofold; it could be used to clearly explain my problem solving methods and thought process, as well as demonstrating the software in action, and the second reason was that my project could then be used as training material to make the software more accessible, as training material, aside from the material that ships with massive, is virtually non existent.

With my new approach in mind I went over all the distinct aspects of my AI agent in development and prepared instructional videos on my methods. It actually got to the point that by remaking all the aspects of my agent one by one in a structured educational manner, the AI I ended up with for purely demonstrative purposes was much cleaner and less buggy than the one I had been using up to that point in development, and I switched over to the demo agent for development.

The first big problem I encountered was implementing pseudo physics non dynamically. As access to positional information was restricted to local space on a per agent basis, calculation requiring knowledge of current world location and orientation weren't easily done. After some time working with matrix algebra I produced a fairly accurate reverse engineered version, but it was prone to crashing and bugs that were hard to track, as it all depended on a complicated and lengthily feedback loop of expressions, commonly sending the agent into outer space. Noting my progress and my priorities I switched my focus to the intelligence side of things. Using foresight I designed the AI so that in theory when the physics was solved I could simply bring in my algorithms and plug them into the AI, and everything would work. I planned to use small stand alone behavioural modules arranged in a control hierarchy in order to gain control of my agent's behaviour easily, and it was a method I'd developed a little before in previous projects. Once this was in place I could focus totally on solving the problems in hand, each one at a time and independently.

Being a system based on expressions being constantly re-evaluated, the design workflow felt less like programming and more akin to that of an electrical engineer, which I have a stronger affinity for in terms of problem solving. What this did mean, however, that it was actually quite a task to implement things that are normally taken for granted in other systems such as latching variables and implementing sequences and loops. While progress was being made on the technical front I really had no expertise in the field of air combat, and so to that end I acquired a copy of "fighter combat: tactics and manoeuvring" by Robert L. Shaw which is widely regarded as the best and most complete text on the subject. Reading it gave me tonnes of ideas for features to add, as well as illuminating things I had not considered, be they problems I would encounter or tactical behaviour I had not considered. It soon became apparent I would have to solve the problem of target leading and collision prediction, as well as developing communication protocols between agents that made good use of the sound channels available in massive. It goes into mindblowing levels of detail about all kinds of engagement scenarios from 1 on 1 fighting and paired wingman tactics to how to approach an enemy formation with inferior numbers.

An agent that quickly became vital to my development was the landing strip, with which I used to explicitly spawn the planes, and employed an initialisation protocol which equipped the launched planes with unique ID's, and information relating to their mission, rank etc, all communicated using the agent's hearing feature. Without this controlled agent creation this would have otherwise been impossible on massive. Making this vital secondary agent also made it possible to test some fairly advanced landing algorithms, and to finish up I took it one step further and developed "launch sequences", where at the touch of a button the landing strip agent would launch a preset configuration of planes with defined missions, ranks and combat loadouts.

What follows are some choice clips from the instructional videos I produced, and a little bit of background into what was involved in developing them.

1.2.1 Controls and death

In this clip I am demonstrating manual control of an agent, using the massive's in built controller channels, where a joy pad can be plugged into the USB and the agent hooked up to respond to the inputs. In an hour I had made a fully controllable plane that fired weapons. I then assembled a simple enemy AI that would fly around randomly, and trigger dynamics if a bullet passed too close.

1.2.2 Collision prediction

This clip demonstrates practically a linear collision prediction algorithm I developed in 2 dimensions. Here I have applied it to a tank column that have to navigate through these swinging green cubes of death. Any agent that comes in contact with one freezes and plays dead. With the 75% survival rate it admittedly has weaknesses, particularly if an agent is caught in a cube's path as it is rapidly changing direction, as the time delayed prediction algorithm takes a couple of frames to settle, and is based on current and previous vectors. Even so, it adequately demonstrates how the problem could be approached, and implemented on a large scale.

1.2.3 3D tracking with vision

This missile demonstrates fuzzy logic and 3D vision tracking with the example of a missile, which is used extensively in subsequent demonstrations. It navigates a field packed with green targets, and nails several with it's specialised fuzzy intelligence network.

1.2.4 Firing mission

This clip demonstrates a firing mission . The plane uses sequence and hierarchical control to achieve its goal of taking off, flying to the target zone, identifying, entering the kill zone, firing, confirming the kill, and returning to base. It also demonstrates an emergency handling system, which in this case drops chaff to confuse missiles fired from the defensive SAM site.

1.2.5 Target leading

Target leading was an intelligence problem that had to be overcome, as the finite speeds of projectile ordnance meant that to hit a target, an estimated lead angle had to be calculated and adjusted to. Using bullet velocity, absolute distance and rotational speeds, I could predict and lead a target using vision, as demonstrated here with an AA gun decimating a squadron of fighters.

1.2.6 Landing

In order for my planes to land I had to develop modules that, using sound, could blend between achieving a parallel bearing and pointing at the landing strip. Once implemented a plane could effectively line up with a landing strip from almost anywhere in the sky relative to the strip.

1.2.7 Physics and rotation

This was by far the hardest part to accomplish - working backwards from the limited and local information pertaining to rotation and translation and using it to figure out an approximation to world space in order to implement gravity and momentum. Here I demonstrate it at work by dropping a plane downwards in the world axis while tumbling, which must be translated into a constantly updating local space if the plane is to follow the correct path.

Chapter 2

Research

In its most simple terms, the aim of aerial combat is to achieve a position where the requirements of the currently equipped weapon system(s) to prove deadly are met, as well as denying the opponent to achieve these parameters. Shaw [1985] (page 1) asserts this in the introduction to his chapter on fighter weapons;

> "The airplane itself may be considered only a weapons platform designed to bring the weapons system into position for firing."

With this in mind I knew I had to develop the manoeuvres and behaviour based primarily on what they would be fighting with. This also meant that before I could start work on these manoeuvres I would have to design the weapons systems. The two obvious ones to implement were homing missiles and guns. Making a bullet agent was trivial, though I did not develop it to the point of simulating the effects of gravity on the trajectory, which made the task of target leading simpler. Shaw [1985] details all considerations when bringing guns to bear upon the enemy and the conditions needed for correct leading. Using this as a guide, I developed my own solution, using fuzzy logic and an equation taking bullet velocity, rate of turn and absolute separation distance into account. The results shown in the demonstrative video of the fixed AA tower firing on the fighter squadron were conclusive.

Producing a missile was a good chance to try out the fuzzy logic tracking techniques in 3D. Shaw [1985] describes a few missile guidance techniques and the popular methods for foiling each. Keeping in mind the tools available to me I decided that using the vision system to emulate a sidewinder style heatseeker. Chaff could be dropped as a decoy, as long as it was the same colour as the target and placed in the flightpath of the missile. Using a fire-and-forget missile system meant avoiding the problem of communication between plane and missile, though problems of this type would need to be solved later regardless, as communication between planes in formations and performing wingman tactics would need attention.

Firing envelopes were a consideration I noted for future use. A firing envelope, as the name suggests, is a set of spatial constraints, as well as constraints like relative speeds which determine the effective killzone of a weapon system, the conditions of which the AI would endeavour to satisfy in order to execute a kill. While I did not implement them, the theory would have been to define a set of rules where the relative angles, velocities, altitudes etc would be resolved into a value which could be fuzzed into a fuzzy rule, and weighed against ammunition, firing orders and other similar limiting factors to influence the decision of whether to fire.

With acquiring and maintaining the firing envelope in mind, I studied the types of pursuit covered in Shaw [1985]. There are three types; lead pursuit, pure pursuit and lag pursuit. Each is defined by the angle of the pursuer's nose relative to the target. To paraphrase, in lead pursuit the nose is pointed ahead of the target aircraft. This provides the fastest closure, but as the target closes, the line of sight (LOS) increases, and so could prove less useful when range is suitable for a gun kill. Pure pursuit would be ideal in a space sim involving laser guns, as the target craft remains zeroed for the duration of the approach (of course the firing envelope of laser guns would be greatly extended and target leading virtually non existent. The range would most likely be determined more by the ability of the pilot or the target system to maintain a lock on the target). It provides reasonable closure, largely dependant on relative speeds, without affecting the LOS much. The lag pursuit points the attacker's nose behind the enemy fighter, giving the gentlest closure and resulting in the smallest LOS angle. It is ideal for maintaining a distance from a target so that the attacker can remain outside the minimum firing range of the equipped ordnance, or to remain in the rear quarter (4 to 8 o'clock) of a target with a much lower airspeed.

Another very important consideration in engagement was that of the energy of a plane. It is not merely a measure of the speed of an aircraft, but also of the altitude, and the art of trading one for the other without a significant loss in energy was one every ace would have to master. Anything that increases drag, such as many sharp turning manoeuvres in quick succession, would contribute to the loss in energy, and it was much easier to lose than to gain, particularly when the pilot was more busy fighting for his life. A loss of energy could mean not having sufficient speed to pull up in time, or achieve complex manoeuvres at low altitudes. It also left an aircraft prone to points where it was a practical sitting duck, stuck trying to gain airspeed to avoid stalling and manoeuvring little.

In massive this sense for potential and actual energy built up could be expressed by calculating roughly the potential energy stored in the altitude summed with the kinetic energy currently in effect. While this was not implemented it would be trivial to include, but highly useful for the AI to make tactical decisions.

After this point the more advanced manoeuvres and tactics became less important to study, as I had to focus on lower level concerns in the time remaining. While not implemented in this project, I still endeavour to study and include the more advanced behaviour involved in complex deployment.

Chapter 3

Conclusion

Overall I feel the project was a success. While I did not accomplish my initial aims, they were fairly unrealistic to begin with, which was all part of what I learnt during the course of development. For every problem I encountered I produced at least a partial solution, and at best one that fit perfectly within the massive environment and solved the problem efficiently and concisely. Being forced to review what I had done when I decided that the production of tutorial videos was the best path for explanation meant that I could rigorously test, check and streamline my already developed modules and behaviour, in all cases increasing the final quality.

During the course of this project I got accustomed to producing networks with nodes so much that I could replicate in a matter of minutes something which had initially taken me hours to develop. I found I began to think in terms of what the system could offer me and what it could not, and it allowed me to tackle the problems of target leading and landing with relative ease. Many of the problems I solved were not specific issues to the project's subject matter, and I will be carrying the solutions I developed over to my future projects with massive.

The overall most limiting factor was the restriction of access to certain information that the system kept, in particular the world space co-ordinates and orientation matrices of agents. In retrospect I spent far too long working on this relatively minor aspect of the agent and even though I mostly solved the problem, I feel that if I had got started on the AI sooner, I would have a more complete final agent, possibly enough progress made to perform one or more simple 1 on 1 manoeuvres. As it stands, the material resembles much more a tool kit with which anyone should be able to take further into a final simulation.

While the channels you do have access to are fairly realistic, there are so many little things that the system knows about which you have no access to, as they are technically unrealistic things for an agent to know in the situation. There are no global variables or explicit message passing in massive - the closest you can get is by getting near to another agent and transmitting a message with sound, which I deal with in the protocol section, where I set up communication between a landing strip and a plane, setting a few variables on the plane itself before allowing it to fly off and complete its mission. It seems like software designed for use in film should allow hacking around and quick dirty fixes for problems like that, as in a film production environment, whatever will get the job done is what will be used, even if it is technically cheating. As it stands massive enforces a very thorough and honest approach, which at times feels over the top in terms of sensory realism, particularly when all you want is agent X to send a message to agent Y.

As a very visual learner I find that the best way for me to pick something up is to have it demonstrated to me, and so the initial learning material package provided with massive was great to work from. This is currently the only material available anywhere, however, and so I feel obliged to report my findings, my mistakes and my breakthroughs. Recording the four and a half hours of tutorials was as much a learning experience for me as it would be for anyone watching, as I had to deal with the challenges of getting a good take, while explaining the tasks clearly and relying on massive to behave as expected.

Given the chance to start over, I would have focussed more on the problems that pertained to the field of air combat and less to fighting the innards of massive. While I learned a lot about both, I feel that my initial aim to produce a shot with combat taking place took a backseat to the problem solving. Of course given more time (and I fully intend to continue with this project) I am confident I would be able to get my AI to the point that it can be used as a convincing background to middle-ground supporting character in a high end visual effects shot in an aerial combat production.

Bibliography

Shaw, R. L. (1985), *Fighter comboat tactics and maneuvering*, Patrick Stephens.