

# Afstudeerproject Opdracht 4

Inge Becht  
6093906

June 2011

## 1 Research Question

In game AI a challenge is to create game bots that are able to make smart decisions based on a game situation, without the human playing having the feel the AI is cheating. This can possibly be done by not giving the AI full knowledge about the game world, like where the opponent may be at all times. In my thesis I want to find out if I can use a model for predicting an opponent's position to perform better in combat situations. This will be tested within a multi-agent game of capture the flag, with a basic AI that is improved upon by adding a opponent prediction model and a reasoner that acts upon a given situation.

## 2 Relevant Works

In the works of (?) two different models for opponent position prediction are tested. The idea of the research is to see how well the predictions can be made, as well as to test how human like the predictions made are. They test this for both Hidden Semi Markov Models and Particle filters, by letting people predict opponent position in the game Counter Strike, and then look how well these models perform in regard to human prediction. The conclusion of the research is that Hidden Semi Markov Models work best, and are similar in accuracy towards how humans would predict opponent behavior. In case mistakes are made by the model, the mistakes are very human-like, making the AI more believable and less perfect. The research is thus mostly working on being human-like, but does not work out how to integrate such a prediction system in a multi-agent AI system. I could possibly use their implementation model of the particle filter or the Hidden Semi-Markov Model to test how well a multi-agent system can be made that reasons about this uncertainty data.

In (?) a same approach is made using a particle model, but this time not the focus on if the predictions made are human-like but if integrating this in an AI system can enhance performance. Essentially this is the same that I want to

do, except in this case it is used on a different game type, a Real Time Strategy game. The game Star Craft was used and the EISBot was enhanced with their developed reasoning capabilities. The outcome was that the model outperformed the other models by 10 %, but that making more game states available to the bot does not always improve the performance. Although this research does sound quite similar to my own research question, the domain is different enough to expect different result, but still their particle model implementation could help me with developing my own way of prediction opponent position.

Instead of trying to predict the opponent position, the authors of (?) focused on creating a reasoner that anticipates behavior of an opponent in Quake. They did this by enhancing the Soar Quakebot with anticipation strategies. These anticipation strategies are used only in case the bot has a high chance of successfully predicting what the opponent is about to do, and in this case reasons what the bot might do if he was in its place. Using this chunking is applied that construct some rules that completely predict the actions of the opponent and how to respond towards this situation. Although they state their additions were fruitful they do not really show experiments that confirm this. Also there are still elements in their reasoning that could be approved upon, by using recursive reasoning and by making it more general for other games. Although the reasoner is developed for a single reasoning agent, the ideas given could help for a multi-agent based system. While the previous articles give inspiration towards how to make models for the prediction of opponent position, this article goes more in depth on how to create the reasoning part of the AI.

Article (?) deals with the same problem of tracking people, but does this in real life situations, with a robot that assists humans in day to day tasks. To keep the robot from interfering it needs to predict where humans are heading and what their intentions are. This paper explains all aspects of these kinds of predictions. For example, using sensor data to sense humans and how to keep track of a single person. Both these aspects are less important for my own research, but there are sections that explain how Hidden Markov Models can be used so that motion patterns can be learned, something that is indeed interesting for me. as well as the use of an Expectation Maximisation algorithm. In section 2 their own way of modelling expectations of human motion is explained, and some basic ideas can be retrieved for own research. In section 3 Hidden Markov Models are used to learn motion patterns. This can come in really handy in case of building a reasoner, which in essence should be able to recognise some patterns and then use these patterns to respond to a given situation. Section 4 is less important, as it is about person detection and identification, something I will presume is given information in the game AI world, or at least does not need any sensor data like in case of a robot. The conclusion of their own research is that motion patterns can indeed be learned using the Hidden Markov Models and that it is able to reason with this knowledge.

In the paper (?) both the combination of making predictive models for op-

ponent positions is discusses as well as a way of intercepting the opponents in game. The predictive models are made using a particle filter both with IRL and Brownian motion to test which works best. Not surprisingly, IRL seems to give the most accurate result and helps to intercept the opponent the best. For interception 3 different heuristics are tested. Although it looks a lot like what I want to do, it is different in multiple aspects. In this case a different game with different goals is used. There is only 1 bot against multiple different opponents, and it is only tried to intercept.