Classifying an Opponent's Behaviour in Robot Soccer

David Ball and Gordon Wyeth
School of Information Technology and Electrical Engineering
University of Queensland
Australia, 4072
{dball, wyeth}@itee.uq.edu.au

Abstract

This paper illustrates the prediction of opponent behaviour in a competitive, highly dynamic, multi-agent and partially observable environment, namely RoboCup small size league robot soccer. The performance is illustrated in the context of the highly successful robot soccer team, the RoboRoos. The project is broken into three tasks; classification of behaviours, modelling and prediction of behaviours and integration of the predictions into the existing planning system. A probabilistic approach is taken to dealing with the uncertainty in the observations and with representing the uncertainty in the prediction of the behaviours. Results are shown for a classification system using a Naïve Bayesian Network that determines the opponent's current behaviour. These results are compared to an expert designed fuzzy behaviour classification system. The paper illustrates how the modelling system will use the information from behaviour classification to produce probability distributions that model the manner with which the opponents perform their behaviours. These probability distributions are show to match well with the existing multi-agent planning system (MAPS) that forms the core of the RoboRoos system.

1 Introduction

When planning in adversarial environments, the performance can be increased by taking into account the predictions of an opponent's behaviour. Predicting an opponent's future in a competitive environment is difficult as generally the opponent's plans are hidden. Robot soccer is an example of a competitive environment where plans for the future must be made in a highly dynamic, multi-agent environment in real time. The ability to predict the opponent's future behaviours during competition allows for more informed decisions.

Prediction of an opponent's behaviour requires that their current behaviours can be recognised and classified and that these behaviours can be modelled and recalled. This paper proposes a layered approach where classification is to reason about the opponent's current behaviour, modelling is to reason about their internal decision process and prediction is to reason about their future behaviour.

An opponent is an agent that has private strategies and has goals that are conflicting to your own. Their world state and behaviour list may be known or assumed but their utility functions (preferences) are not. While this research is focused on modelling and predicting an opponent agent, it can also be applied to the more general case of an agent in any environment.

Representing uncertainty is important when dealing with agents that work in the real world. This is especially important with robotics as there is always some uncertainty in the information provided by their sensors. Probability theory provides a basis for dealing with this uncertainty. As it is not possible to classify or model the opponent's behaviours with complete certainty, a probabilistic approach seems valid. There are many different probabilistic methods and representations, for example; general probability laws, continuous and discrete probability distributions, Bayesian techniques including the Bayesian Classifiers. The theme of this research is:

"How well can probabilistic methods classify, model and predict the behaviours of opponents in a competitive, multi-agent, highly dynamic and inaccessible environment and what effect do these predictions have on the performance of a planning system?"

Utility theory allows different methods of achieving goals and the likelihood of achieving these goals to be weighed up. The combination of probability theory and utility theory constitutes decision theory, and this is important for building robots and agents for the real (uncertain) world. By integrating the predicted behaviours into a planning system more informed decisions about what behaviours to assign to each agent can be made.

1.1 Testing Domain

The testing domain is the RoboCup small size league [Kitano *et al.*, 1995]. The research platform, the RoboRoos, competes in the F180 league (also known as the small size league) of the annual RoboCup competitions. In the F180 league both teams have five robots that each must physically fit inside a cylinder with a diameter of 180mm and a height of 150mm. Devices to dribble and kick the ball are permitted as long as they do not hold the ball and 80% of the ball is kept outside of the convex hull of the robot. The field is approximately 2.3 ×

2.8 metres, with an orange golf ball acting as the soccer ball. Teams use global overhead vision as the primary sensor.

The rules are similar to the human version of the game (FIFA), with exceptions such as the elimination of the offside rule and changes required to make sense for wheeled robots. There are two 10 minute halves. The robots are fully autonomous in the sense that no strategy or control input is allowed by the human operators during play. Humans referee the matches.

In relation to behaviour classification and prediction, the F180 league domain is partially observable, stochastic, sequential, dynamic, continuous and multi-agent which is regarded as the most difficult domain for an intelligence system [Russell and Norvig, 2003]. In regard to the partially observable claim it is true that the overhead camera generally provides the pose state of the robots and the ball. However the behaviours of the opponent robots are not directly observable and are relevant to the choice of action by a system that performs on a behavioural level. The domain is;

- stochastic as the next state cannot be predicted exactly due to the complexity of the environment and the potential inability to perform the behaviours as desired,
- sequential as the current decisions may effect all future decisions,
- highly dynamic due to the high velocities and accelerations that the ball and robots move at,
- continuous due to the continuous nature of the pose inputs and possible wheel velocity outputs,
- multi-agent due the existence of other intelligent agents in the domain.

Due to these properties classifying, modelling and predicting the opponent's behaviour is a difficult task.

1.2 The RoboRoos Team

The University of Queensland's robot soccer team, the RoboRoos [Ball *et al.*, 2003], [Wyeth *et al.*, 2002], [Wyeth *et al.*, 2001] and [Wyeth *et al.*, 1999] is one of the longest standing teams in the small-size league of RoboCup having competed annually since 1998. During these years the performance of the team has been successful, and many research areas explored especially in the areas of multi-robot coordination and navigation in highly dynamic environments. There have been two major generations, the 1998 RoboRoos and the 2001 RoboRoos, both shown in Figure 1. At RoboCup 2003, the RoboRoos came second, beaten 1-0 in the final by Big Red from Cornell University. Aggregate goals for the competition (RoboRoos versus all other teams played) was 63-5.

Figure 1: Action shot of the RoboRoo robots. The 2001 RoboRoos are the larger robots and the 1998 RoboRoos are the other robots.

The RoboRoos system is a layered set of subsystems, where each subsystem performs a different task. The flow diagram in Figure 2 shows how the system is laid out overall. To give an overview of the system the flow of the information from the camera to the robot's actuators is presented.

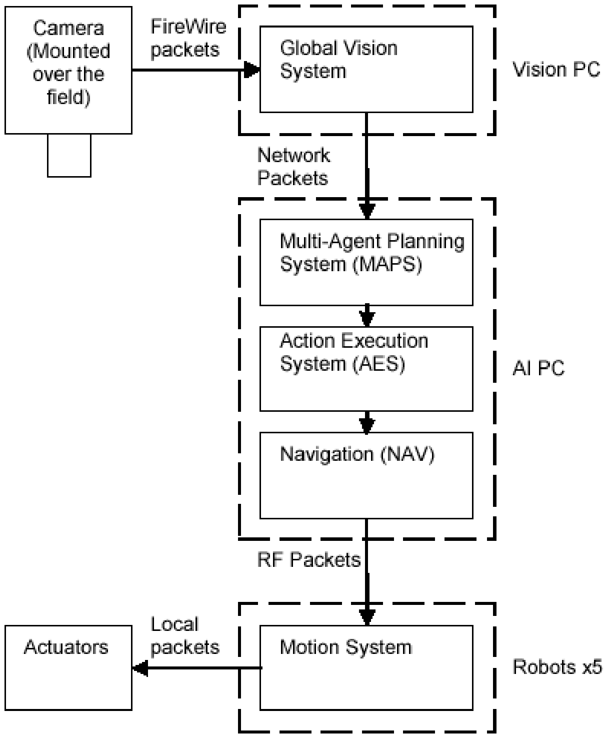

Figure 2: Flow diagram showing the overall RoboRoos software system.

An overhead camera captures global images of the field. The vision system processes the images to identify and locate the robots and the ball. This state of the field is sent to the Multi-Agent Planning System (MAPS) [Tews, 2002]. MAPS is the highest level planner in the RoboRoos system. MAPS coordinates the RoboRoos by selecting a behaviour for each robot. Some example behaviours include KICK and DEFEND. The MAPS behaviours and behaviour parameters are now passed to the Action Execution System (AES).

The MAPS behaviours are interpreted by the AES system. Each behaviour has a set of appropriate parameters and a notion of the overall desired robot motion. The Navigation [Browning, 2000] module attempts to achieve the desired motion behaviour while avoiding obstacles. The Navigation module determines the immediate desired heading and distance for the Motion System. The Motion system accelerates and decelerates the robot to the desired heading and distance by creating force limited trajectories.

1.5 Paper Overview

The next section outlines the complete research proposal for a system to predict the behaviour of opposing agents. Section 3 review the literature for methods for the first part of the proposed system: the Behaviour Classification System. Section 4 shows the results so far in classification of behaviour for the RoboRoos system. Section 5 illustrates how this work will link with the Behaviour Prediction system, to achieve the aim of reading the opponent's play. Section 6 draws conclusions form the work performed and highlights the immediate future work.

2 Predicting Opponent Behaviour

The effectiveness of the RoboRoos intelligence system is potentially limited by its inability to predict the likely behaviour of the opposition. The module that deals with the effects of the opposition – MAPS – currently treats the predicted effect of the opposition as a probability distribution around the opposition robot's current location. In order to allow more powerful planning to be used, a more accurate model of the effect of the opposition agents is required.

This research proposes to allow prediction of an opponent's behaviour in a highly dynamic, multi-agent and inaccessible environment. The predictions of the opponents will be integrated into the current planning system. The uncertainty in the observations by the sensors and the uncertainty in the opponent's predicted behaviours will be accounted for by using probability techniques. The prediction system will not determine a definite behaviour but give probability distributions over the range of possible behaviours. The proposed system will use a layered and modular approach to the task of predicting the opponent's future. The proposed system is shown in Figure 3. There are three parts to the implementation of the proposed final system.

- Behaviour Classification System.
- Behaviour Modelling and Prediction System.
- Integration into the existing planning system.

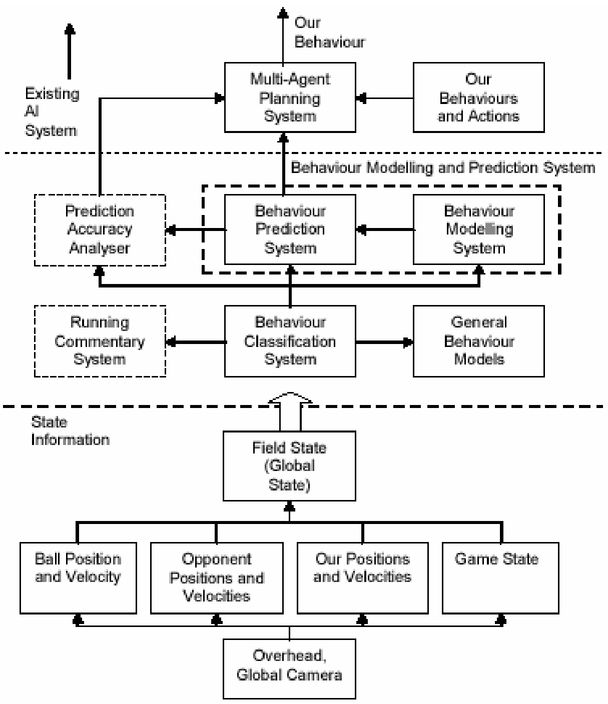

Figure 3: The proposed opponent behaviour classification, modelling and prediction architecture shown integrated into the current planning system. The field state represents the observations that are available to the opponent prediction system.

This paper illustrates the implementation of the Behaviour Classification System and includes initial results. It also details the intended integration into the prediction and modelling system as well as outlining integration into the existing intelligence system.

The Behaviour Classification System uses a Bayesian Classifier to reason about what the opponent's behaviours are. This system aims to, for each agent, give a range of probabilities over the typical behaviours for the environment. A probability technique is used due to the inaccessibility in observing an opponent's behaviours and the uncertainty in the observations.

The Behaviour Modelling and Prediction System consists of two separate (but similar) sub modules, one that models and one that predicts. The opponent's behaviours will be modelled by incrementally building a 2D probability distribution representation of the features of the behaviours. These models will represent the utility functions of the opponents. The opponent's behaviours will be predicted by combining their current behaviour (from the classification system) with the models of their behaviours. The prediction accuracy will be quantitatively measured by comparing the predictions to the actual behaviours as they occur.

Finally these predictions and the estimate of their accuracy will be integrated in to the existing planning system, MAPS [Tews, 2002]. MAPS (Multi-Agent Planning System) is responsible for distributing the overall goal of the team (to score more goals than the opposition) to the robots. It does this by assigning a behaviour to each robot. This system works by overlaying potential fields that represent different physical and abstract features. The predictions of the opponent's behaviour will serve as more abstract potential fields that will be overlayed onto the current fields.

For example, the locations where the goal is covered by the opponent's robots are dependent on the balls location. By learning and modelling these locations the uncovered shot angles at goal (weak spots in the defence) can be determined. These could be extended out and overlayed onto the field that generates locations to dribble the ball to or the field that determines the locations for pass receivers to wait. Note that this would one example of predicting the opponent's future (where their robots will move to dependent on where the ball is moved to).

3 Behaviour Classification

The first part of the proposed opponent prediction system is behaviour classification. This section looks at the state of the art in classifying the behaviour of agents in adversarial environments.

Classification is the process of grouping a distribution into classes according to common relationships. Classification can be a difficult task, especially in complex and inaccessible environments using real sensors with uncertainty in their measurements. First a list of useful classes must be determined that will enable all input data to be allocated. The important features or relationships that represent a particular class must then be determined either by an expert or by training from data. Now the network can be used for classification of data.

3.1 Heuristic

Classification can be performed by quantitatively comparing input features to predefined behaviour models that represent each class [Riley, 1999]. This measures how similar the features in the data are to the features in the predefined models. By adding up the similarities (using an appropriate method and perhaps weighting) the closest class to the input data is determined.

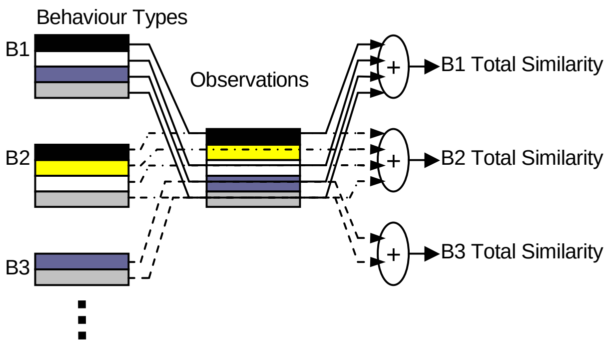

Figure 4: Riley et al [Riley, 1999] behaviour classification system for RoboCup simulation soccer. It uses the weighted sum of the similarities of observations of teams motion to determine the team behaviour type.

3.2 Artificial Neural Network

Costa Florencio [Florencio, 1998] demonstrates using an Artificial Neural Network to recognise rat behaviour from movies. Examples of behaviours that the system recognised are rear, groom, head raise, sit, hunch, head dip and stretched attend. It uses a standard feed-forward neural network that has access to the parameters from three successive frames which allows access to temporal regularities in the data. Examples of parameters that were extracted from the images (pre-processing) were surface of the rat's body, centre of gravity, tail point, nose point, and bounding box. The network had 48 input nodes, 18 hidden nodes and 10 output nodes. The back-propagation rule with momentum was used to a train the network. Classification accuracy was approximately 85%.

3.3 Hidden Markov Model

A Hidden Markov Model represents the relationship between internal states and observations using probabilities, where the states are hidden from an external observer. In this way the state that a system is in or the behaviour that it is executing is represented relative to the observations. States and transitions between states have probabilities associated with them.

A behaviour is represented as a series of states that are transitioned through to complete the behaviour [Han and Veloso, 1999]. These states are not visible to the external viewer. The states are mapped to the nodes that represent observable features of the world. Observable features include the absolute position, object relative positions of the objects as well as their dynamic properties. Four types of states exist: initial, accept, intermediate and reject. Initial states represent the state that the agent will be in at the start of the execution of a behaviour. Accept states represent the successful completion of a behaviour and therefore detection of the behaviour. Intermediate states represent the intermediate stages of a behaviour that the agent must go through from the initial state to the accept state. Reject states represent states the agent should not be in for the current behaviour. Figure 5 shows this HMM type applied to recognising a "Go-To-Ball" behaviour.

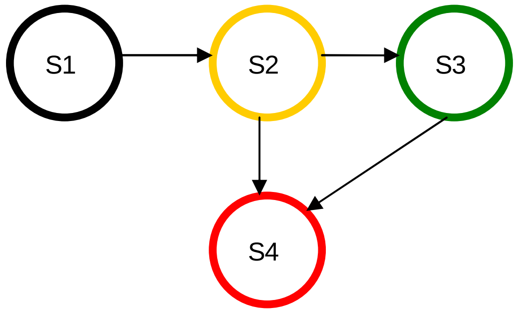

Figure 5: Han et al [Han and Veloso, 1999] Behaviour HMM representing a "Go-To-Ball" behaviour. S1 is the initial state indicating the robot is far away from the ball and S2 represents the robot moving towards the ball. S3 represents accepting this behaviour and S4 represents rejecting this as the behaviour.

3.4 Bayesian

Bayesian networks [Pearl, 1988] are able to reason under uncertainty as they integrate a graphical structure that represents the causal relationships between nodes and have a sound Bayesian foundation.

A Bayesian classifier [Duda and Hart, 1973] is a type of Bayesian network (graphical networks that encode the relationship between nodes) that is able to classify cases of data. It is a probabilistic method of classification. Generally, the task for a classifier is to determine which class a data case belongs to based on several attributes. A Naïve Bayesian classifier [Duda and Hart, 1973] is a type of Bayesian classifier that makes the simplifying assumptions that the classes are mutually exclusive and exhaustive and that the attributes are conditionally independent once the class is known [Cantu, 2000]. Figure 6 shows a diagram of a naïve Bayesian classifier.

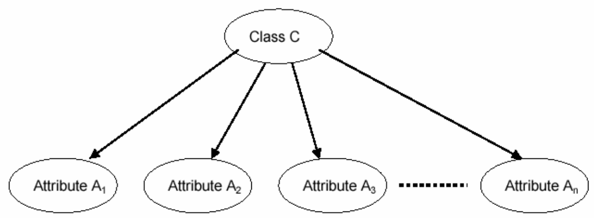

Figure 6: Naïve Bayesian Classifier. A type of Bayesian network where there are no connections between the attributes. This makes it very simple to implement and use.

The conditional probability tables representing the relationship between the attributes and the classes can either be learned from a database of cases or determined by a domain expert. In the learning case, Bayes' rule is used to learn the conditional probabilities from a database of cases. In the domain expert case, all probabilities must be estimated by a human.

There are some situations where the Naïve Bayesian classifier will not give good results. However as they are simple in structure, easy to implement and will often give good results they are worth trying, especially on problems where the independence assumption on attributes is appropriate [Cantu, 2000], [Pazzani and Billsus, 1997]. In a Naïve Bayesian classifier, a set of C classes is defined by a set of A attributes. Assume that the classes are mutually exclusive and exhaustive. Also assume that the attributes are conditionally independent once the class is known. Given a case *j* with *n* values V for the attributes, then:

$$P(C_i \mid A_1 = V_{1j}, A_2 = V_{2j}, \ldots A_n = V_{nj}) \propto P(C_i) \prod P(A_k = V_{kj} \mid C_i)$$

Both of the terms on the right side may be estimated from training data.

Pazzani et al [Pazzani and Billsus, 1997] demonstrates the successful use of a NBC that determines whether a given web page would be interesting to a user based on their previous responses.

Steffens [Steffens, 2002] introduces a Feature Based Declarative Opponent-Modelling method for classification of a team's overall strategy type. This method assumes that a small set of models of team strategies can represent a wide range of opponent strategies accurately enough. This is the same assumption as in Riley et al [Riley, 1999]. The first step was to manually create models of team strategies based on observations of several teams during competition. Then a Bayesian classifier is used to determine the best matching team model based on observations of particular features of the opponent's behaviour during a game. The performance of this method was insignificantly better than random guessing. This research also investigated the development of a counter strategy for each manually created team strategy model.

4 Behaviour Classification Experiments

While each of the systems described in the preceding section have the ability to perform behaviour classification, the Bayesian classifiers have the advantage of being readily understandable while giving a probabilistic output that is suitable for representing the uncertainty in the domain. For this reason, the Behaviour Classification System has been designed around Bayesian classification techniques.

The Behaviour Classification System (BCS) system will determine (as a probability) the opponent's current behaviour based on information about the absolute and relative positions and velocities of the opponent's robots, the ball and some features of the field. The behaviours are complicated and need multiple attributes to separate them. The BCS will be based on a Naïve Bayesian classifier where the classes will represent the behaviours and the attributes will represent the observations of the field state. A probability distribution over all possible behaviours will be determined for each opponent robot. While this system is research worthy itself, the main reason for it is because the modelling and prediction systems need this opponent behaviour input.

Information from the RoboRoos system will be used to train the Bayesian conditional probability tables. These tables will be able to viewed and adjusted by a domain expert, something not possible when using a neural network and weights. Also having the result of the classification as probabilities over the range of possible behaviours gives a level of confidence.

To provide a comparison point for classification results an expert designed fuzzy classification system was developed. It takes the same input from the vision system and outputs the same classifications as the BCS.

Real time opponent classification attempts in robot soccer based on classifying an entire team's strategy have had minimal success. Steffen's team feature based method showed performance that was insignificantly better than guessing. The BCS will instead classify a team by classifying each individual player's behaviour. That way there is no need to attempt to generate models for every different type and combination of opponent strategies.

Lastly the output of the BCS could potentially form the input to an autonomous running commentary system. This could be useful as an entertainment system or in helping a domain expert, 'coach' the team.

Note that a major assumption is that their world state is similar to ours. This assumption is valid as both teams have the same global overhead view of the world.

4.1 Experimental Setup

The classification system is tested on the existing robot soccer system. The results of the classification system during testing are compared to the behaviours that the MAPS planning system is sending to the robots. The system is therefore classifying itself and so the ground truth is the MAPS assigned behaviours. Note that even though the input is the currently assigned behaviour by MAPS, the robots however may then take some time to begin executing the behaviour or may even be unable to execute it. Therefore there is a potential lag between receiving the desired behaviour and execution of the behaviour that is inherent in the training and results.

The Bayesian network was trained using the existing MAPS assigned behaviours as input over two minutes of playing time. The vision system and MAPS provide the input attributes and specified behaviour at 60 Hz. Two minutes of training at 60Hz provides 7200 training cases. During the training time the ball is moved around the field to all locations to ensure that all behaviours have the chance to be active. During this time the network will observe each of the behaviours several times.

The classification performance for both the Bayesian and Fuzzy classifiers was then tested against two further minutes of playing time. This presents completely unseen data to the classification systems for testing.

4.2 Bayesian Classifier Implementation

The experiment is designed to test whether a Naïve Bayesian classifier can determine robot soccer behaviour from a set of simple observations of field attributes. The experiments are run on the existing RoboRoos system. The observations of the field attributes come from the existing RoboRoos vision system. The classification of behaviour can be learnt or tested from the behaviour that is being specified for each robot by the RoboRoos MAPS (Multi-Agent Planning System).

The Bayesian inference engine Netica (GUI) [Norsys, 2003] was used to build the network as shown in Figure 7. This picture shows the state of the network before learning where all behaviours are considered equally likely. The behaviours are described in Table 1. These are the behaviours specified by MAPS, with the exception of the Transition behaviour which was added to capture the behaviour between specific soccer playing roles. The objective of the experiment is to compare the performance of the classifier against the known MAPS assignment of role.

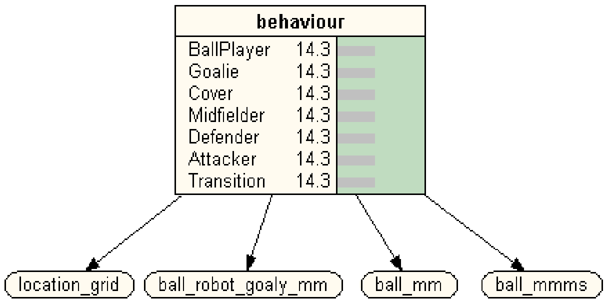

Figure 7: Naïve Bayesian Network for classifying the core behaviours in soccer based on the shown attributes. The input attributes are discretised into appropriate groups. The Netica network drawing package was used for building the network.

Table 1: Table showing the core behaviour list and a description of each.

Behaviour	Description
BallPlayer	Interacts with the ball. Could be acquiring, dribbling or kicking the ball.
Goal Keeper	Covers direct shots on the goal. Also saves shots. Is close or inside the goal.
Cover	Covers direct shots on goal. Is located a within a few robot widths from the goal.
Attacker	For example screening the defence or BallPlayer, waiting in the forward half for a pass or a loose ball.
Defender	For example zone control, marking a player or intercepting a pass.
Midfielder	Maintaining proximity to the centre of the field but also potentially covering the goal.
Transition	Moving at high speed around the field.

The input to the classifier is based on easily obtained attributes from the RoboRoos vision system. The input attributes are:

- MAPS grid location (based on a 15 × 12 grid to cover the field),
- goal coverage (*y* distance to the centre of our goal),
- the distance between the robot and the ball, and
- the velocity of the robot towards the ball.

Each of these attributes is currently available, or easily derived from, the RoboRoos vision system.

4.3 Bayesian Classifier Results

The classification system using this type of learning and network currently achieves ~ 84% correct classifications, 15% unknown classifications and less than 1% false positives. This is using a classification confidence threshold of 70%. Those robots that have no behaviour with a probability over the confidence threshold are given an unknown classification.

Table 2 shows the confusion matrix for the BCS. The matrix shows that the bulk of classification problems revolved around the attacker and, to a lesser degree, defender types. These behaviours are the least specific with respect to the input attributes. They can occur over a large range of possible positions, and do not bear a distinct relationship to the ball or the goal.

The current classification system as it stands is already useful to the MAPS planning system. For example it is useful to know who the opponent's BallPlayer is so they can be screened from reaching the ball. The current system assumes (unrealistically) that the closest opponent to the ball is the opponent's BallPlayer.

4.4 Future work for the Bayesian Classifier

Due the noise inherent in the visual input attributes the output of the classification system will be filtered. This will help reduce the number of unknowns by filtering out short term drops below the classification confidence. This will first be attempted using a Kalman filter. Another attempt will be to filter by providing biased prior probabilities for the behaviours in the network. As a robot continues to execute a behaviour over time the prior probability of it having this behaviour will be increased.

Determining the core behaviour of each opponent is the first stage for the Behaviour Classification System. In the second stage the system will break these down further into their sub behaviours. For example, the opponent that is determined to have the BallPlayer behaviour will be further classified to determine which of the following sub behaviours it is executing.

- Acquiring the ball.
- Shooting the ball at our goal.
- Passing to a team member.
- Dribbling the ball to another location.
- Kicking the ball so as to clear it from a zone.

The methodology of layering the classification process is preferred for robustness as incorrect classifications can be traced more easily than in one large network. It will also be necessary to retrain only one smaller network.

Table 2: Confusion matrix illustrating the classification performance of the BCS. The headings on the left show the behaviour specified by MAPS, while the headings across the top show the classification by the BCS. The percentages indicate the number of times that classification matched the specification, including confused classifications.

	Unknown	Goal Keeper	BallPlayer	Cover	Midfielder	Defender	Attacker	Transition
Unknown	0%	0%	0%	0%	0%	0%	0%	0%
Goal Keeper	1%	99%	0%	0%	0%	0%	0%	0%
BallPlayer	20%	0%	77%	0%	0%	0%	3%	0%
Cover	6%	0%	0%	94%	0%	0%	0%	0%
Midfielder	6%	0%	0%	1%	92%	0%	0%	0%
Defender	34%	0%	0%	0%	0%	62%	3%	0%
Attacker	73%	0%	5%	0%	0%	2%	19%	1%
Transition	0%	9%	10%	3%	2%	2%	24%	50%

Table 3: Confusion matrix illustrating the performance of an expert designed fuzzy classifier. The headings on the left show the behaviour specified by MAPS, while the headings across the top show the classification by the fuzzy classifier. The percentages indicate the number of times that classification matched the specification, including confused classifications. The performance of the fuzzy classifier is similar to that of the Bayesian classifier (shown in Table 2.).

	Unknown	Goal Keeper	BallPlayer	Cover	Midfielder	Defender	Attacker	Transition
Unknown	0%	0%	0%	0%	0%	0%	0%	0%
Goal Keeper	0%	100%	0%	0%	0%	0%	0%	0%
BallPlayer	2%	0%	80%	2%	4%	2%	1%	9%
Cover	1%	0%	1%	78%	0%	12%	0%	8%
Midfielder	0%	0%	4%	15%	54%	0%	1%	25%
Defender	3%	0%	1%	1%	2%	74%	4%	16%
Attacker	15%	0%	6%	0%	10%	6%	38%	25%
Transition	13%	3%	32%	1%	0%	0%	1%	51%

4.5 Expert Designed Fuzzy Classifier Implementation

An expert designed fuzzy behaviour classifier was developed. It uses input from the vision system and outputs the same classifications as the Bayesian Classifier. The input attributes are:

- proximity of the robot to the ball,
- proximity change of robot to the ball,
- velocity of the robot,
- velocity of the ball,
- goal coverage,
- relative heading of the robot to the ball,
- relative heading of the ball from the robot,
- relative velocity of the robot and the ball,
- core regions (Defending, Midfield, Attacking).

The inputs are fuzzified using typical terms such as close, far, same and different. Each behaviour is given a possibility by averaging the membership of multiple appropriate attributes. For the behaviour to be active this possibility must be greater than a confidence threshold of 85%. In this fuzzy classifier there is the possibility of multiple behaviours being active. For example a Ballplayer attempting to acquire the ball is moving a high speed therefore will be likely to have a high membership in the transition behaviour. The behaviours have a predetermined ranking of priority and importance. Only the behaviour with the highest ranking is considered active for these results.

Note that the extra inputs as compared to the Bayesian classifier enable the fuzzy classifier to determine whether the BallPlayer is acquiring, dribbling or kicking the ball.

4.6 Expert Designed Fuzzy Classifier Results

Table 3 shows the confusion matrix for the fuzzy classifier. The results are similar to those of the Bayesian classifier. In the fuzzy classifier there is a higher level of classifications for the transition behaviour and a lower level for the unknown classification. These classifications are interchangeable when the robots are moving at high speed as the behaviour could also be considered unknown.

The fuzzy classifier also had problems classifying the Attacker behaviour, again demonstrating the difficulty in classifying the least specific of the behaviours.

5 Modelling and Prediction

The performance of the Behaviour Classification System is best understood in the context of the next stage of the opponent prediction system. The Behaviour Modelling and Prediction System builds and stores the models of how the opponents perform their behaviours and predicts their future behaviours based on the models and the opponent's current behaviour. It is separated into two similar but separate sub systems; the Behaviour Modelling System and the Behaviour Prediction System.

The Behaviour Modelling System (BMS) will take as an input from the Behaviour Classification System the opponent's behaviours and incrementally build a model of how the opponent's perform their behaviours. Due to the probabilistic nature of the BCS the opponent behaviours used for learning already have a confidence measure associated with them. The probability for each behaviour is a direct confidence measure. This can be used as a multiplication factor for adding new data to the current model. The confidence measure can also be used as a threshold for rejection of data from the learning process.

A 2D probability distribution will model the way in which opponents perform their behaviours. It will model not only which behaviours are active but also the goal states for the behaviours. Which behaviours are active will depend on the state of the game in terms of defensive and offensive player behaviours. The goal state for each behaviour is dependent on features in the environment including the positions of the objects on the field. A discrete or continuous distribution will be appropriately chosen to map each behaviours goal state.

The 2D probability distribution is a simple, discrete representation that is built incrementally and will be computationally fast. It will be possible for a domain expert to interpret the meaning of the distribution. The distribution will be able to handle an input that is uncertain about the opponent's current behaviour. It will also represent the uncertainty in the model of the opponent's behaviours. The models will be robust to 'dirty data' due to the inherent fading effect of noise in probability distributions. Another by product of this fading effect is that the models will adapt if the opponents change their behaviour parameters during a game.

The main purpose of the BMS system is to provide the prediction system with a resource in an appropriate format that it can use for prediction. The output of this system will be a probability distribution across the field of play that will indicate where parameters of the behaviours are likely to be performed based on inputs relevant to the behaviour.

The Behaviour Prediction System (BPS) will take the current behaviour classification and the model of the opponent's behaviour to reason about the opponent's current behaviours and the goal of the opponent's current behaviours. The output of the prediction system will be a 2D probability distribution across the field of play that will represent the predicted goals of the behaviours. This 2D probability distribution will be formed by combining (ANDing or ORing) multiple probability fields that represent the different attributes. The peaks of these fields will represent the BPS's best prediction guess. One field will be from the classification system representing their current behaviour, the second will be the model of the previous goal location for the behaviour.

The coordination between multiple agents is not represented explicitly by this system as the probability distributions represent the individual behaviours of each agent. Coordination is only represented in the modelling of which behaviours are active dependent on the state of the game. The interaction between these behaviours is not explicitly modelled.

6 Conclusion

Previous research in this domain has been on multi-agent coordination and navigation in the highly dynamic and multi-agent domain of the F180 league of RoboCup. This paper has indicated the future research direction for a robot soccer team based in the F180 league. It will address the inaccessible nature of the opponent's strategy. The first stage of the research is to classify and recognise the opponent's current behaviour using a Naïve Bayesian Classifier. Initial results indicate that this is a realistic and worthy area of research. They also indicate that the Bayesian classifier has similar performance to an expert designed fuzzy classifier. These results were shown in the form of a confusion matrix. The second stage is to model the features of the way in which the opponents perform their behaviours and then to predict them into the future. Lastly these predictions will be integrated into the current multi-agent planning system. This future research will enable the current planning system to treat the opponent robots not as only a set of obstacles but as agents that have their own preferences on how to achieve their own (conflicting) goals.

Acknowledgements

The authors would like to acknowledge the work of the 2003 RoboRoos team that included Isaac Linnett, Leighton Carr, Douglas Turk and Joanna Wu.

References

[Ball *et al.*, 2003] David Ball, Gordon Wyeth, et al. *Design of a Team of Soccer Playing Robots*. Autonomous Minirobots in Research and Edutainment (AMiRE), Brisbane, Australia, 2003.

[Browning, 2000] Brett Browning. Biologically Plausible Spatial Navigation for a Mobile Robot. PhD dissertation. *Department of Computer Science and Electrical Engineering*, University of Queensland, 2000.

[Cantu, 2000] Francisco Cantu. Reinforcement and Bayesian Learning in Multiagent Systems: The MACS Project, Center for Artificial Intelligence, ITESM, 2000.

[Duda and Hart, 1973] Richard Duda and Peter Hart. *Pattern Recognition and Scene Analysis*. New York, Wiley, 1973.

[Florencio, 1998] Christophe Costa Florencio. Rats on Film. Utrecht, Rudolf Magnus Institute, 1998.

[Han and Veloso, 1999] Kwun Han and Manuela Veloso. *Automated Robot Behaviour Recognition Applied to Robotic Soccer*. Proccedings of IJCAI-99 Workshop on Team Behaviours and Plan Recognition, 1999.

[Kitano *et al.*, 1995] Hiroaki Kitano, Minoru Asada, et al. *RoboCup: The Robot World Cup Initiative*. IJCAI-95 Workshop on Entertainment and AI/ALife, 1995.

[Norsys, 2003] Norsys. Netica Application, Norsys Software Corp, 2003, www.norsys.com, [17th of November, 2003]

[Pazzani and Billsus, 1997] Michael Pazzani and Daniel Billsus. *Learning and Revising User Profiles: The Identification of Interesting Web Sites*. Machine Learning 27, 1997, Kluwer Academic Publishers.

[Pearl, 1988] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufman Publishers, 1988.

[Riley, 1999] Patrick Riley. Classifying Adversarial Behaviours in a Dynamic, Inaccessible, Multi-Agent Environment, Carnegie Mellon University, 1999.

[Russell and Norvig, 2003] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach, 2nd Edition*, Prentice Hall, 2003.

[Steffens, 2002] Timo Steffens. Feature-based Declarative Opponent-Modelling in Multi-Agent Systems. Masters. *Institute of Cognitive Science Osnabruck*, 2002.

[Tews, 2002] Ashley Desmond Tews. Achieving Multi-Robot Cooperation in Highly Dynamic Environments. PhD dissertation. *School of Computer Science and Electrical Engineering*. Brisbane, University of Queensland, 2002.

[Wyeth *et al.*, 2002] Gordon Wyeth, David Ball, et al. *UQ RoboRoos: Achieving Power and Agility in a Small Size Robot*. RoboCup 2001, 2002, Springer-Verlag.

[Wyeth *et al.*, 1999] Gordon Wyeth, Brett Browning, et al. *UQ RoboRoos: Preliminary Design of a Robot Soccer Team*. Lecture Notes in AI: RoboCup '98, 1604, 1999.

[Wyeth *et al.*, 2001] Gordon Wyeth, Ashley Desmond Tews, et al. *UQ RoboRoos: Kicking on to 2000*. RoboCup-2000: Robot Soccer World Cup IV, 2001, Springer-Verlag.