Modelling the positioning of players in a five-a-side Football game

Amit Shah - 0904778

Abstract

Five-a-side football is a common sport, but has seen little development in terms of strategies to be played, in particular with player formations. In this paper we take a look at the Empirical Modelling approach to building a model that allows for the interaction of players, to be able to view their optimal position in the game.

1 Introduction

Five-a-side football is a variation of the classic game, in which there are only 5 players on each team. The game itself is popular with many people, largely because less people are needed to play the game, making organisation easier. The problem however, is that many teams attempt to use classic strategies from the original game, which do not necessarily apply to the smaller version of the game. Due to the lower amount of players involved, and the smaller pitch, there is a lot more focus on individual abilities, and very little on teamwork.

Typically in many sports, a blackboard is used to show how players should move during a game, depending on the circumstances, such as ball position and the positioning of other players. This is sometimes an ineffective solution, as it quickly becomes difficult to be able to visualise the state of the game.

The focus of this paper will be to provide a demonstration of a model, that allows for a more accurate visualisation of the field of play. This model is built on the techniques of Empirical Modelling, utilising many of the benefits (and drawbacks) of such an approach. The model itself will also allow for a much more modern visualisation of how players should react and move in a game of five-a-side football.

2 Related Work

2.1 Empirical Modelling Approach

Empirical Modelling (EM) offers a unique perspective on being able to interact with models. Trope and Liberman (2010) spoke about being distant from a problem allows for the development of much more creative solutions. In this respect EM allows for the ability to develop abstract models, which can often be used to find innovative solutions. In creating a model, the users will be able to investigate approaches and strategies to a five-a-side game of football, without the need for teams and players to be present.

The principles of Empirical Modelling are based on Observation, Dependency and Agency (Russ, 1997), and this is most suited to the task in hand. The players in a game of football are dependant on each other, in their positioning and their actions, while each plater must be able to react on their own accord if needed. Building a play diagram with Empirical Modelling techniques allows for a user to view the actions of the players, and interact with them also.

While these ideas could indeed implemented in many standard languages, they do not provide the simplicity involved when building a model defined by interactivity. JSEden[1] is an EM tool that is quickly increasing in its abilities in developing interaction based models. Tools such as Java could be suited to building the independence of the models agents, but is lacking in its

ability to easily define the dependencies involved with a model such as this.

2.2 Player Positioning

Reis et al. (2001b) Talked about building multi agent systems that are well balanced between reactivity and social deliberation. It was mentioned that the most effective systems would mix strategies depending on the situation. This allowed them to adapt very quickly to new situations. In the case of this model, most of the agents (the players) will be dependant on (and therefore reacting to) the other players in the model. It will be important to remember, that to make an effective simulation of this game, that the players have the ability to make decisions themselves, based on their given information. Reis et al. (2001a) presented a similar idea in that teams will have a global strategy that each member will follow, but each member will ultimately be left with the ability to perform role exchange, and make decisions dynamically. This is relevant to the game of football, as often player will have to move from their current position, adapting to roles that they do not usually participate in. For instance if a player in front of a defender is not within their normal area, the defender would be required to move forward and assume a position that would provide support to the surrounding players.

In game of five-a-side football, there are lots of different formations a team can take, some offer more fluidity with player movements than others (Pitch-Invasion, 2013). This needs consideration for the model, as user may wish to see how a team should react and play through a particular formation.

3 Model Description

3.1 LSD Description

An LSD (Beynon, 1986) account was written for this system, to save space an example is given for only one player in the system, however it can be generalised to all the players in the model.

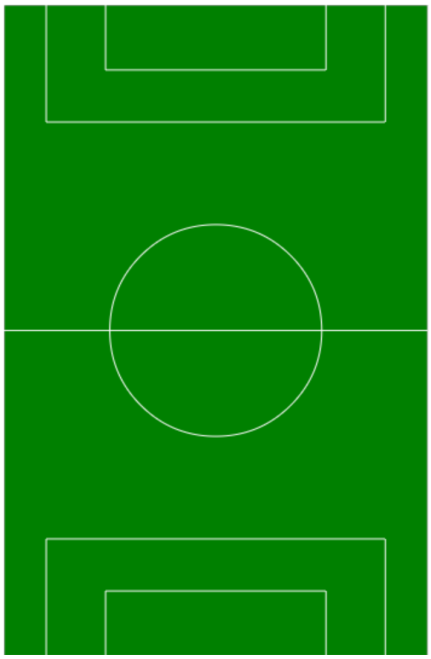

Figure 3.2: The Pitch

3.2 The Model

The model comprises of several items, the first of which is the pitch (Figure 3.2). The pitch itself is a background for the game, but it is important to model properly, as a player positioning can be dependant on the pitch's layout. For instance it would inappropriate for a player to move of the edge of the pitch. It is also useful for players to be placed in positions, not just relative to other team members, but dependant on those team members locations on the pitch (such as within penalty box).

This model only focuses on four of the five players in a team, this is due to the fact a goalkeeper rarely leaves their position, and has little effect on the positioning of other players. As can be seen in Figure 3.2 the players were placed in a diamond formation, this was chosen because the diamond formation allows the demonstration of the players movements, with more fluidity than a simple 2-by-2 formation. Each player is colour coded based on their position, and a key is provided to the right of the pitch. Images were attempt to be used, but they caused problems when trying to drag from one position to another, so

```
agent player() {
        state
                //Current location of the player
                (x,y) current_position = |Coordinates|

                //Determines if the player is being moved
                (bool) being_moved = false
        oracle
                //The pitch boundries
                (x1,x2,y1,y2) pitch = |Rectangle|

                //The other players on the pitch
                (player) players = |Player|
        handle
                //The football, changeable when in contact
                (football) ball = |Coordinates|
}
```

Figure 3.1: The LSD Account for a Player

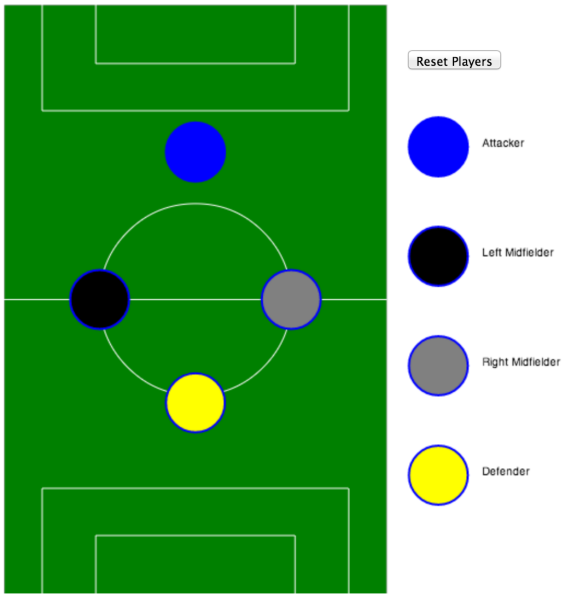

Figure 3.3: The pitch with the players in their starting positions

circle shapes were used instead, to provide the same functionality. It may be useful for a user to be able to reset the players into a default position, so a reset button is provided for that functionality, just above the key.

3.3 Agent Rules

When a player is dragged from a position to another, the model will update the locations of all the other players in the model. The majority of the time this is done by a simple translation, moving a pixels in the x direction, will cause all players to move by a pixels in the x direction. However, a rule was added to slow down the movement of the defender at the back of the pitch. This is identical to a real life game, in which a defender rarely moves forward to much, in order to remain ready for the other team to move forward. When moving backwards, the speed at which the three forward players move is faster than the defender, again this is similar to real life as the three players will attempt to catch up with the opposing team, and join the defender in protecting the goal.

There are simple rules in place to stop each

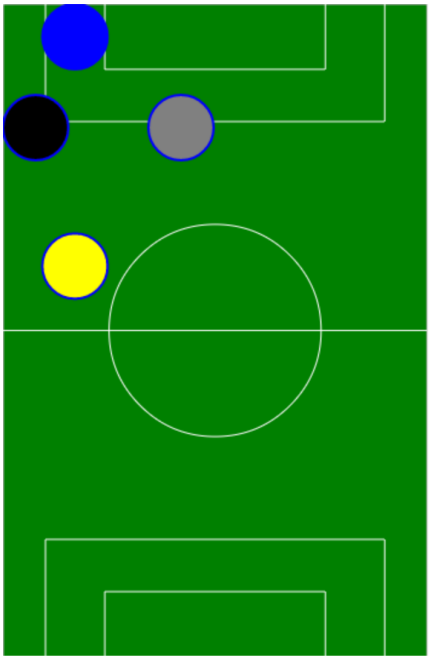

Figure 3.4: The pitch with the players moved towards the edges of the pitch

player from moving off the edge of the pitch (as shown in Figure 3.3, which would be unrealistic if not implemented. On top of this, each midfield player will not move past the other midfield player on the other side, instead remaining in a position in which they can help that player.

There are also rules which force the midfield players to not move past the inner box at the top of the pitch, it is not these player's responsibility to score a goal, only to provide support to the attacker, and therefore they shouldn't need to move any further forward. Likewise when moving to the bottom of the pitch, the striker will not move past the top of the lower box, and the midfielders will not move past partway through the inner lower box.

4 Evaluation

This model provides a start into modelling a five-a-side football match, however there are some issues. The agents currently do not have the ability to react on their own accord, they are very reliant on user in ut from the mouse, before any movement is made. As such the model only shows where player's ideal positioning on the pitch, relative to the other players, ignoring things such as free/empty spaces. It may be difficult to specify rules for the use of things like free space, although this is not limited to Empirical Modeling, as it is also a problem with attempting to use a rule based approach.

It is difficult to model the relationships between the players as being dependent on each other, in particular when there are a large number of cases, in which a simple dependency is not appropriate (such as a player being at an edge of the pitch). In this case the relationships built using EM allowed for quick construction of the model, and the ability to view an abstraction of the situation, however attempting to build a more realistic model, would require a different approach. It is unclear right now as too whether or not this is simply just the need for new Empirical Modelling tools, or a more complex modelling construct, just as an Object Orientated Language. Using something like Java would allow an abstraction of the problem, while allowing the agents to behave semi-autonomously, however creating the dependencies would become complex.

5 Further Work

One way to extend this model would be to add various other formations. This is a good way to demonstrate the capabilities of the model, and provide further ways to interact with the agents. As the code currently stands, it could become very complex handling the different options and the rules associated with them.

The positioning of the ball is very important in a football game, often actions can be almost entirely dependent on the ball. One way to provide this functionality in the model, is to treat the ball in a similar manor to the player agents, and have the players react to movements of the ball, in relation to the other players o. the pitch.

Another useful feature for the model would be an automatic run through of plays, demonstrating how players should be moving, without the

need for direct interaction. This could be quite simple to add in, as the agents already follow the rules needed to adjust themselves.

References

WM Beynon. The lsd notation for communicating systems. *Computer Science Research Report RR87, University of Warwick, Coventry, UK*, 1986.

Pitch-Invasion. Tactics guide, Jan 2013. URL `http://www.pitch-invasion.com/players/tactics-guide/`.

L. Reis, N. Lau, and E. Oliveira. Situation based strategic positioning for coordinating a team of homogeneous agents. *Balancing Reactivity and Social Deliberation in Multi-Agent Systems*, pages 175–197, 2001a.

LusPaulo Reis, Nuno Lau, and EugnioCosta Oliveira. Situation based strategic positioning for coordinating a team of homogeneous agents. In *Balancing Reactivity and Social Deliberation in Multi-Agent Systems*, volume 2103 of *Lecture Notes in Computer Science*, pages 175–197. Springer Berlin Heidelberg, 2001b. ISBN 978-3-540-42327-0.

S. Russ. Empirical modelling: the computer as a modelling medium. *The Computer Bulletin*, 39(2):20–22, 1997.

Y. Trope and N. Liberman. Construal-level theory of psychological distance. *Psychological review*, 117(2):440, 2010.

Notes

[1] http://jseden.dcs.warwick.ac.uk/emile/