OPTIMAL OFFENSIVE PLAYER POSITIONING AND COLLABORATION IN A DIGITAL SOCCER GAME

By

Serguei Razykov
Bachelor of Science, Tashkent State University 1986

THESIS SUBMITTED IN PARTIAL FULLFILMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

In the
School of Interactive Arts and Technology

SIMON FRASER UNIVERSITY
Summer 2006

APPROVAL

Name:	Serguei Razykov
Degree:	Master of Science
Title of Thesis:	Optimal Offensive Player Positioning and Collaboration in a Digital Soccer Game

Examining Committee:

Chair: **Dr. Diane Gromala**
Associate Professor, Interactive Arts & Technology

Dr. Vadim Kyrylov
Senior Supervisor
Associate Professor, Interactive Arts & Technology

Dr. Belgacem Ben Youssef
Supervisor
Assistant Professor, Interactive Arts & Technology

Dr. Chris Shaw
External Examiner
Associate Professor, Interactive Arts & Technology

Date defended: June 30, 2006

ABSTRACT

Player positioning is critical in many sport games; we use soccer as the example. The results of this study will help to improve digital sports games technology.

In existing methods, the player calculates its desired position using current location of the ball and its own role in the team formation.

The existing methods have two disadvantages: neglecting the game dynamics and leaving behind some potentially good positions without consideration; the latter being the common shortcoming of decision tree algorithms.

The proposed approach is taking into account the dynamics by determining the available time horizon which limits the feasible area where the optimal position is located. To make sure that all potential alternative positions in the feasible area have been evaluated and considered, the Pareto optimality approach is used. As a result, the proposed method provides the opportunity to create an optimal dynamic formation for the whole team.

Keywords: Artificial intelligence; computer simulation; multicriteria decision making; Pareto optimality; RoboCup

To my family

ACKNOWLEDGEMENTS

I would like to thank many people whose contribution to this thesis is inestimable. First of all I am grateful to Dr. Vadim Kyrylov who encouraged me to take my graduate studies, shared and discussed his wonderful ideas, and rendered comprehensive support in all stages of my work on the thesis. Also, I would like to express my gratitude to all SFUnleashed project participants – Eddie Hou, David Brokenshire, David Bergman, Martin Greber and Daniel Wardzinsky who deeply examined a number of problems connected to this work and contributed to SFUnleashed soccer simulation team success. The work of Eddie Hou was especially valuable for me, since we explored different aspects of the same problem, and shared ideas and implementation techniques. Beside that, Eddie Hou shared results of his literature search and performed code validation for a number of functions and procedures and made many useful suggestions. My special appreciations for Geoff Brown, Brian Haubrick and all the remarkable people in Academic Computing Services in Surrey who spent their precious time helping me to solve number of technical problems. My colleagues in the School of Interactive Arts & Technology encouraged and supported me during my studies and also deserved my appreciation. Finally, I would like to thank my family for love, understanding, encouragement, and help which made this thesis possible.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 Background

This research is inspired by previous work in the SFUnleashed project. SFUnleashed was the only Canadian team in the soccer simulation league that participated in RoboCup World Competition and RoboCup American Open in 2003 and 2004.

The idea of robotic soccer was first introduced by Professor Alan Mackworth of University of British Columbia in 1992 (Mackworth, 1992). Independently, a group of researchers in Japan after serious investigation decided to launch international robotics competition. In 1993-1995 an official soccer simulator was developed and the first official RoboCup competition was held in 1997. Games involving either physical or simulated autonomous robots have been played in all competitions since then.

RoboCup uses soccer as the primary domain for research in Robotics and Artificial Intelligence. The main activity in RoboCup is international competitions and research conferences in three major domains: RoboCup Soccer, RoboCup Rescue, and RoboCup Junior. This study concentrates exclusively on the soccer domain.

During our interactions with representatives of the *Electronic Arts*, one of the biggest digital games production companies in America, we realised that rational behaviour is one of the main issues in digital sports games design and development. In this thesis, these issues are addressed by using simulated soccer as a test bench. In particular, I rely on the experience with the SFUnleashed simulated soccer project. Some results of this project are outlined in Figure 1.1. SFUnleashed has demonstrated reasonably good performance, especially taking into account that in 2003 this simulated team was a newcomer.

The purpose of the SFUnleashed project was two-fold: (1) to develop approaches for implementing methods of artificial intelligence into robotics soccer and (2) to contribute to the development of digital sports games.

One essential behavioural feature is deciding by the given player where to go during the game when the ball is under the control by somebody else. I call this *player positioning*. On the average, this activity is taking about 90 per cent of the player time. Because of the critical importance of this feature, this thesis is dedicated to the development of methods for improved player positioning. From the main material it will become clear that some theoretical results of this study can be applied in different situations, and not exclusively to player positioning and in different digital sports games. However, in order to keep focused and limit the size of the thesis, I have deliberately narrowed its scope to the *offensive* player positioning. In other words, consideration is given only to situations when the ball is possessed by a team-mate. I hope to demonstrate the applicability of the theoretical models in situations other than offensive positioning in my future work.

So-called *multi-level player architecture* is one of such theoretical models. From the first steps of the SFUnleashed project we had noticed that there were difficulties in player behaviour. In most known from the literature (RoboCup 2000, 2001, ...) implementations of simulated soccer teams every simulation step each player makes a new decision. These decisions are based on the *current* perception information and some model of unobserved aspects of the current state (Russell&Norvig, 2003). Indeed, the environment state in simulated soccer game changes in the real time and a new state can significantly differ from the previous one. Abrupt changes in the simulated player's intentions are obviously counterproductive, as they, sometimes, result in hectic behaviour. These changes are especially noticeable when the perceived information about the world is imperfect due to the presence of random errors. This leads to the presumably false conclusion that the decisions in simulated soccer cannot be persistent in time and long-time planning is impossible.

This contradicts with what takes place in real-life soccer. Human players are normally acting according to some mental plan, having the time horizon up to several seconds. We would like to find a way to modeling this sort of persistent, robust behaviour.

I believe that some decisions can last for several simulation cycles if the environment state does not change or changes 'smoothly'. To model this, we want to know when the decision will change. Determining the time horizon for short-term planning is one of the key issues addressed in this thesis.

To achieve robustness in decision making I am making use of the improved layered agent structure.

The basic idea of layered agent structure itself is not new. Such RoboCup scholars as Peter Stone (Stone, 2000) introduced layered reinforcement learning and Kok and De Boer (Kok & Boer, 2002) also described a sort of layered agent structure. Nevertheless, I believe that this structure can be constructed in different ways. For instance, Kok and De Boer (Kok & Boer, 2002) constructed the layers as agent skills levels in the following way.

- Atomic actions. Atomic actions are the commands which a player can send to the server, like *turn* or *dash.*
- Low-level skills like *searching for the ball*
- Intermediate-level skills like *moving to a position*
- High-level skills like *intercepting the ball*

A skill in every level is a sequence of lower-level skills. This approach appears to be reasonable and proved to be effective in many cases. However, it does not guarantee that any of the higher-level skill sequence of actions will not be interrupted to start a new one.

The improved layered agent structure of a SFUnleashed player differs in that it was about the layers of decision making rather than player actions. The structure involved the following four decision levels:

- Strategic level. Long term plan for all players for the whole game. Strategy was implemented through formations.

– Tactics level. Short term plans for small groups of player to achieve a local goal like an offside trap. This level was not implemented in SFUnleashed as yet.

– Individual level. Individual short term plan of actions for a player is, for instance, to make a leading pass to a particular team-mate. This level was implemented through a persistent action plan with a particular duration. The duration was determined empirically, without proper theoretical analysis.

– Atomic level. Atomic action is an action with duration of one simulation cycle. In fact, atomic actions are the commands which can be sent to the server like *kick* or *dash*.

The experience with SFUnleashed raised many questions and some problems remained unsolved, especially with the soccer player behaviour in the offensive situations. Obviously without reasonably good implementation of such behaviour, the simulated soccer team would be hardly winning in the RoboCup competitions. In particular, it was unclear what the time horizon of an individual player level plan should be and how collaboration with the team-mates should be organized. This study addresses these and related issues. In particular, it raises the following research questions.

1. *What generic decision making framework should be used to achieve rational player behaviour that would be applicable to positioning?*
2. *How to balance rewards, risks, and costs while the player is deciding about its optimal position on the field?*
3. *How to determine the reasonable time frame for positioning planning?*
4. *How to limit the search space for the optimal position and achieve robustness of the player positioning behaviour?*
5. *How to achieve player collaboration with the proposed decision making framework?*

This research answers these questions using Multicriteria Decision Making Theory, prediction methods, and constructing the appropriate criteria for players' behaviour.

Figure 1.1 shows that in American Open 2003 tournament SFUnleashed won four of six played games, in the RoboCup 2003 SFUnleashed only lost one game in the first round and was just one point short to advance from the second round to the final stage. In RoboCup 2004 the team also advanced into the second round.

1.2 Overview of the chapters

Chapter two further elaborates on the background and overviews relevant information used in this research. First of all, a short description of the soccer rules is given. Soccer simulator presumably must implement these rules. Also, soccer strategy and tactics methods are described. These methods are the basis for implementing the rational behavior criteria and evaluation.

Secondly, the chapter provides information about the RoboCup research and educational initiative and the Tao of Soccer simulator. The RoboCup initiative provides a framework and a standard problem for research. The Tao of Soccer is used as the research tool providing more flexible and convenient environment for implementation of the proposed methods. The Tao of Soccer server physics description is provided as the basis for prediction methods proposed in Chapter three. Finally, I present the overview of the theory underlying Multicriteria Decision Analysis. This theory provides methods and approaches for finding the balanced solution with respect to many conflicting performance criteria.

Chapter three describes the main ideas and proposed methods of this research. I propose methods for determining the time horizon for planning player positioning and calculating the area for feasible positions. Also, the criteria for the positioning problem, the place of the problem in the multicriteria problem classification, and algorithms for finding the best compromise solution are described.

Chapter four presents statistical results and analyses and discusses the experimental teams' performance in different settings.

Finally, in chapter five, the research questions are revisited, the conclusion is made about the current research contribution, and future research directions are outlined.

Table 1.1 SFUnleashed in international competitions

AmericanOpen 2003, Pittsburgh, USA (http://www.cs.cmu.edu/~AmericanOpen03/results/simulation_r1.html, 2003)

	Group A	1	2	3	Wins	Losses	Draws	Points	Rank
1	UT Austin Villa	---	0:0	0:1	0	1	1	1	2*
2	Iranians	0:0	---	0:3	0	1	1	1	3*
3	SFUnleashed'03	1:0	3:0	---	2	0	0	6	1

Winner's Bracket

A1 - SFUnleashed	0	Helli-Amistres	2	Helli-Amistres	0	UvA Trilearn
B2 - Helli-Amistres	2					Winner's Bracket Champion
C1 - Aria	7	Aria	0			
D2 - Red Sky Koblenz	0					
A2 - UT Austin VIlla	0	UvA Trilearn	1	UvA Trilearn	1	
B1 - UvA Trilearn	24					
C2 - Caspian	0	Brainstormers	0			
D1 - Brainstormers	1					

Loser's Bracket

(from winners) SFUnleashed	5	SFUnleashed	0	BrainStormers	0; 3	Brainstormers	2	Brainstormers
(from winners) Red Sky Koblenz	2	(from winners)Brainstormers	1			(from winners)Helli-Amistres	1	Loser's Bracket Champion
(from winners) UT Austin Villa	0	Caspian	4	Caspian	0; 1			
(from winners) Caspian	2	(from winners)Aria	1					

RoboCup 2003, Padua, Italy (http://www.uni-koblenz.de/%7Efruit/orga/rc03/, 2003)

Results from the first level group games

1	Brainstormes03	---	4:0 F L	17:0 F L	5:0 F L	8:0 F L	17:0 F L
2	**SFUnleashed03**	**:**	**---**	**2:0** F L	**4:0** F L	**4:0** F L	**2:0** F L
3	hana	:	:	---	0:0 F L	2:0 F L	2:0 F L
4	Robolog2k3	:	:	:	---	1:0 F L	0:0 F L
5	Avan	:	:	:	:	---	0:0 F L
6	VirtualWerder	:	:	:	:	:	---

Results from the second level group games Group A_2

	Group A_2	1	2	3	4	5	6
1	FC Portugal	---	5:0	5:0	2:0	1:0	16:0
2	Oxsy	:	---	0:4	0:5	0:3	2:0
3	**SFUnleashed**	**:**	**:**	**---**	**0:1**	**2:2**	**5:0**
4	Cyberoos	:	:	:	---	1:2	12:0
5	Zenit-NewERA	:	:	:	:	---	17:0
6	Amistres	:	:	:	:	:	---

RoboCup 2004, Lisbon, Portugal (http://www.robocup2004.pt/docs/pdfs/SoccerSimulation2D.pdf, 2004)

Round 1, Group H results

Rank	*Team*	*Points*	*Goal Diff*	*Goals Scored*	*Goals Conceded*	*Games*
1	RoboSina	9	22	24	2	3
2	**SFUnleashed**	**4**	**0**	**9**	**9**	**3**
3	Kshitij	4	-2	4	6	3
4	RoboLog2D	0	-20	0	20	3

Round 2, Group C results

Rank	*Team*	*Points*	*Goal Diff*	*Goals Scored*	*Goals Conceded*	*Games*
1	STEP	15	32	34	2	5
2	TsinghuAeolus	12	38	42	4	5
3	YowAI2004	9	12	17	5	5
4	**SFUnleashed**	**4**	**-23**	**4**	**27**	**5**
5	Impossibles	3	-16	7	23	5
6	hana	1	-43	2	45	5

2 LITERATURE REVIEW

2.1 Strategy and tactics of soccer

Soccer is one of the oldest sports games in the world. The first official soccer association was created in 1863 in England. Soccer is now considered the most popular sport on earth.

2.1.1 Rules of modern soccer

This section overviews only those parts of soccer rules which are of interest for simulation and research purposes. (www.fifa.com, Official site of FIFA, 2006)

A soccer game is played on a rectangular field about 100 by 64 meters. Two goals are placed on the opposite sides of the field at the centre of each goal line. The distance between the goal posts is 7.32 meters. Two teams play with a spherical ball with a circumference of about 70 centimetres. Each team consists of not more than eleven players, one of whom is the goalkeeper. A goal is scored when the ball completely passes over the goal line, between the goalposts. The team scoring the greater number of goals during a match is the winner. If both teams score an equal number of goals, or if no goals are scored, the match is drawn. A number of special situations are recognized in soccer. We are interested only in some of them:

- Offside. A player is in an offside position if he is nearer to his opponents' goal line than both the ball and the second last opponent; a player in an offside position is only penalized if, at the moment the ball touches or is played by one of his team, he is involved in active play
- Throw-in. A throw-in is a method of restarting play when the whole ball passes over the touch line (the side line of the field).
- Goal kick. A goal kick is a method of restarting play when the whole ball, having last touched a player of the attacking team, passes over the goal line outside the goal.

- Corner kick. A corner kick is a method of restarting play when the whole ball, having last touched a player of the defending team, passes over the goal line and a goal is not scored

Current rules of soccer open a wide range of action for teamwork and player collaboration referred as soccer strategy and tactics.

2.1.2 Soccer strategy

Strategy is a long-term plan of action to achieve the particular goal. This goal in soccer could be to win the game or not to lose the game. Strategy defines overall team behavioural pattern. In soccer, strategy is mainly achieved through formations.

A formation defines the players' roles in the team and their location on the field. The player roles are as follows:

- Goalie. The player who defends the goal - the only player in the team who is allowed to touch the ball with his or her hands.
- Defenders. The players located close to the goal that prevent the opponents from scoring. There are wing defenders and center defenders. Defenders mostly are situated in the *defensive zone* and create the *line of defense.*
- Midfields. The players located in the middle area of the field. Midfields support defenders in defense and forwards in attack. Also, they serve as a bridge between defenders and attackers. There are wing midfields and central midfields. Midfields are situated in the *middle zone* and form the *middle line.*
- Forwards or attackers. The players located near the opponents' goal. Forwards try to come closer to the opponents' goal and score goals. There are wing and center forwards. Forwards are situated in the *offensive zone* and form the *line of attack.*

Players act on the field according to their role in the formation. If there are more defenders in the formation the team plays a defensive game. In the opposite case, if there are more forwards in the formation the team plays an offensive game. Many formations were developed in the history of soccer. Usually, different formations are denoted as a set of three numbers representing number of defenders, number of midfields, and number of forwards. Some formations have their own names. For instance, 4-2-4, also known as "Brazilian" formation is the formation with four defenders, two midfields, and four forwards.

Figure 2.1 Early 3-2-5 "WM" offensive formation (Beim, 1977)

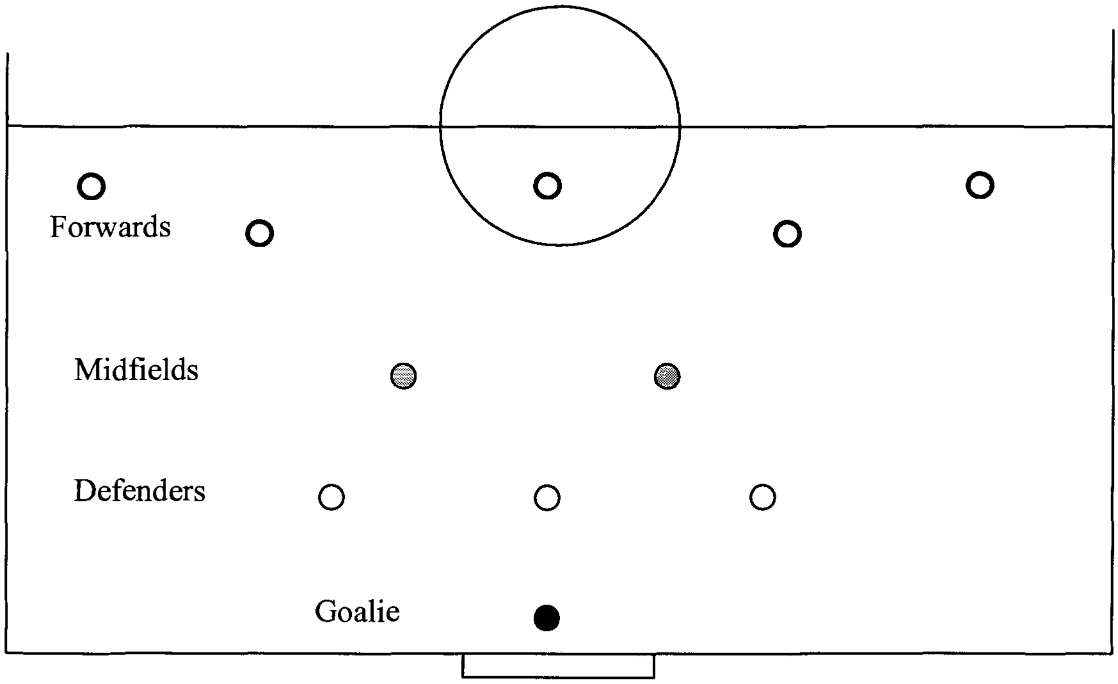
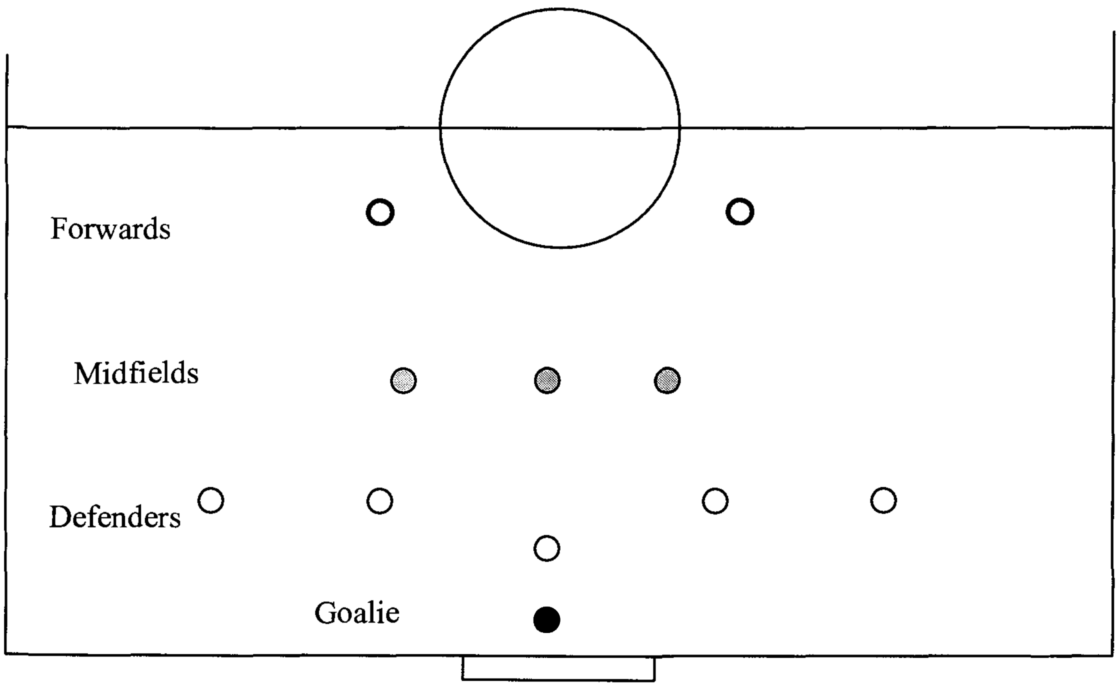


Figure 2.2 5-3-2 "Catenaccio" or "Italian bolt" defensive formation (Beim, 1977)



Nowadays, balanced and flexible formations are widely used. This means that teams use one of the "balanced" formations 4-3-3 or 3-3-4 but can change them to an offensive formation when in attack and to a defensive formation while in defence. Peter Stone facilitated the idea of flexible formation in soccer simulation in the form of role exchange (Stone, 2000). The formations show that the soccer strategy is essentially all about positioning. The tactics of soccer are achieved through positioning, as well.

2.1.3 Soccer tactics

Tactics is a short term plan to achieve an interim goal and support the strategy. The interim goal can be to destroy an opponent's attack or to penetrate the defence. Different tactics methods are used in attack and defence. In attack, the tactics are as follows (Beim, 1977; Vogelsinger, 1973)

- Space. Space is extremely important in attack. Forwards must use the space between and behind the defenders. If a player has no space he or she most likely will not be considered as a potential pass taker or may lose the ball control in case of pass. The second aspect of the free space principle is keeping free space between the player and partners and/or the goal to be able to receive a pass.

- Attack depth and support. Forwards coming closer to the opponent's goal with the ball are attacked by defenders. It is always easier for a defender to intercept the ball than for an attacker to keep it. For this reason midfields must support the attackers. Quick short passes back and forth between midfields and forwards can disorient the defenders and create a chance to penetrate the defence.

- Penetration. The principle of penetration requires the players, especially forwards, to move as deeply as possible into the opponent's defence. Such moves lead to destroying the defence and chance to score a goal.

- Width. If all forwards were to concentrate in the middle of the field before the goal, defenders could easily outplay them. Supporting the width of attack causes defenders to move closer to the touch lines, thus opening up space for penetration.

The objectives of defence are to prevent opponents form scoring, regain possession of the ball, and initiate an attack. Defence can take several forms like man-to-man defence, zone defence, and combined defence. The tactics methods, though, are the same for all defence forms. They involve:

- Delay. This method is opposite to penetration in attack. When an opposing player gains possession of the ball the team needs time to restructure for defence. A defender must position himself to eliminate as many forward passing opportunities as possible to prevent the defence penetration.

- Support. While some players directly oppose the player with the ball the others must block other opposing players to eliminate passing opportunities. Sometimes this technique is referred as *marking*.

- Balance. Defending players must provide cover for as much space as possible which means that they must be distributed evenly across the field.

- Pressuring. Pressuring is an active defensive tactics. It can be thought of as attack in defence. The goal of pressuring is to restrict space for the attacking opponents. The pressuring players must keep as close as possible to the attacking opponents, remaining goal-side.

- Control. Defending player must maintain its role in the whole defensive structure.

- Offside trap. When an opponent is ready to make a forward pass, defending players can move in such a way that one or several opposing players will find themselves in offside position. This method requires full concentration and strong coordination since, if applied inaccurately, it can easily lead to a goal.

2.1.4 RoboCup: robotic soccer as a research tool. TAO of Soccer

"By the year of 2050, develop a team of fully autonomous humanoid robots that can win against the human world soccer champion team"

This is the motto of the RoboCup international research and education initiative as it is stated in the official RoboCup site (www.robocup.org, 2006).

2.1.4.1 RoboCup overview and research objectives

RoboCup Soccer consists of five leagues.

- Simulation league. In this league eleven independent artificial agents play as a team using computer simulation. Each player is a computer program. Players may communicate using a simulation server protocol but any direct communication outside the server is prohibited. The players get visual information from the server and send back commands representing their actions. Matches have two 5-minute-long parts.
- Small size robot league
- Middle size robot league
- Four-legs robot league
- Humanoid robot league

An interested reader can obtain more information about real robot leagues in the official RoboCup website mentioned above.

While the last four leagues of real robots deal with many technical problems like mechanics and sensors, the simulation league mainly develops methods for rational player behaviour. This research falls into the domain of computer soccer simulation.

RoboCup provides a standard framework and standard problem for research in AI. The objectives of the research are real-time sensor development, rational behavior, strategy acquisition, learning, real-time planning, multi-agent systems, collaboration, context recognition, vision, strategic decision-making, motor control, intelligent robot control, and many more. In this research I concentrate on rational behaviour and collaboration.

2.1.4.2 Simulated soccer and digital sports games

Soccer inspired many digital computer games, such as FIFA Soccer by Electronic Arts. By the multi-player nature of the game, the human player is unable to control all the characters in his own team. This means that several team-mates are computer-driven non-player characters. They must be designed to make the impression of real independent soccer players. In simulated soccer each player is an independent computer program not controlled by a human. This feature connects simulated soccer and digital sports games. Methods developed for simulated soccer can be successfully applied to digital sports games.

2.1.4.3 The simulation environment - TAO of Soccer

The simulation server used in RoboCup is a sophisticated tool intended to simulate a real soccer game as closely as possible. It is written in C++ and operates under Linux. The server brings in some random errors into visual information and players' actions. Also, the visual information is restricted by some view angle and distance. These features make some research tasks difficult. For instance, if problems with positioning are revealed, it is hard to say whether they are a result of a poor positioning algorithm, a wrong world model, or inaccurate visual information. This can only be determined with sufficiently long simulations. Unfortunately, thousands of games are required for gaining reasonably precise results, which prolongs experiments too much. For this reason, another soccer simulator, TAO of Soccer, was chosen for conducting this research.

TAO of Soccer was developed by Yu Zang in 2001 as an alternative to the RoboCup soccer simulator (Zang, 2005). TAO of Soccer has all the features of the RoboCup simulator but it is written in Java, has a simpler client-server protocol, and can be used both as a simulator and an interactive game environment. I used it as a simulator only. Using TAO of Soccer gave us the opportunity to use full information about the environment and concentrate on problems of rational behaviour. Actuator random errors are the only source of randomness in TAO of Soccer.

Table 2.1 Differences and similarities between RoboCup simulator and TAO of Soccer

Characteristics	RoboCup simulator	TAO of Soccer
Environment	Dynamic	Dynamic
State change	Real time	Real time
Environment information	Incomplete	Complete
Actuator error	Present	Present
Information errors	Random for visual information. Random for players' actions	No errors in visual information (except small rounding errors). Random for players' actions
Control	Distributed	Distributed, human interaction possible

Table 2.1 shows that the main difference between the RoboCup simulator and TAO of Soccer is that the information available to the artificial player is complete and precise. This substantially reduces the number of simulation runs that are necessary for evaluating different player behaviours.

2.1.4.4 TAO of Soccer server physics

For determining the time horizon available for planning the player behaviour, we need to be able to predict situations on the field rather precisely. To construct prediction algorithms, we should use some laws of physics. The TAO of Soccer server simulates physics as follows (Zang, 2005):

- Soccer field is rectangular. The touch line is 100 meters long and goal line is 65 meters long. The distance between goalposts is 8 meters. Each point p on the field is represented by rectangular Euclidean coordinates (x,y), where x is measured along the touch line and y is measured along the goal line. The center of the field is set to (0,0); Y axis goes up and X stretches to the right.
- The players and the ball are represented by circles and are the only dynamic objects of the environment. The motions of the dynamic objects are simulated stepwise every 50 milliseconds.

– Motion of a player is calculated every simulation step as follows

- $p_i = p_{i-1} + v_{i-1}$
- $v_i = v_{i-1} + a_{i-1}$
- $a_i = FORCE * K_1 - v_{i-1} * K_2$

where i is current simulation step number, p_i is player's current position, p_{i-1} is player's previous position, v_i is player's current velocity, v_{i-1} is player's previous velocity, a_i is player's current acceleration, a_{i-1} is player's previous acceleration.

FORCE is set by the client (player agent program). Coefficient K_1 is the force factor. K_2 is the friction factor. They are calculated by setting constants *MaxSpeed* and *TimeToMax*.

- $K_1 = MaxSpeed * TimeStep^2 / (TimeToMax * MaxForce)$;
- $K_2 = MaxForce * K_1 / (MaxSpeed * TimeStep)$;

MaxSpeed is the maximum speed the player can reach. *TimeToMax* is the amount of time a player needs to reach full speed without friction. *TimeStep* is the length of one simulation step, defaulted to 0.05 sec.
MaxForce is the maximum force a player can apply, defaulted to 100.

– Motion of the ball is calculated every simulation step as follows:

- $p_i = p_{i-1} + v_{i-1}$
- $v_i = v_{i-1} + a_{i-1}$
- $a_i = KICKFORCE * K_1$ and $v_i = 0$ if kicked by a player
- otherwise $v_i = -FRICTIONFACTOR * v_{i-1}$

K_1 is the kick force factor. It is calculated as:

- $K_1 = MaxSpeed * TimeStep / MaxKick$;

MaxKick is the maximum kick force a player can apply, defaulted to 100.

– When there are several players very close to the ball, one of them is randomly chosen as the controller of the ball. The controller of the ball can kick the ball by sending the *kick* command, or he can dribble the ball by sending normal *drive* command. Hence, the acceleration is reduced as:

- $a_i = \{FORCE * K_1 - v_{i-1} * K_2\} * DRIBBLEFACTOR$

where *DRIBBLEFACTOR* is the maximal dribble force factor when a player is dribbling.

In order to reflect unexpected movements of objects in real world, TOS adds random error to the movement of objects and to the parameters of commands.

- As for player movements, noise is added as follows:
  - $A_1 = (FORCE * K_1 - V_0 * K_2) * (1 +/- RandomFactor);$
- As for the free ball movement, noise is added as follows:
  - $A_1 = -FRICTIONFACTOR * V_0 * (1 +/- RandomFactor);$
- When the player kicks the ball, noise is added to the kicking direction as follows:
  - $KickDir_1 = KickDir_0 +/- KickRandom;$

2.2 Aspects of the player rational behaviour. Why positioning?

We want our player agents to behave rationally. It is normally believed in the AI community that an ideal rational artificial agent is defined as follows.

"For each possible percept sequence, an ideal rational agent should do whatever action is expected to maximize its performance measure, on the basis of the evidence provided by the percept sequence and whatever built-in knowledge the agent has."

(Russell&Norvig, 2003, p.36)

An *agent* is defined as some entity that perceives its environment through sensors and acts upon this environment trough some actions (Russell&Norvig, 2003).

In the simulated soccer environment an agent perceives the following information

- its own position and the orientation of its body and its parts,
- the positions of other players,
- the position of the ball.

An agent can perform the following actions

- kick the ball

- move (dash, turn)
- talk (send messages).

These simple actions can be combined in more complex actions which make sense from the soccer point of view as follows:

- position itself (move)
- chase the ball (move)
- pass (kick)
- shoot(kick)
- dribble (kick +move).

Russell&Norvig outline four basic types of agent (Russell&Norvig 2003).

- Simple reflex agents
- Model-based reflex agents
- Goal-based agents
- Utility-based agents

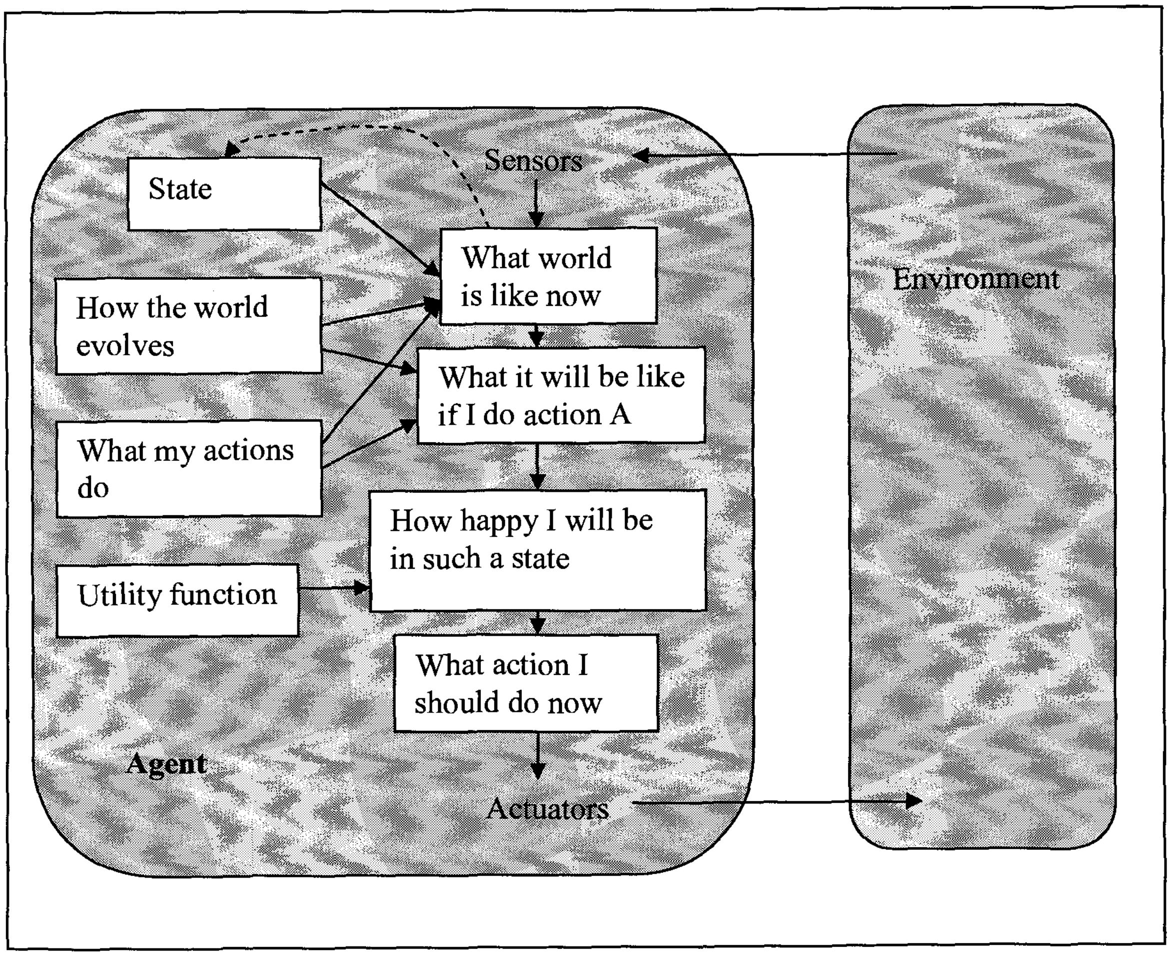
Simple reflex agents select actions using some condition-action rules according to current perception.

Model-based reflex agents maintain some model of the world. They also use some condition-action rules but act according to the current state of world using an internal model.

Goal-based agents store goal information and the information about the results of possible actions in order to choose actions that achieve the goal. Goal-based agents reason about the future (Russell&Norvig 2003). The goals for an agent can be set manually by the designer.

Utility-based agent can set the goals for itself by defining a utility function. A utility function is a function that maps a state to a real number, which represents the associated degree of happiness (Russell&Norvig 2003). In other words, each simulation step, an agent must perceive the environment information, create possible plans of actions, and evaluate these plans according to the utility function.

Figure 2.3 Structure of a utility-based, goal-oriented agent



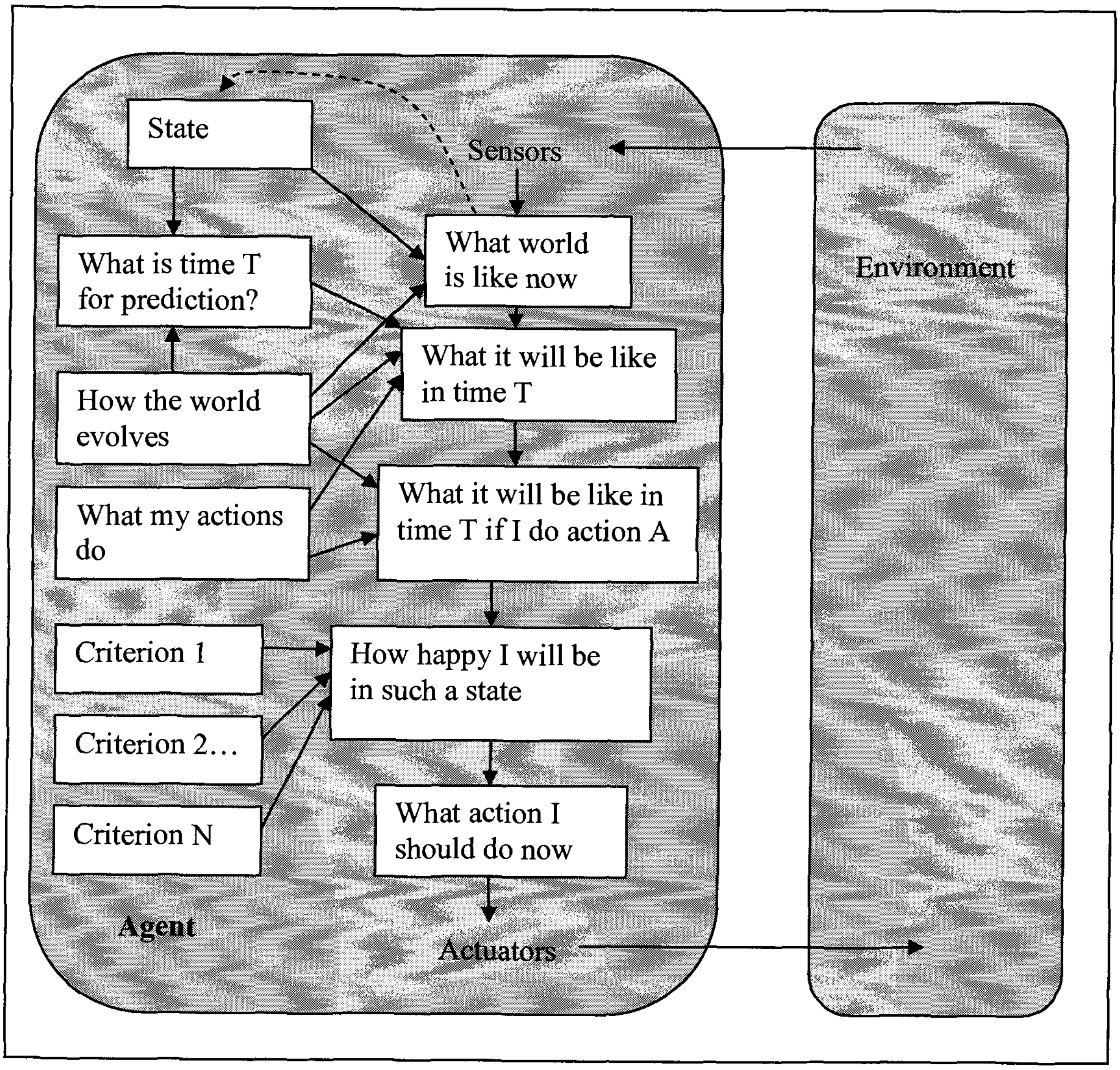
While dealing with the model in Figure 2.3, we must bring attention to two problems. Firstly, the environment state in soccer simulation is changing in real time. Since the plan of actions depends on the perceived environment state, this plan can change significantly from one simulation step to another. This renders some plans useless since any plan which lasts more than several cycles is unachievable because the state would change before the plan completion. For instance, passing the ball requires just one or very few simulation steps and the appropriate plan probably would not change significantly during the pass execution. On the other hand, implementing a positioning plan can take several dozens of cycles. It would be naïve to expect that the state would not substantially change during such period of time. Secondly, the state may be characterized by several parameters. Utility functions are normally used for evaluating states. However, it is not always possible to create a utility function properly mapping the environment state into a real number. In what follows, I will show that these problems were not addressed in existing player positioning methods.

To solve these problems, I will consider an enhanced structure of a utility-based agent. This structure involves the predicted state of the world and multi-criteria decision making.

So, why positioning? First of all, positioning occupies most of a player's time. More than 90% of the player's time is devoted to deciding where to go to and moving to this destination. Secondly, soccer strategy and tactics are mostly achieved through player positioning. A team in a soccer simulation is a multi-agent system which requires collaboration. Positioning, if executed purposefully, is the key to collaboration. At last, positioning requires the longest action plans compared to other player behaviors.

There is a conflict between the real time change of the environment state and the necessity to create long term plans for player positioning. This conflict has not been addressed in existing positioning methods. This study investigates this problem in a systematic way and provides the solution.

Figure 2.4 Enhanced structure of utility-based, goal-oriented agent which is using prediction and multicriteria decision making



2.3 Existing player positioning methods

In some existing player positioning methods, a soccer agent perceives the current situation and calculates its desired location on the pitch by taking in consideration the current or predicted location of the ball and its own 'home' position in the formation. Each player determines its destination as a weighted sum of these two points. In some cases, current positions of other players are also taken into account. This approach was implemented in some simulated soccer teams participating in the international RoboCup initiative, in particular, *FC Portugal* and *UvA Trilearn* (Kok & Boer, 2002) who were the winners in some worldwide and regional competitions.

An alternative approach can be found in the descriptions of the *CS Freiburg* middle-size robots team (2001) and the *CM-United* small-size robots team (Stone at. al, 1998). With this approach, the field is divided into small rectangles and each rectangle is evaluated against some utility function. This approach involves some multi-criteria evaluation similar to the simple weighting method for multicriteria optimisation problems described later. As both teams were world champions in their leagues, these positioning methods produce satisfactory results.

Nevertheless, I see some disadvantages of the existing positioning methods. First of all, common for both approaches, is that these two approaches neglect the game dynamics. When the game is in process, the ball is in motion almost all the time. Therefore, the calculated player position is a moving target too and often is too far away from the player. If the player cannot reach the target before the situation changes, it will waste its effort. For the first, weighed ball-home position approach, one more disadvantage is in the decision making method used. It is based on a decision tree with heuristic rules balancing the anticipated rewards and risks. In some cases, these conflicting criteria are even not explicitly specified by the creator of the decision making algorithm. This is leaving behind some potentially good target positions without proper consideration.

For the second approach based on fixed rectangular zones, another disadvantage is that the utility function cannot always be properly constructed. For conflicting criteria, mapping the multicriteria optimisation problem into single criteria optimisation problem may be inappropriate and give unexpected results. This effect was observed when a similar algorithm was implemented for the SFUnleashed team.

The proposed method eliminates these disadvantages by using a more elaborate prediction of the situation in combination with the multi-criteria decision analysis (MCDA).

2.4 Overview of Multi-Criteria Decision Analysis Theory

Real-life optimisation problems often require solutions which are characterised by several incomparable and often competing performance indicators, or criteria. Informally, the problem can be defined as a search for the optimal solution among a number of possible solutions characterised by several criteria. The Multi Criteria Decision Analysis theory is well developed by many authors and is applicable to many areas from economics to engineering. It is also called Multicriteria optimisation (Stadler, 1988; Ehrgott, 2005), Multiobjective optimization (Liu, Yang, Whidborne, 2003) or Vector optimization (Kolbin, 2003). Below I will describe the basics of the theory following the concepts provided by these authors.

2.4.1 Problem formulation

It is always possible to construct the criteria as assumed for minimisation, so formally the multicriteria optimisation problem can be formulated as the problem of simultaneously minimising the n criteria functions $x_i(p)$, $i = 1, 2, \ldots n$ where p is a variable vector from the space of vectors p called *decision space* F, or, find

$$\min_{p \in F} \left(x_1(p), x_2(p), \ldots x_n(p) \right) \tag{2.1}$$

In general, the problem does not have a unique optimal solution which means that we cannot minimize all the criteria simultaneously because of the inherent conflict. Nevertheless, we should find some solution which we will call optimal in the sense of the most suitable compromise.

We will call a set of accessible alternatives for the decision problem a feasible set $F_s \subseteq F$. We denote the space of vectors $c(x_1, x_2, \ldots x_n)$ as the *criteria space* C and the image of F_s under $X = (x_1, x_2, \ldots x_n)$ as $C_s \subseteq C$ - the image of the feasible set, or the feasible set in the criteria space.

2.4.2 Definitions

To introduce the concepts of non-dominated points and efficient solutions we need some definitions.

2.4.2.1 Relations

A *Cartesian product* $A \times B$ of two sets A and B is the set of all ordered pairs (a,b) where a is in A and b is in B. That is $A \times B = \{(a,b) \mid a \in A, b \in B\}$.

Let S be a set. A subset R of $S \times S$ is a *binary relation* on S. A binary relation R on S is called

- *reflexive* if $(s,s) \in R$ for all $s \in S$
- *irreflexive* if $(s,s) \notin R$ for all $s \in S$
- *symmetric* if $(s^1,s^2) \in R \Rightarrow (s^2,s^1) \in R$ for all $s^1,s^2 \in S$
- *asymmetric* if $(s^1,s^2) \in R \Rightarrow (s^2,s^1) \notin R$ for all $s^1,s^2 \in S$
- *transitive* if $(s^1,s^2) \in R$ and $(s^2,s^3) \in R \Rightarrow (s^1,s^3) \in R$ for all $s^1,s^2,s^3 \in S$
- *negatively transitive* if $(s^1,s^2) \notin R$ and $(s^2,s^3) \notin R \Rightarrow (s^1,s^3) \notin R$ for all $s^1,s^2,s^3 \in S$
- *connected* if $(s^1,s^2) \in R$ or $(s^2,s^1) \in R$ for all $s^1,s^2 \in S$ with $s^1 \neq s^2$
- *strongly connected* (or total) if $(s^1,s^2) \in R$ or $(s^2,s^1) \in R$ for all $s^1,s^2 \in S$

2.4.2.2 Ordering

Strict preference. A binary relation R on set S is a strict preference on S if and only if R serves to introduce a hierarchy among the elements of S. In this case R is denoted as $\prec$.

Indifference. A binary relation R on set S is an indifference on S if and only if R serves to introduce a notion of equality among the elements of S. In this case R is denoted as ~.

Preference. A binary relation R on set S is a preference on S if and only if $R = R_1 \cup R_2$ is the disjoint union of a strict preference R_1 and an indifference R_2. In this case R is denoted as $\leq$.

Ordering relations. A binary relation R on set S is:

- a *partial preorder* if and only if it is reflexive and transitive
- a *partial order* if and only if it is reflexive, transitive, and asymmetric
- a *complete preorder* if and only if it is reflexive, transitive, and complete
- a *linear order* (or simply order) if and only if it is reflexive, transitive, asymmetric and complete
- an *equivalence* if and only if it is reflexive, transitive, and symmetric

2.4.2.3 Cones and lexicographical order

Often partial orders and preorders are generated by cones.

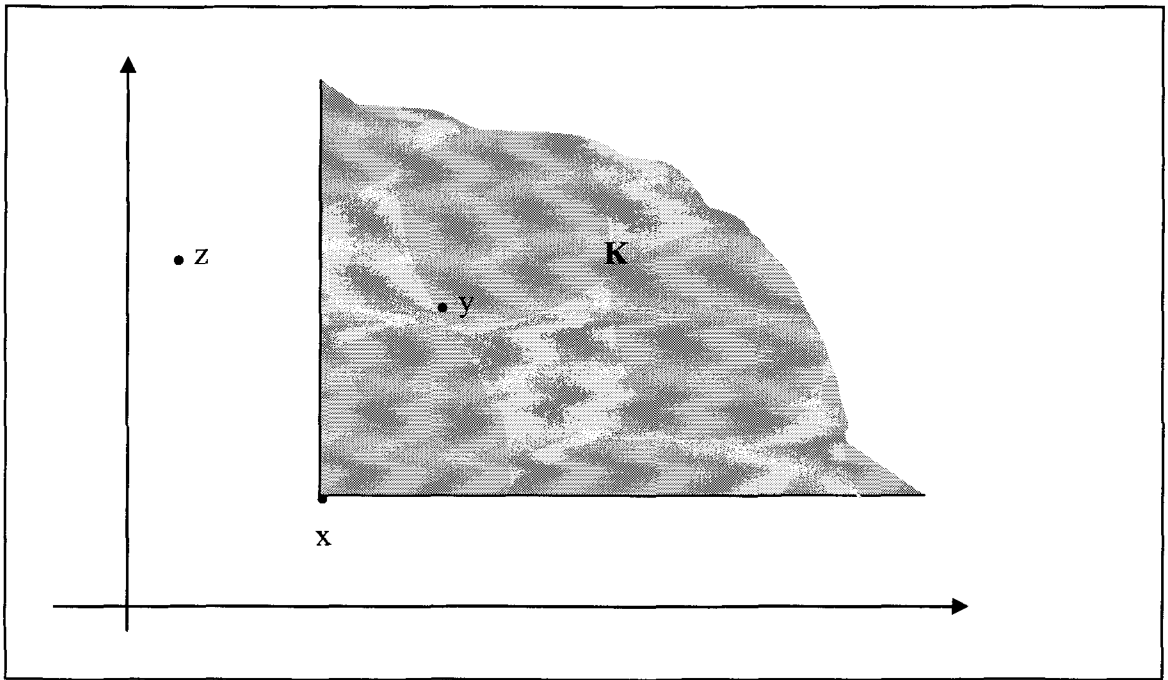
A subset $K \subseteq W$ of a vector space W is a *cone*, if and only if $\alpha p \in K$ for all $p \in K$ and for all $\alpha > 0$.

A cone $K \subseteq W$ is called:

- *nontrivial* or proper if $K \neq \emptyset$ and $K \neq W$
- *convex* if $\alpha p^1 + (1-\alpha)p^2 \in K$ for all $p^1, p^2 \in K$ and for all $0 < \alpha < 1$
- *pointed* if for $p \in K,\ p \neq 0,\ p \neq -p$ i.e. $K \cap (-K) = \emptyset$

Convex pointed cones generate partial orders. Non-convex and non-pointed cones generate only partial preorders since they contain subspaces destroying the asymmetry property.

Figure 2.5 Natural order in R^2 (Stadler, 1988)



A cone **K** is associated with each point x in R^2 and $\forall\, x \in \boldsymbol{K},\ x \leq y$. The point z is not comparable to x with this order.

Another ordering widely adopted on practice is the lexicographical order. A lexicographical order is similar to the order of words in a dictionary (as assumed for maximization): $p^1 \succ p^2$ if and only if:

- $x_1\left(p^1\right) > x_1(p^2)$

  or

- $x_i\left(p^1\right) = x_i(p^2),\quad i = 1,\ldots,k$ and $x_{k+1}\left(p^1\right) > x_{k+1}(p^2)$ for some $k = 1,\ldots,n-1$.

This means that the criteria $x_1,\ldots,x_n$ are ordered according to importance. p^1 is preferred to p^2 if its criterion ranged first is greater regardless of the values of other criteria. Only if the values of the first criterion are equal for both points, the next criterion is taken into consideration. An important property of the lexicographical order is that two distinct points in the decision space cannot be indifferent with this order.

2.4.3 Non-dominance and efficiency

2.4.3.1 Concepts

In the feasible area in the criteria space not all alternatives deserve equal consideration. There is only a small subset of so-called *non-dominated* alternatives where the solution to the optimisation problem should be sought. All the rest of the alternatives could be just ignored, which substantially simplifies the search. Below I explain this idea in a more formal way.

In terms of the decision space and criteria space we can compare two points p^1 and p^2 in the decision space the following way (minimisation assumed):

- either $p^1 \succ p^2$ if and only if $X(p^1) \leq X(p^2)$ (Strict inequality for at least one criteria), that is $\exists \quad j$ such as $x_j(p^1) < x_j(p^2)$ and $x_i(p^1) \leq x_i(p^2)$ for all $i \neq j$
- or $p^1 \prec p^2$ if and only if $X(p^1) \geq X(p^2)$ (Strict inequality for at least one criteria) that is $\exists \quad j$ such as $x_j(p^1) > x_j(p^2)$ and $x_i(p^1) \geq x_i(p^2)$ for all $i \neq j$
- or $p^1 \sim p^2$ if and only if $X(p^1) = X(p^2)$ that is $x_i(p^1) = x_i(p^2)$ for all i.
- or $p^1 \prec\succ p^2$ if and only if $X(p^1) <> X(p^2)$ that is $\exists \quad j$ such as $x_j(p^1) > x_j(p^2)$ and $\exists \quad i \neq j$ such as $x_i(p^1) < x_i(p^2)$

Notice, that we compare the images of the points from the decision space in the criteria space and, for instance, $p^1 \sim p^2$ does not mean that $p^1 = p^2$ in the decision space.

Here $p^1 \prec\succ p^2$ are incomparable or not dominating each other because p^1 is better by some criteria and p^2 is better by some other criteria. $p^1 \succ p^2$ means that p^1 is better than p^2 by some criteria and not worse by the others. It is said that p^1 *dominates* p^2 or p^2 is dominated by p^1.

When solving the multicriteria optimisation problem, we are not interested in dominated points as possible solutions because for any dominated solution there is at least one solution which is better by at least one criterion and is not worse by any of the other criteria. We need to find the solutions which are not dominated by others. Such solutions are called non-dominated or Pareto-optimal. Some authors also call them Edgeworth-Pareto optimal (Stadler, 1988), non-inferior, or efficient (Ehrgott, 2002). Pareto (Pareto, 1906 as cited in Stadler, 1988, p.2) defined optimal decision as:

> "We will say that the members of a collectivity enjoy maximum ophelimity in a certain position when it is impossible to find a way of moving from this position very slightly in such a manner that the ophelimity enjoyed by each of the individuals of the collectivity increases or decreases. That is to say, any small displacement in parting from that position necessarily has the effect of increasing the ophelimity that certain individuals enjoy, of being agreeable to some and disagreeable to others."

In a set of non-dominated solutions the improvement of some criterion can be achieved only by deterioration of some other criteria. The definition of efficient solutions and non-dominated points can be stated as (Ehrgott, 2002):

A feasible solution $p^s \in F_s$ is called *efficient* or Pareto-optimal, if there is no other $p \in F_s$ such that $X(p) \leq X(p^s)$. If p^s is efficient, $X(p^s)$ is called non-dominated point. If $p^1, p^2 \in F_s$ and $X(p^1) \leq X(p^2)$ than p^1 dominates p^2 and $X(p^1)$ dominates $X(p^2)$. The set of all efficient solutions $p^s \in F_s$ is denoted F_p and called the efficient set. The set of all non-dominated points $c^s = X(p^s) \in C_s$, where $p^s \in F_s$ is denoted as C_p and called the non-dominated set, Pareto-frontier, or efficient solution frontier (Liu, Yang, Whidborne, 2003).

There are several equivalent definitions, in particular:

$p^s \in F_s$ is efficient if there is no such $p \in F_s$ such that $x_i(p) \leq x_i(p^s)$ for $i = 1,\ldots,n$ and $x_j(p) < x_j(p^s)$ for some $j \in \{1,\ldots,i\}$. In other words there is no point p such as $p \succ p^s$.

A feasible solution $p^s \in F_s$ is called *weakly efficient* or weakly Pareto-optimal, if there is no $p \in F_s$ such that $X(p) < X(p^s)$, i.e. $x_i(p) < x_i(p^s)$ for all $i = 1,\ldots,n$. The point $c^s = X(p^s)$ is then called weakly non-dominated.

Figure 2.6 Pareto frontier

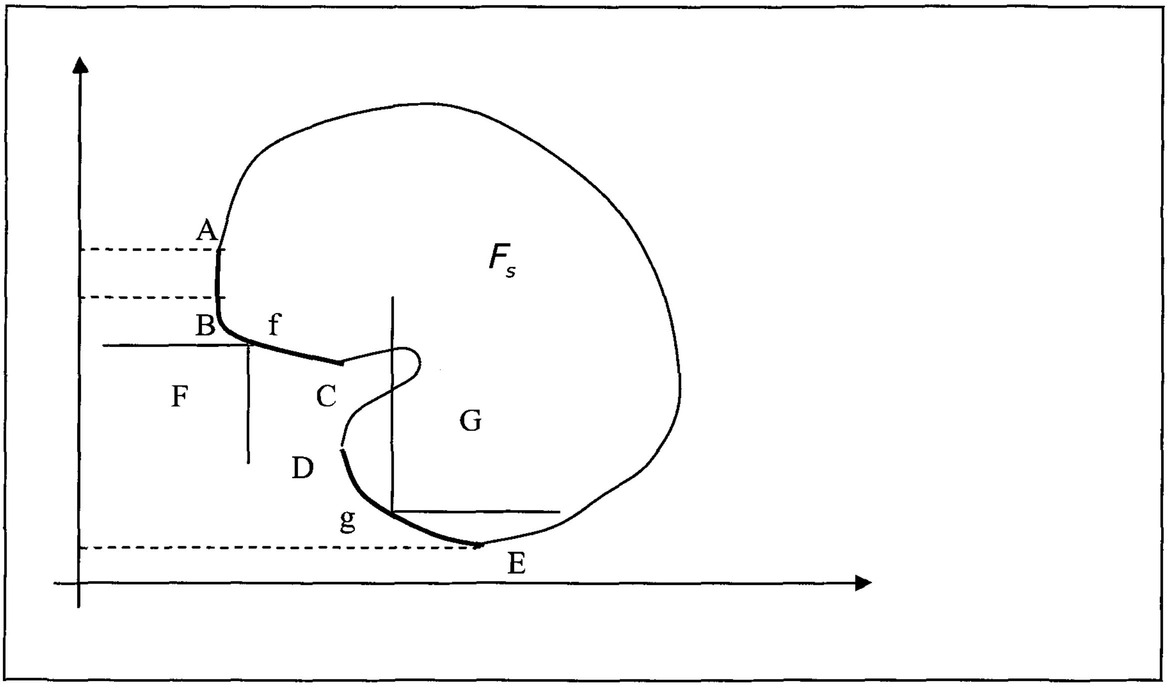


Figure 2.2 illustrates efficient solutions in the two-dimension criteria space where $x_1(p)$ and $x_2(p)$ are assumed for minimization. Segments AC and DE represent the Pareto frontier. Segment AB represents weakly non-dominated points, segments BC and DE represent strictly non-dominated points. We can see that the Pareto frontier is non-convex and disconnected. All points in cone G are dominated by point g and point f is non-dominated because cone F contains no points from F_s.

2.4.3.2 Non-dominated set bounds

An indication of maximal and minimal values of non-dominated points is given by the ideal and nadir points (Ehrgott, 2002). These points are used in many methods, for instance, for minimax (ideal point) method (Liu, Yang, Whidborne, 2003) of finding the most preferred solution from a set of efficient solutions.

If a set of efficient solutions is nonempty and bounded, we always can find real numbers $\underline{c}_i, \bar{c}_i$, $i = 1,\dots,n$ such as $\underline{c}_i \le x_i \le \bar{c}_i$ for all $c(x_1, x_2, \dots, x_n) \in C_s$.

The *ideal point* $c^I = (x_1^I, x_2^I, \dots, x_n^I)$ of multicriteria optimisation problem (2.1) is given by $x_i^I = \min_{p \in F_s} x_i(p)$.

The *nadir point* $c^N = (x_1^N, x_2^N, \dots, x_n^N)$ of multicriteria optimisation problem (2.1) is given by $x_i^N = \max_{p \in F_E} x_i(p)$.

Figure 2.7 Ideal and nadir points for a two-dimensional criteria space

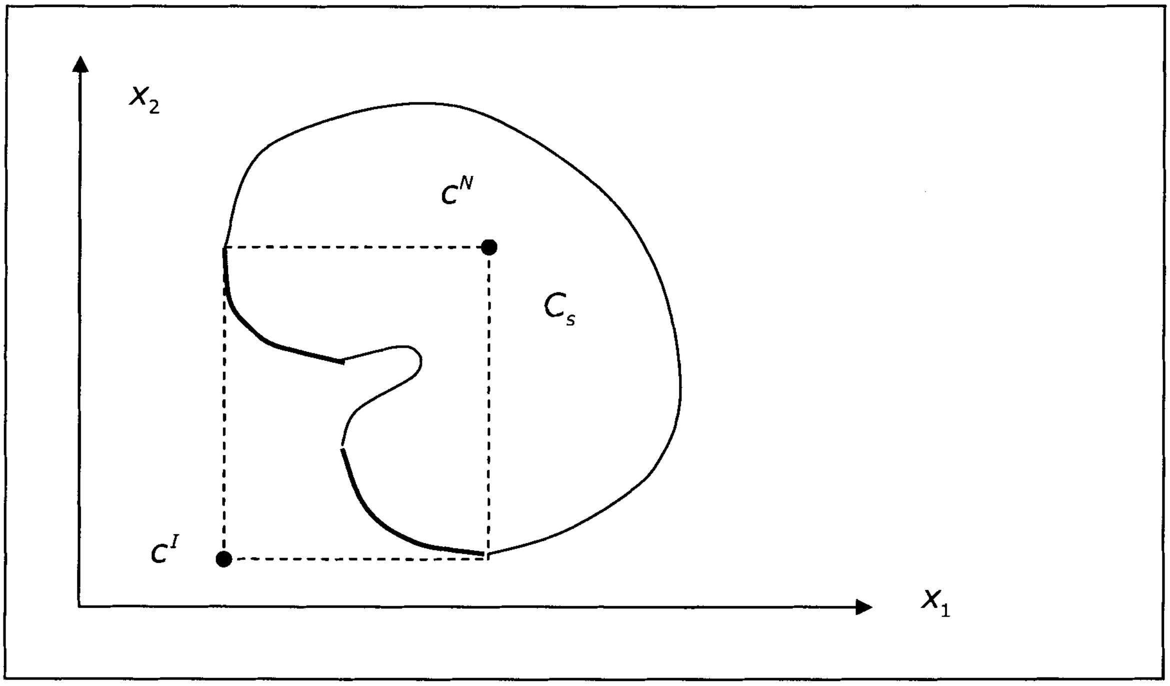


Figure 2.3 shows the ideal and nadir points for the non-convex problem depicted in Figure 2.2. Notice, that we do not need to calculate the efficient set of solutions to find

the ideal point. This fact makes this point particularly useful for *a priori* methods described below.

2.4.4 Methods

Sometimes it is possible to reduce the multi-criteria optimization to single criterion. This is achieved by constructing a utility function for the multicriteria optimization problem (2.5.1) in the form $u(X(p)) = u(x_1(p), x_2(p), \dots, x_n(p))$. In this simple case the optimal solution can be found as the solution that minimizes the utility function $u(X(p))$ for all $p \in F_s$. The simple weighting method was developed to serve this purpose.

2.4.4.1 Simple weighting method

We describe the simple weighting method using an example of a two criteria optimization problem, as it is given in (Liu, Yang, Whidborne, 2003) and (Ehrgott, 2003).

In general, a two criteria optimisation problem can be stated as follows:

- minimize $x_1(p)$
- minimize $x_2(p)$
- given $p \in F_s$

Notice, that if some criterion $x_i(p)$, $i = 1,2$ assumed for maximization we always can replace it by the $-x_i(p)$ equivalent for minimization. Without loss of generality, we can assume that both criteria are measured using the same scale. If the image C_s of the feasible set F_s in the criteria space is convex and compensation between the two criteria is allowed, the simple weighting method can be applied to generate efficient solutions. In this case, we create a utility function and the problem can be thought of as a single criteria optimization problem in form:

$$\text{Minimize } f(p) = \alpha_1 f_1(p) + \alpha_2 f_2(p) \tag{2.2}$$

where $\alpha_1 \geq 0$ and $\alpha_2 \geq 0$ are weighting factors. For a single criteria problem, dividing the criteria by a positive real number does not change the optimum. If we assume that $\alpha_1 > 0$ we can divide both sides of (2.5.2) by α_1 and denote $\alpha = \alpha_2 / \alpha_1$. Then we can consider the equivalent problem

$$\text{Minimize } f(p,\alpha) = f_1(p) + \alpha f_2(p) \tag{2.3}$$

since C_s is convex. For a given α, the optimal solution of (2.5.2) is an efficient solution of the stated multicriteria problem. Using different values for α, we can generate different efficient solutions. Since we are not looking for a specific efficient solution but for a set of efficient solutions, here α is just a parameter that does not represent the decision-maker preferences. The graph of the utility function is a line in the criteria space given by the formula

$$f_1 + \alpha f_2 = a \text{ or } f_2 = -\frac{1}{\alpha} f_1 + \frac{a}{\alpha} \text{ where } a \text{ is a constant} \tag{2.4}$$

So, the slope of the line is $-\frac{1}{\alpha}$ and its ordinate intercept is $\frac{a}{\alpha}$.

Figure 2.8 Indifference lines for the simple weighting method

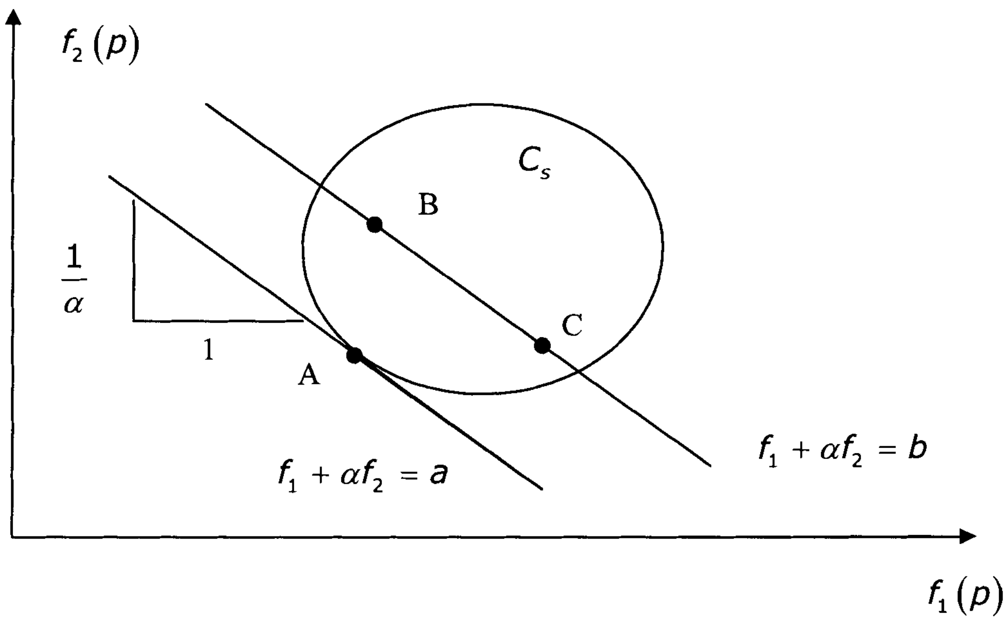


Figure 2.9 Change weight for simple weighting method when $\alpha \neq 0$ **and** $\alpha \neq \infty$

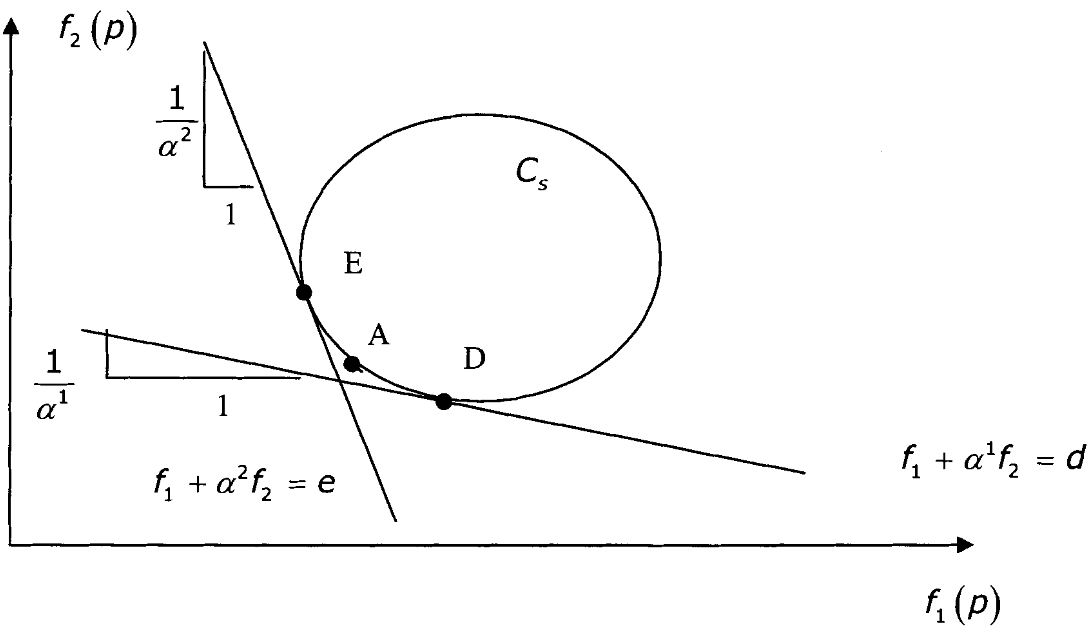
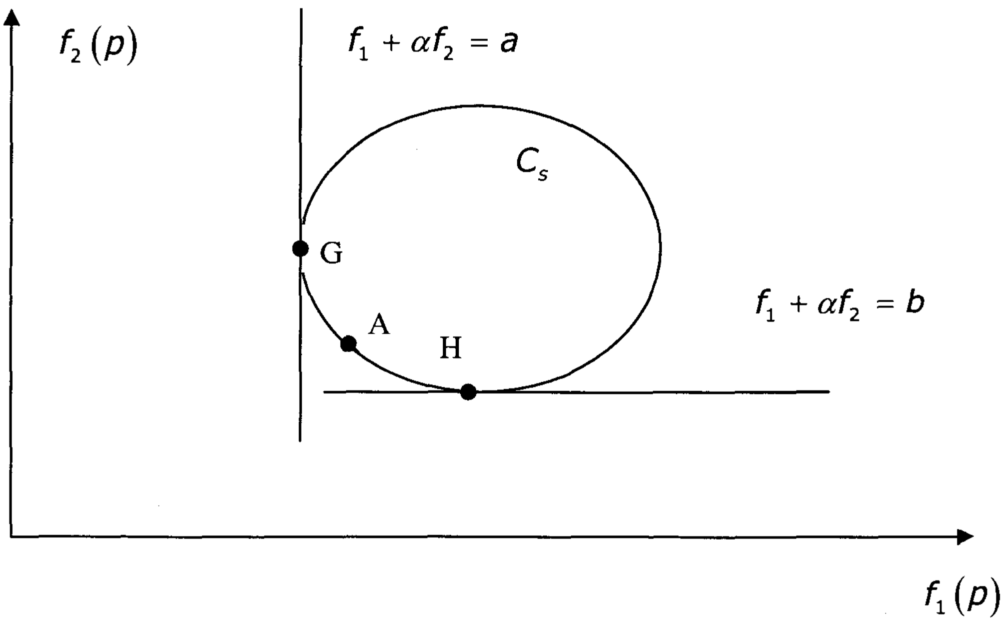


Figure 2.10 Change weight for simple weighting method for $\alpha = 0$ and $\alpha = \infty$



All points of the line located inside the image of the feasible set have the same value of the utility function. Therefore, the line is a linear indifference curve. Points B and C on Figure 2.3 represent two solutions in which the utility function has the same value, i.e. $f_1(B) + \alpha f_2(B) = f_1(C) + \alpha f_2(C) = b$. This means that the two solutions in the feasible set represented by points B and C are indifferent regarding this utility function. The solution of the single criteria problem is to move the line to the direction of the origin in parallel until it becomes the tangent line to the image of the feasible set in the criteria space. Point A in Figure 2.3 represents the tangent point. We can see that point A is in Pareto frontier and represents an efficient solution in the feasible set. If the coefficients of the utility function represent preferences on the criteria and the linear utility function is acceptable, then the point in F_s represented by A in the criteria space would be the best compromise point.

If we change α, the line will rotate and following cases are possible:

- if the new weight is $\infty > \alpha^1 > \alpha$, the representation of the best compromise solution point will change from A to D as shown in Figure 2.5. Increasing α means that the weight of f_2 is increasing but the weight of f_1 is decreasing

- if the new weight is $\alpha > \alpha^1 > 0$, the representation of the best compromise solution will change from A to E as shown in Figure 2.5. Increasing α means that the weight of f_1 is increasing but the weight of f_2 is decreasing
- if $\alpha = 0$,the representation of the best compromise solution will change from A to G as shown in Figure 2.6. This means that f_2 is not considered anymore and we only want to minimize f_1. The solution may be weakly efficient.
- if $\alpha = \infty$, the representation of the best compromise solution will change from A to H as shown in Figure 2.6. This means that f_1 is not considered any more and we only want to minimize f_2. The solution may be weakly efficient.

The simple weighting method is natural but the utility function approach can be applied only to a particular type of multicriteria optimization problems. A large number of methods for different types of problems have been developed. We must classify the methods and problems to be able to choose the appropriate methods.

2.4.4.2 Optimization method and problem classification

Multiple criteria optimization methods can be divided into three main classes (Liu, Yang, Whidborne, 2003):

- Efficient solution generation methods with preferences provided after optimisation.
- Methods for generating the best solutions based on preferences provided *a priori.*
- Interactive methods with preferences extracted progressively in decision analysis process.

In the first class of methods, the set of desirable efficient solutions is generated first. Then, according to the decision maker preferences the best compromise solution is found. An advantage of these posterior methods is that there is no need to involve the decision maker in the generation of the set of efficient solutions. The disadvantages of these methods are as follows: they usually require a large number of calculations and, sometimes, the set of efficient solutions is too large which complicates finding the best compromise solution. The simple weighting method described above is a widely used but only applicable to problems with a convex image of the feasible set and a smooth Pareto-frontier. In the case of a non-convex image of the feasible set in the criteria space, this method may fail to produce a correct set of efficient solutions.

The second type of methods, often referred to as "*a priori*" methods, require some global preference information in advance. Using the preferences, a multicriteria optimisation problem can be transferred into a single criteria optimisation problem. Then, the solution for the single criteria optimisation problem is the best compromise solution for the original problem. For these methods, optimisation only needs to be conducted once and the number of calculations is relatively small but it could be difficult to provide the global preference information in advance. The ideal point method is one of the widely used methods of this group. In fact, this method serves as the base for a number of other methods; the goal attainment method is one of them, using canonical weights to represent the decision maker preferences. Goal programming is only applicable to convex problems; the minimax reference point method extends goal programming to non-convex cases and provides a basis for generating efficient solutions in both convex and non-convex Pareto-frontiers.

The third type of methods requires providing some local preference information progressively in an interactive optimisation and decision making process. The main idea is to construct a series of single criteria optimisation problems related to the original multicriteria optimisation problem. The solutions of the single criteria problems will approach the best compromise solution for the multicriteria optimisation problem. These methods are referred to as interactive methods. Among the methods of this type Geoffrion's method has been introduced the earliest. Again, this method is applicable to convex problems only.

We also need to classify the multicriteria optimisation problems to be able to apply appropriate methods to different types of problems. The formal classification of multicriteria optimisation problems is as follows (Ehrgott, 2003):

Usually, vectors in the criteria space can not be compared directly. To be able to compare them we introduce some ordering on the criteria space. The ordering maps the criteria space into some ordered criteria space. This ordering is called model map and denoted as θ. A multicriteria optimisation problem has the following elements:

- the feasible set F_s,
- the criteria vector $X = (x_1, x_2, \ldots, x_n)$,
- the criteria space C,
- the ordered image of the feasible set in the criteria space, and
- the model map θ.

The feasible set, criteria vector and criteria space are the data of the multicriteria optimisation problem. These five features exhaustively describe a multicriteria optimisation problem.

For practical purposes, I will classify multicriteria optimisation problems on the basis of the features of the data of multicriteria optimisation problems.

- Depending on the properties of the feasible set, I will distinguish between continuous and discrete; infinite and finite problems.
- Depending on the type of objective functions, I will distinguish between linear, non-linear and non-smooth problems
- Depending on the form of Pareto-frontier, I will distinguish between convex and non-convex problems, and problems with disconnected Pareto-frontier.

Thus, the multicriteria optimization theory provides powerful methods for solving the problems involving multiple parameters evaluation. We should carefully evaluate the nature of the problem to apply an appropriate methodology. In many cases, the optimal solution is non-feasible and we can find only the best compromise solution. This solution always belongs to the Pareto-frontier of the feasible set.

3 METHODS

3.1 Determining the time horizon for decision making by the player

The new method for predicting situation in the soccer game with reasonably high precision is one of the central ideas of this study. It is based on determining the available time horizon until the situation is expected to change abruptly. This section provides description of methods for defining the time horizon and other prediction methods. These methods supply information for multicriteria optimization methods for player positioning, also described in this chapter.

Soccer is a dynamic game with rapidly changing environment state. The simulation environment reflects this property by having a simulation cycle length of 50-100 ms. In every simulation cycle, the player receives an update about the environment state and must inform the simulation server about its decision by sending control commands. Some important information like the direction and magnitude of the ball's velocity can change significantly from cycle to cycle. For this reason, it is often difficult to precisely predict the situation and create any short-term plans even for several cycles. If the decision differs significantly from cycle to cycle, the dynamics prevents the player from performing all necessary actions to actually carry out the decision. For instance, if the calculation of the player position on the field is based on the current location of the ball, the player would very rarely reach the desired position.

While the actual environment state changes every cycle, some *predicted* environment state can be relatively stable for several cycles. I will define the time span with stable predicted environment state as the *time horizon* for prediction or *prediction period* and denote it as T, expressed in the number of simulation cycles. Notice that there is no need for the player to remember the first cycle of current prediction period. From the player's point of view, every simulation cycle is the first cycle of the time horizon for prediction. If $T = 0$ for the previous cycle, the player calculates a new time horizon and makes a new prediction. Otherwise, the player merely decrements the calculated time horizon and updates the prediction by utilizing current information about the environment.

Table 3.1 shows sequence of time horizon T calculations by one of the players. When the ball is under control of a player the time horizon is zero since the players' action is unpredictable. As soon the ball is kicked and leaves the control area, the player is able to evaluate the time needed for the ball interception. This time becomes the time horizon for other predictions. Every simulation cycle the player recalculates the time horizon and refines the other predictions using new information about the environment.

Since the time horizon of zero length makes no sense, in the implementation I make a guess about the behaviour of the player controlling the ball. I suppose that the player will continue to move with the ball maintaining the same velocity for at least 10 simulation cycles, and extend the time horizon accordingly. Prediction of the ball motion helps us to define the length of T.

Table 3.1 Action sequence for prediction

Simulation cycle #	Calculated T	Player's action
800	...	
801	5	Recalculate T, refine the prediction
802	4	Recalculate T, refine the prediction
803	3	Recalculate T, refine the prediction
804	2	Recalculate T, refine the prediction
805	1	Recalculate T, refine the prediction
806	0	Recalculate T, refine the prediction
807	9	Calculate new T, make new prediction
808	8	Recalculate T, refine the prediction
809	6	Recalculate T, refine the prediction
810	5	Recalculate T, refine the prediction
811	4	Recalculate T, refine the prediction
812	3	Recalculate T, refine the prediction
813	2	Recalculate T, refine the prediction
814	1	Recalculate T, refine the prediction
815	0	Recalculate T, refine the prediction
816	25	Calculate new T, make new prediction
817	...	

3.1.1 Ball motion prediction

In the soccer game the situation prediction is possible with reasonable precision when the ball is outside the reach of all players. I assume that in each team the player who can reach the ball in the shortest time will be trying to get control of the ball. The other players will be just moving to some positions on the field which are good from their point of view. This fact allows predicting the situation while the ball is moving freely. So, determining the ball motion is the critical task that must be addressed.

Prediction of the ball motion and location is the base for defining feasible area and the time horizon for other predictions. We can identify two distinct states of the ball:

- the ball is controlled by a player
- the ball is not controlled by a player

In the first case, the ball is situated inside some kickable area around a player. The kickable area is a circular space around the player inside which the player can kick the ball. The diameter of the kickable area is defined by the simulation server settings and represents a distance in which a real human player can reach the ball without changing his or her own position. The diameter of a kickable area in the simulation server implementation used is 1.5 metre.

In the second case, two types of action are possible:

- the player kicks the ball
- the player dribbles the ball

If the player makes the decision to shoot the ball, the ball leaves the kickable area of the player and we are faced with the situation where the ball is not controlled by any player until it arrives into the kickable area of another player. The shooting itself takes very little time, usually one or two simulation steps.

In the case where the ball is dribbled, the player moves along the field keeping the ball inside the kickable area. It is hard to predict which decision the player will make next and for how long it will dribble the ball. For this reason, in the case of controlled ball, it is sufficient to suggest that the player will dribble the ball for some empirically defined time horizon. Since a player has some inertia, the dribbling player is unable to change the velocity or direction of the motion abruptly; therefore, the vector of the predicted location of the ball can be defined as the sum of current position vector of the player and current velocity vector of the player multiplied by the time horizon:

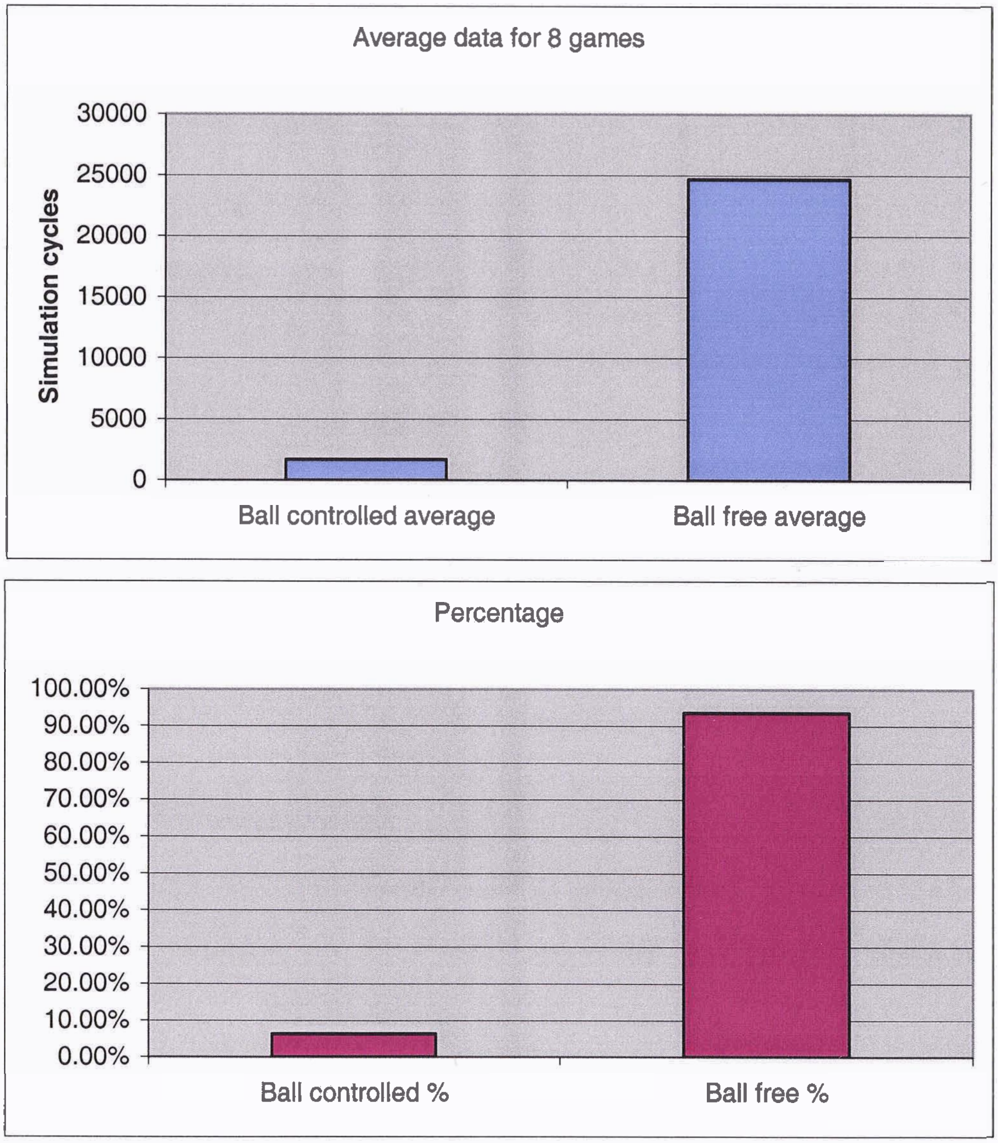
$$\vec{p}_{predicted} = \vec{p}_{current} + \vec{v}_{current} \times n,$$

where n is the time horizon empirically defined in this implementation, as $n = 20$. We must not use position and velocity of the ball here since the ball has much less inertia than a player and its velocity can significantly change from one simulation step to another causing confusion in the predicted position.

This prediction can be inaccurate if the player decides to shoot the ball; anyway, it leaves only a few simulation cycles and other players are unable to relocate before they realise its inaccuracy. Another shortcoming of this method is the fact that we can only guess the time horizon for dribbling which forces us to take the current position of the player as a base for the prediction. This makes the predicted position of the ball a "moving target" since the current position of the player controlling the ball will change every simulation step. Nevertheless, the maximal velocity of dribbling is sufficiently less than the maximal velocity of a player without the ball which means the players will be able to successfully relocate using the predicted position of the ball.

The case when the ball is not controlled is much more interesting and useful. The situation on the field depends mostly on the ball location and speed. If the ball is controlled, its velocity can change abruptly when the player kicks it. The ball velocity change forces all the players to change their location accordingly, so the entire agent environment will change in the time horizon of several cycles while the ball is moving. If it is moving freely, we can predict with high accuracy where, when and by which player the ball will be likely intercepted using the laws of physics and the standard simulation model features. Having these predictions, we can determine the time horizon for other predictions and decision making since the behaviour of all the players somehow depends on the state of the ball. Moreover, experiments show that the ball is uncontrolled more than 90% of the time of the game (Fig. 3.1). This means that we are able to divide the time span of the game into periods significantly larger than one simulation step. These periods have a stable environment state given by prediction inside the time horizon defined for each period. This makes the player's decisions about positioning persistent during the period.

Figure 3.1 Ball control during the game.



To predict the time and the place of where the interception occurs, we must determine which player is able to reach the ball first because the fastest players to the ball in both teams are most likely to be chasing it. To do that, we must estimate the interception time for all the players on the field. The algorithm should also define the location of the interception point. Remco de Boer and Jelle Kok (Kok&Boyer, 2002) proposed an appropriate algorithm for determining the time and location of the interception point. This algorithm, however, contains three nested loops, which is rather time consuming. For the purposes of this research a simplified algorithm using only two nested loops and some heuristics was implemented. The simplified algorithm provides accurate results for the time and location of the interception point.

Figures 3.2-3.4 illustrate the process of the interception point prediction. The magenta coloured circle with a dot represents the interception point as it is predicted by the yellow player #11. We can see that the prediction is refined as the ball is approaching the interception point. Nevertheless, the predicted interception point remains in close area of the actual interception point (Figure 3.4). Table 3.2 presents data on the predicted situation while passing the ball.

Figure 3.2 The ball has just started to move freely

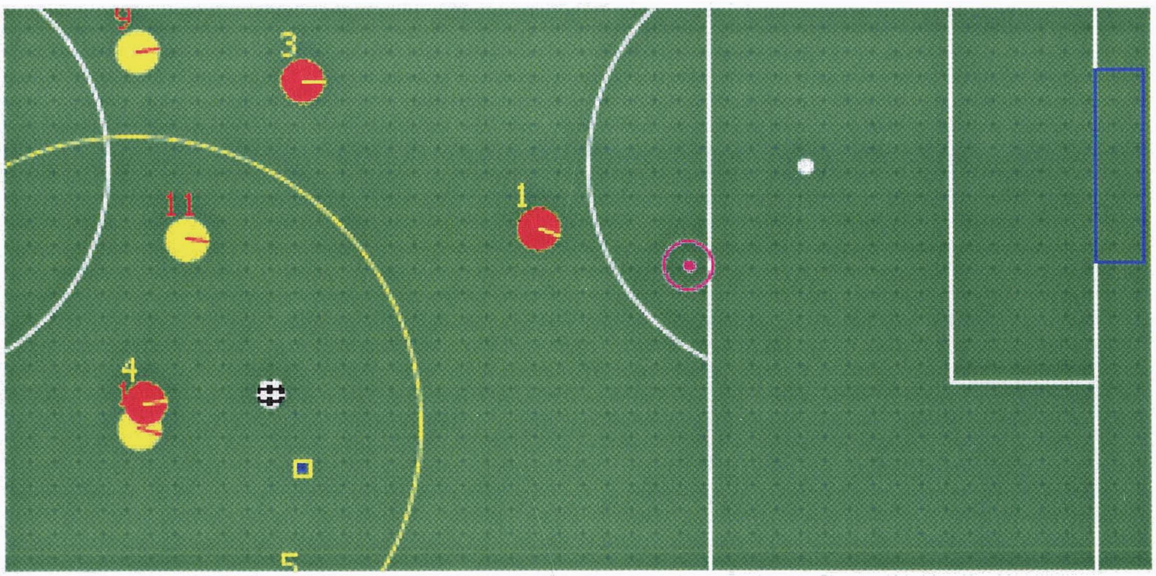


Figure 3.3 The ball is halfway the way to the interception point

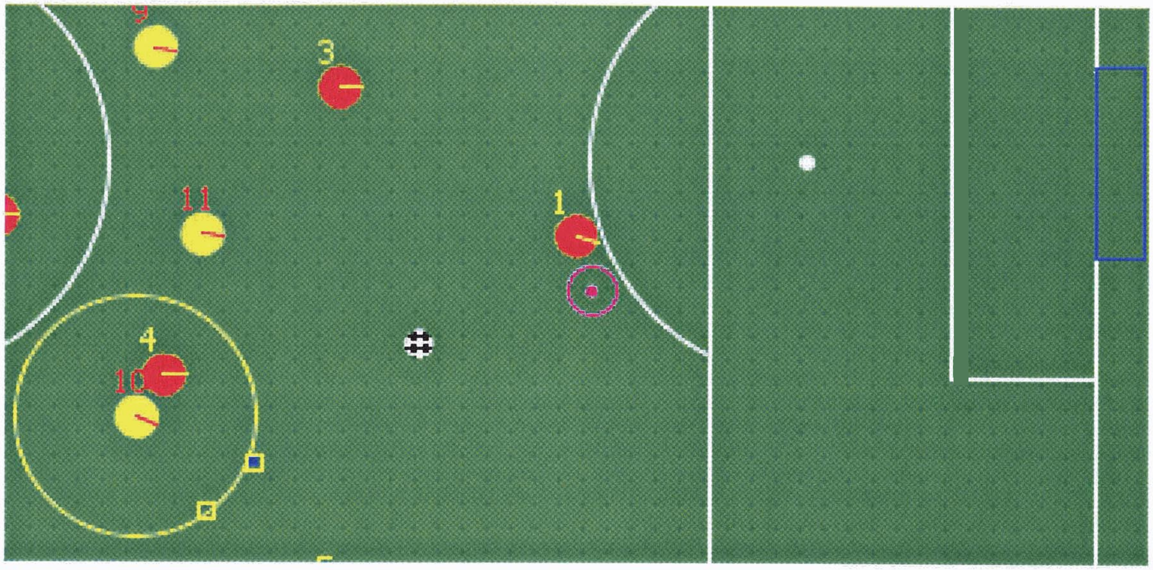


Figure 3.4 The ball is intercepted.

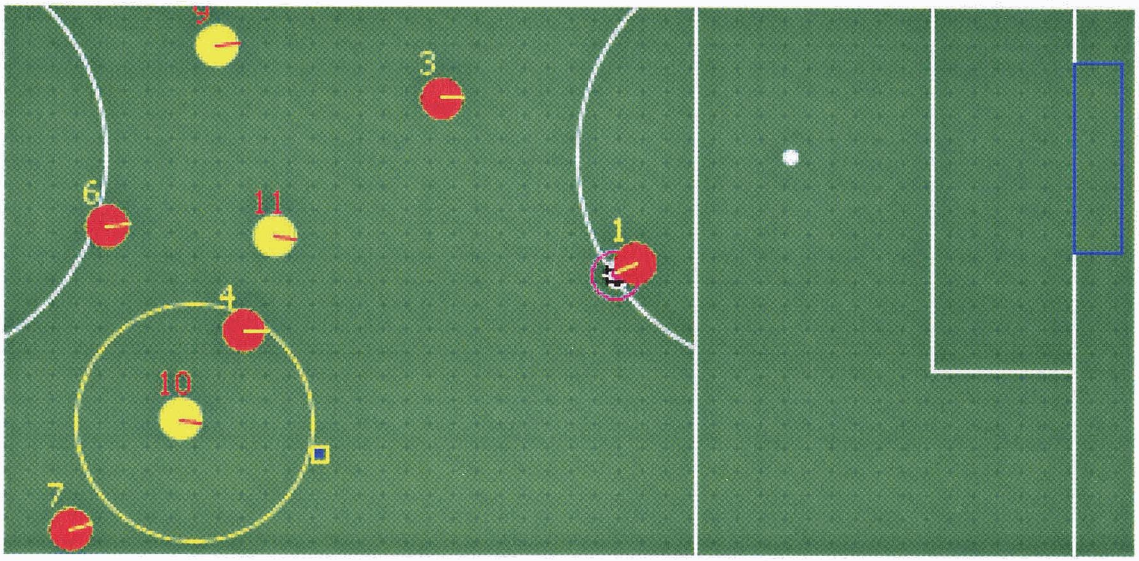


Table 3.2 shows an example of the ball interception point and time prediction. The maximal and average deviations of the predicted interception points from the actual interception point are sufficiently less that 1 meter. This is enough accuracy for decision making.

Table 3.2 A player reports its prediction data (Fig.3.2-3.4).

X coordinate of predicted interception point	Y coordinate of predicted interception point	Predicted number of cycles	Deviation from actual interception point in meters	Comment
-27.20	0.78	16	0.27	
-27.18	0.68	15	0.27	
-27.43	0.68	14	0.02	
-27.53	0.70	13	0.08	
-27.67	0.71	12	0.23	
-27.78	0.64	11	0.33	
-27.75	0.67	10	0.30	
-27.37	***0.75***	***8***	***0.09***	Time horizon refined
-27.36	0.71	7	0.09	
-27.40	0.66	6	0.06	
-27.46	0.69	5	0.02	
-27.51	0.71	4	0.06	
-27.48	0.69	3	0.03	
-27.47	0.72	2	0.03	
-27.48	0.71	1	0.03	
-27.45	**0.70**	**0**	**0.00**	**Actual interception point**
Average deviation	0.13	Maximal deviation	0.33	

The player predicted, at some point in the game, that the ball would be intercepted in 16 cycles at the point with coordinates (-27.20, 0.78). The actual interception happened in 15 cycles at point with coordinates (-27.45, 0.70). TAO of soccer simulator provides almost precise visual sensor information which helps to make the prediction more accurate. The RoboCup simulation requires some additional methods to enhance the prediction accuracy.

3.1.2 Players' motion prediction

The ball motion prediction gives us the time horizon for other predictions. The prediction of the player motion can be based on two types of information:

- the player's physical state
- the player's decision making mechanism

In both cases several levels of prediction are possible. For instance, during the game we can try to infer the opponents' decision making scheme and use it for prediction, or use one of the already known decision making schemas. For the physical state prediction we can use information about the players' velocity and acceleration. In this case the prediction is based on the fact that the players possess some amount of inertia and are not able to change their velocity abruptly. In the current implementation only player velocity was used for prediction.

The vector of the player's predicted location can be defined as the sum of current position vector of the player and current velocity vector of the player multiplied by the time horizon:

$$\vec{p}_{predicted} = \vec{p}_{current} + \vec{v}_{current} \times n_p, \tag{3.1}$$

This formula is similar to the above formula of the prediction of the motion of controlled ball, but this time the time horizon is substantially greater; it is calculated based on the prediction of the ball free motion.

Since the player can change its velocity applying some force, the given formula can produce inaccurate results when the velocity is changing over time. To reduce this inaccuracy the exponential smoothing with coefficient 0.5 was applied. This means that the predicted position is given by formula:

$$\vec{p}_{new} = \vec{p}_{predicted} \times \alpha + \vec{p}_{old} \times (1 - \alpha) \tag{3.2}$$

where $\vec{p}_{old}$ is the player's position predicted in the previous simulation cycle, $\vec{p}_{predicted}$ is the predicted position given by (3.1) and α is a smoothing coefficient.

In general, the accuracy of the prediction in any given prediction interval grows in the end of the interval because the player gets desired speed and acceleration approaches zero. Also, the longer the prediction interval, the more precise the prediction at the end of the interval is.

In Figures 3.5-3.7 the white circle with a dot represents the anticipated position of red player #8 at the moment of the ball interception as predicted by yellow player #8.

Figure 3.5 The ball is in motion

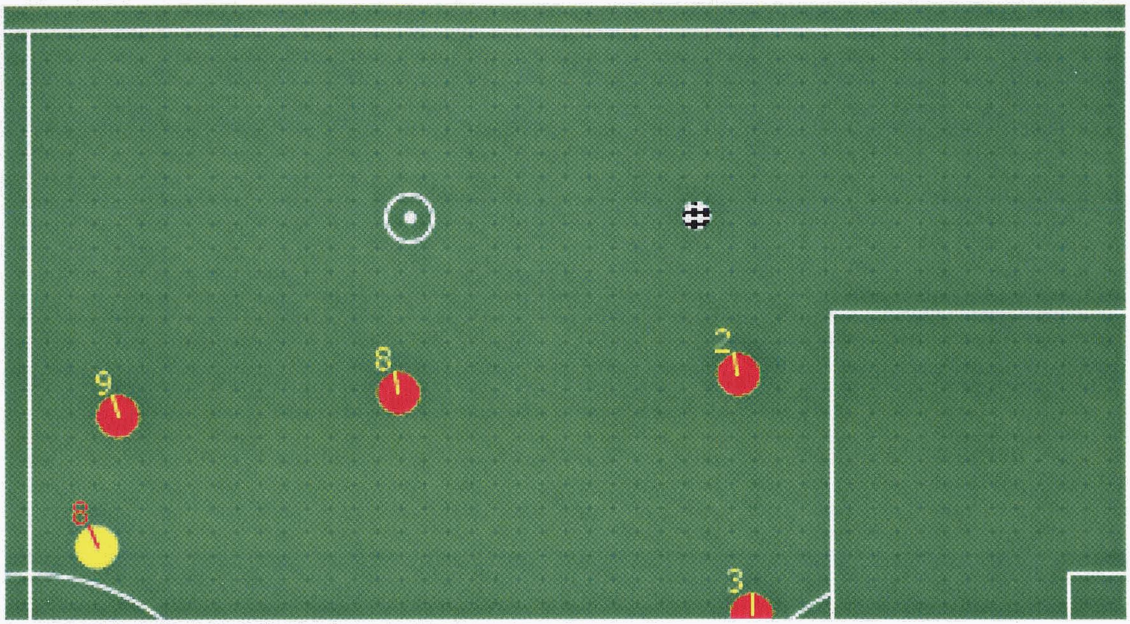


Figure 3.6 The ball is about to be intercepted

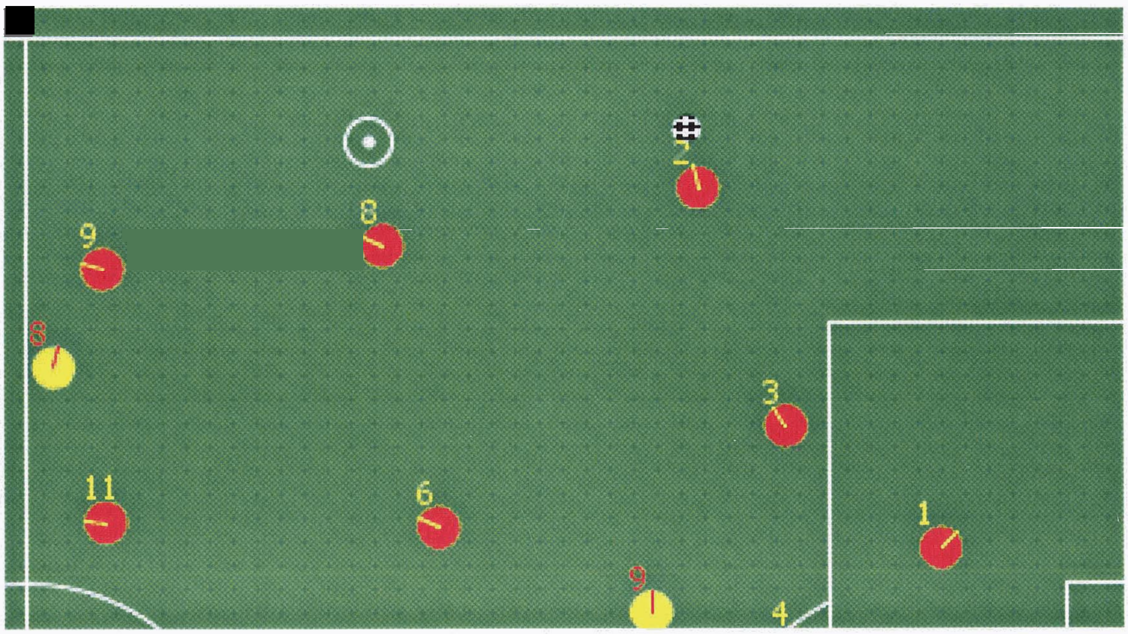


Figure 3.7 The ball is intercepted

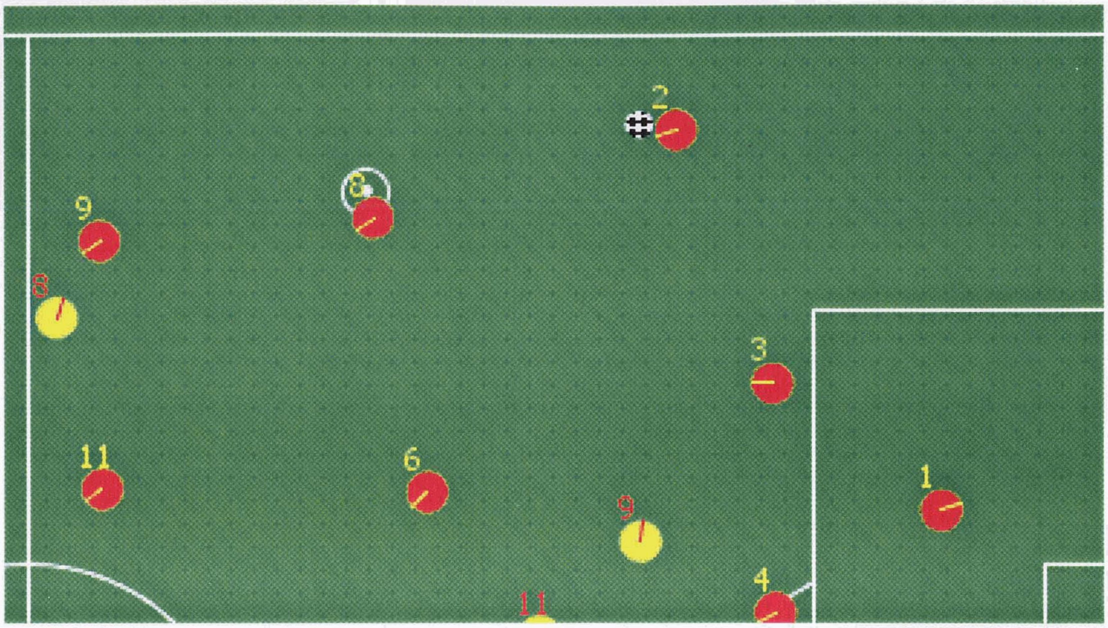


Table 3.3 shows an example of the prediction of a player position at the moment of the ball interception. The maximal and average deviations of the predicted positions from the actual position are close to 2 metres. This prediction is less precise than the prediction of the ball interception point but still accurate enough to make a decision. It can be seen from the table that the prediction precision grows as the prediction period comes to its end.

Table 3.3 A player reports its prediction data.

X coordinate of the predicted position	Y coordinate of the predicted position	Predicted number of cycles	Deviation from the actual position in meters	Comment
5.57	-7.11	19	1.46	
5.16	-7.38	18	1.71	
4.94	-7.73	17	1.86	
4.80	-8.02	15	1.99	
4.79	-8.21	14	2.03	
4.78	-8.32	12	2.06	
4.81	-8.38	11	2.04	

X coordinate of the predicted position	Y coordinate of the predicted position	Predicted number of cycles	Deviation from the actual position in meters	Comment
4.85	-8.42	10	2.02	
5.19	-8.34	9	1.66	
5.82	-8.15	8	1.01	
6.25	-7.98	7	0.55	
6.57	-7.91	6	0.23	
6.71	-7.89	4	0.08	
6.79	-7.88	3	0.02	
6.83	-7.87	2	0.04	
6.82	-7.87	1	0.04	
6.80	-7.90	0	0.00	**Actual interception point**
Average deviation	1.25	Maximal deviation		2.06

At some point in the game, player #6 predicted that when the ball is intercepted in 19 cycles, the opponent player will be located at the point with coordinates (5.57, -7.11). The actual interception happened in 16 cycles and at that moment the opponent player was located at the point with coordinates (6.80, -7.90).

More sophisticated algorithms can be developed using the acceleration data, but the development of such algorithms is beyond the scope of this research. The ball and the players are the dynamic parts that form the soccer simulation environment. Once the positions of the ball and the players are predicted, the predicted state of the environment is defined and the player is able to look for an optimal position.

Further improvements could be made with reasonably good models of player behaviour. It is possible in principle to predict actions by team-mates; prediction for the opponents requires modeling their decision making. In this study, I do not address this problem, though. It was left as part of future work instead.

3.1.3 Feasible area and area of responsibility

In general, the player can consider any point on the field as a potential destination. Since the coordinates of the points on the field are represented by pairs of real numbers, there is an infinite number of location options. To make the problem tractable, I will be using a discrete representation of the field in the form of a grid of points covering the entire field. To preserve precision, the distance between points should not be too large; too small distance would result in prohibitively long computations. So I set this distance at 2 metres in each dimension, which is comparable with the player size and provides sufficient precision for positioning. Since the field size is 100 by 65 meters, the total number of point in the grid is:

$$\left(\frac{100}{2}+1\right)\times\left(\frac{64}{2}+1\right)=1683 \text{ points.} \qquad (3.3)$$

We will consider this grid as the set representing the decision space and denote it as F. Having information about the time horizon for the planning of positioning and the player role in the formation, we can define the area on the field where the player will be searching for the optimal position. To make this positioning decision, the player must be able to eventually reach the desired position in given time T. This means that the optimal position can not be just any point in the decision space - it must be some point that the player can reach in the given prediction period. In other words, many points in F can be eliminated as unfeasible.

Thus we define *the feasible area* as the area containing all the points the player can reach in the given time horizon. We denote the set of points inside the feasible area as

$F_f \subseteq F$.

In practice, at any given simulation cycle, for each of the players, the feasible area is a circle with radius $R_f = V_{\max} \times T$ where $V_{\max}$ is the maximal player velocity and T is the current time horizon for decision making. The player has some inertia; so the centre of the circle is defined as $\vec{p}_{current} + \vec{v}_{current}$. Consequently, we can define F_f as

$$F_f = \left\{ \vec{p} \in F \mid \left| \left(\vec{p}_{current} + \vec{v}_{current} \right) - \vec{p} \right| \le R_f \right\} \quad (3.4)$$

To maintain simple collaboration with team-mates, every player must obey team formation. This means that the player must occupy a particular part of the field, according to its role in the formation and current or, in our case, predicted location of the ball. In fact, there are many algorithms, calculating the point where the player must be located based on its role and the ball location. We will call this point the *recommended point*. Jelle Kok and Remko de Boer described a simple algorithm for calculation of such point (Kok&Boyer, 2002). Essentially, this is a weighted sum of the player "home" position defined in the formation and the location of the ball. We will use a predicted ball position for this calculation.

$$\vec{p}_{rec} = \vec{p}_{home} \times \alpha + \vec{p}_{ball} \times (1 - \alpha) \quad (3.5)$$

We define *the responsibility area* as some circular area with centre in the recommended point. We denote the set of points inside the responsibility area as $F_r \subseteq F$.

At any given simulation cycle, for each of the players, the responsibility area is a circle with heuristically defined radius $R_r = 10$ meters. We can define F_r as

$$F_r = \left\{ \vec{p} \in F \mid \left| \vec{p}_{rec} - \vec{p} \right| \le R_r \right\}.$$

The player must be seeking some position inside the responsibility area at any time of the game to maintain the team formation. Note that the responsibility area can take a geometrical form other than a circle. This form was merely chosen as the most natural in the context of the soccer game.

Since the player must seek the position inside the responsibility area and must be able to realise its positioning plans, the set of points $F_s \subseteq F$ where the player must search for the optimal position is the intersection of the sets of feasible area and the responsibility area:

$F_s = F_f \cap F_r$. This is the feasible set in the decision space.

Figure 3.8 Search space

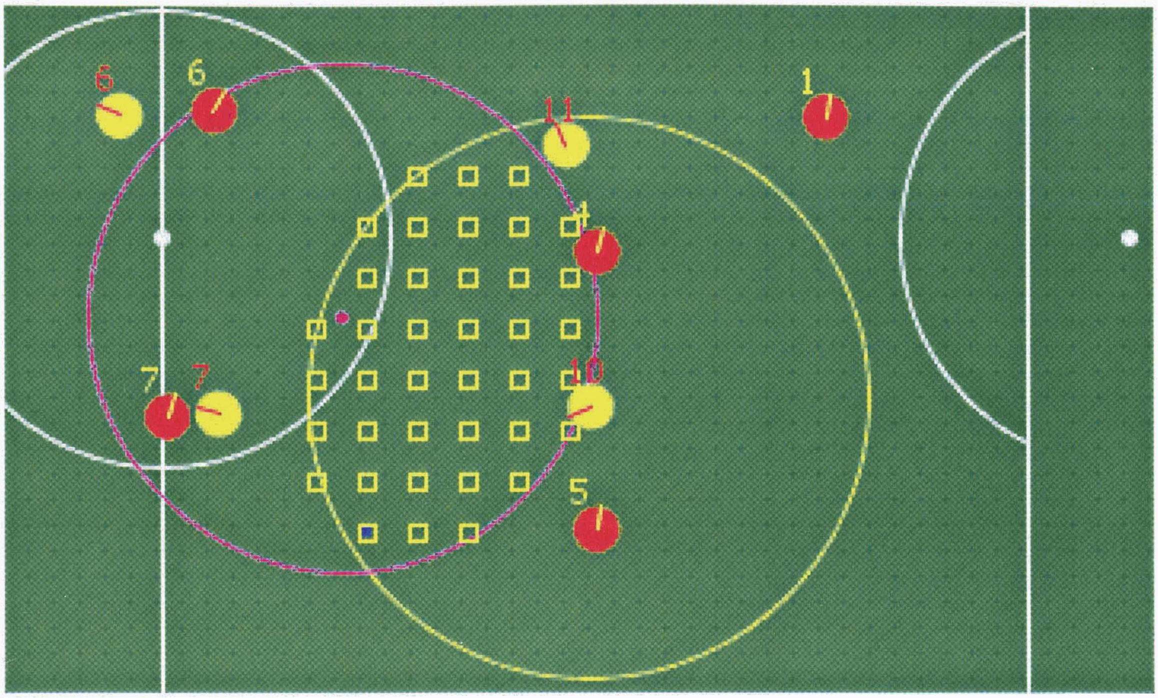


Figure 3.8 shows an example of the search space for yellow player #10. Yellow circle represents the feasible area, magenta circle represents the responsibility area and yellow squares represent the feasible set F_s.

In the case when F_s is an empty set, I consider the player being too far away from the responsibility area and establish the only solution to be the center of the area of responsibility. The player will move to the area of responsibility as quickly as possible.

3.2 Criteria for general positioning in attack

To keep the size limit of this thesis, the application part of this study is deliberately limited to player positioning in the situations when own team is in attack. Still I believe that a similar approach is also possible to address positioning in defence. However, the recent study conducted by Eddie Hou, my fellow graduate student at the same school, has shown that some defence related tasks require different approach. Therefore, in this thesis, I elected to concentrate solely on the offensive player positioning. This section elaborates on the criteria used for decision making.

Now that we constructed the feasible set, we must create an appropriate objective or criterion space to use multicriteria optimisation methods. To each point $\vec{p}_i$ in the feasible set we assign the vector $\vec{c}_i\left(x_{1i}, x_{2i}, \ldots, x_{ni}\right)$ where $x_{1i}, x_{2i}, \ldots, x_{ni}$ are some *characteristics* of the point in the form of real values. The source of knowledge for the criteria construction is the strategy and tactics of the soccer game.

In this research I consider only general positioning for attack. General positioning means that in this work I am not considering any 'special' situations or tasks as personal marking or offside trap. Attack means that we consider criteria only for situations where the team controls the ball with one exception. In defence, either a regular simple positioning algorithm or the same criteria as for the attack is used. The team controls the ball if one of the team players actually controls the ball or the prediction shows that the ball will likely be intercepted by one of the team players. I also describe different sets of criteria for simple and advanced team collaboration.

3.2.1 Simple team collaboration

Simple team collaboration is achieved through team formation. With the simple team collaboration, each player must only maintain the team formation and search for the optimal position for itself disregarding positions of other players in the team. The characteristics of the recommended point depend on the ball state, the state of opponents, and the state of the player itself.

For simple collaboration, the criteria are the same for all stages of attack but different for different groups of players.

3.2.1.1 Criteria for attackers

1. All players must maintain the formation. This means the player must keep as close as possible to the "recommended" point which represents the responsibility area centre. So the first criterion is the distance between the point in the feasible set and the centre of the responsibility area $x_{1i} = \left\|\vec{p}_i - \vec{p}_{rec}\right\|$. The smaller this number, the better the point is.

2. All attackers must be open for a forward pass (Beim,1977). This means that the player must keep the path (straight line) between itself and the predicted position of the ball from being blocked by an opponent or opponents. The characteristic of this path can be the widest open angle (Kok, 2002) or the distance from the line segment to the closest opponent. We use the distance from the line segment (the point of the predicted location of the ball) to the closest opponent as the base for the second parameter. The greater this number, the better the point is. There exists such threshold value of this distance that for any values greater than this threshold the actual distance does not matter. For instance, if the distance to the closest opponent is greater than 5 metres it does not matter if it equals to 10 metres or 50 metres because the opponent is still unable to intercept the pass regardless of the pass distance and shooting direction error. We call this threshold *distance tolerance threshold* d_{tr}. Using this threshold we can invert the parameter, so that $x_{2i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{ball}}, \vec{p}_{closest\ opponent}\right)\right)\right)$. The smaller the value of this criterion is, the better the point. Notice that for all points with the distance greater than the threshold, the value of the parameter is zero.
3. All players must maintain open space (Vogelsinger, 1973; Beim, 1977). This means that the player must keep as far away as possible from surrounding opponents. We use the distance from the point to the closest opponent as the base for the second parameter. The greater this number, the better the point is. Again, as we did for the second parameter, we invert this criterion using the distance tolerance threshold $x_{3i} = \max\left(0, \left(d_{tr} - \left\|\vec{p}_i, \vec{p}_{closest\ opponent}\right\|\right)\right)$. The threshold value for the first parameter can differ from the threshold value for the second parameter. For all points with the distance greater than the threshold, the value of the parameter is zero.
4. The attackers must be ready for defence penetration (Beim, 1977). This requirement means that the player must keep an open path to the opponent's goal and keep as close as possible to the offside line, so we can construct two criteria. The player must keep the path (straight line) between itself and the opponent's goal from being blocked by an opponent or opponents. The characteristic of this

path can be the widest open angle [Kok] or the distance from the line segment to the closest opponent. We use the distance from the line segment (the point – the center of the goal) to the closest opponent (except the goalie) as the base for the second parameter. The greater this number, the better the point is. We use the distance tolerance threshold to inverse the parameter, so $x_{4i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{goal}}, \vec{p}_{closest\ opponent}\right)\right)\right)$. The smaller the value of this criterion, the better the point. Notice that for all points with the distance greater than the threshold the value of the parameter is zero.

5. The player must keep as close as possible to the opponent offside line to be able to penetrate the defence. The offside line is the line going through the position of the opponent defender closest to the goal and parallel to the goal line. So, the next criterion is the distance between the point in the feasible set and the offside line by the X coordinate $x_{5i} = \left|\vec{p}_i^{\,x} - X_{offside}\right|$. The smaller this number, the better the point is.

Thus, we have five criteria altogether to evaluate the potentially optimal location points for attackers.

3.2.1.2 Criteria for midfields

The criteria for the midfields are similar to the criteria for attackers with some exceptions.

1. All players must maintain the formation, so $x_{1i} = \left\|\vec{p}_i - \vec{p}_{rec}\right\|$.
2. All midfields must be open for a forward pass from the defenders (Beim, 1977), so $x_{2i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{ball}}, \vec{p}_{closest\ opponent}\right)\right)\right)$. For all points with the distance greater than the threshold the value of the parameter is zero.
3. All players must maintain open space (Vogelsinger, 1973; Beim, 1977). This means that the player must keep as far as possible from surrounding opponents. We use the distance from the point to the closest opponent as the base for the second parameter. The greater this number, the better the point is. Again, as we did for the second parameter, we inverse this criterion using a distance tolerance

threshold $x_{3i} = \max\left(0, \left(d_{tr} - \left\| \vec{p_i}, \vec{p}_{closest\ opponent} \right\|\right)\right)$. The threshold value for the first parameter can differ from the threshold value for the second parameter. For all points with the distance greater than the threshold the value of the parameter is zero.

4. The midfields act in the central zone of the field and usually have many opponents around them. To be able to develop an attack they must have some open space before them when they get the ball (Beim, 1977). This requirement means that the player must keep an open path in the direction of the opponent's goal. The direction of the path is not the direction to the center of the opponents' goal, since the midfields are not going to penetrate the defence line. While experimenting, we empirically discovered that one of the appropriate paths is a line segment about 10 meters long, parallel to the side line of the field. The player must keep this path from being blocked by an opponent or opponents. The characteristic of this path is the distance from the line segment to the closest opponent. We use the distance tolerance threshold to inverse the parameter, so $x_{4i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i(x,y)p_i(x+10,y)}, \vec{p}_{closest\ opponent}\right)\right)\right)$. The smaller the value of this criterion is, the better the point. For all points with the distance greater than the threshold the value of the criterion is zero.

Altogether, we constructed four criteria for estimation of the possible location points for the midfields in attack.

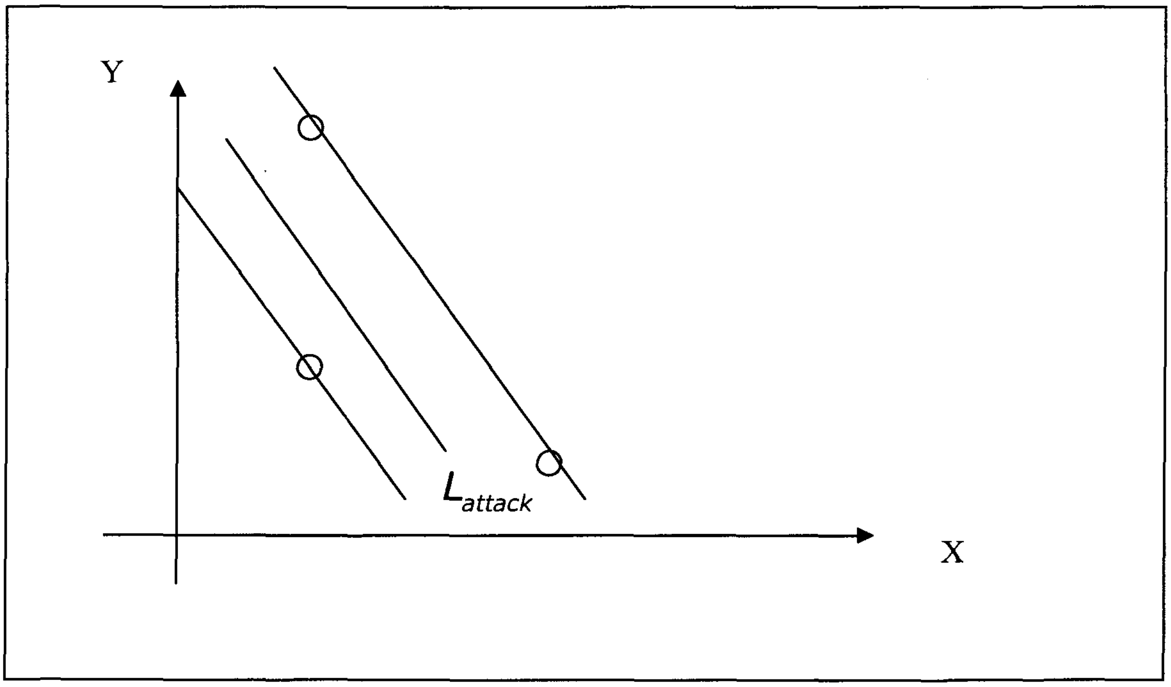
The midfields must interfere with the opponents' activity to prevent the development of an attack. Generally speaking, this responsibility of the midfields is considered to be a part of the defensive tactics. Nevertheless, there is a method of defence that can be thought of as active defence or attack in defence. This method is called "pressuring". The essence of this method is to interfere with all the actions of as many opponent players as possible at the same time forcing them to make mistakes (Beim, 1977).

Since in defence neither open space nor receiving a pass are of any concern, the set of criteria for it would be different:

1. All players must maintain the formation, so $x_{1i} = \left\| \vec{p}_i - \vec{p}_{rec} \right\|$.
2. We use only a part of pressure technique, namely preventing cross passes between opponent attackers. To achieve this goal the midfields must keep closer to the line defined by positions of the opponent forwards. We construct this line as follows (in case of three opponent forwards):
    - o Construct the line connecting the positions of two wing opponent forwards
    - o Construct a line parallel to it through the position of the opponent central forward
    - o The line between two previously constructed lines, and parallel to them will be the desired line

   If we denote the described line as L_{attack} the last criterion can be expressed as $x_{2i} = d(L_{attack}, \vec{p}_i)$. The smaller the value of the criterion is, the better the point.

Figure 3.9 L_{attack} **construction**



3.2.1.3 Criteria for defenders

The criteria for the defenders are essentially the same as the criteria for midfields.

1. All players must maintain the formation, so $x_{1i} = \left\| \vec{p}_i - \vec{p}_{rec} \right\|$.
2. All defenders must be open for a forward pass from the goalie, so $x_{2i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{ball}}, \vec{p}_{closest\ opponent}\right)\right)\right)$. For all the points with the distance greater than the threshold the value of the parameter is zero.
3. All players must maintain open space, so $x_{3i} = \max\left(0, \left(d_{tr} - \left\| \vec{p}_i, \vec{p}_{closest\ opponent} \right\|\right)\right)$. The threshold value for the first parameter can differ from the threshold value for the second parameter. For all points with the distance greater than the threshold the value of the parameter is zero.
4. The defenders must keep an open path in the direction of the opponent's goal, so $x_{4i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i(x,y)p_i(x+10,y)}, \vec{p}_{closest\ opponent}\right)\right)\right)$. The smaller the value of this criterion, the better the point. For all points with the distance greater than the threshold, the value of the criterion is zero.

Altogether, we constructed four criteria for estimation of the possible location points for the attacking defenders.

3.2.2 Advanced team collaboration

As previously mentioned, simple team collaboration is achieved through team formation. Each player purely maintains the team formation and searches for the optimal position for itself disregarding positions of other players in the team. Further improvement is possible through advanced team collaboration. This is achieved trough collective decisions, when the players look not only for the optimal positions for themselves but for mutually optimal positions for some group of players. This approach can be used for general positioning as well as for special actions like offside trap. However, in this research I construct the criteria for general positioning only.

When thinking about mutually optimal positions for the group of players, we can reconstruct the decision space or use the same decision space and effectuate additional criteria. The first approach seems to be more promising but poses some problems, which makes it more difficult to implement in the given simulation.

3.2.2.1 Decision space reconstruction

Let's consider a case of two players looking for mutually optimal location points. Let the first player have the feasible decision set F_s^1 and the second player have the feasible decision set F_s^2. Then the new feasible decision set for mutually optimal location points will be some set of ordered pairs or the Cartesian product of these two sets

$$F_s^{12} = F_s^1 \times F_s^2 = \left\{ \left(\vec{p}, \vec{q} \right) \mid \vec{p} \in F_s^1, \vec{q} \in F_s^2 \right\}.$$

Having this set, we can try to construct the criteria space for it. Some criteria are individual for each point and some should be applied to both points, such as the most wide open path between them. The problem of this approach is in its high computational cost. For instance, if each initial feasible set consists of N points the resulting set will consist of $N \times N = N^2$ points. The dimension of the criteria space will be greater than the sum of dimensions of individual criteria spaces since it must include all the individual criteria and some aggregate criteria.

For the reason of computational complexity, I utilized the additional criteria approach for advanced collaboration.

3.2.2.2 Construction of additional criteria

The idea of constructing additional criteria for advanced collaboration is based on taking in the consideration not only the state of the ball and the opponents but also some team-mates. In the case of two players trying to find mutually optimal positions each of them is taking in consideration the predicted position of the other. Eventually, they will adjust their positions according to the positions aggregated criteria values. We can think about this approach as a *reflection* of the first degree, when each of the partners takes in consideration the position of the other but ignores the fact that the partner takes in consideration its own position.

The problem in this approach is that we cannot be sure that the process of adjusting always converges. The benefit of the method is that we do not substantially increase the computational complexity and simply use different sets of criteria. The implementation of the method indeed showed improvement in the team performance.

Unlike the simple collaboration case, advanced collaboration requires different sets of criteria for different stages of attack as well as introducing the notion of designated partner.

Let A and B be two team-mates. Player B is the designated partner of the player A if the latter takes into consideration the location of B in its positioning process. If the player has more than one designated partner, we will call them the first designated partner, the second designated partner, and so on.

For the 4-3-3 formation, the partner designations are as follows:

- For lines forwards - midfields
    - The left-wing forward to the left-wing midfield and vice versa
    - The center forward to the center midfield and vice versa
    - The right-wing forward to the right-wing midfield and vice versa
- For lines midfields – defenders
    - The left-wing defender to the left-wing midfield and vice versa
    - The right-wing defender to the right-wing midfield and vice versa
    - The center right defender to the center midfield and vice versa
    - The center left defender to the center midfield and vice versa

3.2.2.3 Criteria for the attackers

3.2.2.3.1 Criteria for the case when the ball is in the defensive zone

The ball is in the defensive zone when it is controlled or will be intercepted by the goalie or a defender. In this case the criteria for the forwards are as follows:

1. All players must maintain the formation, so $x_{1i} = \left\|\vec{p}_i - \vec{p}_{rec}\right\|$.
2. All players must maintain open space, $x_{2i} = \max\left(0, \left(d_{tr} - \left\|\vec{p}_i, \vec{p}_{closest\ opponent}\right\|\right)\right)$. For all points with the distance greater than the threshold, the value of the parameter is zero.
3. The attackers must be ready for the defence penetration, so $x_{3i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{goal}}, \vec{p}_{closest\ opponent}\right)\right)\right)$ and $x_{4i} = \left|\vec{p}_i^{\,x} - X_{offside}\right|$. The smaller these numbers are, the better the point.
4. Since the ball is too far from the attackers, they can not expect a direct pass. This means they must keep open span towards designated partners rather than to the ball. Also, the wing forwards must try to stay open towards the center midfield but they are not designated partners because the central midfield will not adjust its position according to the positions of the wing forwards. The criterion for all the forwards to keep open for a direct pass from the designated partner is based on the distance from the line segment connecting the point and the predicted position of the designated partner to the closest opponent. We use the distance tolerance threshold to inverse the criterion $x_{5i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{desig\_patner}}, \vec{p}_{closest\ opponent}\right)\right)\right)$.
5. The wing forwards must keep open to direct pass from the central midfield $x_{6i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{desig\_patner}}, \vec{p}_{closest\ opponent}\right)\right)\right)$.

Altogether, we have five criteria for the central forward and six criteria for the wing forwards when the ball is in the defensive zone.

3.2.2.3.2 Criteria for the case where the ball is in the middle zone

The ball is in the middle zone when it is controlled or will be intercepted by a midfield. In this case the criteria for the forwards are as follows:

1. All players must maintain the formation: $x_{1i} = \left\| \vec{p}_i - \vec{p}_{rec} \right\|$.
2. All players must maintain open space: $x_{2i} = \max\left(0, \left(d_{tr} - \left\| \vec{p}_i, \vec{p}_{closest\ opponent} \right\|\right)\right)$. For all points with the distance greater than the threshold the value of the parameter is zero.
3. The attackers must be ready for the defence penetration: $x_{3i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{goal}}, \vec{p}_{closest\ opponent}\right)\right)\right)$ and $x_{4i} = \left| \vec{p}_i^{\,x} - X_{offside} \right|$. The smaller these numbers are, the better the point.
4. If the ball is controlled or is going to be intercepted by the designated partner or (for the wing forwards only) by the central midfield, any of the forwards must keep open for a direct pass. Notice that the player must keep open to the predicted ball position, not the designated partner position since the partner is chasing the ball and does not adjust its position $x_{5i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{ball}}, \vec{p}_{closest\ opponent}\right)\right)\right)$.
5. If the ball is controlled or is going to be intercepted by wing midfield that is not the designated partner (and not the central midfield for the wing forwards), any forward must keep open for the designated partner rather than for the ball. $x_{5i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{desig\_patner}}, \vec{p}_{closest\ opponent}\right)\right)\right)$.

Altogether, we have five criteria for all forwards when the ball is in the middle zone and the 5th criterion differs depending on the situation.

3.2.2.3.3 Criteria for the case when the ball is in the offensive zone

The ball is in the offensive zone when it is controlled or will be intercepted by a forward. In this case, the criteria for the forwards are as follows:

1. All players must maintain the formation: $x_{1i} = \left\| \vec{p}_i - \vec{p}_{rec} \right\|$.

2. All players must maintain open space: $x_{2i} = \max\left(0, \left(d_{tr} - \left\|\overrightarrow{p_i}, \overrightarrow{p}_{closest\ opponent}\right\|\right)\right)$. For all points with the distance greater than the threshold the value of the parameter\ is zero. In the final stage of attack, the forwards are concerned more about the defence penetration than about maintaining the wide open space, so the distance tolerance threshold must be significantly reduced.
3. The attackers must be ready for the defence penetration: $x_{3i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{goal}}, \overrightarrow{p}_{closest\ opponent}\right)\right)\right)$ and
4. $x_{4i} = \left|\overrightarrow{p}_i^{x} - X_{offside}\right|$. The smaller these numbers are, the better the point.
5. The forwards should not bother to be open to the direct pass since they are very close to the line formed by opponent defenders and direct pass is likely to be intercepted. Instead, together with the midfields the forwards must create "attack depth" (Vogelsinger, 1973; Beim, 1977). All forwards must keep open for the designated partners again $x_{5i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{desig\_patner}}, \overrightarrow{p}_{closest\ opponent}\right)\right)\right)$.

Altogether, we have five criteria for all forwards when the ball is in the offensive zone.

3.2.2.4 Criteria for midfields

3.2.2.4.1 Criteria for the case when the ball is in the defensive zone

When the ball is in the defensive zone the criteria for midfields are as follows:

1. All players must maintain the formation: $x_{1i} = \left\|\overrightarrow{p}_i - \overrightarrow{p}_{rec}\right\|$.
2. All players must maintain open space: $x_{2i} = \max\left(0, \left(d_{tr} - \left\|\overrightarrow{p_i}, \overrightarrow{p}_{closest\ opponent}\right\|\right)\right)$. For all points with the distance greater than the threshold the value of the parameter is zero.

3. The midfields must be ready to get a direct pass from defenders or the goalie. If the ball is controlled or is going to be intercepted by the designated all midfields must keep open regarding the predicted ball position.

$x_{3i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{ball}}, \vec{p}_{closest\ opponent}\right)\right)\right).$

If the ball is controlled or is going to be intercepted by a player which is not the designated partner all midfields must keep open for the designated partner in the defensive line rather than for the ball.

$x_{3i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p^1_{desig\_patner}}, \vec{p}_{closest\ opponent}\right)\right)\right)$. Notice, that the central midfield has two designated partners and has one more similar criterion, accordingly.

4. Midfields must keep open to the predicted position of the designated partner in the forward line to create a "bridge" between defenders and forwards. This "bridge" allows the team to quickly deliver the ball from the defensive zone to the offensive zone $x_{4i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p^2_{desig\_patner}}, \vec{p}_{closest\ opponent}\right)\right)\right).$

Altogether, we have four criteria for the wing midfields and five criteria for the central midfield when the ball is in the defensive zone.

3.2.2.4.2 Criteria for the case when the ball is in the middle zone

The ball is in the middle zone when it is controlled or will be intercepted by a midfield. In this case, the criteria for the forwards are as follows:

1. All players must maintain the formation: $x_{1i} = \left\|\vec{p}_i - \vec{p}_{rec}\right\|.$
2. All players must maintain open space: $x_{2i} = \max\left(0, \left(d_{tr} - \left\|\vec{p}_i, \vec{p}_{closest\ opponent}\right\|\right)\right)$. For all points with the distance greater than the threshold the value of the parameter is zero.

3. The midfields must be ready to support the attackers for the defence penetration, so they must move forward $x_{3i} = \left| \vec{p}_i^{\,x} - X_{offside} \right|$. The smaller these numbers are, the better the point.

4. Midfields must keep open to the predicted position of the designated partner to be ready to make a forward pass, $x_{4i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p^2_{desig\_patner}}, \vec{p}_{closest\ opponent}\right)\right)\right)$.

5. Midfields must be open to a direct pass to support the player controlling the ball if a forward pass is impossible, $x_{5i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{ball}}, \vec{p}_{closest\ opponent}\right)\right)\right)$.

Altogether, we have five criteria for all midfields when the ball is in the middle zone.

3.2.2.4.3 Criteria for the case when the ball is in the offensive zone

The ball is in the offensive zone when it is controlled or will be intercepted by a forward. The main task for the midfields in this situation is to support the forwards creating "depth" for the attack. Criteria for the midfields are as follows:

1. All players must maintain the formation: $x_{1i} = \left\| \vec{p}_i - \vec{p}_{rec} \right\|$.
2. All players must maintain open space: $x_{2i} = \max\left(0, \left(d_{tr} - \left\| \vec{p}_i, \vec{p}_{closest\ opponent} \right\|\right)\right)$. For all points with the distance greater than the threshold the value of the parameter is zero.
3. The midfields must be ready to support the attackers for the defence penetration, so they must move forward $x_{3i} = \left| \vec{p}_i^{\,x} - X_{offside} \right|$. The smaller these numbers are, the better the point.

4. If the ball is controlled or is going to be intercepted by the designated partner or (for the wing midfields only) by the central forward, all midfields must keep open for direct pass. Notice that the player must keep open to the predicted ball position, not the designated partner position since the partner is chasing the ball

and does not adjust its position $x_{4i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{ball}}, \vec{p}_{closest\ opponent}\right)\right)\right)$.

If the ball is controlled or is going to be intercepted by the wing forward that is not the designated partner (and not the central forward for the wing midfields), all midfields must keep open for the designated partner rather than for the ball. $x_{4i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{desig\_patner}}, \vec{p}_{closest\ opponent}\right)\right)\right)$.

Altogether, we have four criteria for the midfields when the ball is in the offensive zone and the 4th criterion differs depending on the situation.

3.2.2.4.4 Pressure

Just like in the case of simple collaboration, the midfields perform pressure when the ball is controlled by the opponents. The criteria are the same as for simple collaboration.

3.2.2.5 Criteria for defenders

Since the defenders rarely participate in attacking actions the criteria for defenders are similar to the criteria for simple collaboration.

1. All players must maintain the formation: $x_{1i} = \left\|\vec{p}_i - \vec{p}_{rec}\right\|$.
2. All midfields must be open for a forward pass from the goalie: $x_{2i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{ball}}, \vec{p}_{closest\ opponent}\right)\right)\right)$. For all points with the distance greater than the threshold the value of the parameter is zero.
3. All players must maintain open space: $x_{3i} = \max\left(0, \left(d_{tr} - \left\|\vec{p}_i, \vec{p}_{closest\ opponent}\right\|\right)\right)$. For all points with the distance greater than the threshold the value of the parameter is zero.

4. The defenders must keep open to the designated partner: $x_{4i} = \max\left(0, \left(d_{tr} - d\left(\overline{p_i p_{desig\_patner}}, \vec{p}_{closest\ opponent}\right)\right)\right)$. For all points with the distance greater than the threshold the value of the criterion is zero.

Altogether, we have established four criteria for estimation of the possible location points for the defenders in attack.

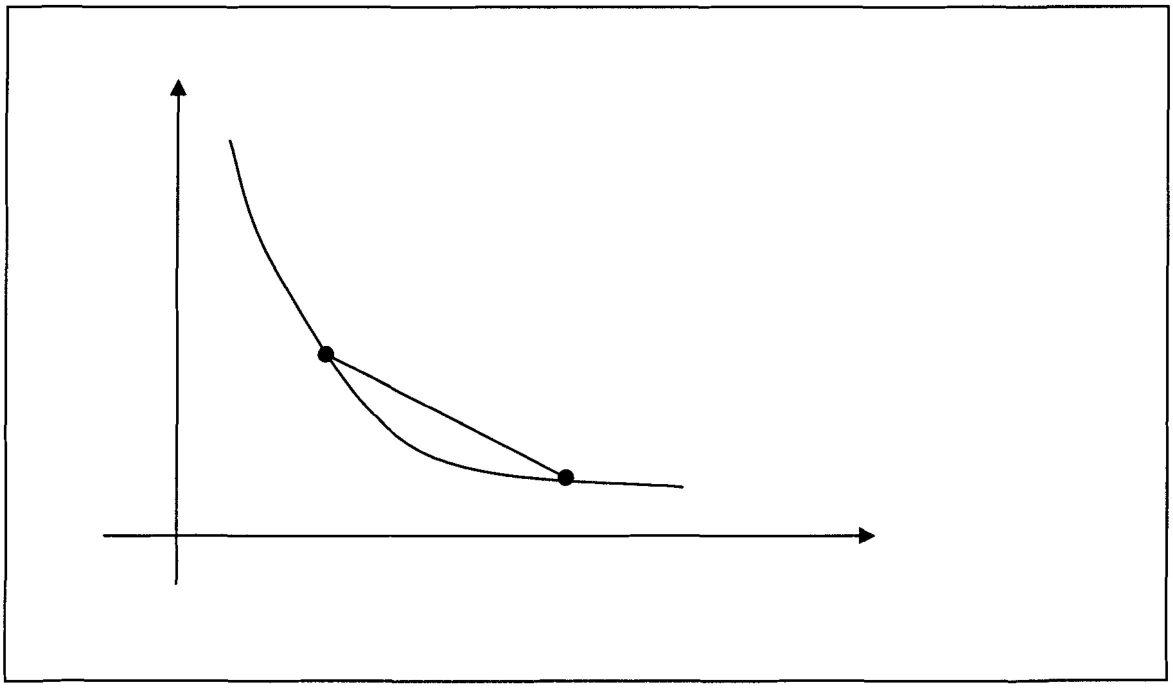
3.3 The decision making algorithm

We have specified the positioning problem as a Multicriteria Decision Making (MCDM) problem. The player has a feasible set of points on the field which is a subset of the decision space. Each point in the feasible set is mapped into the objective or criteria space. We constructed all the criteria for minimisation in the sense that the smaller the value of a criterion is, the better the position. The criteria, in general, are incomparable and conflicting; this means we are unable to minimise all the criteria simultaneously. For this type of problems, the general approach is to find the set of non-dominated or Pareto-optimal points and then apply some method for choosing the best compromise point from this set. We refer to this set as the Pareto-set. Many methods have been developed for different types of Multicriteria Optimization problems. To choose the suitable method, we must analyse the type of the problem we have.

3.3.1 Problem analysis

The type of the problem depends, in particular, on the type of the Pareto-set, so the Pareto-set types must first be described. A Pareto-set can be convex or non-convex. For a convex Pareto-set any two points in the set can be connected by a straight line segment which does not cross the Pareto frontier (Fig. 3.1).

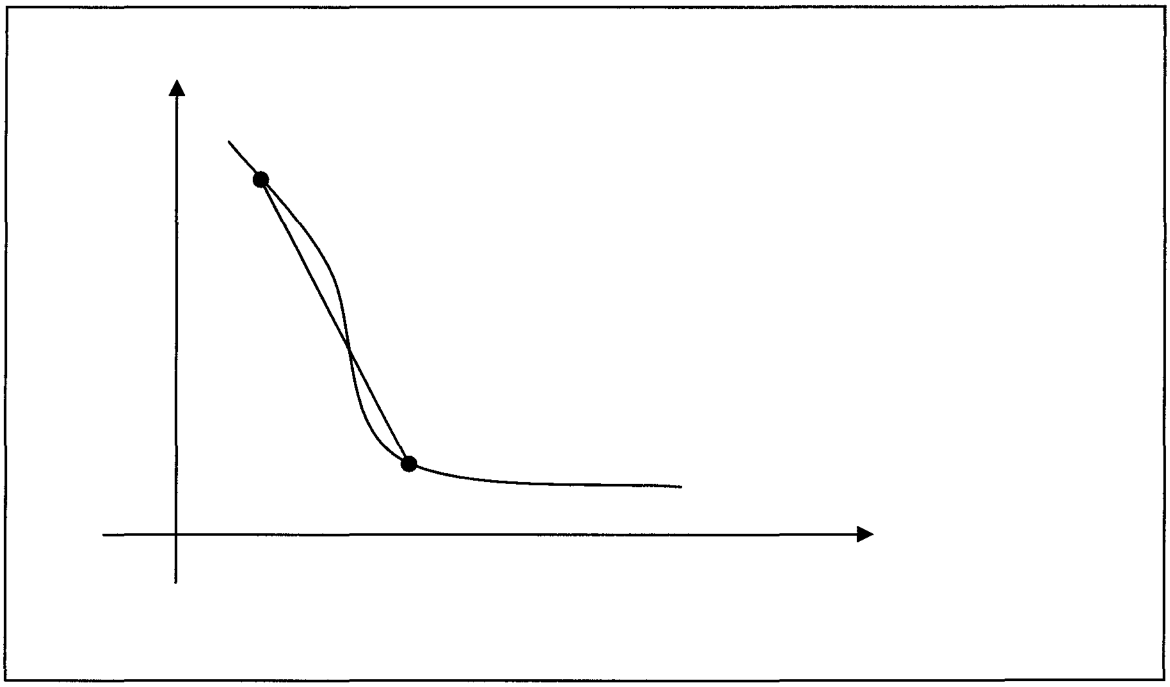
Figure 3.10 Example of a convex Pareto-set



For problems with convex Pareto-set, the weighted sum method can be applied.

For a non-convex Pareto-set, there are at least two points in the set which can be connected by a straight line which does cross the Pareto frontier.

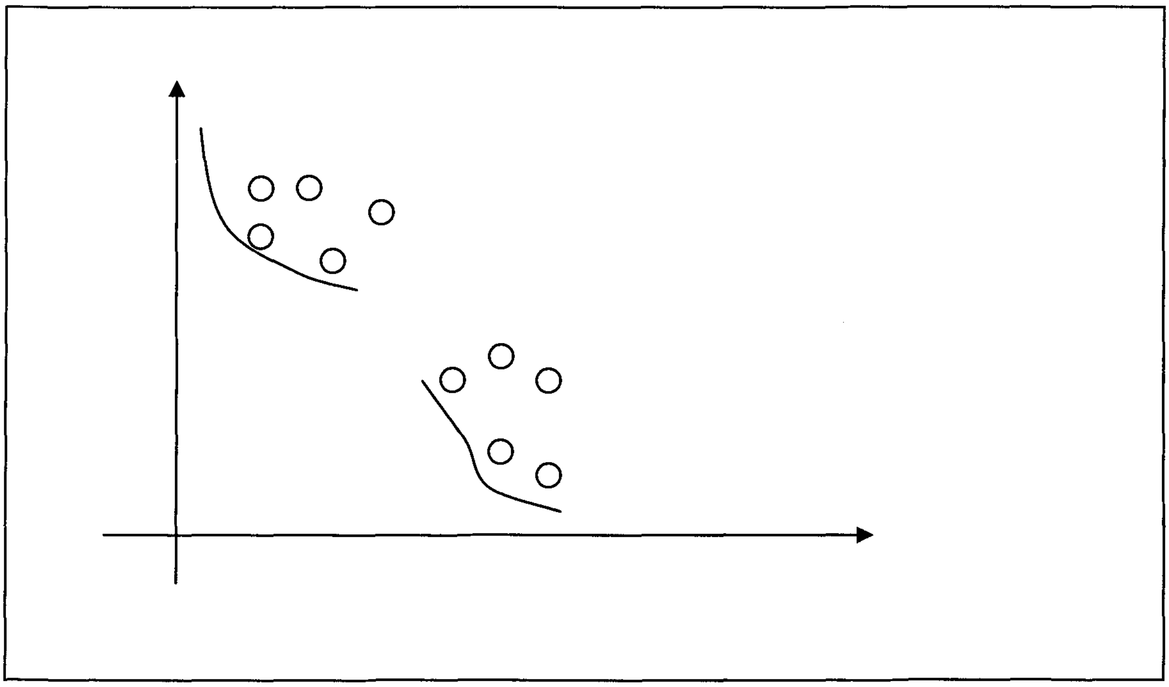
Figure 3.11 Example of a non-convex Pareto-set



For problems with non-convex Pareto-set, several methods were developed, for instance the minimax reference point method (Yang, 2000 in Liu, Yang, Whitborn, 2003). However, it requires some preference information in advance.

In the soccer game the Pareto-set also can be disconnected (Kyrylov, 2005). The feasible set itself can be connected but not necessarily convex. This non-convexity can make the Pareto frontier disconnected.

Figure 3.12 Example of a disconnected non-convex Pareto-set



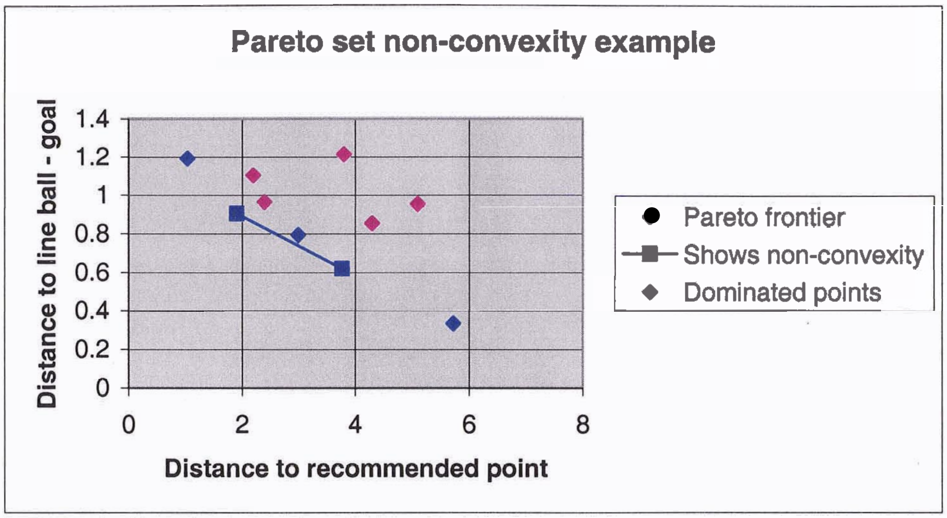
In case of the disconnected Pareto set, the minimax reference point method may fail to produce a unique solution.

Having this classification of Pareto sets, we can then classify our problem. We replaced the continuous field space by a grid of points and restricted the decision space of a player to a feasible set, so our problem is discrete and finite.

In this implementation, the objective functions for the points in the set are some distances. In general, for different points in the set a particular criterion can be a distance to different objects like the distance to the closest opponent. For different points, the closest opponents could be different opponent players. Having only this reason we can conclude that the objective functions are not only non-linear but also non-continuous; therefore the problem that we are dealing with is a non-linear non-convex MCDM problem.

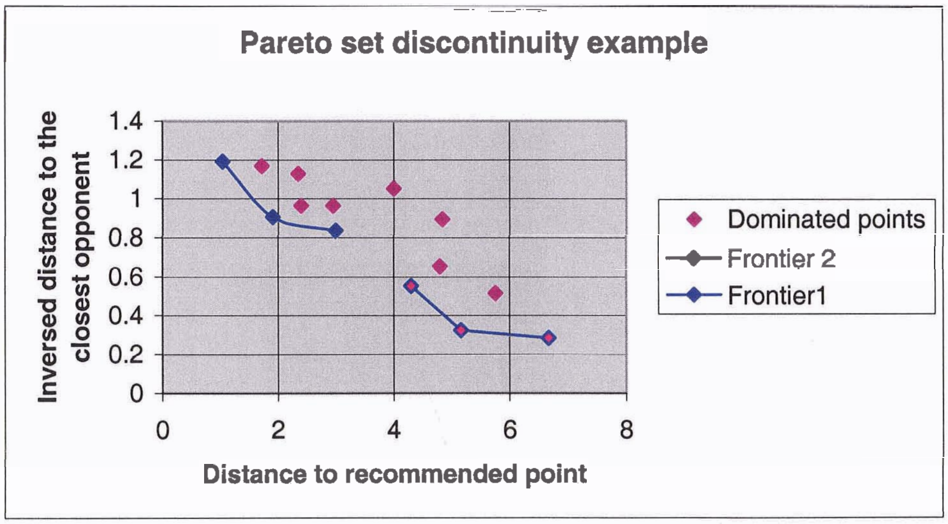
To prove that some Pareto sets for the problem are non-convex and disconnected, we first make the assumption that all the Pareto-sets for the problem are connected and convex. Then, we present some counterexamples to show that is not the case.

Figure 3.13 Example of a non-convex Pareto-set for an attacker



Figures 3.13 and 3.14 present examples of non-convex Pareto-sets in case of two parameters for an attacker trying to keep close to the recommended point and to the offside line.

Figure 3.14 Example of a disconnected Pareto-set for an attacker



Finally, we can classify the problem as a discrete, finite, non-linear, non-continuous, and non-convex problem.

3.3.2 Pareto-set construction and sequential elimination

To make the final choice from the feasible set, the player can first find the points which are definitely "better" than the others. Following the Pareto optimality principle, the "better" points are the points which are not dominated by the others. I use the definition of non-dominancy to find the set of Pareto-optimal solutions or Pareto-set.

3.3.2.1 Pareto-set construction algorithm

I used a simple and straightforward algorithm for the Pareto-set construction implemented by Dr. V. Kyrylov. The algorithm is based on the definition of strong non-dominance for two points:

Definition 3.1 A point p^1 is not dominated by point p^2 if there is at least one criterion $c_i(p)$ such as $c_i(p^1) \le c_i(p^2)$, where $c_i(p)$ $(i = 1,2,\ldots k)$ are assumed for minimisation.

This definition allows for an easy comparison between two points according to dominance. Using this definition, which was implemented as a function, the following Pareto-set construction algorithm was implemented (V.Kyrylov, 2005):

```
set ParetoSetConstructinAlgorithm(set FEASIBLE SET)
BEGIN
     Create set ParetoSet (empty set)
     FOR (every element A in FEASIBLE SET)
          Mark A as nondominated
          FOR (every element B in FEASIBLE SET)
               IF (A is dominated by B)
                    Mark A as dominated
                    Break the loop
               END IF
          END FOR
          IF(A is nondominated)
               Add A to ParetoSet
```

```
        END IF
    END FOR
    RETURN ParetoSet
END
```

3.3.2.2 Sequential elimination algorithm

Once we have obtained the Pareto-set, the last step in the process of finding the optimal position for the player is choosing the point from the constructed Pareto-set. Professor Kyrylov, the primary academic supervisor of this study, has recently proposed a method called "*the sequential elimination of the poorest alternative*" (Kyrylov, 2006). Because this algorithm does not rely on any information about objective functions, it is applicable to any MCDM problem having a finite Pareto-set. The computational complexity of the algorithm is $O\left(K^2\right)$, where K is the number of elements in the Pareto-set. Each criterion is given some relative weight. Kyrylov describes the algorithm as follows:

> "The key assumption is that each criterion has its relative weight; in our case this information is reflecting the preferences of the developer of the decision making algorithm. So let **X** be the set of all alternatives, P $\subset$**X** be the Pareto set, **x**$\in$**X** be a decision vector, $g1(\mathbf{x}),\ldots \mathbf{gn}(\mathbf{x})$ be the criteria functions (all of which we want to minimize), and $w1,\ldots,wn$ be the non-negative weights whose sum is 1. The algorithm is ...

```
S := P;
for ( k := 1 to K-1 )
{
 With probability wj, randomly select j-th criterion;
 Find the element x ∈ S having the maximal value of
gj(x);
 remove x from S;
}
return the last remaining element in S"
```

(Kyrylov, 2006, p. 9)

The algorithm eliminates one element from the Pareto-set at a time and there are $K-1$ iterations. In every step, one criterion is randomly selected according to its weight. Since the weights are used as the probability distribution, criteria that have greater weight are chosen more frequently. On each step the point having the greatest value of the current criterion is removed from the resulting set. The last remaining element is the approximation of the optimal solution of the problem. When K is increasing, the approximation converges to the optimal solution.

In the current implementation, it is almost impossible to decide which criterion is more important than others. For this reason, a simplified version of this algorithm was used. Criteria for elimination were not given any weights and were just used in turns, starting always from distance to the recommended point. This is similar to assuming that their weights are equal.

If we are able to achieve complete precision of the predicted state of the environment, we could argue that the computed optimal point would stay the same in every step in the prediction period. Unfortunately, such exact precision is impossible and the predicted state is refined with each simulation step. Decision robustness is very important for the good performance of the team. To increase the robustness of the decision the following method was applied: let the current simulation step be step number i, current Pareto–set P_i, and the optimal point for the previous simulation cycle $\vec{p}_{i-1}$. Then, for every simulation step i:

if($\vec{p}_{i-1} \in P_i$)

{

$\vec{p}_i = \vec{p}_{i-1}$

}

In other words, if the optimal point of the previous simulation cycle is in the Pareto-set for the current simulation cycle, it is thought to be good enough to serve as the new optimal point.

Figure 3.15 Example of the Pareto-set and the optimal point

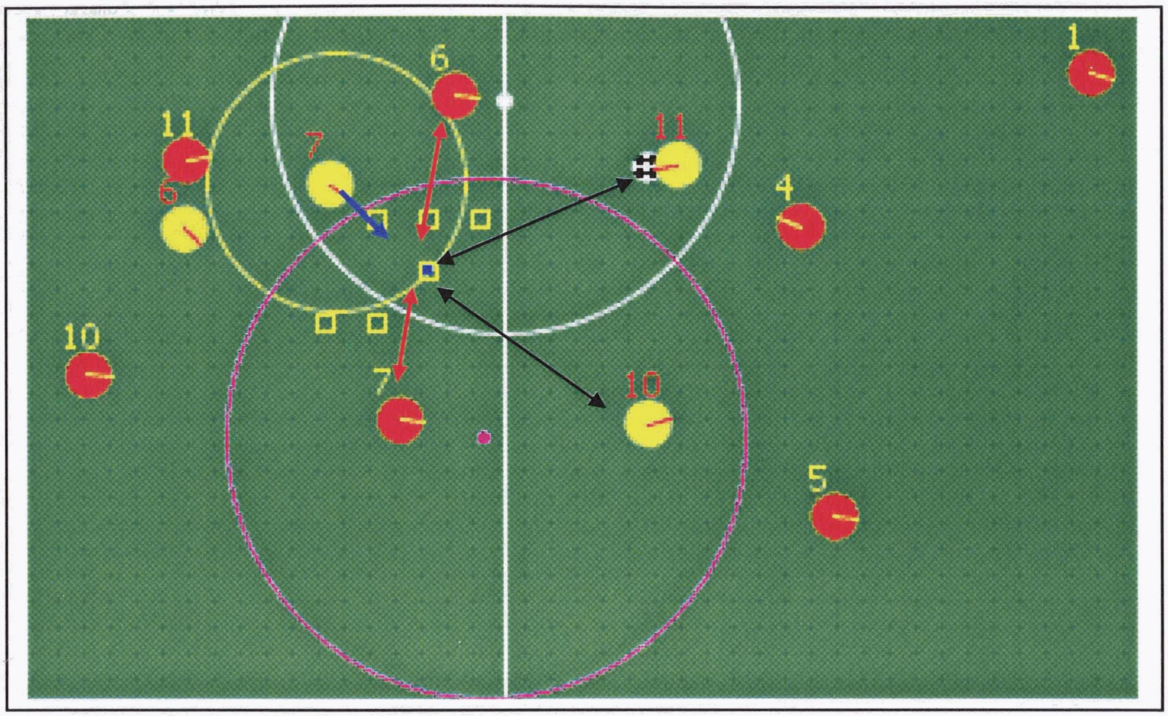


Figure 3.15 shows the Pareto set and the optimal point for the yellow midfielder #7 which is looking for the optimal point by using the criteria for advanced collaboration. Currently, the player supports the attack. Empty yellow squares represent the Pareto-set; the yellow square with a blue dot inside represents the optimal point. We can see that the optimal point is open to the ball and the designated partner (black arrows), far enough from the closest opponents (red arrows), and takes into consideration the recommended point (white arrow). Yellow player #11 is about to intercept the ball. It takes about 4-6 cycles for a player to perform an action like a pass. Player #7 is about 3 metres from the desired position heading directly to it (blue arrow). This means that when the player controlling the ball is ready to finalize its action the positioning yellow player #7 will be at or very close to the optimal position, ready to support the attack.

3.4 Research tools - visualization

Even a perfect theory can produce unexpected results if it is applied incorrectly. This research is partly a study of simulation, so validation and verification processes must be applied to the model. In particular, it is necessary to verify the prediction methods and the choice of the criteria. The easiest and the most efficient way to do that is to observe visually the predicted positions for dynamic objects (for instance the ball or a player), the Pareto-set, and the optimal point.

With this purpose an additional tool was added to the simulation monitor. The menu item "Show Pareto" turns on the visual representation of the Pareto-set with the optimal point and, by default, the responsibility area for one of the players. Since any changes in the user interface are time consuming, I did not introduce any other user controls. Nevertheless, by changing several lines of code we can replace the responsibility area representation by the predicted position of the ball or the predicted position of any of the players. Also, the player whose Pareto-set is displayed can be changed in the same way.

The standard player communication system was used as a channel for communicating the information about the player's world model.

The communication system is designed to provide some restricted communication between players. According to the RoboCup rules, the players can not communicate directly. A player sends its messages to the server, which broadcasts them to other players.

In the current implementation, the player simply sends the information about the Pareto-set as a text message in a particular form to the server. The server, in turn, relays it to the monitor, and the monitor displays the received information.

Figure 3.16 Visualization

4 EXPERIMENTAL RESULTS AND ANALYSIS

4.1 Performance indicators

Before we choose the performance indicators for evaluating the proposed method, we should ask ourselves what exactly is going to be measured. The soccer game doesn't have any explicit measurements that would fully characterise player positioning performance. Even the game score often can be deceiving; it often happens that an undeniably stronger team loses to a weaker team even if it has apparent prevalence during the game. Using the score for measuring player positioning performance makes sense only when it is possible to run at least 50-100 games for each set of conditions. Luckily, over the years several indicators have been developed that are implicitly related to player positioning. Each of them was intended to measure some aspect of the team performance. I have used the same indicators, which are: game score, territorial prevalence, ball possession, and number of shots to goal.

In all experiments, the same simulated team played on both sides. The only difference between the two teams was that one team had the improved player positioning algorithm. All the remaining features in both teams were same. This remark is important because the performance indicators that we have selected should be able to measure the difference in player positioning rather than other features, such as goal scoring or ball passing algorithms.

4.1.1 The game score

The game score is the overall indicator of team performance. As previously mentioned, it can be deceiving; mostly score is applied to official competitions, when the competing teams do not have a chance to play against the same opponent more than one or two times. On the contrary, if the same two teams play a series of games, as the common practice during Stanley Cup play-offs goes, it is more likely that the better team wins more games.

4.1.2 The territorial prevalence

The territorial prevalence shows better team organisation. While the game score strongly depends on the quality of the scoring algorithm and the quality of defence, the territorial prevalence relies more on positioning and passing. I will measure the territorial prevalence in the number of simulated seconds which the ball was located on the left or the right half of the field.

4.1.3 The ball possession

The ball possession also shows the quality of team organisation. In some sense, it is the complement to the territorial prevalence. This indicator reflects mostly the quality of the passes, and to some extent player positioning. We can imagine that with both good passing and positioning a team can quickly deliver the ball to the opponents' goal, organise an attack, and try to score. However, after an attempt to score the team often loses the control of the ball and it can happen that the team that is unable to quickly penetrate the opponent's defence will have better ball possession time. For this reason, I will use this indicator to evaluate player positioning only in combination with the others. Since all the teams don't use dribbling, this indicator is measured in number of kicks made by the players of the team.

4.1.4 The number of shots to goal

The number of shots to goal shows the quality of team organisation in the final stage of the attack. This organisation includes the ability to penetrate the opponent defence using positioning and passing. I did not implement any special tactic schemes for the defence penetration and all the teams have the same simple passing algorithm. For this reason, in this implementation I have concentrated on offensive positioning. The number of shots to goal is the second most important measurement of the team performance. This performance indicator includes four cases: (1) all situations when the player has shot at the goal, but the ball was intercepted by any opponent except the goalie; (2) all cases when the goalie caught the ball; (3) all cases when the ball crossed the goal line outside the opponents' goal but close enough to a goal post (within the distance equal to the width of the goal), and (4) the actual goals.

4.2 Performance analysis methods

4.2.1 Experiments

For the experiments three different teams: control team (team 1), experimental team with so-called 'simple' collaboration (team 2), and experimental team with 'advanced' collaboration (team 3). The optimality criteria used in teams 2, 3 are explained in Section 3.2. All the teams are identical except the *positioning of players in the attack*. The control team players used for positioning the respective *recommended* locations calculated as weighted sum of the home position and the location of the ball. Players in the experimental team with simple collaboration used the optimality criteria for simple collaboration and multicriteria decision analysis methods while performing attacks; in all the rest situations they were using the recommended positions. Players in the experimental team with advanced collaboration were using the optimality criteria for advanced collaboration and multicriteria decision analysis methods while performing attacks; otherwise the recommended positions were used.

To gather statistics, 100 games have been run in each pair: team 1 vs. team 2, team 1 vs. team 3, and team 2 vs. team 3. Each game was of the RoboCup format (two halves 5 minutes long each). Each team played 50 games on the left side of the field and 50 games on the right side of the field. It is natural to assume that, by design, the measurements of the performance indicators obtained in different games are statistically independent.

4.2.2 Hypothesis testing

To make sure that the proposed methods actually yield improved performance, we must show that the difference between values of the measured performance indicators, if any, is statistically significant. The nature of these indicators suggests the equal number of measurements for each team. Therefore we are interested in the construction of a confidence interval for the difference of the mean values of each indicator. If zero value lies outside the confidence interval, there is a statistically significant difference between the measured performance indicators. This is exactly the hypothesis we want to test.

This pattern fits for the Paired-*t* Confidence Interval Method, as neither the expectations nor the variances of the performance indicator probability distributions are known. This method is especially useful when the expectations are different, so the null hypothesis stating that the expectations are equal is false (Law&Kelton, 2000). We will use this method when comparing an improved team against the control team. In contrary, when we compare two improved teams it is sufficient to perform a hypothesis test to show that the observed difference is significantly different from zero. In this case we will apply the T-Test: Paired Two Sample for Means, included in the Excel statistics package. Since the variances of the performance indicator probability distributions are unknown, the test for uneven variances will be applied.

For $i = 1,2$ let $X_{i1}, X_{i2}, \ldots, X_{in}$ be a sample of n independent and identically distributed observations collected from i-th system. If $\mu_i = E(X_{ij})$ is the expectation we are interested in, we want to construct a confidence interval $\xi = \mu_1 - \mu_2$. Thus we define new set of observations, $Z_j = X_{1j} - X_{2j}$, for $j = 1,2,\ldots,n$; let their expectation be $\xi = E(Z_j)$.

We use the average to estimate the latter expectation:

$$\overline{Z}(n) = \frac{\sum_{j=1}^{n} Z_j}{n}. \tag{4.1}$$

The variance of this estimate is,

$$\overline{Var}\left[\overline{Z}(n)\right] = \frac{\sum_{j=1}^{n}\left[Z_j - \overline{Z}(n)\right]^2}{n(n-1)}. \tag{4.2}$$

Thus the $100(1-\alpha)$ percent confidence interval can be formed as

$$\overline{Z}(n) \pm t_{n-1,1-\alpha/2}\sqrt{\overline{Var}\left[\overline{Z}(n)\right]}. \tag{4.3}$$

or

$$\overline{Z}(n) \pm t_{n-1,1-\alpha/2} \frac{S}{\sqrt{n}} \tag{4.4}$$

where S is the standard deviation for the estimate.

Notice, that X_{1j} and X_{2j} do not need to be exactly normally distributed or have equal variances. If Z_j's are normally distributed, this confidence interval is exact. Otherwise, the Central Limit Theorem implies that it is near $1 - \alpha$ for large n.

4.3 Offensive positioning with simple collaboration

4.3.1 Statistics

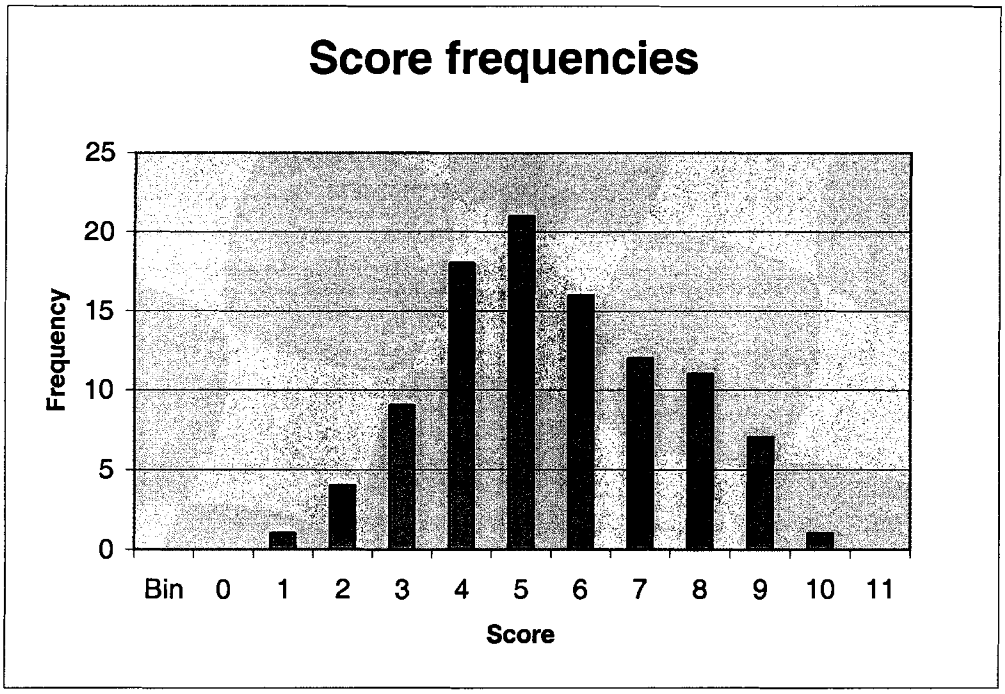
The games results by the team with simple collaborative player positioning (experimental team) versus the control team with basic positioning are presented in Table 4.1.

Table 4.1 Game statistics for the team with simple collaboration positioning vs. control team

	Score	Ball on side	Shots to goal	Ball possession
Mean experimental team	**5.54**	**175.02**	**20.68**	**466.93**
control team	0.34	299.24	12.17	348.62
Median experimental team	**5**	**172**	**21**	**464**
control team	0	301	11	347
Standard Deviation experimental team	**1.95**	**25.39**	**5.72**	**56.75**
control team	0.55	24.71	4.16	29.89
Sample Variance experimental team	**3.81**	**644.40**	**32.75**	**3220.11**
control team	0.31	610.57	17.29	893.27
Range experimental team	**9**	**195**	**33**	**581**
control team	2	198	25	172
Minimum experimental team	**1**	**88**	**3**	**340**
control team	0	193	5	283
Maximum experimental team	**10**	**283**	**36**	**921**
control team	2	391	30	455
Game count	100	100	100	100

Data histogram show that the distribution of each performance indicator is close to normal distribution. One example is presented in and Figure 4.1.

Figure 4.1 Score frequencies histogram for the experimental team.

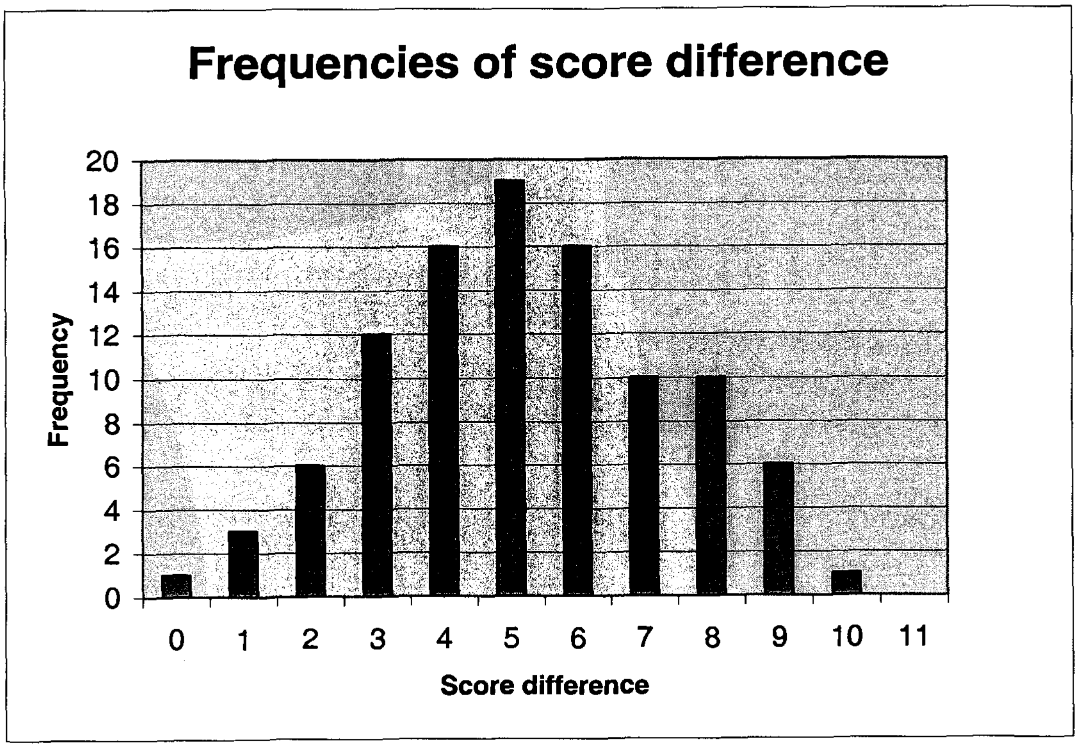


Since we are going to apply the Paired-*t* Confidence Interval Method, we need to construct data for the performance indicators differences. These data are given in Table 4.2. Their distribution is also close to normal (Figure 4.2).

Table 4.2 Games statistics for the team with simple collaboration positioning vs. control team

	Score difference	*Territorial prevalence*	*Shots to goal difference*	*Ball possession difference*
Mean	5.20	124.22	8.51	118.31
Median	5.00	127.00	8.00	121.00
Standard Deviation	2.14	49.59	7.58	69.34
Sample Variance	4.59	2458.84	57.46	4807.91
Range	10	393	57	647
Minimum	0	-90	-27	-115
Maximum	10	303	30	532
Game count	100	100	100	100

Figure 4.2 Score difference frequencies histogram.



4.3.2 Confidence interval calculation

We will construct the confidence intervals according to (4.3). For 99% confidence, we have $t_{100-1,1-0.05/2} = 2.626$.

For the score difference, we have $\overline{Z}(100) = 5.2$ and $\sqrt{Var\left[\overline{Z}(100)\right]} = \frac{2.14}{\sqrt{100}} = 0.214$. According to (4.3), the 99% confidence interval is $\left[4.64;5.76\right]$. Since zero is outside this interval, with 99% confidence we can say that the score difference is statistically significant in favour of the experimental team playing against the control team.

For the territorial prevalence, we have $\overline{Z}(100) = 124.22$ and $\sqrt{Var\left[\overline{Z}(100)\right]} = \frac{49.59}{\sqrt{100}} = 4.96$. According to (4.4), the 99% confidence interval is $[111.20;137.25]$. Since zero value is outside the interval, with 99% confidence we can say that the territorial prevalence is statistically significant for the experimental team.

For the ball possession difference, we have $\overline{Z}(100) = 118.31$ and $\sqrt{Var\left[\overline{Z}(100)\right]} = \frac{69.34}{\sqrt{100}} = 6.93$. According to (4.3), the 99% confidence interval is $[100.11;136.51]$. Since zero value is outside the interval, we can say that for the experimental team the ball possession prevalence is statistically significant with 99% confidence.

For the shots to goal difference we have $\overline{Z}(100) = 8.51$ and $\sqrt{Var\left[\overline{Z}(100)\right]} = \frac{7.58}{\sqrt{100}} = 0.76$. According to (4.3), the 99% confidence interval is $[6.51;10.50]$. Since zero value is outside the interval, we can say that for the experimental team the prevalence in shots to goal is statistically significant with 99% confidence.

4.3.3 Conclusion

The presented statistical data indicate that the experimental team with simple collaboration outplays the control team with basic player positioning in all aspects of the game.

4.4 Offensive positioning with advanced collaboration

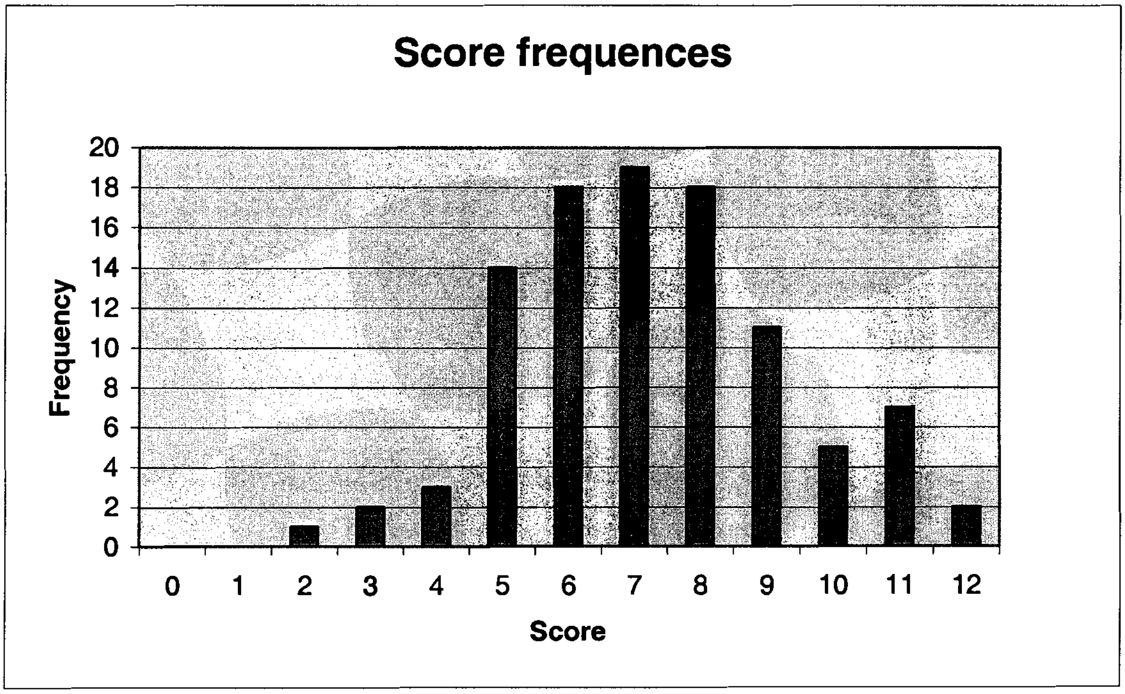
4.4.1 Statistics

The games by the team with advanced collaborative player positioning (experimental team) versus the control team with basic positioning showed results which are presented in Table 4.3. Data histograms show that the distributions of all measured performance indicators are close to normal (see an example in Figure 4.3).

Table 4.3 **Game statistics for the team with advanced collaboration positioning vs. the control team**

	Score	*Ball on side*	*Shots to goal*	*Ball possession*
Mean experimental team	**7.25**	**189.41**	**23.9**	**424.61**
control team	0.43	281.56	13.13	355.61
Median experimental team	**7**	**191.5**	**24**	**423.5**
control team	0	281	13	355
Standard Deviation experimental team	**2.06**	**22.61**	**5.72**	**24.12**
control team	0.62	21.06	4.14	30.41
Sample Variance experimental team	**4.25**	**511.21**	**32.68**	**581.86**
control team	0.39	443.60	17.14	924.48
Range experimental team	**10**	**150**	**30**	**114**
control team	3	147	25	203
Minimum experimental team	**2**	**137**	**6**	**371**
control team	0	188	4	290
Maximum experimental team	**12**	**287**	**36**	**485**
control team	3	335	29	493
Count	100	100	100	100

Figure 4.3 **Score frequencies histogram for the experimental team.**



Since we are going to apply the Paired-*t* Confidence Interval Method, we need to construct data for the differences. These data are given in Table 4.4. Their values obtained in different games are statistically independent. The distributions are also close to normal.

Table 4.4 Game statistics for the team with advanced collaboration positioning vs. control team

	Score difference	*Territorial prevalence*	*Shots to goal difference*	*Ball possession difference*
Mean	6.82	92.15	10.77	69.00
Median	7	89.50	9.50	68.50
Standard Deviation	2.25	43.25	8.12	46.12
Sample Variance	5.08	1870.96	65.88	2127.09
Range	11	297	51	303
Minimum	1	-99	-23	-116
Maximum	12	198	28	187
Count	100	100	100	100

Figure 4.4 Score difference frequencies histogram.

Error! Objects cannot be created from editing field codes.

4.4.2 Confidence interval calculation

We will construct the confidence intervals according to the equation (4.4). For the score difference we have $\overline{Z}(100) = 6.82$ and $\sqrt{Var\left[\overline{Z}(100)\right]} = \frac{2.25}{\sqrt{100}} = 0.26$. According to (4.4), the 99% confidence interval is $\left[6.14;7.50\right]$. Since zero is outside the interval, with 99% confidence we can say that the score difference is statistically significant in favour of the experimental team. This advantage is somewhat greater than for the team using simple collaborative positioning.

For the territorial prevalence, we get $\overline{Z}(100) = 92.15$ and $\sqrt{Var\left[\overline{Z}(100)\right]} = \frac{43.25}{\sqrt{100}} = 4.33$. According to (4.3), the 99% confidence interval is $[80.78;103.52]$. Since zero value is outside the interval, we can say that the territorial prevalence is statistically significant for the experimental team with 99% confidence.

For the ball possession difference, we have $\overline{Z}(100) = 69.00$ and $\sqrt{Var\left[\overline{Z}(100)\right]} = \frac{46.12}{\sqrt{100}} = 4.61$. According to (4.3), the 99% confidence interval is $[56.89;81.10]$. Since zero value is outside the interval, with 99% confidence we can say that the experimental team has prevailing ball possession.

For the shots to goal difference we have $\overline{Z}(100) = 10.77$ and $\sqrt{Var\left[\overline{Z}(100)\right]} = \frac{8.12}{\sqrt{100}} = 0.81$. According to (4.3), the 99% confidence interval is $[8.64;12.90]$. Since zero value is outside the interval, we can say that the experimental team has prevalence in the number of shots to goal with 99% confidence.

4.4.3 Conclusion

The statistical data indicate that the experimental team with advanced collaboration outplays the control team in all aspects of the game. This advantage appears to be greater than that of the team with simple collaborative player positioning for scoring and shooting but less in territorial prevalence and ball possession. However, without additional experiments and testing we cannot say for sure that these differences are statistically significant. This issue is addressed in the following sections.

4.5 Advanced collaboration/simple collaboration compared with control team

4.5.1 Hypothesis testing

Essential statistical data for the team with simple collaboration and the team with advanced collaboration when playing against the control team are given in Tables 4.1 and 4.3. Since we compare two experimental teams we are interested in testing the hypothesis if the means of the performance indicators are different. So we apply t-Test: Paired Two

Sample for Means with hypothesis of equivalence of two means. T-Test: Paired Two Sample for Means with 95% confidence produces following results.

Table 4.5 T-test results for score

	Advanced collaboration team score	*Simple collaboration team score*
Mean	7.25	5.54
Variance	4.25	3.806465
Observations	100	100
Pearson Correlation	-0.04646021	
Hypothesized Mean Difference	0	
Df	99	
t Stat	5.889482279	
P(T<=t) two-tail	5.33529E-08	
t Critical two-tail	1.9842169	

Table 4.6 T-test results for shots to goal

	Advanced collaboration team shots to goal	*Simple collaboration team shots to goal*
Mean	23.9	20.68
Variance	32.67676768	32.74505
Observations	100	100
Pearson Correlation	-0.017663162	
Hypothesized Mean Difference	0	
df	99	
t Stat	3.946322211	
P(T<=t) two-tail	0.000148466	
t Critical two-tail	1.9842169	

Table 4.7 T-test results for territorial prevalence

	Ball on side of the control team when playing against advanced collaboration team	***Ball on side of the control team when playing against simple collaboration team***
Mean	281.56	299.24
Variance	443.6024242	610.5681
Observations	100	100
Pearson Correlation	-0.006258204	
Hypothesized Mean Difference	0	
df	99	
t Stat	-5.428616481	
P(T<=t) two-tail	4.05127E-07	
t Critical two-tail	1.9842169	

Table 4.8 T-test results for ball possession

	Ball played by advanced collaboration team (times per game)	***Ball played by simple collaboration team (times per game)***
Mean	424.61	466.93
Variance	581.8564646	3220.106
Observations	100	100
Pearson Correlation	0.015019097	
Hypothesized Mean Difference	0	
df	99	
t Stat	-6.900854655	
P(T<=t) two-tail	4.96374E-10	
t Critical two-tail	1.9842169	

The test results show that the mean difference is statistically significant for all performance indicators with the confidence at least 95%.

4.5.2 Conclusion

The statistical data indicate that the team with advanced collaboration outplays the team with simple collaboration in the goals scored and shots to goal. However, the team with simple collaboration 'outplays' the team with advanced collaboration in terms of ball possession and territorial prevalence, which appears to be counter-intuitive.

These results and results of visual observations allow us to affirm that the team with advanced player collaboration acts in a more effective way. It takes less time for this team to deliver the ball into the attack zone and to an attacker to get the ball in the shooting position. This is the reason why the team with advanced collaboration yields in territorial prevalence and ball possession to less sophisticated team. The former makes smaller number of passes and spends less time on the opponents' half of the field before one of the players is able to shoot to goal, while the latter tends to have more chances to get the ball possession after the successful shots on the goal by the opponent.

4.6 Advanced collaboration vs. simple collaboration

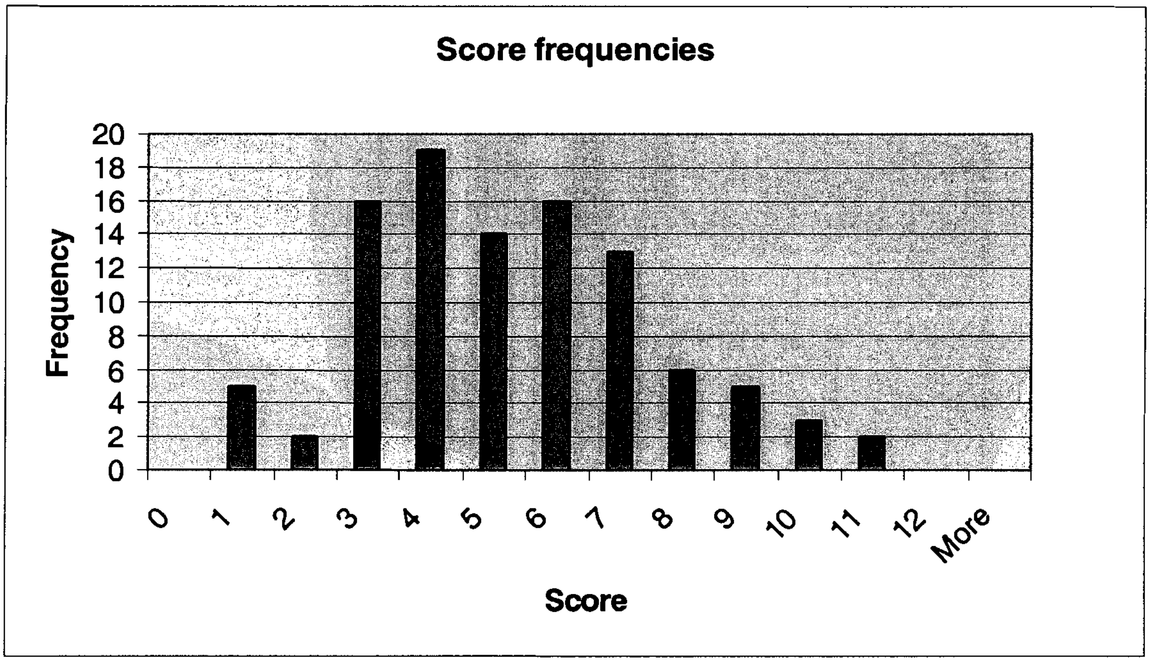
4.6.1 Statistics

The games of the team with advanced collaborative player positioning versus the team with simple collaboration showed the results presented in Table 4.9. Sample distributions of all performance indicators appear to be close to normal (see example in Figure 4.5).

Table 4.9 **Game statistics for the team with advanced collaboration vs. the team with simple collaboration**

	Team score	*Ball on team side of the field*	*Number of shots to goal*	*Ball played by team (times per game)*
Mean advanced collaboration team	5.30	223.03	22.97	375.75
simple collaboration team	2.88	246.50	16.97	401.08
Difference for means	2.42	-23.47	6.00	-25.33
Median advanced collaboration team	5	223	24	374
simple collaboration team	3	248	17	399
Standard Deviation experimental team	2.28	21.59	5.93	26.89
control team	1.64	20.33	5.36	27.60
Sample Variance experimental team	5.21	465.99	35.17	722.97
control team	2.69	413.17	28.77	761.55
Range experimental team	10	123	27	125
control team	8	113	25	140
Minimum experimental team	1	148	9	315
control team	0	200	5	326
Maximum experimental team	11	271	36	440
control team	8	313	30	466
Count	101	101	101	101

Figure 4.5 Score frequencies histogram for the advanced collaboration team.



4.6.2 Hypothesis testing

T-Test: Paired Two Sample for Means with 95% confidence produces the results shown in Tables 4.10 -4.13.

Table 4.10 T-test results for score

	Advanced collaboration team score	*Simple collaboration team score*
Mean	5.29703	2.881188
Variance	5.210891	2.685743
Observations	101	101
Pearson Correlation	-0.19095	
Hypothesized Mean Difference	0	
Df	100	
t Stat	7.95054	
P(T<=t) two-tail	2.9E-12	
t Critical two-tail	1.983971	

Table 4.11 T-test results for shots to goal

	Advanced collaboration team shots to goal	*Simple collaboration team shots to goal*
Mean	22.9703	16.9703
Variance	35.16911	28.76911
Observations	101	101
Pearson Correlation	-0.194	
Hypothesized Mean Difference	0	
Df	100	
t Stat	6.904092	
P(T<=t) two-tail	4.72E-10	
t Critical two-tail	1.983971	

Table 4.12 T-test results for territorial prevalence

	Ball on side of advanced collaboration team	*Ball on side of simple collaboration team*
Mean	223.0297	246.495
Variance	465.9891	413.1725
Observations	101	101
Pearson Correlation	-0.94049	
Hypothesized Mean Difference	0	
Df	100	
t Stat	-5.71199	
P(T<=t) two-tail	1.15E-07	
t Critical two-tail	1.983971	

Table 4.13 T-test results for ball possession

	Ball played by advanced collaboration team (times per game)	*Ball played by simple collaboration team (times per game)*
Mean	375.7525	401.0792
Variance	722.9681	761.5537
Observations	101	101
Pearson Correlation	-0.35137	
Hypothesized Mean Difference	0	
Df	100	
t Stat	-5.68301	
P(T<=t) two-tail	1.31E-07	
t Critical two-tail	1.983971	

The test results show that the mean difference is statistically significant for all performance indicators with confidence at least 95%.

4.6.3 Conclusion

The statistical data indicate that the team with advanced collaborative player positioning outplays the team with simple collaboration in goals scored and shots to gaol but yields in the ball possession and the territorial prevalence. Since the team using the advanced collaboration has better values for score and shots to goal, we can conclude that this team plays more effective way; it tends to execute a smaller number of passes before creating a shooting opportunity.

5 CONCLUSION

5.1 Research questions revisited

Now we can return to the research questions asked at the beginning of the paper. The first research question was stated as: *What generic decision making framework should be used to achieve rational player behaviour that would be applicable to positioning?* The definition of rational behaviour gives the answer to this question. Russell& Norvig (Russell& Norvig, 2003, p.972) gives the following definition for perfect rationality:

"A perfectly rational agent always acts in every instant in such a way as to maximize its expected utility given the information it has acquired from the environment"

In spite of the fact that the perfect rationality is unachievable, having this definition we can consider rational player positioning as the process of finding a point on the field which would be optimal in the sense of balancing risks and rewards which are some objective functions or criteria.

The second research question was stated as: "*How to balance rewards, risks, and costs while the player is deciding about its optimal position on the field?*" Multicriteria Decision making analysis theory can be used to solve this problem. If we are able to define some area on the field where a player will look for the solution and make the number of alternatives finite, we can state the problem of finding the optimal position on the field as a MCDA problem. We can define the set criteria for every point in the feasible set creating the criteria space and apply MCDA methods to solve the problem.

The third question was stated as: "*How to determine a reasonable time frame for positioning planning?*" The answer to this question is one of the central ideas of this research. A player is unable to plan anything using rapidly changing information about the environment. The soccer game is so dynamic that it seems impossible to recognise any period of stability. Fortunately, it appears, that not actual but some predicted state of the system can be stable for a considerable period of time. Ball motion prediction, which is rather precise, gives the time horizon for positioning planning. The prediction is easier to make when the ball is rolling free. When the ball leaves a kickable area, a player can predict the state of the environment at the moment of interception and calculate the period for planning. Then, during every simulation cycle the player simply refines the prediction and can adjust the decision. In most cases these adjustments proved to be only minor, which provides the good base for the robustness of the decisions made with the new method.

The fourth question was stated as: "*How to limit the search space for the optimal position and achieve robustness of the player positioning behaviour?*" The time horizon for positioning planning gives us a tool for substantially limiting the search space. When the time for planning is known, the player can calculate the feasible area which contains the alternative points reachable in the given time. Since every player has some area of responsibility, which it is not supposed to leave, the intersection of these two areas gives rise to the restricted search space, or the *feasible set*. To make the decisions robust, we use the predicted state of the game environment instead of the state perceived in every simulation cycle. The perceived state is used to just refine this prediction. Thus the persistence of the player behaviour is achieved.

The fifth question was stated as: *How to achieve player collaboration with the proposed decision making framework?* We see two methods to achieve the collaboration. The first method is to create a more complex decision space, considering a possible solution not as a single location on the field for a single player but as a set of locations for a group of players. This approach is a subject of future work. The second method is to introduce criteria taking in the consideration of the positions of some partners or *designated team-mates*. The method produces promising results but needs further investigation.

Now, having the research questions answered, we can pose an additional, final question: *What are the achievable benefits of the proposed methods?* One of the central benefits of the proposed methods is the option to translate humanly-formulated requirements into programming logic. For example, imagine a coach who gives one of his players the following instruction: "Keep away from the opponents, stay as close as you can to the offside line and do not forget about your base position". These instructions seem to be difficult to be implemented using traditional programming methods. Using MCDA we can translate these requirements into criteria like: "maximise the distance to the closest opponent", "minimise the distance to the offside line", and "minimise the distance to the recommended position". Then, the player can search for the point in the feasible set using the methods described above.

5.2 Future work

Some directions for future work were already mentioned in the previous section. This research did not elaborate much on the collaboration problem. The approach using the feasible set of locations for a group of players seems to be promising but requires highly efficient algorithms to overcome the computational complexity.

The prediction methods used here are rather simplified, especially methods for predicting player positions. More sophisticated methods based on opponent behaviour modeling can significantly improve the decision making mechanism and make the decisions more robust.

The soccer game simulation, as the other sports games, is about making decisions and carrying them out. Most of these decisions must be taken regarding many objectives or criteria. Professor V.Kyrylov (Kyrylov, 2006) has already performed research about application of MCDA methods for carrying out decisions. Many other types of decisions are yet to be explored. Especially interesting are the decisions involving actions of different types like the decision to dribble or to make a pass.

5.3 Conclusion

This study has shown that the MCDM methods can be successfully applied to achieve rational behaviour and multi-agent collaboration in sports game simulation. The results of the research can be used in the industry of digital games. In one of the conferences the Sr. Art Director of Electronic Arts Frank Vitz admitted that

"Nowadays we have achieved complete photo realism in the game character appearance. What we have not achieved yet, is the realism of its behavior. We just do not know how to do that."

(Frank Vitz, Sr. Art Director, Electronic Arts Canada, *New Media BC Games Workshop panel discussion*, Vancouver, BC, March 16, 2006)

Scientific research can make contribution to the solution of this problem.

APPENDIX: RAW STATISTICAL DATA

Games statistics. Simple collaboration (experim.) team vs. control team

Game #	Experim. team score	Control team score	Ball on the side of the experim. team (secs. per game)	Ball on the side of the control team (secs. per game)	Shots to goal by the control tem	Shots to goal by the experim. tem	Kicks by the experim. team	Kicks by the control team
1	4	0	170	297	12	25	456	349
2	6	0	172	295	10	18	470	321
3	4	0	202	280	19	18	412	366
4	8	0	159	306	9	18	468	319
5	4	1	144	335	7	18	474	348
6	6	0	170	295	10	18	503	293
7	6	0	177	299	15	21	474	343
8	8	0	163	307	7	21	468	324
9	5	0	190	289	9	15	455	374
10	2	1	212	272	13	12	427	382
11	9	0	188	280	12	16	442	355
12	6	1	178	294	9	28	456	381
13	2	1	211	274	14	13	415	373
14	8	0	147	317	9	24	449	322
15	4	0	178	304	8	18	408	388
16	4	0	198	284	17	21	451	372
17	8	0	178	293	13	18	476	343
18	7	0	161	304	12	21	442	338
19	4	0	185	296	11	15	475	375
20	6	0	164	313	20	18	519	344
21	5	2	206	268	20	13	425	367
22	5	0	137	342	12	18	527	331
23	7	0	156	317	13	18	462	335
24	5	0	177	302	11	21	459	364
25	4	2	201	275	26	24	456	366
26	4	0	184	298	13	36	473	364
27	3	0	208	277	14	18	424	419
28	3	0	177	306	9	12	503	368
29	6	1	142	326	6	24	486	309
30	5	0	169	311	10	30	458	347

Game #	Experim. team score	Control team score	Ball on the side of the experim. team (secs. per game)	Ball on the side of the control team (secs. per game)	Shots to goal by the control tem	Shots to goal by the experim. tem	Kicks by the experim. team	Kicks by the control team
31	4	0	283	193	30	3	340	455
32	7	0	147	314	13	21	451	311
33	10	0	167	301	14	22	476	332
34	6	1	186	287	16	27	441	320
35	3	0	217	268	19	21	413	400
36	5	0	163	316	16	23	499	338
37	6	0	173	304	10	14	502	362
38	3	0	163	322	9	21	478	364
39	5	0	204	275	15	21	437	380
40	5	0	88	391	6	36	578	283
41	3	0	152	310	7	18	476	292
42	8	0	137	334	9	24	491	311
43	6	0	166	310	11	18	523	338
44	7	1	159	307	11	26	492	345
45	6	1	153	313	13	18	475	296
46	5	0	176	303	8	24	428	349
47	8	0	154	317	9	14	462	335
48	9	0	167	301	16	22	508	311
49	6	0	159	317	8	31	419	374
50	8	0	147	324	8	27	464	334
51	4	2	213	263	17	17	423	375
52	6	0	194	283	11	20	429	375
53	5	0	193	286	11	21	443	395
54	7	0	156	314	11	21	466	352
55	9	0	179	289	15	11	423	337
56	8	0	197	269	9	24	407	351
57	6	0	129	347	12	28	511	295
58	4	1	192	287	10	15	486	375
59	5	0	161	318	9	17	483	362
60	7	0	161	312	11	16	468	330
61	8	1	178	290	12	29	439	322
62	6	0	166	310	12	28	486	358
63	4	0	194	288	19	23	442	367
64	1	1	216	271	22	21	410	408
65	4	0	193	287	12	21	478	337
66	9	1	184	280	11	18	434	321
67	3	0	200	285	13	25	446	396
68	4	1	156	323	14	21	475	316
69	9	0	157	313	9	29	449	320

Game #	Experim. team score	Control team score	Ball on the side of the experim. team (secs. per game)	Ball on the side of the control team (secs. per game)	Shots to goal by the control tem	Shots to goal by the experim. tem	Kicks by the experim. team	Kicks by the control team
70	5	1	206	271	18	21	410	331
71	4	0	150	332	9	21	508	342
72	4	1	182	277	12	27	459	305
73	7	1	175	296	12	21	465	333
74	8	1	171	297	11	21	453	344
75	6	0	166	311	11	24	473	348
76	2	0	152	319	7	17	507	300
77	6	0	172	299	11	16	481	345
78	3	0	153	332	13	18	502	341
79	3	1	152	330	9	33	447	326
80	7	1	177	294	9	12	477	347
81	5	0	178	301	8	18	494	373
82	5	0	177	301	11	23	484	363
83	9	0	157	310	11	21	453	325
84	9	0	155	296	10	15	458	316
85	5	0	164	315	8	16	441	349
86	3	1	206	275	17	26	413	398
87	7	0	168	305	14	21	489	365
88	8	0	164	306	9	14	499	327
89	7	0	161	313	15	36	496	342
90	4	0	162	319	10	15	457	391
91	5	0	165	314	5	18	488	368
92	5	0	172	286	12	18	438	341
93	7	0	176	288	17	19	464	335
94	7	1	181	290	8	28	436	372
95	5	1	254	212	14	30	921	389
96	5	1	197	269	16	15	435	352
97	4	2	165	311	16	20	483	319
98	5	1	200	276	13	12	436	348
99	5	1	161	316	8	21	496	348
100	2	1	199	285	15	25	466	377

Games statistics. Advanced collaboration (experim.) team vs. control team.

Game #	Experim. team score	Control team score	Ball on the side of the experim. team (seconds per game)	Ball on the side of the control team (seconds per game)	Shots to goal by the control tem	Shots to goal by the experim. tem	Kicks by the experim. team	Kicks by the control team
1	7	1	209	247	14	20	392	358
2	6	0	215	261	15	16	429	381
3	10	0	164	300	8	24	418	326
4	6	0	166	311	8	24	479	327
5	7	0	177	295	7	32	441	348
6	8	2	174	290	12	22	420	316
7	11	1	161	298	11	28	454	344
8	7	0	207	267	15	26	418	360
9	8	1	185	283	10	20	411	344
10	5	1	183	293	10	34	423	383
11	7	1	171	300	14	16	405	333
12	7	0	192	281	13	22	409	350
13	4	0	224	258	15	16	416	390
14	3	2	193	286	16	30	410	372
15	7	1	206	264	15	22	432	344
16	6	0	200	276	17	28	431	345
17	8	0	163	308	10	32	478	365
18	5	0	186	293	11	24	431	329
19	5	0	237	242	20	22	390	420
20	8	0	178	294	18	24	404	347
21	12	0	163	296	5	22	446	294
22	9	0	199	271	13	18	416	355
23	6	1	202	271	14	30	380	380
24	8	0	190	280	15	22	410	360
25	6	0	218	258	19	24	393	363
26	6	1	174	299	9	30	451	327
27	12	0	182	278	17	20	422	332
28	8	0	205	266	12	16	442	352
29	7	1	200	271	18	24	439	367
30	8	0	207	266	20	18	426	341
31	5	3	173	297	15	22	414	328
32	6	0	189	287	7	14	408	355
33	8	0	177	292	14	30	450	348
34	8	0	176	295	8	36	413	381
35	7	0	207	264	15	20	427	370
36	5	1	204	272	15	24	435	366
37	8	1	154	314	12	32	432	330
38	11	0	214	247	17	14	433	334
39	11	0	139	323	5	32	465	350
40	4	0	201	280	14	20	427	384

Game #	Experim. team score	Control team score	Ball on the side of the experim. team (seconds per game)	Ball on the side of the control team (seconds per game)	Shots to goal by the control tem	Shots to goal by the experim. tem	Kicks by the experim. team	Kicks by the control team
41	6	0	217	259	22	28	421	361
42	6	0	203	273	12	24	382	373
43	9	0	156	311	9	28	389	354
44	6	0	195	282	19	26	452	384
45	6	1	176	298	13	36	428	349
46	9	1	202	263	9	32	409	403
47	8	0	181	290	9	22	454	359
48	7	0	202	271	12	20	391	364
49	5	1	209	266	20	20	411	366
50	6	1	209	264	18	16	421	382
51	7	1	167	303	19	26	456	336
52	4	0	181	301	8	30	438	361
53	10	0	182	283	16	20	422	326
54	7	0	151	322	4	30	437	355
55	8	0	196	274	11	20	371	364
56	9	0	185	281	18	26	464	354
57	2	0	199	289	10	24	470	386
58	9	0	187	281	14	20	410	365
59	8	0	137	335	8	26	485	298
60	9	0	193	275	15	16	410	322
61	5	1	287	188	29	6	377	493
62	10	0	195	262	8	28	417	355
63	7	1	189	281	17	28	399	337
64	7	0	160	313	13	30	468	290
65	5	0	197	281	18	24	397	381
66	10	0	191	274	16	28	415	343
67	5	0	230	249	15	16	416	394
68	11	2	153	304	10	22	429	319
69	3	1	215	267	14	22	423	431
70	6	0	219	257	15	18	424	370
71	11	0	148	314	10	32	462	295
72	7	1	206	264	9	24	395	373
73	10	0	167	300	14	32	448	316
74	5	1	195	281	15	22	416	375
75	7	1	194	278	19	26	426	365
76	5	0	202	277	13	28	430	354
77	11	0	165	297	14	20	468	304
78	9	0	213	254	16	18	425	342
79	6	0	179	297	11	20	439	336
80	9	0	204	263	9	18	386	376
81	9	0	182	285	11	36	450	348
82	5	0	172	307	8	18	431	371

Game #	Experim. team score	Control team score	Ball on the side of the experim. team (seconds per game)	Ball on the side of the control team (seconds per game)	Shots to goal by the control tem	Shots to goal by the experim. tem	Kicks by the experim. team	Kicks by the control team
83	7	0	204	270	13	20	440	370
84	8	0	170	300	12	20	434	312
85	6	0	229	247	17	22	415	392
86	11	1	170	286	10	22	398	326
87	7	0	167	297	10	34	467	331
88	9	1	161	304	8	22	421	319
89	6	0	175	301	12	32	443	320
90	6	1	196	277	11	22	390	406
91	5	2	182	292	10	26	404	344
92	8	1	182	286	11	28	428	363
93	7	1	193	277	15	18	430	385
94	9	0	210	257	9	22	428	407
95	7	1	203	268	12	18	413	370
96	5	1	193	276	8	26	392	367
97	8	1	189	278	16	28	417	338
98	6	0	168	309	13	32	432	357
99	8	1	194	274	17	24	435	381
100	8	1	199	269	16	18	392	344

Games statistics. Advanced collaboration (advanced) team vs. simple collaboration (simple) team.

Game #	Advanced team score	Simple team score	Ball on the side of the advanced team (secs per game)	Ball on the side of the control team (secs per game)	Shots to goal by the control team	Shot to goal by the advanced team	Kicks by the advanced team	Kicks by the control team
1	6	4	213	242	12	24	364	399
2	2	7	264	204	30	13	345	418
3	7	4	208	253	17	24	365	374
4	3	1	235	248	14	24	356	416
5	5	3	201	270	14	27	440	406
6	5	2	223	249	18	20	358	395
7	5	3	219	251	20	27	390	395
8	1	1	234	253	18	31	364	399
9	10	1	201	261	8	26	418	381
10	7	2	216	250	15	26	374	388
11	8	2	237	228	23	26	349	398
12	3	2	219	261	20	24	397	419

Game #	Advanced team score	Simple team score	Ball on the side of the advanced team (secs per game)	Ball on the side of the control team (secs per game)	Shots to goal by the control team	Shot to goal by the advanced team	Kicks by the advanced team	Kicks by the control team
13	3	1	215	267	17	35	395	388
14	5	2	191	282	5	20	409	363
15	9	2	227	236	23	27	392	395
16	9	4	195	260	9	26	380	380
17	4	2	207	270	17	27	400	375
18	3	8	262	200	24	15	315	420
19	3	4	234	239	27	31	330	420
20	7	3	220	245	21	27	354	417
21	8	3	234	228	14	20	386	407
22	2	5	240	233	26	22	360	423
23	5	2	241	232	17	13	364	424
24	3	6	211	256	12	24	345	431
25	7	2	213	254	11	22	340	374
26	4	2	207	269	12	24	354	379
27	4	3	221	252	21	31	357	397
28	4	1	234	244	9	22	406	395
29	6	4	238	227	24	20	346	394
30	6	2	222	248	18	35	369	388
31	6	4	248	215	15	22	362	426
32	3	6	245	223	23	17	368	417
33	1	4	271	207	20	18	337	466
34	7	4	233	228	11	15	357	382
35	4	2	223	252	15	36	382	413
36	3	2	215	266	15	29	406	403
37	5	5	213	251	18	24	381	386
38	1	1	251	235	24	18	401	444
39	10	3	204	252	15	20	380	363
40	5	3	259	211	18	18	356	448
41	4	4	235	236	17	15	383	400
42	4	1	243	236	26	17	406	382
43	7	0	231	242	15	18	376	398
44	6	3	245	223	29	22	341	430
45	5	0	218	261	15	18	403	390
46	3	4	241	232	17	22	372	464
47	4	3	249	224	21	24	374	462
48	4	4	234	237	26	24	394	405
49	6	7	239	216	17	18	335	411
50	6	3	210	257	26	29	417	400
51	6	4	178	286	14	24	394	376
52	5	2	212	262	20	26	369	381
53	9	2	204	258	12	15	404	353
54	1	3	249	233	23	15	397	452

Game #	Advanced team score	Simple team score	Ball on the side of the advanced team (secs per game)	Ball on the side of the control team (secs per game)	Shots to goal by the control team	Shot to goal by the advanced team	Kicks by the advanced team	Kicks by the control team
55	6	3	219	249	14	22	398	421
56	5	0	236	243	12	17	363	398
57	6	3	245	222	18	11	355	413
58	3	3	251	225	15	26	355	445
59	8	3	253	209	12	20	345	411
60	4	1	211	266	17	24	388	420
61	7	3	198	267	20	27	407	386
62	6	6	240	219	20	17	366	398
63	5	2	202	271	12	26	409	415
64	5	3	240	230	6	26	328	422
65	6	5	218	244	20	22	380	367
66	4	6	214	251	12	29	342	418
67	9	2	235	227	20	15	357	399
68	4	1	207	272	11	31	387	424
69	6	3	227	242	24	17	382	366
70	7	2	214	254	17	18	381	419
71	3	6	231	237	11	17	342	407
72	8	1	189	279	17	27	410	373
73	3	2	226	253	12	33	363	429
74	3	3	225	241	15	20	365	399
75	4	3	238	234	27	15	397	431
76	8	2	209	255	14	26	408	375
77	10	1	148	313	6	27	438	326
78	5	1	218	258	11	26	350	391
79	7	1	229	242	24	9	420	373
80	11	4	177	276	9	36	350	343
81	4	2	249	227	17	13	367	418
82	4	2	210	265	9	35	386	413
83	9	3	190	268	23	27	358	347
84	1	4	228	251	21	29	377	369
85	3	3	256	220	20	17	372	440
86	3	4	211	261	17	31	353	394
87	4	4	238	232	14	24	345	411
88	5	3	222	248	11	24	367	381
89	7	2	200	265	15	20	431	391
90	11	3	158	295	12	24	406	326
91	7	0	208	264	20	24	432	382
92	6	2	222	248	14	22	349	437
93	3	7	248	217	17	18	338	460
94	4	1	229	249	20	17	390	385
95	6	1	243	230	23	27	348	395
96	4	4	230	241	17	15	394	414

Game #	Advanced team score	Simple team score	Ball on the side of the advanced team (secs per game)	Ball on the side of the control team (secs per game)	Shots to goal by the control team	Shot to goal by the advanced team	Kicks by the advanced team	Kicks by the control team
97	7	3	208	257	17	33	409	409
98	6	2	193	276	9	17	374	366
99	4	3	234	239	23	29	343	419
100	7	2	195	272	14	27	397	379
101	8	4	222	237	17	26	412	394

BIBLIOGRAPHY

Baba, N at al. (2001). *Computational Intelligence in Games.* Heidelberg, Germany: Physica-Verlag.

Beim, G. (1977). *Principles of Modern Soccer.* Boston, MA: Houghton Mifflin Company.

Erghott, M. (2005). *Multicriteria Optimization.* Berlin, Germany: Springer.

Gandibleux, X at al (2004). *Metaheuristics for Multiobjective Optimization.* Berlin, Germany: Spronger-Verlag.

Grosz, B., Kraus, S. (1996). Collaborative plans for complex group action. *Artificial Intelligence.*, 86(2), 269.

Hunter, M., Hu, H. (2002) A Generalized Approach to Position Selection for Simulated Soccer Agents. *Lecture Notes In Computer Science; Vol. 2377 RoboCup 2001: Robot Soccer World Cup V*, 380

Law, A., Kelton, W. (2000). *Simulation Modeling and Analysis.* New York, NY: McGraw-Hill.

Kaliszewski, I. (1994). *Quantitative Pareto Analysis by Cone Separation Technique.* Boston, MA: Kluwer Academic Publishers.

Keeney, R.L. (1976). *Decisions with Multiple Objectives.* New York, NY: John Wiley & Sons

Kelton, W. at al.(2004). *Simulation with Arena.* New York, NY: McGraw-Hill.

Kok, J., De Boyer, R. (2002). The Incremental Development of a Synthetic Multi-Agent System: The UvA Trilearn 2001 Robotic Soccer Simulation Team. Master's Thesis. Faculty of Science. University of Amsterdam.

Kok, J. Vlassis, N. Groen, F.(2003) UvA Trilearn 2003 team description. *Proceedings CD RoboCup 2003*, Springer-Verlag.

Kok, J. Spaan, M.,Vlassis, N. (2003). Multi-robot decision making using coordination graphs. *Proceedings of the International Conference on Advanced Robotics (ICAR)*, Coimbra, Portugal, 1124.

Kok, J. Vlassis, N. (2005). UvA Trilearn 2005 team description. *Proceedings CD RoboCup 2005*, Springer-Verlag.

Kok, J., Matthijs T., Vlassis, N. Non-communicative multi-robot coordination in dynamic environments. *Robotics and Autonomous Systems*, 50(2-3):99–114, Elsevier Science, February 2005

Kolbin, V. (2003). *Decision Making and Programming.* Singapore: World Scientific Publishing Co. Pte. Ltd.

Kyrylov, V. (2006). Balancing Rewards, Risks, Costs, and Real-Time Constraints in the Ball Passing Algorithm for the Robotic Soccer. Accepted for presentation at the *RoboCup 2006 Research Symposium*

Kyrylov, V., Greber, M., Bergman, D. (2006). Multi-Criteria Optimization of Ball Passing in Simulated Soccer. To appear in *Journal of Multi-Criteria Decision Analysis*, 2006, issue 2/3

Kyrylov, V., Brokenshire, D., Hou, E. (2005). Optimizing Precision of Self-Localization in the Simulated Robotics Soccer. *RoboCup 2004: Robot Soccer World Cup VIII.* Lecture Notes in Artificial Intelligence 3276, Springer-Verlag, pp. 249-257.

Lau, N., Reis, L.(2002) FC Portugal 2001 Team Description: Flexible Teamwork and Configurable Strategy. *RoboCup-2001: Robot Soccer World Cup V*, Andreas Birk, Silvia Coradeshi, Satoshi Tadokoro editors, Springer Verlag, Berlin.

Liu, G.P., Yang, J.B., Whidborne, J.F. (2003). *Multiobjective Optimisation and Control.* Baldoc, Hertfordshire, England: Research Studies Press Ltd.

Reis, L, Lau, N.(2001) FC Portugal Team Description: RoboCup 2000 Simulation League Champion, *RoboCup-2000: Robot Soccer World Cup IV*, Peter Stone, Tucker Balch and Gerhard Kraetzschmar editors, Springer Verlag, Berlin, 29.

Reis, L, Lau, N. Oliveira, E.(2001) . Situation Based Strategic Positioning for Coordinating a Team of Homogeneous Agents, *Balancing Reactivity and Social Deliberation in Multi-Agent Systems*, Markus Hannebauer, Jan Wendler, Enrico Pagello, editors, Springer Verlag, 175.

Russell, S., Norvig, P. (2003). *Artificial Intelligence. A Modern Approach.* Upper Saddle River, NJ: Pearson Education, Inc.

Stadler, W. (1988). *Multicriteria Optimization in Engineering and in the Sciences.* New York, NY: Plenum Press.

Stone, P. (2000) *Layered Learning in Multiagent Systems: A Winning Approach to Robotic Soccer.* MIT Press.

Stone, P. at al. (2006) From Pixels to Multi-Robot Decision-Making: A Study in Uncertainty. *Robotics and Autonomous Systems* , 2006. Special issue on Planning Under Uncertainty in Robotics

Stone, P., McAllester, D.(2001). An Architecture for Action Selection in Robotic Soccer. *Proceedings of the Fifth International Conference on Autonomous Agents*, ACM Press, New York, NY, 316

Stone, P. (2002) ATTUnited-2001: Using Heterogeneous Players. In Andreas Birk, Silvia Coradeschi, and Satoshi Tadokoro, editors, *RoboCup-2001: Robot Soccer World Cup V*, Springer Verlag, Berlin

Stone, P., Riley, P., Veloso, M. (2000) The CMUnited-99 Champion Simulator Team. M. Veloso, E. Pagello, and H. Kitano, editors, *RoboCup-99: Robot Soccer World Cup III*, pp. 35–48, Springer, Berlin

Veloso, M, Stone, P. Han, K.(2000). The CMUnited-97 Robotic Soccer Team: Perception and Multi-agent Control. *Robotics and Autonomous Systems*, 29(2-3)

Vogelsinger, H. (1973). *The Challenge of Soccer*. Boston, MA: Allyn and Bacon, Inc.

Wang, C.; Chen, X.; Zhao, X. & Ju, S. (2004) Design and Implementation of a General Decision-making Model. *RoboCup Simulation, International Journal of Advanced Robotic Systems*, 1(3),207

Weigel, T at al.(2002). CS Freiburg 2001. *Lecture Notes in Computer Science. RoboCup 2001: Robot Soccer World Cup V.* A. Birk, S. Coradeschi, S. Tadokoro (Eds.), 26

Zang, Y. (2005). Tao of Soccer Overview. *Electronic resource.* http://e-graviton.com/ird/.