Master Thesis
Software Engineering
Thesis no: MSE-2001-03
March 2001

An Electric Field Approach

- A Strategy for Sony Four-Legged Robot Soccer

John Johansson

Department of
Software Engineering and Computer Science
Blekinge Institute of Technology
Box 520
SE – 372 25 Ronneby
Sweden

This thesis is submitted to the Department of Software Engineering and Computer Science at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Software Engineering. The thesis is equivalent to 20 weeks of full time studies.

Contact Information:
Author:
John Johansson
E-mail: John.Johansson@home.se

University advisor(s):
Stefan Johansson
Department of Software Engineering and Computer Science

Department of
Software Engineering and Computer Science
Blekinge Institute of Technology
Box 520
SE – 372 25 Ronneby
Sweden

Internet : www.bth.se/ipd
Phone : +46 457 38 50 00
Fax : + 46 457 271 25

ABSTRACT

Using physical analogies when solving computational problems is not uncommon. The Electric Field Approach is such an analogy using potential fields for describing the situation in a robotic soccer match and for proposing the next step in optimizing the situation. The approach was developed and implemented during the summer of 2000 and was later used and tested during the fourth Robot Soccer World Cup in Melbourne, Australia. The results of the games prove that the approach is applicable in this narrow but interesting domain, combining artificial intelligence and robotics. The theory of the approach is general and can also be applied on various other domains. This is not the first potential field approach, but the ability of both handling navigation and manipulation of the environment is unique.

Keywords: robocup, potential fields, real-time decision-making, legged robot, AIBO

CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

During the last year I have had the opportunity of developing, implementing and competing with the Electric Field Approach and the AIBO Entertainment Robot. Little did I suspect when I took on this master thesis project, that it would bring me halfway around the world and back, and that I would meet so many interesting people on the way.

The field of Artificial Intelligence and Robotics is a young and interesting one, and it raises many questions that, when answered, not only provides an easier and safer world for us to live in, but also brings a deeper understanding about how the world around us really works.

1 INTRODUCTION

This chapter is an introduction to the Electric Field Approach, the RoboCup domain and describes the problems, methods and results of this master thesis.

1.1 Background

To fully understand the problem description, a brief introduction to each of the most important concepts can be read in the following subsections.

1.1.1 The Electric Field Approach

The main topic of this thesis is the development and evaluation of the Electric Field Approach (EFA), implemented as a high-level strategy decision-maker on the AIBO platform. The Electric Field Approach is a logic replication of a real world phenomenon. The environment is abstracted to a representation of a virtual electric field with areas of positive and negative potential, used as a heuristic function; Dangerous or bad places are represented by negative virtual charges while positive ones represent good places. There are already techniques using potential fields for path planning and navigation, but this approach is more general and can handle navigation as well as manipulation of the environment.

1.1.2 The RoboCup Domain

In the end of August 2000 the Third International World Cup in Robot Soccer, RoboCup, took place in Melbourne, Australia. The RoboCup Domain is artificial intelligence and robotics combined in an entertaining and fun form. Team Sweden, one of the Swedish contributions to RoboCup 2000, participated in the Sony Legged Robot League competing with the AIBO ER-110 entertainment robot from Sony. RoboCup will be further described in section 2.1.

1.1.3 AIBO

Today the entertainment industry is an important application for microprocessors [Sakamura, 1999]. As the traditional game consoles are getting more and more advanced, with more computation power, and the mobile games are combined with mobile phones and portable devices, we also see a trend of technical advancements of traditional toys. The AIBO ER-110 (see Fig. 1), the dog-like robot produced by Sony, was used in the Sony Legged Robot League in RoboCup 2000.

Fig. 1 The AIBO ER-110 robot from Sony

1.1.3.1 Hardware

The AIBO ER-110 hardware features a 100 MHz 64-bit Mips RISC processor with 16 Mb of RAM. The dimensions of the robot are 156 x 266 x 274 mm and it weighs 1.6 kg (batteries included). It has 18 degrees of freedom (DOF), which is the number of movable joints. Three DOF in each leg (four legs), three in the head, two in the tail and one in the mouth.

The robot has a color video camera in the nose, one microphone in each ear for stereo recording and a miniature speaker in the mouth. Two heat sensors, an acceleration detector and a rotational detector are installed in the torso, and the head contains an infrared range detector. It has also one touch sensor under each foot and one on the head.

1.1.3.2 Software

The software on the AIBO platform consists of a real-time operating system and an Application Programmers Interface (API).

- **Aperios.** Aperios is a completely object oriented real-time operating system that is used for the AIBO robot, developed by Sony.
- **Open-R.** Open-R is an API for robots and can be seen as a middle layer between the Aperios operating system and the programmer. Open-R is proposed by Sony as a standard architecture for robot entertainment systems and specifies standards for hardware, electronics, and software architectures.
- **Programming Language.** The programming language is C and C++. Most of the framework is developed in C++.

1.2 Problem Description

When we started developing the software implementation of the Electric Field Approach we did not know whether the approach would be suitable for the RoboCup Domain or not. Team Sweden has defined four requirements for developing the soccer playing software: scientific value, generality, effectiveness and robustness.

The main problems of this thesis are formulated as follows:

- Is the Electric Field Approach suitable for the RoboCup Domain?
- Does the implementation of the EFA fulfill the four requirements of Team Sweden?
- What are the strengths and weaknesses of the Approach?

The method of evaluating the approach involved the development of prototypes to demonstrate the approach, and the design and implementation, and the real life testing of the system on the AIBO platform before and during the world cup.

1.3 Results

The results from the RoboCup competition verify the applicability of the Electric Field Approach for the RoboCup Domain. Theoretically this approach is also applicable in other domains as well. The implementation of the approach proved to be highly configurable, allowing the high-level strategy to be altered in an easy way, thus making it able to adapt to the different tasks of the RoboCup competition.

1.4 Terms and Abbreviations

AI	Artificial Intelligence
AIBO	Artificial Intelligence roBOt. The similar Japanese word 'aibou' translates to 'partner' or 'friend'
Aperios	Sony real-time object oriented operating system.
API	Application Program Interface
Azimuth	The horizontal angular distance from a reference direction, measured from the right
BDI	Belief – Desire – Intention, a term in artificial intelligence
BTH	Blekinge Tekniska Högskola. (Blekinge Institute of Technology, Sweden)
CMD	Commander Module
CMU	Carnegie Mellon University, USA
DOF	Degrees of Freedom. Measure how flexible a robot is in the number of joints
DQ	Disqualified
EFA	Electric Field Approach
FIFA	Fédération Internationale de Football Association
GM	Global Map Module
GO	Global Object
GS	Global Space
HBM	Hierarchical Behavior Module
HK-R	The University College of Karlskrona/Ronneby, a former name of BTH
KTH	Kungliga Tekniska Högskolan. (Royal Institute of Technology, Sweden)
Landmark	Visual object used for localization
LPO	Local Perceptual Object
LPS	Local Perceptual Space
Mb	Mega byte
MHz	Mega Hertz. Commonly used to express microprocessor clock speed
Mips	Million Instructions Per Second. General measure of computing performance
NDA	Non Disclosure Agreement
Open-R	An API and standard architecture for robot entertainment from Sony
ORU	Örebro University, Sweden
p	Penalty (when used in RoboCup result tables)
PAM	Perceptual Anchoring Module
RAM	Random Access Memory
RISC	Reduced Instruction Set Computer
RMIT	Royal Melbourne Institute of Technology
RP	Reactive Planner Module, for real-time strategic decisions
UNSW	University of New South Wales, Australia

Table 1. Terms and abbreviations

1.5 Outline of the Thesis

The next chapter introduces RoboCup and describes the purpose and history of the initiative. The theory of the Electric Field Approach is explained in chapter 3, and the implementation on the AIBO platform is described in chapter 4. In chapter 5 the calibration of the software implementation is described, and also the performance of Team Sweden in RoboCup 2000. Further chapters discuss the attributes of the approach and conclude the thesis.

2 THE TEAM SWEDEN EFFORT IN ROBOCUP

Team Sweden was one of Sweden's contributions to RoboCup 2000. This chapter aims to describe the competition in general and the Team Sweden effort in particular.

2.1 RoboCup

The Robot World Cup Initiative (RoboCup) is a soccer world cup in which researchers compete with fully autonomous robots or software agents in different leagues. The challenges range from building hardware to implementing software and artificial intelligence.

2.1.1 Purpose of RoboCup

RoboCup was started in order to foster and promote Artificial Intelligence (AI) research internationally by providing a standard problem that also is fun to work with. Researchers founded the RoboCup Federation in order to push the borders of science and technology using soccer games played by robots and software agents. This international organization involves national research institutions, private corporations and over 150 universities in 35 countries.

The aim is to in the year 2050 get a team of fully autonomous robot soccer players to win a soccer game (complying with the official FIFA rules) against the human winners of the World Cup [RoboCup, 2000]. This might seem a bit ambitious, but considering the great improvements in the domain from the start in 1997 up until today it is not an unreachable goal. The humanoid robot 'PINO' was demonstrated at RoboCup 2000, and the RoboCup Federation is planning to include a humanoid league in the competition in the year of 2002.

It took only 50 years from the Wright Brother's first aircraft the to Apollo mission to send man to the moon and safely return them to the earth. Although the early work in the area of Artificial Intelligence research made a more optimistic prediction of the advancements of the field [Russel & Norvig, 1995], it took only 50 years from the invention of the digital computer to the chess playing IBM computer Deep Blue.

Before the goal can be achieved there is a wide range of research areas to explore. Building a world-class roboteam will according to [Reese Hedberg, 1997] require:

- Capable single players (autonomous agents, robotics, vision, and real-time sensor fusion)
- Teamwork (multi-agent collaboration, context recognition)
- Understanding the competition (cognitive modeling)
- The ability to develop and execute plays and strategies in real-time (strategy acquisition, real-time reasoning and planning, and reactive behavior
- Pre- and post-game training (machine learning)

2.1.2 History

Before the first RoboCup in Japan 1997 the preparations had already been going on for four years. The idea was first discussed in 1993. Two years later the RoboCup Federation made a public announcement followed by a preliminary contest in 1996.

The first real RoboCup event was held in Nagoya, Japan 1997 including more than 40 participant teams in two leagues: the Simulation League and the Middle-Size Robot League. 1998 RoboCup was held in Paris, France with more than 50 teams. When Stockholm hosted the third RoboCup event in July 1999 there were over 120 competing teams. The Electric Field Approach was developed for RoboCup 2000, which was hosted by RMIT in Melbourne, Australia where over 150 teams attended. RoboCup 2001 is to be held in Seattle, USA.

The robot soccer domain is not just fun and games. The competition promotes improvement of technology, engineering and science, which can be used in various domains in the future. The technical innovations driven by RoboCup can be used in the following areas [RoboCup, 2000]:

- Robots for dangerous tasks: clean-up of radioactive sites, bomb/mine disposal, underwater and space exploration, software players.
- Search and Rescue robots: post earthquake survivor location, rescue of swimmers in danger, mine rescue, avalanche-search and rescue.
- Future transport systems.
- ToyBots: for entertainment and education.
- Robots for assisting aged and disabled.
- Teams of industrial robots.
- High efficiency batteries and energy use.
- New materials and devices.
- Evolvable software.
- Artificial Life.

2.1.3 The Leagues

The leagues in the RoboCup competition can be seen in Table 2.

Name	Description
Small-Size Robot League	Small robots in shoebox sizes competing on a table tennis sized field. Each team creates both hardware and software.
Middle-Size Robot League	Larger robots. Hardware and software engineering.
Simulator League	Software agents competing on a big screen. Software interfacing with software.
Junior League	A league for children using robots built in Lego.
Sony Legged Robot League	Described in section 2.1.4. Software interfacing with hardware.
Humanoid League	To be introduced 2002. In RoboCup 2000 the humanoid Honda robot was demonstrated.

Table 2 The leagues in the RoboCup competition.

2.1.4 Sony Legged Robot League

The Sony Legged Robot League, first introduced at RoboCup'99, uses the Sony AIBO ER-110 entertainment robot platform. In contrast to the other leagues the robot hardware may not be changed, which makes it one of the more interesting leagues in a software engineering perspective, since the games are decided by whom has developed the best software. The fully autonomous robots compete in three-on-three soccer matches divided in two 10-minute halves with a 10-minute brake.

The soccer field is 2800 x 1800 mm with goals with a width of 600 mm. In each corner of the field and on either side of the half way line landmarks are placed. These are for navigation and localization of the robots and consist of two color combinations to uniquely identify their position (see Fig. 2).

The league is divided into two major parts. The soccer matches, where the robots compete three-on-three, and the RoboCup Challenge, where the robots are to perform certain tasks on time.

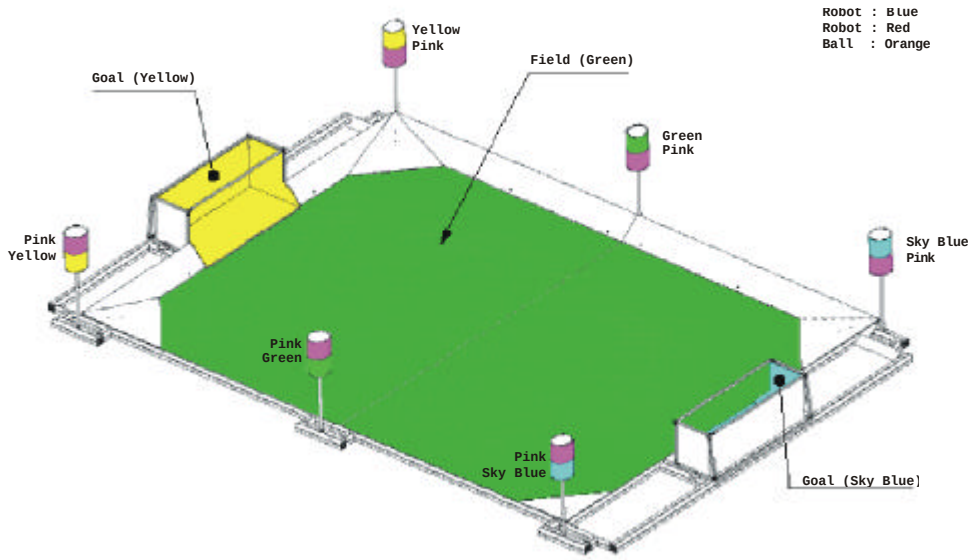

Fig. 2 The soccer field of the Sony Legged Robot League

2.1.4.1 The Match

In the match each team player wears blue or red stickers, hence the blue team and the red team. The blue team tries to score a goal in the yellow goal, and the red team in the blue goal. During the 10 minute break the teams switch team color and sides. No more than one player of the defending team may be in the penalty area. If more than one defender is in the penalty area, the last one to cross the penalty line will be picked up by a referee and put on the half way line.

If a player has the ball and the opponent pushes or hinder the player from getting to the ball, the opponent is picked up and put on the half way line (the obstruction rule). A complete description of the rules can be found in the rulebook [Sony Corp., 2000].

2.1.4.2 The RoboCup Challenge

RoboCup Challenge consists of three challenges.

- In the first one, *the Striker Challenge*, there is only one player and one ball in the field. The player should score the ball into the right goal within three minutes. The position and orientation of the player and the position of the ball are decided at random a few minutes before the challenge to encourage general solutions to the problem. This challenge might seem a simple task, yet no one could do this at RoboCup'99 and surprisingly only half of the participants in the 2000 version of the competition.
- Challenge two, *the Collaboration Challenge*, consists of two robot players and one ball. The first player, positioned in the own half of the field, should try to pass the ball over to the other player without crossing the half way line and without scoring a goal. The other player should, also without crossing the half way line, receive the ball and score, all this within four minutes. Penalty of one additional minute to the result is given for crossing the lines.
- In *the Collision Avoidance Challenge*, which is the third and last challenge, there are three robots on the field. One active (the player), and two passive ones acting as obstacles. The player should navigate past the two obstacles and score the goal within four minutes. Penalty is given if the player touches the obstacles.

If the player does not score the ball within the time limit, the distance between the ball and the goal is measured and set as result. See Appendix B for the result of the RoboCup Challenge 2000.

2.2 Team Sweden

The section describes Team Sweden, the Swedish team participating in the Sony Legged Robot League in RoboCup '99 and RoboCup 2000.

Fig. 3 The Team Sweden logotype of 1999

2.2.1 History

"Team Sweden" was created in 1998 when universities from three Swedish cities, Stockholm, Örebro and Ronneby, joined forces to compete as one team in the newly formed Sony Legged Robot League at RoboCup competition, which was held in Stockholm 1999. The four main requirements for the development of a soccer playing software for the Sony legged robots was [Saffiotti et al., 2000]:

- **Scientific value:** the software should illustrate our scientific approach to autonomous robotics, and demonstrate our research lines in this field.
- **Generality:** the software should embody general principles that are needed to achieve autonomous robot operation, and that can be reused in different robots operation in different environments.
- **Effectiveness:** the software should effectively address the specific challenges present in the RoboCup domain in general, and in the legged robot league in particular; and
- **Robustness:** the software should degradate smoothly in face of errors and imprecision in perception and execution; in particular, the lower layers should still provide some reasonable response even when the higher layers cannot compute a reliable course of action.

Team Sweden has approached the RoboCup problem as an "entertaining way to do team research that involves some important scientific challenges" [Saffiotti et al., 2000]. The main challenges proved to be the problems of how to cope with large uncertainty inherent in the domain of robotics, the cognitive processes with lower level sensori-motor process and the integration of actions and perception in a complex and dynamic environment.

The second time around, when work was focused on RoboCup 2000 in Melbourne Australia, the configuration of the team was slightly changed. An almost completely new crew from Ronneby joined up with parts of the former Örebro team, which also provided a new team leader in Alessandro Saffiotti. Team Sweden 2000 started working on the new version of the system in the spring of 2000. The software architecture developed for RoboCup'99 was enhanced with a more stable Global Map module, new refined kicks and behaviors in the Hierarchical Behavior Module and Commander Module, and a completely new Reactive Planner Module (see Fig. 6).

2.2.2 Members

Alessandro Saffiotti was the team leader, and the team members on location in Melbourne during the competition were John Johansson, Pernilla Johnson, Kevin LeBlanc, Britt-Marie Lindberg, Alessandro Saffiotti and Zbigniew Wasik. See Table 3 for full listing of the members of Team Sweden 2000.

ORU	BTH	KTH
Pär Bushka	Ina Bergman	Magnus Boman
Mikael Karlström	Magnus Broberg	
Kevin LeBlanc	Paul Davidsson	
Alessandro Saffiotti[1]	John Johansson	
Zbigniew Wasik	Stefan Johansson	
	Pernilla Johnson	
	Richard Krejstrup	
	Britt-Marie Lindberg	
	Elisabeth Smeds	

Table 3 Members of Team Sweden 2000

[1] The Team Leader of Team Sweden 2000

3 THE ELECTRIC FIELD APPROACH

The Electric Field Approach it an approach similar to the traditional potential field approaches that have been used for controlling navigation and path planning in autonomous robotics [Payton et al., 1990]. These approaches assign negative charges for the real world objects that it should avoid, and positive charges for the goal, creating a simulated potential field. By using this field the robot can plan a route avoiding obstacles and navigate safely to the goal.

3.1 Introduction

The EFA was implemented in the Reactive Planner Module (RP) of the Team Sweden software architecture for the AIBO Robot. In order to distinguish between the theory and the implementation, this section will describe the general principles of the approach while section 4 will describe how the EFA was implemented in the RP dealing with the issues of uncertainty and the AIBO platform.

The EFA is a reactive planner with one step look-ahead. It was developed for the RoboCup domain, but it can be used in other domains as well. [Johansson et al., 2001]

Compared to existing approaches using potential fields the EFA is more general due to the fact that it is not restricted only to navigation but can be used in various other domains as well. By assigning charges to the objects in the environment in real-time, a simulated 2-dimensional electric field is created for the current situation. This field is then used to evaluate the situation and reacting in the optimal way according to the charges.

Apart from the different types of charges, the probe and the behaviors (the actions) play an important roll in the EFA. These are in turn described in the subsequent sections.

3.2 Static charges

Consider a physical map of the environment in which the robot should operate. Intuitively we can often see which geographical areas in the environment that are better to be in, and which areas that are not so good to be in.

Example 1:

Task: Go from point A to point B (Fig. 4).

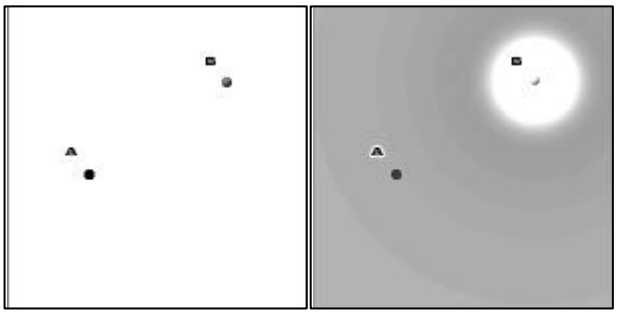

Fig. 4 Navigate from point A to point B. Physical representation (left) and potential representation (right).

The worst geographical position be in is of course as far from point B as possible while the optimal situation is to be exactly in point B. This can easily be seen using common sense, but how can this be represented heuristically? The answer to this question is *virtual charges* (later referred to only as "Charges"). Putting a single positive charge in point B will indicate that this is the best place to be on the map, regardless of the actual potential as long as it is positive (Fig. 4).

In some cases there are areas on the map that should be avoided. These areas are represented by negative charges. When using several charges, the potential of the charges must be calibrated relative to each other in order to achieve the desired field.

3.3 Dynamic charges

A real world environment, in which a robot should work in or interact with, is never completely static. There are always moving or changing elements to consider such as e.g. other robots (friends or foes), moving obstacles, or the robot itself. These elements may have the possibility to move/be moved around the environment, which will affect the state of the situation, bringing it closer or farther from the goal. Together with the static charge set, the dynamic charges make up the complete potential field for a specific situation.

3.3.1 Undirected charges

The undirected dynamic charge is used in the representation of moving/movable objects where the physical orientation is unimportant or cannot be determined. By attaching this type of charge to the object, the potential of the charge contributes to the electric field in a single point in the exact position of the object.

3.3.2 Agent-oriented charges

When facing bi-symmetrical dynamic objects, the physical orientation is of great importance for the situation. For instance, when facing the ball in a soccer game you can see the ball, and maybe kick it if it is close enough, but when the ball is behind you, you cannot do very much without turning around. In terms of EFA we would say that the potential, the "goodness" of the situation, should be different depending on the orientation of the player in respect to the ball. When the ball is within the same range but behind the player, the potential should be smaller than when facing the ball. Therefore we have to introduce the concept of "agent-oriented charges". These charges are positioned with an offset from the center of the player in the direction in which the object is facing. This causes the player to turn towards the ball.

3.3.3 Environment-oriented charges

While agent-oriented charges are positioned with respect to the orientation of the player, environment-oriented charges are oriented with respect to the electric environment made up by the charges. An environment-oriented charge is placed with an offset of the player position and in the direction of the right angle to the electric field lines directed towards the positive area.

These type of charges are particular useful when using navigational behaviors (see section 3.5). E.g. in a robot soccer game the player avoids to position itself between the ball and the opponent net, which could hinder a mate from scoring, or in a worst case scenario unintentionally score an own goal.

3.4 The probe

The probe is an object, real or virtual, that should be put in the most positively charged area as possible. In Example 1 above, the actual robot represents the probe, navigating to point B, thus trying to put itself near the positive charge. This is however only one scenario. Another scenario is the robot soccer game where the ball is the probe and the robot is to move the ball into the opponent goal, into the most positively charged area of the soccer field.

3.5 Behaviors

Every kind of physical robot has a set of predefined motions making it able to interact with the environment. Several motions executed in parallel or sequence defines a behavior. Kicking a ball, walking in a certain way or to a specific place are

examples of behaviors. From the EFA's point of view a behavior is defined as *"an action that changes the configuration of charges in the electric field or the position of the probe"*, thus changing charge potentials or moving charges or the probe(s). By moving towards point B in Example 1, the probe is moved and therefore this should be considered a behavior.

In order to decide what to do for a particular situation, the EFA must be provided with a set of *predefined behavior descriptions*. Each behavior in this set is, as the name states, a description of a physical behavior that can be executed on the robot. The impact on the electric field can now be predicted for each behavior, by using its description, without actually executing the behavior on the robot.

Considering the static and dynamic charges, the situation is evaluated and the best behavior is executed. The best behavior is defined as *"the behavior that increase the potential of the position of the probe the most"* of all the available behaviors. This can be done in two ways.

- **Navigation**. Moving the positively charged dynamic objects towards the probe, (often the robot/agent itself).
- **Manipulation**. Moving the probe(s) closer to a positively charged area in the potential field.

3.6 Behavior Prediction

In order to extract the best behavior from the behavior set, each behavior must be predicted, and the effect on the probe evaluated. Most behaviors are situation dependant, which means that certain *preconditions* must be fulfilled to enable the behavior. Some behaviors are not meaningful to execute in certain situations, and some are simply not possible. An example of such a behavior in the RoboCup domain is kicking the ball when the ball is not within reach. A kick is theoretically executable, but it will not have the predicted effect, (i.e. moving the ball some distance in the direction of the kick). Such a behavior is considered invalid for that particular situation, and cannot be executed because its precondition is not fulfilled.

In some situations a predicted behavior can result in an invalid situation, e.g. if the behavior puts the robot/agent outside the field, or puts the probe in an unacceptable state. This behavior does not fulfill the *post-conditions* and is also invalid.

At every evaluation each behavior is predicted, and each prediction requires geometrical calculations of distances and angles. These types of calculations are considered computationally costly due to the frequent use of floating-point functions.

3.7 The Evaluation Loop

The Evaluation Loop is a loop, which is called with a certain frequency when the current situation should be evaluated. This loop contains all the behaviors in the behavior set, and when executed it loops through every behavior and evaluates them sequentially. Those behaviors that do not fulfill their pre- or post-conditions are considered not available in the current evaluation and are temporarily excluded from the calculation. The behavior that results in the highest probe potential wins the evaluation and is executed until another behavior results in a higher probe potential or the pre- or post-conditions for the current behavior no longer are fulfilled. The evaluation loop expressed in pseudo code may look like Fig. 5 [Johansson et al., 2001].

```
while not reached_goal do
     highest_charge := -∞;
     generate(list_of_possible_behavior)
     foreach behavior in list_of_possible_behavior
          predict(behavior);
          charge := calculate(probe_charge);
          if charge > highest_charge then
               highest_charge := charge;
               best_behavior := behavior;
          end_if;
     end_foreach;
     execute(best_behavior);
end_while;
```

Fig. 5 The evaluation loop in pseudo code

4 THE EFA AS IMPLEMENTED IN THE REACTIVE PLANNER MODULE

In the previous section the theory behind the EFA was explained. The aim of this section is to describe the implementation of the EFA on the AIBO platform, using the Team Sweden architecture. When developing a robotic system there are certain aspects that have to be considered that differ from "regular" software development, such as incomplete and incorrect inputs and uncertainty.

This implementation of the EFA and does not handle special cases such as ball holding or obstruction [Sony Corp., 2000].

In the first subsection the software architecture of Team Sweden 2000 is described in short, to make the reader aware of the environment in which the EFA was implemented. The second subsection describes the Reactive Planner Module in detail where the EFA is implemented.

4.1 Software Architecture of Team Sweden 2000

This is a schematic picture of the architecture used by Team Sweden (Fig. 6) in the RoboCup 2000 competition. It can briefly be described as a layered architecture, inspired by the Thinking Cap [Saffiotti, 2000] for autonomous robotic software design.

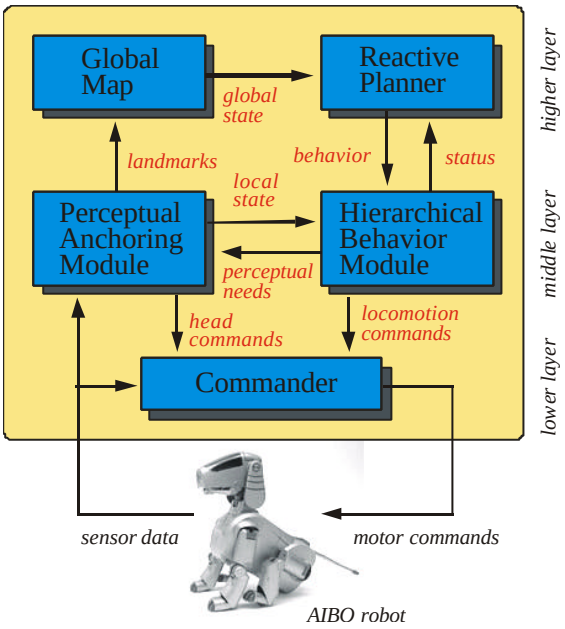

Fig. 6 Team Sweden 2000 Architecture

4.1.1 Global Map

The Global Map module (GM) provides a birds-view of the soccer field. By using the perceptual information provided by the Perceptual Anchoring Module (PAM) the GM is able to calculate the robots own position and other objects, such as other robots and the ball relative to the field. By calculating and storing the transient information from the PAM, a global view of the field is modeled, which results in a much stronger awareness of the environment than pure object detection.

4.1.2 Perceptual Anchoring Module

Perception is the only physical input used at this level of the system, which puts a lot of importance of the correctness of this module. The PAM acts as a short-term memory of the location of the objects around the robot. An object is considered anchored when the robot's camera has identified it. Then the distance and the angle of the object in respect to the robot is calculated and provided to the system by the Local Perceptual Space (LPS) which is used by the GM and the Reactive Planner Module (RP).

Head motions for scanning the different objects and camera image processing is done here. The PAM also uses a fuzzy position grid [Buschka et al., 2000], which accounts for errors and imprecision in the visual recognition, which is suitable for the poor odometry of the legged robot and its limited computational power.

4.1.3 Hierarchical Behavior Module

The set of behaviors the robot is able to execute is defined in the Hierarchical Behavior Module (HBM) using hierarchical fuzzy control [Saffiotti, 1997]. When the RP has decided the best behavior for the current situation, a high-level behavior is called in the HBM. The HBM is in turn responsible for calling the right low-level behavior in the Commander Module (CMD).

The hierarchy in the HBM can be described by a tree structure. The most basic behaviors, which can be used in the CMD, are the building blocks of the slightly more complex ones, which in turn are used by the even more complex behaviors, etc, until a high level complex behavior is defined and is ready to be used by the RP. Each level of the hierarchy uses fuzzy logic rules applied on the LPS for choosing between the behaviors in the lower levels. Typical high-level behaviors defined in the HBM are "Go to ball", "Face ball" and "Self localize".

4.1.4 Commander

The CMD interfaces both the PAM and the HBM, and is the main communication channel between the Team Sweden architecture and the frameworks on the AIBO platform (Open-R and Aperios). The instructions of the low-level behaviors from the HBM and the head motion commands from the PAM are translated into a format comprehensible for the frameworks. Data from the sensors such as the gyroscope, the motor sensors and the head sensor are communicated from the frameworks to the system through the CMD.

The framework can be called at different levels, depending on how detailed the own description of walking styles and behaviors are. In RoboCup 2000, Team Sweden used tuned walking styles included in the framework and own kicks and head motions [Lindberg, Johnson, 2000].

Making own walking styles from scratch is a very complex task, involving automatic control engineering and other advanced techniques. Tries have been made to imitate the walking style of the University of New South Wales (UNSW), winners of the competition in the year of 2000 [Forslund et al., 2001].

4.2 The Reactive Planner Module

The responsibility for control of motion and manipulation is divided between the HBM and the RP. The RP is the upper level of the behavior elicitation. The author was responsible for the development of this module in the RoboCup 2000 competition.

4.2.1 Inputs and outputs (communication with other modules)

As the picture of the module design (see Fig. 6) shows, a "global state" is submitted from the GM to the RP. This happens every second, and it is the cue for the RP to start evaluating the situation. The GS (Global Space) and the LPS are C structures public to the RP module, and are used as storage of information about the current situation. The LPS is a world representation from the point-a-view of the robot.

The LPS is an array of Local Perceptual Objects (LPO) which each contain information about the object it represents, such as the type of object, the distance and angle from the robot, and perceptual information in the degree of anchoring. The LPS is generated and updated by the PAM, and consists of the LPOs which can be seen in Fig. 7

```
// Structure which holds information about object
typedef struct
{
  short int id;            // who I am (index in lpo array)
  short int type;          // can be ball, dog, ...
  int rho;                 // distance of object, mm
  float theta;             // azimuth of object, rads
  float anchored;          // degree to which the object is
                           //  anchored
  unsigned int last_anchored;   // timestamp when object was
                                //anchored
} Lpo;
```

Fig. 7 The LPS is an array of LPOs

```
// Global map object
typedef struct
{
  short int id;
  short int type;
  int x;
  int y;
  float anchored;
} Go;
```

Fig. 8 The GS is an array of GOs

The GS structure is generated and updated by the GM and consists of the robot position with uncertainty measures and an array of Global Objects (GO) which can be seen in Fig. 8. It contains a world representation from a birds-eye view, a map of the soccer field. As the LPO it describes the position of the objects and the degree of anchoring. It also contains the position of the robot itself, with a certainty measure, in respect to the soccer field. The LPS is considered the most reliable information of the two, since the global localization data in the GM is derived from the LPS. The GM is however useful for the global view of the field and contains data about objects not currently in the camera view, in contrast to the LPS.

When the process of deriving the best behavior is done, the output of the RP to the HBM is a simple code that tells the HBM which behavior to execute.

4.2.2 Static Charge Configuration

The configuration of the static charges in a robot soccer game is very simple. The opponent goal should be positively charged and the own goal negatively charged. Five charges are placed in each goal ranging from goal post to goal post with an equal distance to each other. The charges in the opponent goal all have the same potential except for the middle charge, which is slightly higher. The own goal has exactly the same configuration except for inverted potentials (Fig. 9).

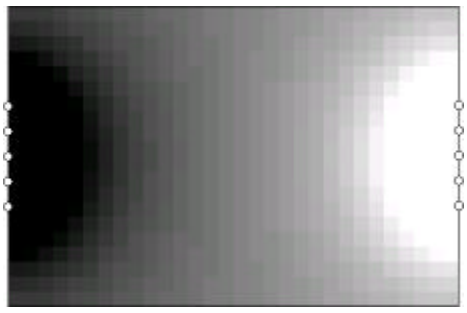

Fig. 9 Configuration of static charges

This configuration was chosen because it contains a minimal number of charges to decrease the execution time and sufficient coverage of potential to represent the goals in the soccer field. The difference in potential of the charge in the middle of the goal is

to further emphasize that this point is preferred as aim for shots because the probability of scoring in an open goal is the highest when aiming for the middle.

In earlier versions of the approach we discussed a number of small charges around the field marking the field boundaries. If the computation of the static charges is to be made on-line, this approach would however require more computing power, and it would not completely represent the lines of the borders unless the charges was set unlimitedly dense, and was therefore discarded in favor of the pre/post-condition approach. Methods of using pre-calculated matrices for the static charges were discussed but not implemented at this time for consistency reasons. In the RoboCup 2000 implementation of the approach no off-line calculations was done besides the rotation angles (see Fig. 11). By being consistent in computing all the charges on-line, changes in the charge configuration could easily be done, which proved successful during the competition.

4.2.3 Dynamic Charge Configuration

To represent the robot, four dynamic charges are used:

- One head charge, an agent-oriented charge with an offset of 130 mm from the center of the robot (the radius of the robot).
- Three environment-oriented charges; one positive charge placed straight in front, and two negative charges on either side marking the rear of the robot (Fig. 10). These charges are placed around the robot in the direction of the positive static charges.

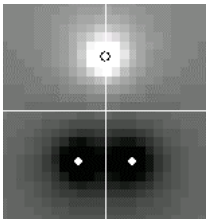

Fig. 10 Dynamic Charge Configuration

This particular placement of dynamic robot charges makes the robot move itself in to a strategically correct position with respect to the ball at all times. Due to the equivalent placement and potential of negative and positive static charges, the field is divide in to one positive and one negative side. At the own side the robot tends to position itself between the own goal and the ball, defending the own goal. At the opponent side it moves behind the ball in line with the opponent goal preparing to shoot. The direction of the environment-oriented charges can be seen in Fig. 11.

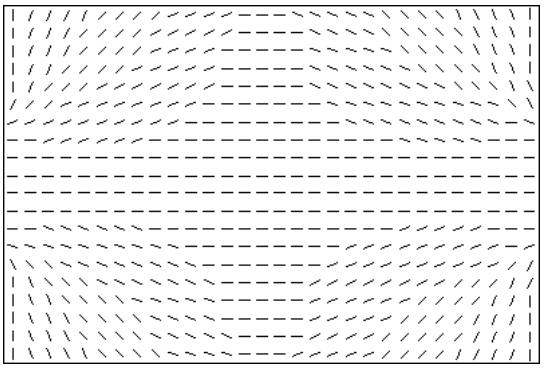

Fig. 11 Rotation angles for environmental-directed charges

Charges generated by mates and opponents where not used during the matches due to poor recognition capabilities, but the RP is prepared for calculating with other robots on the soccer field. During the matches there were at most five other robots on the soccer field and the robot detection feature was considered too unreliable to use in this situation. The visual recognition of other players was however successfully used during RoboCup Challenge Two and RoboCup Challenge Three, a part of the RoboCup separated from the actual matches. From the observations of the other teams'

strategies, Team Sweden seemed to be the only team in the Sony Legged Robot League who used visual robot detection.

Both the opponents and the mates are represented by one charge each. The opponent charge is of course negative while the mate charge is positive and the potential of the charges is proportional to the anchoring level of the object, i.e. the degree of vision. This makes the approach less error prone when provided with erroneous or insufficient perceptual input. When passing a mate, the best position of the ball for the other robot is immediately in front of it, towards the opponent goal, therefore the mate charge is placed 130 mm (the radius of the robot) in the direction of the opponent side. The position of the opponent charge is in the calculated center of the opponent.

4.2.4 Available Behaviors

The behaviors provided by the HBM and predicted in the RP in the matches are the following:

Behavior name	Description
FaceBall	Face in the direction of the ball
GoToBall	Go to the position of the ball
AlignBallAndNet1	Face the ball and the opponent goal
GoBehindBallNet1	Get in position behind the ball in line with opponent goal, not necessarily facing the goal or the ball
GoBetweenBallNet2	Get in position between the own goal and the ball, not necessarily facing the goal or the ball
PushBallNet1	Lightly push the ball towards the opponent goal and following it
KickBall	Kick the ball in the current facing direction
StealRight	Kick the ball to the right with the head
StealLeft	Kick the ball to the left with the head
DiveKick	Kick the ball with the head by diving forward

Table 4 Behaviors used in RoboCup 2000

4.2.5 The Evaluation Loop

```
// Behavior evaluation loop
while(behaviorArray[index] != END_OF_ARRAY) {

      // Evaluate each behavior in list. True if applicable
      if(EvaluateBehavior(behaviorArray[index], ballPot)) {

            // Add some more potential to the current behavior
            //  for 3 seconds.
            if(  (behaviorArray[index] == curBehavior) &&
                 ( curBehaviorCount < 3) ) {
                 ballPot *= 1.10;
            }

            // If higher charge, use this behavior
            if(ballPot > maxBallPot) {
                 maxBallPot = ballPot;
                 bestBehavior = behaviorArray[index];
            }
      }
      index++;
}
```

Fig. 12 Implementation of the evaluation loop in C++ code.

The “behaviorArray” contains the numeric values for each behavior in the HBM that should be evaluated. In turn each behavior is evaluated by the “EvaluateBehavior”

function returning true if the behavior is applicable for the current situation i.e. fulfilling the pre- and post-conditions for the situation, for instance assuring that the player is not set outside the field or in forbidden areas when executed. “EvaluateBehavior” predicts the behavior sent as argument, and returns the predicted potential of the ball when applicable. The behavior with the highest predicted ball potential wins the poll and is sent to the HBM for execution.

When the prediction of two or more behaviors results in approximately the same potential value there is a risk that these are starting to alternate between each other between evaluations. This causes the player to start executing different behaviors every second, which gives the impression of a lost and vacillating player and results in bad performance. That is why we add 10 % to the current behavior for the first three seconds of execution. The value of these numbers were calibrated during empirical studies of the robot in action, and proved sufficient.

As mentioned previously, the robot uses landmarks and goals for “self-localization”, i.e. finding it’s own position in the soccer field. Using the wrong behavior, such as a kick, in a time when it is not properly localized could be disastrous for the match, meaning a kick in the wrong direction or even scoring an own goal. No evaluation is therefore done when the localization is poor. The priority in such situation is to scan for landmarks to increase the certainty of the own position.

The GS structure contains, as mentioned earlier in this report, a global representation of the soccer field and the current position of the different objects in the match. When predicting a behavior, a copy of the GS is used and the objects, usually the ball and/or the player, are moved as though the behavior was really executed. Then the potential field in the position of the ball is calculated, and this value is the predicted potential of the behavior in the current situation. This is done for every behavior in the behavior set. The procedure is described schematically in Fig. 13.

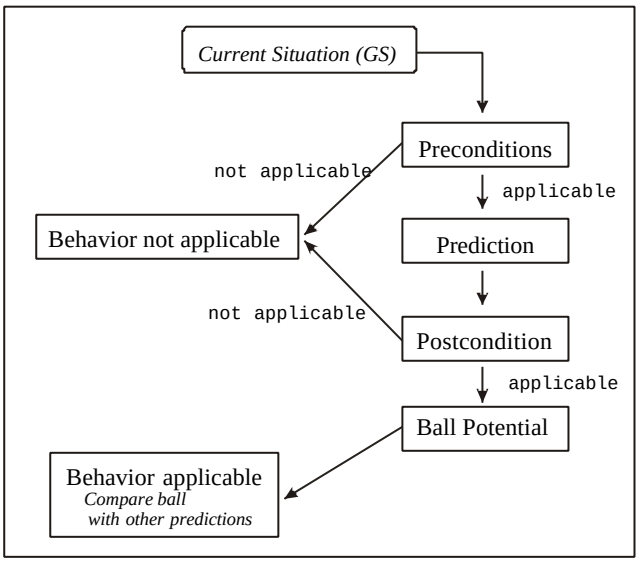

Fig. 13 Behavior prediction

5 EVALUATION OF THE ELECTRIC FIELD APPROACH

This chapter describes the evaluation of the Electric Field Approach on the AIBO platform, and the tools used in doing so. The performance of Team Sweden is also evaluated.

5.1 Testing the Reactive Planner Module

Testing and debugging on the AIBO platform can be both time consuming and tedious because of the complex relation between the modules of the system and the problem of programming real time systems with limited input and output capabilities.

5.1.1 Prototypes

During the development of the new RP with the EFA, three prototypes were produced. The first one was completely separated from the AIBO platform and was made to visualize the implications of different static charge configurations with colors and field lines. In time when the EFA was discussed, it was further developed and used to persuade that this approach was capable of fulfilling the requirements of the RP module in the Team Sweden system. It was also used for demonstrating the impact of the use of certain behaviors, and establishing the interface between the HBM and the RP, which at the time was loosely defined.

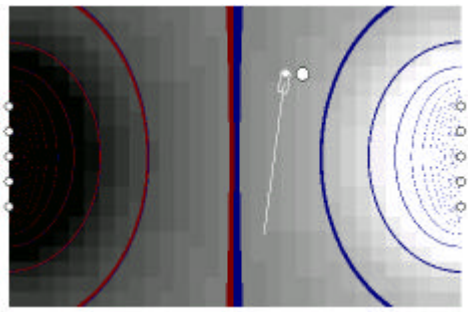

Fig. 14 The first prototype

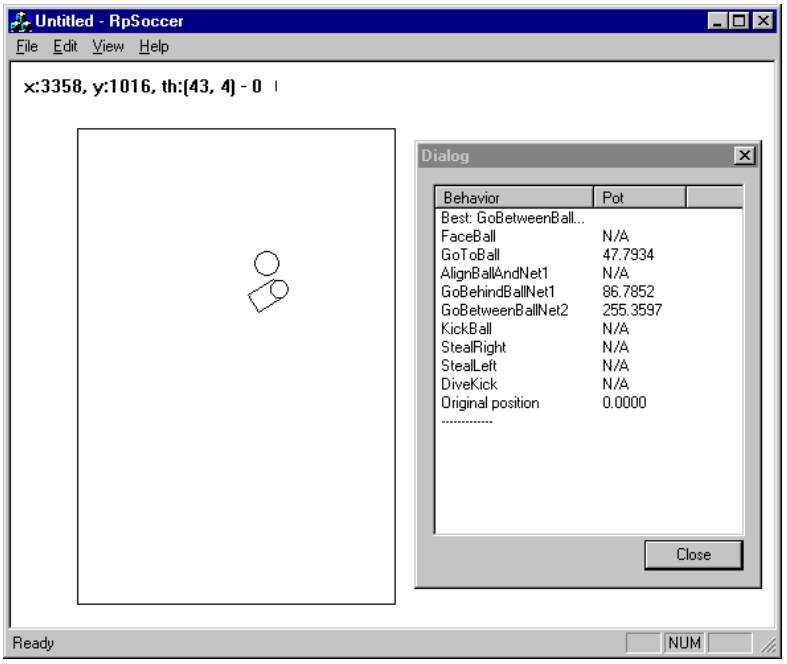

Fig. 15 Prototype for testing the EFA

The second prototype (Fig. 15) was using the RP source code and was used as support for the development. The lighting conditions of the field must be constant and very precise as well as the colors of the goals and landmarks. The condition of the

environment on location in Ronneby was not optimal because of the homemade soccer field that we had to paint ourselves, the shifting lighting conditions, and the color of the walls reflecting down on the field objects interfering with the color recognition. Therefore the second prototype was developed, simulating the other modules of the system and running the RP in the same way as on the actual platform. In this way behaviors could be tuned and pre- and post-condition established.

The third prototype was implemented after the competition mainly to visualize the RP functionality and to produce images for demonstration of the Electric Field Approach, but also to backtrack and see the disadvantages of the approach. Most of the images of the EFA in this thesis are produced by this application (Fig. 16).

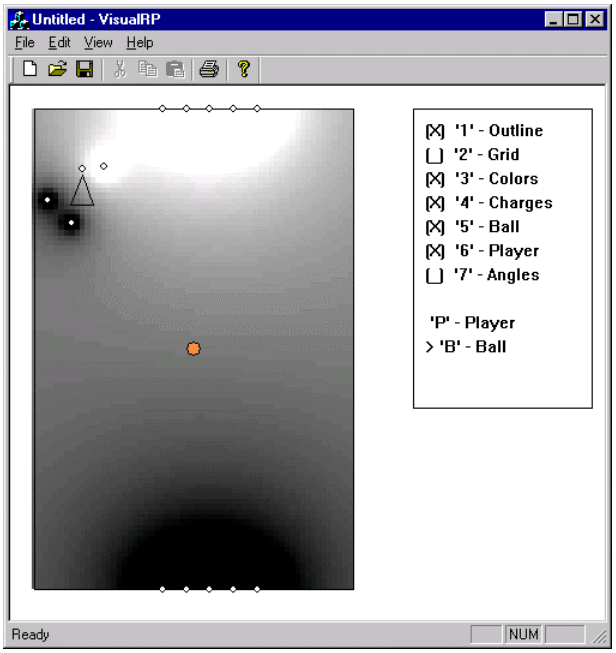

Fig. 16 Prototype for demonstrating the EFA

5.1.2 Tools

In the time of the games on location in Melbourne, the whole team put a lot of effort in fine tuning the system, and to adapt it to the current conditions there. By then we realized that we had to calculate the off-line variables of the RP in a more suitable way than before, so the Magnetic Field Generator, MagFieldGen, was developed[2].

The configuration of static charges could easily be changed, as well as the rotation angles reliant of the static charges. The position of the dynamic player charges could also easily be changed by this application.

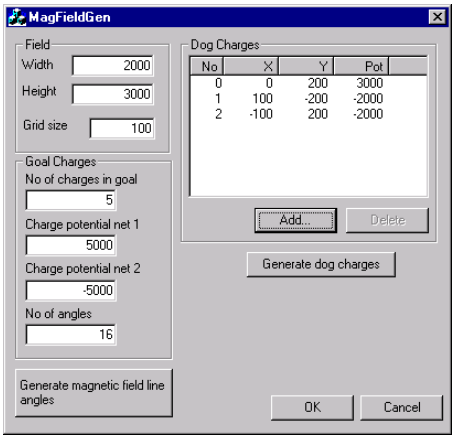

Fig. 17 Tool for calibrating the EFA

5.1.3 Debugging

There are no debugging capabilities in the development tools as such, so this had had to be done manually in the code by using old-fashioned "cout" statements

[2] At the time the approach was called "the Magnetic Field Approach" which was later changed to "the Electric Field Approach" to better reflect the method used.

controlled by "#define" pre-compile directives. An additional module was created for the particular purpose of debugging, making it possible to communicate with the robot at run-time using the serial port. This proved to be a very useful technique, which provided a fast way of debugging not only the RP but all the other modules as well.

5.1.4 Calibration

The prototypes developed were not sufficient for calibrating the charges and the pre- and post-conditions on the AIBO platform due to ever changing lighting conditions. The reliability of the distance measures to objects in the soccer field, such as the ball and landmarks were simply not good enough. The prototypes' measures though were exact at all times, which forced us to calibrate these variables manually on the robot.

5.2 Performance of Team Sweden

Team Sweden came on 5:th place of the 12 competing teams in the soccer matches of RoboCup 2000. Winner of both the matches and the compiled results of the three challenges were the University of New South Wales, who had a fast and effective walking style, good perceptual capabilities and an excellent method of controlling the ball and shoot it in a precise direction. In the RoboCup Challenge Team Sweden came in on 4:th place in the compiled result of the three competitions.

5.2.1 The matches

In total Team Sweden played 3 matches in the competition. The first one, played against Osaka was won with 3 to nil. The second match against Tokyo ended with 4-2 to Tokyo, and in the quarterfinals the Swedish team lost against CMU with 4-0.

During the games the configuration of charges where changed slightly. Instead of having five charges with equal potential on either side of the field, the middle potential was increased for the opponent goal and decreased for the own goal. New and better rotation values of the environment-oriented charges were also added. Fine-tuning of the preconditions, the perception and the behaviors was however a continuing and time consuming task, which was performed during the whole competition. The work was further delayed because of the lack of training fields to test the robots. The ones provided had not the same lighting conditions as the competition field, and was therefore not completely sufficient for testing the performance of the system.

5.2.2 RoboCup Challenge 1

For RoboCup Challenge (described in section 2.1.4.2) number one, *the Striker Challenge*, the RP was not configured differently than when playing the matches. The detection of the other robots, mates and opponents, where turned off because the player was alone on the field with the ball.

In this part of the competition Team Sweden came on 6:th place with a time of 2 minutes and 17 seconds. The winners, UNSW, made it on 46 seconds.

5.2.3 RoboCup Challenge 2

In *the Collaboration Challenge*, the RP had to be modified, but this was done in such a matter that the same system was used, but different compilation flags were set.

For robot number one (the passer), the static charges were completely removed, leaving only the mate charge active (indicating the position of the other player). This charge dominates the whole electric field and causes the passer to aim for his mate instead of the opponent goal. A pre- and post-condition was also set, so that no behavior executed should bring the passer over the half way line, which would result in a penalty.

The second robot, the shooter, had a similar pre- and post-condition forcing it to stay on the opponent side of the field. The static charges were also altered. By clearing

the own charges the opponent goal was made more dominating resulting in a more offensive strategy.

In RoboCup Challenge 2 Team Sweden was the fourth and last team to fully complete the challenge on time (2:34). One of our players happened to cross the half way line, resulting in a penalty of one additional minute to the resulting time. This happened because the pre- and post-conditions was set too optimistically and the self-localization was poor at the time. The winners, UNSW, made the challenge on 1 minute and 12 seconds.

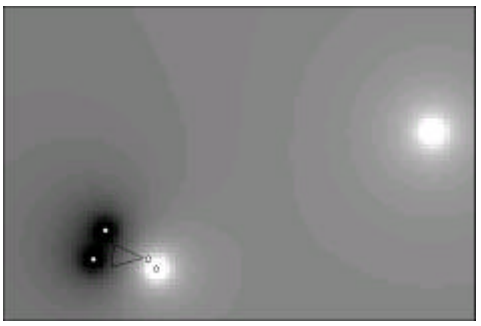

Fig. 18 Collaboration Challenge, passers view point.

5.2.4 RoboCup Challenge 3

In *the Collision Avoidance Challenge* the RP was also slightly reconfigured. Here the two mates have one negative charge each. Robot detection is switched on and the dark areas in the image are the mate charges. This causes the player to avoid the mates when going for the ball and trying to score.

In this challenge the Team Sweden came on second place after CMU, which was the best position for Sweden in a Challenge this year. The time was 1:15, six seconds from the winners.

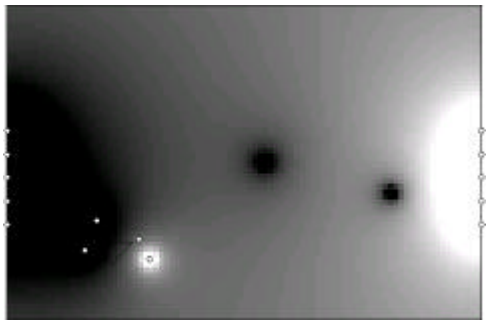

Fig. 19 Collision Avoidance challenge

5.3 Other Strategies

The purpose of this section is to show the different kinds of strategies used in the Sony Legged Robot League. All team strategies are not covered, but the descriptions are sufficient for showing the level of intelligence and logic for high-level strategy decisions. The documentation of the other team strategies used at RoboCup 2000 was not available at the time of writing this report. This section is be based on empirical studies supported by the RoboCup-99 papers for the Sony Legged Robot League [Veloso et al., 2000].

5.3.1 The matches

These are some of the strategies used:

- The strategy of *UNSW* [Lawther & Dalgliesh, 2000] is similar to the approach in Team Sweden in that the robot team consists of two forwards and one goalie. The fast and precise movements and ball handling, makes higher level of AI somewhat unnecessary, but the method used is a “Plans and Roles”. Four high-level behaviors, plans, seem to make up the Forward role; “Look for ball”, “Go to ball”, “Turn with ball” and “Shoot”.
- The –99 version of the *CMU* team strategy is described as a finite state machine: “*Each state corresponds to a set of behaviors that all accomplish the same goal. Each behavior expects a different amount of information to be available. The best behavior that has all its required information available is chosen to be executed.*” (p. 768, [Veloso et al., 2000b])

- The robot team of the *Baby-tigers, Osaka University*, had a very different walking style called "Trotting" a stiff walk, which proved to be precise during the RoboCup Challenge but rather unstable in the matches playing with other robots. The strategy, according to the team description paper from RoboCup-99 [Mitsunaga & Asada, 2000], consists of two programs; One for chasing the ball, and the other to play back a taught behavior. The system seems to have self-learning capabilities as they state that "*[the] strategy mostly depend on the human trainer*".
- In 1999, the *Humboldt Heroes* [Burkhard et al., 2000] were to use a Belief-Desire-Intention architecture (BDI [Russel & Norvig, 1995]), but took instead a simple reactive approach with five behaviors. These were "Look for the ball", "Run to the ball", "Search and positioning for the opponent goal", "Go with the ball to the opponent goal" and "Kick if you are near the opponent goal".

5.3.2 The RoboCup Challenge

The Challenges is considered only as a subordinate task of the competition, and is therefore rarely mentioned in the papers. The empirical studies is the only source of information in this case which maybe can be considered a partial and unreliable source, but still it is worth mentioning.

Generally the walking styles of the RoboCup Challenge did not differ from the styles used in the matches. The strategies however, where of course altered. According to the empirical studies of the games, Team Sweden seems to have been the only team using actual robot detection. In both Challenge 2 and 3 the problem of actual robot detection can be avoided by using strategies tailored for the rules of the specific challenges;

- In Challenge 2 the passer can shoot the ball in one of the opponent corners. This assures that the ball is within reach for the shooter, at the same time as it avoids an accidental goal from the passer. The shooter can wait for a period of time before starting to execute, and there by avoid stepping in to the forbidden side of the field, looking for the ball.
- Patrolling the field from side to side until the ball is detected visually can solve the Collision Challenge problem, thus making it able to go in a straight line to the ball and score. This was done by the CMU team, and possibly a few others.

6 DISCUSSION

In this chapter the quality attributes and the uniqueness of the Electric Field Approach are discussed.

6.1 The Team Sweden Requirements

The requirements for Team Sweden were the following: Scientific value, Generality, Effectiveness, and Robustness (see section 2.2.1). How well do these comply with the attributes of the Electric Field Approach?

6.1.1 Scientific value

The Team Sweden effort in RoboCup 2000 is a proof of that the approach is applicable on the RoboCup domain, which is a sub domain of autonomous robotics. The theory of using potential fields in navigation and path planning is not unique [Payton et al., 1990], but the combination of the one step look-ahead of the Reactive Planner Module, and the behavior elicitation not only restricted to pure navigation are new contributions. The Scientific value of the approach is therefore without question.

6.1.2 Generality

The approach is not only applicable in the RoboCup domain, but in other domains as well [Johansson et al., 2001]. One example is the Stigmergy and Collective robots [Beckers et al., 1994] which imitates the way termites (!) build nests. This is a method in the multi agent domain, which replicates a natural system of collaboration of social insects. Theoretically the EFA is applicable for this particular problem, though no practical simulation has been made to support this statement.

6.1.2.1 Stigmergy

The termites modify their environment by making little mud balls, which initially are placed at random. Each mud ball is marked with a pheromone. Other termites sense the pheromone impregnated mud balls, and prefer to put their mud balls in the vicinity, thus increasing the pheromone concentration at this particular place. After a while of carrying around the mud balls, the mud starts appearing in small heaps which becomes larger and larger until there is only large heap forming a pillar or an arch in the termite nest. This impressive work is made by tens of thousands of small termites using a simple but genial system of collaboration.

Collective agents in a multi-agent environment can with a few small modifications to the current Electric Field Approach use this method. Each “mud ball”, or in this case more likely objects that need to be gathered together, is a probe, thus introducing multiple probes in the system. Each probe has an undirected charge attached to it, making it “attractive” to other probes. The additive characteristic of the charges makes the area of gathered objects a more attractive than a single one. The same effect as with the pheromone in the social insect community is thus achieved by using multiple potential probes in the autonomous agent domain.

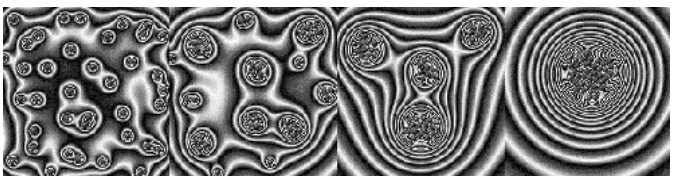

Fig. 20 Four levels of Stigmergy

6.1.3 Effectiveness

The effectiveness of the Electric Field Approach in the RoboCup domain can be argued. Other team strategies do not seem to use such complicated systems as the EFA with greater success. Of course many other parameters must be taken into account when analyzing why we did not win the RoboCup Sony Legged Robot League and it is not necessary because of the EFA.

One weakness of the RP is the slow update. Each second the situation is reevaluated and in worst case a whole second or more is spent executing the wrong behavior. A purely reactive strategy could compensate for this drawback. The performance of the Sony Legged Robot League participants is in the current situation much dependant on effective walking styles and good perceptual abilities, and not so much reliant on the high-level strategy decisions.

6.1.4 Robustness

Robustness as defined (section 2.2.1 and [Saffiotti et al., 2000]), is traceable in the Electric Field Algorithm. Because of the proportional decrement of charge potential for poorly anchored objects, and the level of localization that must be reached before evaluating behaviors, the RP makes the lower levels perform reasonable actions although provided with slightly erroneous or vague input.

6.2 Other Attributes

6.2.1 Configurability

In the RoboCup Challenge we changed the charge configuration, hence changing the strategy of the robot to better suite the needs of the specific tasks. This could be done fast and effective, and required hardly any alterations to the code and no alterations to the theory of the Electric Field Approach what so ever.

This shows that consistency in the application of the theory pays of in the end, and that the algorithm as implemented in the RP possesses a high level of configurability, or maintainability when defined as “the ability to change the system to new requirements”. The most common definition of maintainability is “Mean time to repair” [Gilb, 1988], and Boehm decomposes the attribute into “Testability”, “Understandability”, and “Modifiability” [Boehm et al., 1978] and the component “Modifiability” is the attribute which describes the EFA best in that respect. The way the strategy easily and quickly could be modified to suite the different tasks in the Sony Legged Robot League in RoboCup verifies this quality.

6.3 Similar Approaches

As mentioned earlier, the strategies of the other participating teams did seemingly not contain such an advanced approach of the high-level decision making as the Electric Field Approach. The other methods in the Sony Legged Robot League were more reactive and less planning, although the implementation of our approach has reactive components as well. The common method used were variations of the finite state machine, where the states define a few different situations in the soccer game.

The potential field approaches are traditional used for navigation and path planning [Payton et al., 1990], [Valavanis et al., 2000] and do not possess the ability to manipulate the environment the way the Electric Field Approach does. In robot navigation the own position is to be optimized, i.e. the robot itself should get from point A to point B. In terms of EFA the robot is also the probe in this case. The EFA is however more general due to the separation of the robot and the probe. During a robot soccer match for instance, the robot and the probe are separated, leading to a situation where the robot, unselfishly, strives to optimize the situation of the ball, thus placing it into the opponent goal.

7 CONCLUSION

These are the conclusions made from the development of the EFA and the results from the RoboCup competition.

7.1 The Electric Field Approach in the RoboCup Domain

The Electric Field Approach is well suitable for the RoboCup domain. It has the ability to chose between a set of behaviors in accordance to virtual charges representing the environmental objects. In the procedure of eliciting the best behavior to be performed for the current situation, the approach only provides a one step look-ahead. Due to the frequent changes of the situation in a robot soccer match, further planning than the one step look-ahead is not necessary.

The effectiveness in terms of response time of the RP is not sufficient. In the current version (used at RoboCup 2000) the RP is called every second, which is a too low frequency compared to purely reactive systems. The response time should be shortened, and one way of doing this is to call the RP more often. Such a modification could have impact on the performance of the system, due to the costly behavior predictions.

The implementation of the Electric Field Approach in the RoboCup domain has proven to be easy to configure, in that the configuration of charges is changed. An alteration in strategy and tasks is easily done with very little modification of the actual source code. This is particularly useful adapting to the different tasks in the Sony Legged Robot League.

7.2 The Electric Field Approach as Autonomous Agent Control

A more general case than the RoboCup domain is the field of Autonomous Agent Control. The Stigmergy example (see section 6.1.2.1) is one possible application of the Electric Field Approach.

The focus on optimizing the situation of the probes, which can be the agent itself as well as some other object, further enhances the agent abilities for performing other tasks than purely navigation. This feature is unique amongst the existing potential field approaches, which is restricted to path planning and autonomous navigation [Beckers et al., 1994].

Enhancements to the approach could be applied that further extends the one look-ahead to longer plans, as well as self-learning capabilities, although not included in the current version of the approach.

7.3 Strengths and Weaknesses

These are the strengths and weaknesses that have been identified during development of the theory and the implementation:

+ A high level of configurability for change of strategy and tasks.
+ Good performance in spite of a high degree of uncertainty.
+ The approach can be applied on various domains.
+ Errors in perceptual input have low impact on the produced result.
+ The approach is not limited to physical motion only, but can also manipulate its environment.

- One step look-ahead might lead to myopic behavior.
- Low response time. Evaluation of the current situation is only performed once a second.
- Expensive calculations. Predicting every behavior each second requires more computational power than purely reactive approaches.

7.4 Future Work

These are proposals for future work on the EFA:

- Extend the Approach for dealing with longer plans than the current one step look-ahead.
- The calculation of the electric field requires a considerable amount of computing power. How can the algorithm be optimized to be able to perform elicitation of more behaviors and/or with higher frequency?
- The Approach does not include agent collaboration. How can this be included in the EFA?
- Theoretically the Approach can be applied on other domains than the RoboCup domain, but this is yet to be proven.
- The Approach does not include any self-learning processes at this time, predefined behaviors are therefore necessary as input to the EFA. Is it possible to extend the Approach with this type of functionality?

ACKNOWLEDGEMENTS

The author would like to thank the following persons for the help and support in working with and writing this thesis.

- Stefan Johansson, my supervisor and the originator of the Electric Field Approach.
- Alessandro Saffiotti, team leader of Team Sweden, for guidance and interest in every thing from the theory to the implementation.
- Department of Software Engineering and Computer Science at BTH, for sponsoring the AIBO-robots and the trip to Australia, thus making it possible for us to participate in RoboCup 2000.
- Britt-Marie Lindberg, Pernilla Johnson, Ina Bergman and Elisabeth Smeds, for the support and the laughs during the dark hours in the RoboCup room.
- Kevin LeBlanc, Zbigniew Wasik and Mikael Karlström, for the help with color tables, compilers and every thing else that we did not have a clue about when we started.
- The Sony Corp, for sponsoring the Sony Legged Robot League in RoboCup and developing the fantastic AIBO entertainment robot.
- My family, who has always supported me in my studies whether comprehensible or not.

REFERENCES

[Beckers et al., 1994]	R. Beckers, O.E. Holland, and J.L. Deneubourg. From local actions to blobal tasks: Stigmergy and collective robotics. In R.A. Brooks and P. Maes, editors, Artificial Life IV, Proceedings of fourth international workshop on the synthesis and simulation of living systems, pages 181-189. MIT Press, 1994.
[Boehm et al., 1978]	B.W. Boehm, J.R. Brown, J.R. Kaspar et al., Characteristics of Software Quality, TRW Series of Software Technology, Amsterdam, North Holland, 1978.
[Burkhard et al., 2000]	H-D. Burkhard, M. Werner, M. Ritzschke, F. Winkler, J. Wendler, A. Georgi, U. Düffert, H. Myritz, Humboldt Heroes in RoboCup-99, In M. Veloso, E. Pagello, H. Kitano, editors, RoboCup-99: Robot Soccer World Cup III, pages 770-773, Springer-Verlag, 2000
[Buschka et al., 2000]	P. Buschka, A. Saffiotti, and Z. Wasik. Fuzzy Landmark-Based Localization for a Legged Robot. Proc. of IROS-2000 pp. 1205-1210, Takamatsu, Japan, 2000
[Forslund et al., 2001]	E. Forslund, Å. Gustavsson, C. Lilja, The process of creating new walking styles for the Sony Four Legged Robot, BTH, 2001
[Gilb, 1988]	T. Gilb, Principles of Software Engineering Management, Addison Wesley Longman Limited, 1988
[Johansson et al., 2001]	S. Johansson, J. Johansson, A. Saffiotti. An Electric Field Approach to Autonomous Robot Control, TNr: 412, BTH, 2001
[Lawther & Dalgliesh, 2000]	M. Lawther, John Dalgliesh, UNSW United, In M. Veloso, E. Pagello, H. Kitano, editors, RoboCup-99: Robot Soccer World Cup III, pages 788-791, Springer-Verlag, 2000
[Lindberg, Johnson, 2000]	B-M Lindberg, P. Johnson, Motions to make autonomous four legged robots play soccer in RoboCup, BTH, 2000
[Mitsunaga & Asada, 2000]	N. Mitsunaga, M. Asada, BabyTigers-99: Osaka Legged Robot Team, In M. Veloso, E. Pagello, H. Kitano, editors, RoboCup-99: Robot Soccer World Cup III, pages 762-765, Springer-Verlag, 2000
[Payton et al., 1990]	D.W. Payton, J.K. Rosenblatt, and D.M. Keirsey. Plan guided reaction. IEEE Transactions on Systems, Man, and Cybernetics, 20(6): 1370-1382, Nov/Dec 1999.
[Reese Hedberg, 1997]	Sara Reese Hedberg, Robots playing soccer? RoboCup poses a new set of AI research challenges, IEEE Concurrency, vol5 no 4, Oct/Dec, 1997
[RoboCup, 2000]	http://www.robocup2000.org, last checked March 2001.
[Russel & Norvig, 1995]	Stuart Russel and Peter Norvig, Artificial intelligence: a modern approach, Prentice Hall, 1995
[Saffiotti et al., 2000]	Alessandro Saffiotti et al, The “Team Sweden” Entry at RoboCup 2000 Confidential Report, 2000

[Saffiotti, 1997]	A. Saffiotti, The Uses of Fuzzy Logic in Autonomous Robot Navigation, Soft Computing, 1(4):180-197, 1997. http://aass.oru.se/Living/FLAR, last checked March 2001
[Saffiotti, 2000]	A. Saffiotti, The thinking cap home page, http://www.aass.oru.se/~asaffio/Software/TC/, last checked March 2001
[Sakamura, 1999]	K. Sakamura, Entertainment and Edutainment, IEEE Micro, pages 15-19, Nov/Dec 1999
[Sony Corp., 2000]	Sony Four Legged Robot Football League Rule Book (Draft on 11/15/2000), http://www.robocup.org/regulations/images/LeggedRule2000.pdf, last checked March 2001.
[Valavanis et al., 2000]	K.P. Valavanis, T. Hebert, R. Kolluru, N. Tsourveloudis, Mobile Robot Navigation in 2-D Dynamic Environments Using an Electrostatic Potential Field. IEEE Transactions on Systems, Man, and Cybernetics, 30(2): 187-196, March 2000.
[Veloso et al., 2000a]	M. Veloso, E. Pagello, H. Kitano, editors, RoboCup-99: Robot Soccer World Cup III, pages 758-798, Springer-Verlag, 2000
[Veloso et al., 2000b]	M. Veloso, Scott Lenser, E. Winner, J.Bruce, CM-Trio-99, In M. Veloso, E. Pagello, H. Kitano, editors, RoboCup-99: Robot Soccer World Cup III, pages 766-769, Springer-Verlag, 2000

APPENDICES

Appendix A

Appendix A contains a list of the participating teams in the Sony Legged Robot League in RoboCup 2000.

Team Name	University	Location	Team Leader
CM-Pack'00	Carnegie Mellon University	Pittsburgh, Pennsylvania, USA	Prof. Manuela Veloso
Les 3 Mousquetaires	Laboratorie de Robotique de Paris	Velizy, France	Prof. Pierre Blazevic
Baby Tiger	Osaka University	Osaka, Japan	Prof. Minoru Asada
ARAIBO	University of Tokyo	Tokyo, Japan	Prof. Tamio Arai
UNSW United	University of New South Wales	Sidney, Australia	Prof. Claude Sammut
Upennalizers	University of Pennsylvania	Philadelphia, Pennsylvania, USA	Prof. Jim Ostrowski
McGill	McGill University	Montreal, Quebec, Canada	Prof. Jeremy Cooperstock
Humboldt Heroes	Humboldt University Berlin	Berlin, Germany	Prof. Hans-Dieter Burkhard
Team Sweden	ORU, BTH, KTH	Örebro, Ronneby, Stockholm	Prof. Alessandro Saffiotti
Robo Mutts	University of Melbourne	Melbourne, Australia	Prof. Alan Blair, Dr. Nicholas Barnes
SPQR	Universita' di Roma "La Sapienza"	Rome, Italy	Prof. Daniele Nardi
Essex Rovers	University of Essex	Colchester, England, UK	Dr. Huosheng Hu

Table 5 Participating teams in RoboCup Sony Legged Robot League

Appendix B

Appendix B contains a full listing of the final results for the Sony Legged Robot League in RoboCup 2000.

RoboCup Matches

Position	Team	Location
1.	UNSW	Australia
2.	CM-Pack'00	Pittsburgh, Pennsylvania, USA
3.	Les 3 Mousquetaires	France
4.	ARAIBO	Tokyo, Japan
5.	**Team Sweden**	**Sweden**
6.	Baby Tiger	Osaka, Japan
7.	SPQR	Italy
8.	Humboldt Heroes	Germany
9.	McGill	Canada
10.	Essex Rovers	England
11.	Upennalizers	Philadelphia, Pennsylvania, USA
12.	Robo Mutts	Melbourne, Australia

Table 6 RoboCup match results

RoboCup Challenge 1

Position	Team	Location	Result
1.	UNSW	Australia	0:46
2.	Baby Tiger	Osaka, Japan	0:14
3.	Les 3 Mousquetaires	France	1:20
4.	Essex Rovers	England	1:34
5.	ARAIBO	Tokyo, Japan	1:45
6.	**Team Sweden**	**Sweden**	**2:17**
7.	Humboldt Heroes	Germany	39 cm
8.	CM-Pack'00	Pittsburgh, Pennsylvania, USA	42 cm
9.	Upennalizers	Philadelphia, Pennsylvania, USA	60 cm
10.	SPQR	Italy	68 cm
11.	Robo Mutts	Melbourne, Australia	90 cm
12.	McGill	Canada	DQ

Table 7 RoboCup Challenge 1 results. (DQ = disqualified)

RoboCup Challenge 2

Position	Team	Location	Result
1.	UNSW	Australia	1:12
2.	Baby Tiger	Osaka, Japan	1:26
3.	CM-Pack'00	Pittsburgh, Pennsylvania, USA	1:14
4.	**Team Sweden**	**Sweden**	**2:34 + p**
5.	Robo Mutts	Melbourne, Australia	217 cm
6.	ARAIBO	Tokyo, Japan	4 cm + p
7.	Upennalizers	Philadelphia, Pennsylvania, USA	133 cm + p
8.	Humboldt Heroes	Germany	190 cm + p
9.	Les 3 Mousquetaires	France	250 cm + p
10.	Essex Rovers	England	204 cm + p
11.	SPQR	Italy	DQ
12.	McGill	Canada	DQ

Table 8 RoboCup Challenge 2 results. (p = penalty)

RoboCup Challenge 3

Position	Team	Location	Result
1.	CM-Pack'00	Pittsburgh, Pennsylvania, USA	1:09
2.	**Team Sweden**	**Sweden**	**1:15**
3.	McGill	Canada	2:31
4.	ARAIBO	Tokyo, Japan	2:35
5.	Baby Tiger	Osaka, Japan	1:07 + p
6.	UNSW	Australia	1:16 + p
7.	Robo Mutts	Melbourne, Australia	13 cm
8.	Humboldt Heroes	Germany	57 cm
9.	Essex Rovers	England	66 cm + p
10.	SPQR	Italy	150 cm + p
11.	Upennalizers	Philadelphia, Pennsylvania, USA	DQ
12.	Les 3 Mousquetaires	France	DQ

Table 9 RoboCup Challenge 3 results

RoboCup Challenge overall results

Position	Team	Location
1.	UNSW	Australia
2.	Baby Tiger	Osaka, Japan
3.	CM-Pack'00	Pittsburgh, Pennsylvania, USA
4.	**Team Sweden**	**Sweden**
5.	ARAIBO	Tokyo, Japan
6.	Robo Mutts	Melbourne, Australia
7.	Humboldt Heroes	Germany
8.	Essex Rovers	England
9.	McGill	Canada
10.	SPQR	Italy
11.	Upennalizers	Philadelphia, Pennsylvania, USA
12.	Les 3 Mousquetaires	France

Table 10 RoboCup Challenge overall results