

# Strategies of Teams in Soccerbots

**R. Geetha Ramani<sup>1</sup>, Dr.R.Subramanian<sup>2</sup> and M.Sindurathy<sup>3</sup>**

<sup>1</sup>Assistant Professor, Department of Information Technology, Pondicherry Engineering College

<sup>2</sup>Professor, Department of Computer Science, Pondicherry University

<sup>3</sup>Department. Of Computer Science, Pondicherry Engineering College

Corresponding author E-mail: rgeetha@yahoo.com

**Abstract:** TeamBots was designed and developed by Tucker Balch at Carnegie Mellon University, as development and testing environment for studying collaborative robot behaviour. This paper presents the strategies behind all the robot soccer teams available in the Teambots (soccerbot) package. The performances of the various soccer teams were analysed. This paper will aid the researchers to develop the soccer team with best strategies.

**Keywords:** Teambots, Robocup, Simulation League, Strategies, Soccerbots, Teams.

## 1. Introduction

The Robot World Cup Initiative known as RoboCup ([www.robocup.org](http://www.robocup.org)) is proposed as a standard problem for research in the areas of AI and robotics, requiring the use of several technologies and research in a wide range of areas in AI and robotics. It can be seen as an international research and education initiative, which attempts to foster artificial intelligence and intelligent robotics research by providing a standard problem where wide range of technologies can be integrated and examined.

## 2. Robocup

The main focus of the RoboCup organization is competitive soccer. The competition has several leagues for different sizes of robots and a simulation league, which uses the SoccerServer (Mackworth, A., 1992, David Andre, 1998); allowing software agents to compete. Additionally, there are leagues to accommodate different number of players on the field. The soccer domain provides a very complex problem requiring robust solutions. The RoboCupSoccer has the following leagues:

- Simulation League
- Small-size robot league
- Middle-size robot league
- Four-legged robot league
- Humanoid league

One among the main events of the RoboCup Simulation League is the simulated soccer matches. Both 2D and 3D simulation leagues are conducted. In the 2D Soccer Competition of the RoboCup Simulation League, teams of 11 autonomous software agents per side play each other using the RoboCup soccer server simulator. There are no actual robots in this league but spectators can watch the action on a large screen, which looks like a giant computer game. Each simulated robot player may have its own play strategy and characteristic and every simulated team actually consists of a collection of programmes.

## 3. TeamBots

TeamBots is a Java-based collection of application programs and Java packages for multiagent mobile robotics research ([www.teambots.org](http://www.teambots.org));. The TeamBots distribution is a full source-code release.

Originally titled JavaBots, due to its being written entirely in Java, the simulator is designed to provide a rapid prototyping environment for multirobot logic development (Balch and Ram, 1998).

The simulation environment written entirely in Java. TeamBots is not the only soccer simulation environment available for use. The official RoboCup Soccer Simulator is, in fact, the more popular and well known, owing to its use in the yearly RoboCup tournaments.

TBSim is the title given to the TeamBots simulator. TBSim was designed such that it would simulate the operations of a variable number of robots on a field of arbitrary size. Obstacles and spherical objects are also simulated in 2D.

## 4. TeamBots distribution :

The TeamBots distribution includes several example systems that can run in simulation. The following are the domains which are included in the teambots package.

- Cy: this is a simulation of Probot's Cy robot.
- SoccerBots: this is a simulation of RoboCup F-180 league soccer.
- Book: includes example exercises being developed for the book *Designing Robot Behavior* by Tucker Balch. These primarily illustrate principles of behavior-based control.
- CTF: an implementation of Capture-The-Flag, a multi agent adversarial game.
- Forage: a basic implementation of multi agent foraging.
- ForageAAAI97: the actual source code used to win the AAAI-97 mobile robot competition.

- Roads: two example uses of simulation environments including roads. One example includes two different types of roads and a bridge. The other is an entire city map. Any arbitrarily complex environment including different traversability constraints can be built.
- Walls: an example simulation environment with walls. The official RoboCup Soccer Simulator is, in fact, the more popular and well known, owing to its use in the yearly RoboCup tournaments [Anon., 2003]. Out of these domains, the one which supports soccer domain is the soccer bots. Given below is a brief overview of SoccerBots.

## 5. SoccerBots

SoccerBots is a simulation of RoboCup F-180 league soccer. SoccerBots simulates the dynamics and dimensions of a regulation RoboCup small size robot league game. Two teams of five robots compete on a ping-pong table by pushing and kicking an orange golf ball into the opponent's goal.

SoccerBots runs in the TBSim application using a configuration file that tailors it for soccer games. The same control systems that run in simulation can also operate real robot hardware using the TBHard application.

Soccerbots has a total of 21 teams in it. Each of the team follows different strategies and this paper explains in detail about the strategy used by each of the 21 teams.

### 5.1 Soccerbots Rules and Dynamics

These rules were drawn from RoboCup's regulations for small size robot league with differences noted.

*Field dimensions:* The field is 152.5cm by 274cm. Each goal is 50cm wide. The defense zone, centered on the goal, is 100cm wide by 22.5cm deep. In the coordinate system used in the simulation, the center of the field is (0, 0) with +x to the right or east and +y up or north. The team that starts on the left side is called the west team and the team on the right is called the east team.

*Robots:* Robots are circular and 12cm in diameter. In RoboCup 15cm is the maximum size. The simulated robots can move at 0.3 meters/sec and turn at 360 degrees/sec.

*Team:* A team consists of no more than 5 robots.

*Ball:* The simulated ball represents an orange golf ball. It is 40mm in diameter and it can move at a maximum of 0.5 meters/sec when kicked. It decelerates at 0.1 meters/sec/sec. Ball collisions with walls and robots are perfectly elastic.

*Coloring:* Even though coloring of the field doesn't really matter in the simulation, the image drawn on the screen matches the RoboCup regulations.

*Length of the game:* For now, once the game starts, it runs forever.

*Wall:* A wall is placed around the entire field, except the goals. Ball collisions with walls are perfectly elastic. Robots may slide along walls if they aren't headed directly into them.

*Defense zone:* A defense zone surrounds the goal on each side. It is 22.5cm from the goal line and 100cm wide. Only one defense robot can enter this area.

*Robot marking:* Robots are marked with a two-color checkerboard pattern. In RoboCup rules they are marked with two colored ping-pong balls.

*Communication:* There is no facility for robot-to-robot communication, but this may be added in it.

*Goal keepers:* At present, the goal keeper has no means for grasping the ball. The system may be improved to allow for this.

*Kick-off/Restart/Stop:* Kick-off, restart and stop of the game are managed automatically. The west team gets to kick-off first, with subsequent kick-offs made by the scored-against team.

*Robot positions at kick-off:* The robots are automatically at kick-off.

Specific positions are not specified in the RoboCup rules, except that only the kick-off team can be within 15cm of the ball.

*Fouls/Charging:* Currently no fouls are enforced by SoccerBots except a shot clock. At the beginning of a point, the clock is set to countdown from 60 seconds. If no score occurs in that time, the ball is returned to the center of the field. The ball is also returned to the center if it gets stuck for a period of time. RoboCup enforces several fouls and charging violations.

### 5.2 How Soccerbots Works:

*The description file:* SoccerBots runs under the TBSim simulator. At runtime, TBSim reads in a description file that tells it what sorts of objects will be in the present simulation and where they should appear in the environment. Obstacles, small soccer robots, Nomad 150 robots, golf balls and squiggle balls have all been implemented. By making simple modifications to it, the team or the colors of the robots can be changed.

*The simulation kernel:* Each object in the simulation includes a drawing method and a dynamics simulation method. The simulation kernel, or core, runs in a loop calling the two methods for all the objects in the simulation at each step. It also keeps track of time, and passes elapsed time to the dynamics methods, so they know how far to move their associated simulated objects. The simulation can run in real-time or faster or slower than real-time according to how the time variable is set in the description file.

*Robot control systems:* In addition to the drawing and dynamics methods, robots have a control system object that considers sensor inputs and makes actuator outputs. The control system has to be modified to experiment with different soccer strategies. The control system has no way of knowing whether it is running in a simulation or on real robot hardware. This will make it convenient to modify the control software between simulation and real runs.

*Capabilities of a SoccerBots robot:* The interface between the robot's control system, its sensors and actuators is

provided by the API called SocSmall, a synopsis of which is:

#### 5.3 Sensors :

Detect whether the robot is in a position to kick the ball.

- Get a vector pointing to the ball.
- Get a vector pointing to the team's goal.
- Get a vector pointing to the opponents' goal.
- Get an array of vectors pointing to all the other players.
- Get an array of vectors pointing to the robot's teammates.
- Get an array of vectors pointing to the robot's opponents.
- Get the player's number on the team (0 - 4), 0 is the goalie.
- Get the player's position on the field.
- Get the player's heading.
- Get the time in milliseconds since the game started.

#### 5.4 Actuators:

- Kick the ball at 0.5 meters/sec.
- Push the ball by driving over it.
- Set desired heading. The robot can turn at 360 degrees/sec.
- Set desired speed, up to 0.3 meters/sec.

To develop a soccer team using SoccerBots, the soccer team should extend the ControlSystemSS class to define the team strategy. The robocup.dsc file should be modified to load the particular team for simulation.

#### 5.5 Actions

The following are the possible actions in robotic soccer.

- Drive: The attacker players can drive if their front is clear.
- MoveToBall: The closest player which is close enough moves to the ball
- GetBall: If the player is in a good position and angle, he gets the ball.
- TakePositionAttack: Depending on the strategy, player takes attack position.
- TakePositionDefense: Depending on the strategy, player takes attack position.
- PlayMidfield: Plays as a midfield player
- Playdefender: Plays as a defense player.
- KickBallToGoal: Kicks the ball towards the goal.
- KickBallToPlayer
- Stop: It is selected to halt a robot.
- Wandering: The robot moves to and fro in parallel to the length of the ground.
- SweepBall: It is selected for a robot to attempt to take out a ball bumped into a corner of the playground.
- Position\_To\_SweepBall: It is selected to put the robot in a new posture in anticipation of performing the SweepBall action.
- KickAway:
- Block : It keeps the ball off from opponent robots.

## 6. Teams in Soccerbots

SoccerBots has a number of soccer teams in the teams package. Out of those teams one of the best performing team was used in the project for evaluating the individuals. This team, namely AIKHomoG is a homogeneous team, which uses dynamic role assignment for assigning the roles to the agent. There are both heterogeneous and homogeneous teams in the package, most of which have hand-coded strategies.

1. AIKHomoG
2. BasicTeam
3. BrianTeam
4. BriSpec
5. CDTeamHetero
6. CommTeam
7. DaveHeteroG
8. DoogHeteroG
9. DoogHomoG
10. DTeam
11. FemmeBotsHeteroG
12. GoToBall
13. JunTeamHeteroG
14. Kechze
15. LoneForwardTeamHomoG
16. MattiHetero
17. PermHomoG
18. SchemaDemo
19. SchemaNewHetero
20. SibHeteroG

## 7. Strategy of teams

After closely analysing the various teams, their strategies[16\*] were identified. This includes the author, type of team and the strategy they follow. The teams can either be homogeneous or heterogeneous.

### 7.1 AIKHomoG

Author : Håkan L Younes.

Type : Homogeneous (All individuals behave using the same logic.)

Strategy : If the robot is not the closest to the ball, a force calculation is made. Negative forces are added for teammates and walls, and a positive force is added for the opponent goal's position. Lastly, a positive force is added for the two offensive positions just outside the corners of the opposition's goal.

A check is made to see if there is already someone tending goal, and if there is not, then the robot heads back to perform this task. Otherwise, the robot travels in the direction indicated by the force calculation. If the robot is the closest to the ball, then the robot calculates the best position to kick the ball from, goes there, and kicks it.

### 7.2 BasicTeam

Author: ejona

Type: Homogeneous

**Strategy:** The strategic places regarding the position of the robots are first computed. Then they are directed to the correct positions according to the number assigned to them. The goalkeeper moves to the goalie position if he is far away from the ball. Similarly the other players move towards the midback, northspot and lead forward position if far away from the ball.

### 7.3 BrianTeam

**Author :** Brian McNamara.

**Type :** Homogeneous

**Strategy :** The strategy is documented in the source code as follows:

**Drive:** If the ball is in the player's hands and if the front is clear he drives the ball.

**KickAway :** If the ball is in our goal, then the kick the ball away from the goalpost

**GoToBall :** If the player is closest to the ball or closest to the goal, then move behind the ball.

**GoToBall :** If the player is closest to our goal, then move to the point between the ball and our goal.

**TakepositionAttack :** If the player is closest to opponent then take the attacking position.

### 7.4 BriSpec

**Author :** Brian McNamara

**Type :** Unknown

**Strategy:** The members of this team play three different positions. One player always remains at the back of the field; it aligns itself between the ball and the goal. Three players play mid-field positions; they stay behind the ball and attempt to spread out from one another. The remaining agents stay in front of the ball, in expectation of a pass.

### 7.5 CDTeamHetero

**Author :** Cristian Dima

**Type :** Heterogeneous

**Strategy:** If the robot is closest to the ball, it attempts to kick the ball into the opponent's net. Otherwise, it behaves according to its role on the field, "forward," "assistant forward," "blocker," "defender" or "goalkeeper."

If the ball is in its team's end, the "forward" waits at centre for the ball to come out; then it heads to its kicking location just outside the opponents net. The "assistant forward" does the same, only it hangs back slightly from the forward. The "blocker," "defender" and "goalkeeper" all employ the same strategy; keep their bodies in between the ball and the centre of the net. The blocker does it at three goal widths out, the defender at two, and the goalkeeper does it from flush against the net.

### 7.6 CommTeam

**Author :** Unknown

**Type :** Heterogeneous

**Strategy :** In this team, the robots communicate with each other by sending occasional messages back and forth. All

the robots listen for incoming messages but only one of them actually sends them. The message indicates whether the robot can kick the ball or not. Broadcasting and Multicasting of messages are also available.

### 7.7. DaveHeteroG:

**Author:** Unknown

**Type:** Heterogeneous

**Strategy:** The DaveHeteroG team follows this simple tactic; figure out who on each team is closest to the ball, and then match up the rest of our players to block theirs. For this strategy to work, three classes of individuals are employed, the "goalie", "offender", and general purpose. The "goalie" acts as the team's goalkeeper, staying flush against the goal, and attempting to stay at the same position north and south as the ball. Should the ball's position exceed the limits of the goal, the goalie stops at the corner of the goal it's tending. Of note is the fact that the goalie is not a set individual; in the event that another robot gets closer to the goal than the current goalie, they become the new goalie. This is highly effective in combating teams that employ a strategy of pinning the goalkeeper away from the net. The "offender" constantly tries to get behind the ball and kick it towards the opponent's goal. The rest of the team's players are general purpose. Their strategy is to match up against the opposition's players, each one taking on the closest opponent to them not already taken. They then attempt to block these players from doing their jobs by pushing them away from everyone else, in a scattering attempt. Should one of these players get too close to the goal, they will swap in and become the new goalie.

### 7.8. DooGHeteroG

**Author:** Doog Fubar

**Strategies:** Attack, Goon, Goalie, Tocchet.

The players strategy changes according to the condition specified. (Goon is not Goon always)

**Attack:**

If ball near the play attacker and kick it.

**Goalie:**

If goalie is too far out of goal in x dir, get back to the position. Otherwise, calculate projected ball trajectory.

If ball is headed into goal, block it! - done by finding current position and ball position

**Goon:**

Attack the undefended opponent.

### 7.9. DooGHomoG:

**Author:** Doog Fubar

**Strategy :**

Same as DoogHetero but no Tocchet.

Attack, Goon, Goalie

The players strategy changes according to the condition specified. (Goon is not Goon always)

**Attack:**

If ball near the play attacker and kick it.

Goalie:

If goalie is too far out of goal in x dir, get back to the position. Otherwise, calculate projected ball trajectory.

If ball is headed into goal, block it! - done by finding current position and ball position

Goon:

Attack the undefended opponent.

#### 7.10. DTeam:

Author: David H. Johnson

Type: A reflexive, heuristic, heterogeneous soccer team.

Strategy: the following are the strategies of all the players.

Goalie:

- The goalie is responsible for keeping itself between the ball and the goal at all times. T
- The goalie must also deflect a ball if it is dangerously close to the goal.
- If the goalie is too far from the goal, or outside the width of the goal, it will return to the center as quickly as possible. Otherwise
- it will try to keep the ball from going north or south along the y axis.
- It stops when the ball is sufficiently close to its x-axis to avoid moving past the ball. The goalie moves toward the wall of the goal at all times.
- If the ball is behind the goalie the goalie will chase it down and kick it out as quickly as possible

Offside:

- This player targets the opponent who is closest to the opponents goal.
- Once the opponent is targeted "Offside" calculates and goes directly toward the point directly behind the opponent toward his goal.
- On the way to this point "Offside" avoids collision with any players other than the target.
- Offside serves two purposes.
- One is to serve as a blocker to keep the target from rushing an incoming teammate driving to score.
- Another is to effectively combat the traditional goalie.

Designated Driver:

The Designated Driver tries to continually drive the ball.

Backup:

This robot calculates a spot three robot radii from the ball towards his own side and tries to get behind it with respect to the opponents goal. If "Backup" is the closest robot to the ball it will try to drive the ball. This serves as a backup for the designated driver, hence the name "Backup".

Center:

This robot tries to get behind the center of the field and the opponent's goal.

If it is the closest robot to the ball then it tries to drive the ball. The main purpose of this robot is to sit at center field and wait for the fast break to come.

#### 7.11. FemmeBotsHeteroG :

Author: M. Bernardine Dias Robotics Institute Carnegie Mellon University

Type : Heterogeneous

Strategy:

Players: Center, Attack (2), Goalkeeper, Defender.

Goalkeeper: The "goalkeeper" waits at the net for the ball, tracking its movements back and forth (north and south) across the field, as far as the net goes. As the ball draws closer to the net, the goalkeeper continues to track the ball, attempting to punt it away from the net as it comes within range. Should the goalkeeper find itself out of the net, perhaps as a result of being pushed by an opposing player, it attempts to return to position as quickly as possible. Since a goalkeeper is deemed necessary regardless of perceived demand, much like an insect queen to a nest, the goalkeeper is considered independently of the specialization algorithm.

Center: The "center" hangs around the centre of the field, waiting until the ball comes within its sphere of influence. Then it heads straight for the ball. Once it is the closest one to the ball, the centre tries to get it as close as it can to the opponent's net, and potentially score.

If it is no longer the closest one on its team to the ball, it returns to its wait position in the centre of the field.

Attacker: The "attacker" waits around centre while the ball is in its own end, and attempts to score while the ball is in its opponent's end. There are two attackers, the north and the south. The only difference between the two is where on the field they wait, the north attacker stays just north of centre, while the south stays just south of centre. When attempting to score, the two work together to get the ball into the opponent's net. If they are unable to get it in by kicking, they work together, and use their brute strength to push any defender out of the way as they take the ball to the goal.

Defender: The "defender" waits near the goalkeeper for an opponent to enter the two-thirds of the field closest to its team's net. The defender then rushes to get into position between the ball and its net. If the defender is the closest one to the ball, and it has a clear shot, it kicks the ball towards the opponent's goal. If the ball travels into one of the close corners, it is the defender's job to brace against the goalkeeper so the ball can't be pushed into the net.

#### 7.17. GotoBall

Author: Tucker Blach, Georgia Tech University

Type: Homogenous

Strategy: This is the simplest possible soccer strategy.

GoToBall & Kick : All the players move behind the ball and try to kick it.

#### 7.13 .JunTeamHeteroG

Author: Unknown

Type: Heterogeneous

Strategy: It works by selecting the strategy according to its opponent teams.

It has no zones and has constant functions for the players. If selected strategy is Dteam strategy then it plays like Dteam Homogeneous strategy. It has the following players :

Goalie: If not close to the goal then act as Dribbler else if inside the goal zone play goal, else if in the defender zone play as the defender.

Defender: Defends the goal.

Center: Looks for the ball in the center of the field.

If closest to the ball, it drives it to the goal avoiding collision.

Tackler: The tackler tries to get between the opponent and the goal.

Dribbler: Tries to move the ball towards the goal.

Stopper: Tries to prevent the opponent goal keeper from doing the goal.

#### 7.14. Kechze

Author: KEnt, CHris, and ZELlyn's JavaSoccer Team (Based on BrianTeam.java by Brian McNamara)

Type : Heterogeneous (acc to quadrants)

Strategy :

There are no specific players. Any body can become anybody depending on their positions.

Kick : If the player is near the ball or goal kick it

GoToBall & Drive : the players try to get behind the ball, and try to turn it towards their goal

Goalie:

Has top and bottom part for goal post.

Get behind the ball, and try to turn it towards their goal

Stay beside the goal, in line with the ball, but not past the ends of the goal, when the ball is far away.

When it gets close, run to a spot near the ball, horizontally in line with it, but still between it and the goal

Blocks the opponent goal keeper in the left side.

#### 7.15. LoneForwardTeamHomoG

Author: Bryan Nagy

Type : Homogenous Team

Strategy: Unknown

Retrieve all the possible sensor information like vector to the ball, vector to our and their goal, a list of the positions of our teammates, etc. Compute some strategic places to go a point one robot radius behind the ball, a position between the ball and defended goal, a direction away from the closest teammate, etc. Find the closest teammate n find if we are the robot on our team closest to the ball.

If I am the closest member of my team to the ball, then it will be my job to put it in. If I am between the ball and their goal, that is, on the wrong side of the ball, I will first 'swirl' around the ball. If I am above the ball, I will swirl north else I will swirl south otherwise I will head for the kickspot behind the ball. If I am already at the kickspot, I will continue toward it. And if I can kick the ball and I am

close enough to score, I will kick the ball and continue toward it. But if I am not the closest member of my team to the ball, I will go defend the goal by driving towards the center of the goal. But if I am within a certain distance of the x value of the goal, and within a certain distance of my teammates, detect if we are being trapped and if so, circle around the trapper.

#### 7.16. MattiHetero

Author: Matthias Felber

Type : Heterogeneous Team

Strategy:

This is a team with four roles; "left wing," "right wing," "back wing," and "goalie". All behave the exact same way if they possess the ball; they attempt to get in close enough to get a clear shot at the opponent's net, and then they take it. The "goalie" follows the ball back and forth on the field, staying within a defensive radius of its home goal. Should the ball go further north or south of another player on the field, the goalie stays inside of that player's position, so as to avoid preventing them from becoming a left- or right-winger. If the ball is in close, the goalie stays close to the net; if the ball is out a fair distance, the goalie comes out, so as to avoid any goalie interference tactics presented by the opposing team. "Left wing" and "right wing" players stay behind the ball. Should it veer too much to one side, they serve to guide it back towards the centre of the field, and keep it from going out of play. "Back wing" players avoid their closest teammate, and attempt to stay behind the ball.

#### 7.17. PermHomoG

Author: Gita Sukthankar

Type : Homogenous team

Strategy:

Retrieve all the possible sensor information. Heading is given in radians. Compute useful features in the environment. Determine most dangerous opponent (opponent closest to ball). Compute a point one robot radius behind ball as best kickspot. Get setup for iterating through the permutation array. Only kick if behind the Ball.

#### 7.18. SchemaDemo

Author: Tucker Balch

Type: Homogenous Team

Strategy:

This is an example of a simple control system for soccer robots designed using Clay. The strategy is to get behind the ball, then move to it, while keeping away from teammates. One of the robots remains near the goal to act as a goalie. The Clay package works on a system of pre-defined schemas, weighted together to yield a decision result. There are two roles available to the SchemaDemo team, the "goalie" and "forward." The "goalie" has two

embedded steering schemas; get behind the ball, and move to the back of the field. The get behind the ball schema is primarily a swirl around the ball schema, weighted slight more than a move to halfway between the ball and our goal schema, and an avoid bumping into teammates schema. The move to back of the field schema is a move to goal schema, as well as to a lesser extent a move to ball schema. The “forward” also has two embedded steering schemas; get behind the ball, which is the same as the one for the goalie, and a move to ball schema. The move to ball schema is primarily a move to sweet spot near goal schema, with a slightly lesser degree of the avoid teammates schema. Whether or not the ball is kicked, in either of these roles, is determined by a kick ball schema built into Clay.

#### 7.19. *SchemaNewHetero*

Author: Tucker Balch

Type: Heterogeneous Team

Strategy:

This is an example of a simple control system for soccer robots designed using Clay. The strategy is to get behind the ball, then move to it, while keeping away from teammates. One of the robots remains near the goal to act as a goalie.

#### 7.20. *SibHeteroG*

Author: Mark Sibenac

Type: Heterogeneous Team

Strategy:

Sib's Heterogeneous Team derived a lot from the BasicTeam by Tucker Balch.

Major changes include a cherry picker and a goalie who hugs the goal. If the player is pushing the ball towards his own goal, he is forbidden to kick. Only the goalie is allowed to kick when the ball is close to our goal. The cherry picker waits in the middle of the field for the ball to appear after a reset. It is taking advantage of the system, and it works!

Retrieve all the possible sensor information like vector to the ball, vector to our and their goal, a list of the positions of our teammates, etc. Compute some strategic places to go a point one robot radius behind the ball, a position between the ball and defended goal, a direction away from the closest teammate, a point three robot radii behind the ball, the cherry picker's spot, a north spot, south spot etc. Find the closest teammate n find if we are the robot on our team closest to the ball.

The players go to one of the places depending on the number of the player

If the number is 1, then he moves towards the goalie position.

If the number is 1, then moves towards the midback position and performs actions such as Drive, KeepAway.

If the number is 2, then he moves to the northspot position.

If the number is 4, then he moves towards the southspot position.

If the number is 3, then he goes to a good kicking position and Drives the ball. The actions used here are GoToBall and Drive.

### 8. Tournament:

The teams were divided into 2 groups as follows:

Group A:

1. AIKHomoG
2. BasicTeam
3. BrianTeam
4. BriSpec
5. CDTeamHetero
6. CommTeam
7. DaveHeteroG
8. DoogHeteroG
9. DoogHomoG
10. DTeam

Group B:

11. FemmeBotsHeteroG
12. GoToBall
13. JunTeamHeteroG
14. Kechze
15. LoneForwardTeamHomoG
16. MattiHetero
17. PermHomoG
18. SchemaDemo
19. SchemaNewHetero
20. SibHeteroG

Many one minute games were conducted among the various team. The games are represented in the form of a chart. Only one score is shown for each game. Quarter-finals, semi-finals and finals were conducted in order to identify the top leading 4 teams.

From Group A, the first 5 teams were chosen and the best 2 from those were selected.

The teams selected were :

1. AIKHomoG
2. BasicTeam
3. BrianTeam
4. BriSpec
5. CDTeamHetero

From the above Fig. 1, it is evident that top 2 teams among those 5 teams were:

1. AIKHomoG
2. BriSpec

From Group A, the last 5 teams were chosen and the best 2 from those were selected.

The teams selected were :

1. CommTeam
2. DaveHeteroG
3. DoogHeteroG
4. DoogHomoG
5. Dteam

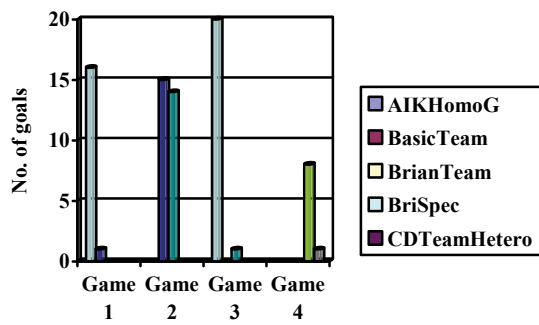


Fig. 1. AIKHomoG Vs BasicTeam Vs BrianTeam Vs BriSpec Vs CDTeamHetero

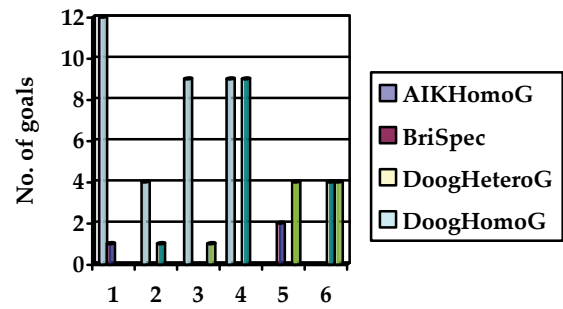


Fig. 3. AIKHomoG Vs BriSpec Vs DoogHeteroG Vs DoogHomoG

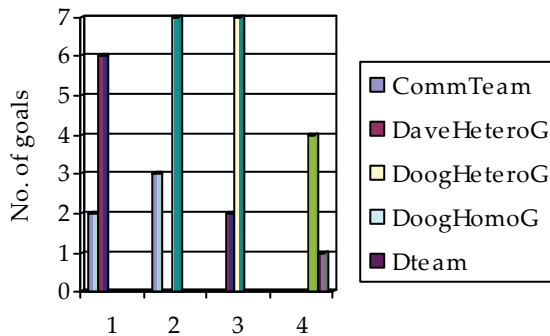


Fig. 2. CommTeam Vs DaveHeteroG Vs DoogHeteroG Vs DoogHomoG Vs Dteam

From Fig. 2, it is evident that top 2 teams among the 5 teams were:

1. DoogHomoG
2. DoogHeteroG

From the top four teams of Group A, three teams were selected as shown in Fig. 3.

Top leading 3 teams from Group A:

1. AIKHomoG
7. DoogHomoG
3. DoogHeteroG.

Group B:

From Group A, the first 5 teams were chosen and the best two teams were selected.

The teams selected were:

1. FemmeBotsHeteroG
2. GoToBall
3. JunTeamHeteroG
4. Kechze
5. LoneForwardTeamHomoG

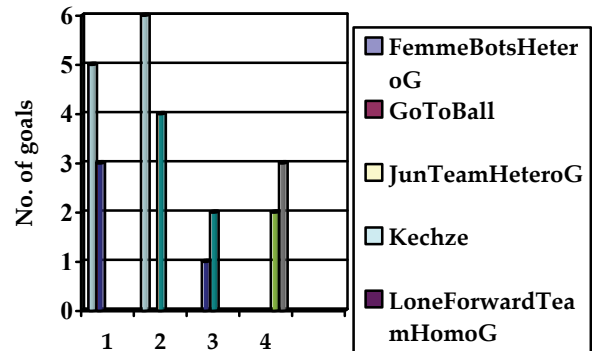


Fig. 4. FemmeBotsHeteroG Vs GoToBall Vs JunTeamHeteroG Vs Kechze Vs LoneForwardTeamHomoG

From Figure 4, it is evident that the top leading two teams are FemmeBotsHeteroG and JunTeamHeteroG.

From Group B, the last five teams were chosen and the best two teams were selected.

The teams selected were:

1. MattiHetero
2. PermHomoG
3. SchemaDemo
4. SchemaNewHetero
5. SibHeteroG

From Fig. 5, it is evident that the top leading two teams are:

1. MattiHetero
2. PermHomoG

Now again, the top 4 teams of Group B were selected and tournament was conducted as shown in the Fig. 6:

The teams were:

- FemmeBotsHeteroG



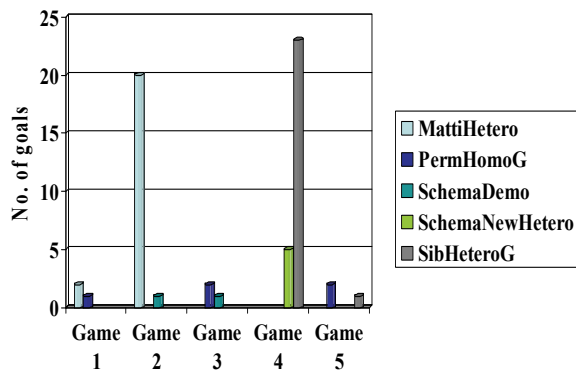


Fig. 5. MattiHetero Vs PermHomoG Vs SchemaDemo Vs SchemaNewHetero Vs SibHeteroG

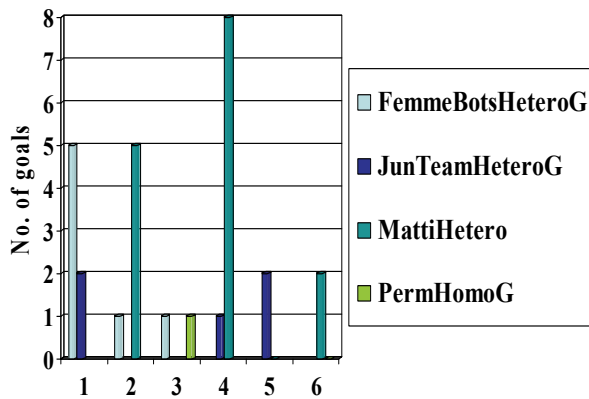


Fig. 6. FemmeBotsHeteroG Vs JunTeamHeteroG Vs MattiHetero Vs PermHomoG

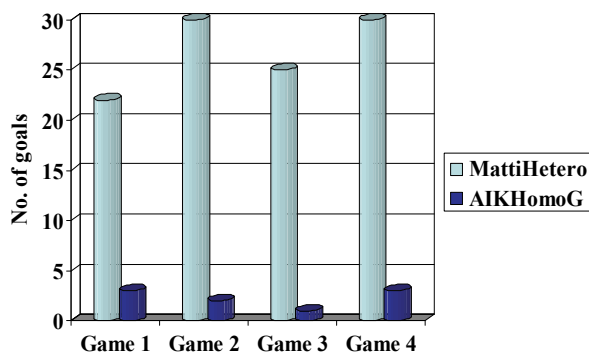


Fig. 7. MattiHetero Vs AIKHomoG

- JunTeamHeteroG
- PermHomoG
- MattiHetero

Therefore the 3 leading teams of Group B were found to be MattiHetero, FemmeBotsHeteroG, PermHomoG.

The top leading teams of Group A and Group B were found to be the following:

- AIKHomoG
- DoogHomoG
- DoogHeteroG
- MattiHetero
- FemmeBotsHeteroG
- PermHomoG

Finals:

Out of these, the final game was conducted between AIKHomoG and MattiHetero as shown in the figure below:

Therefore the leading 4 teams out of the 20 participating teams are:

1. MattiHetero
2. AIKHomoG
3. DoogHomoG
4. DoogHeteroG

## 9. Conclusion

The strategies and tactics of various teams in Soccerbots have been identified. Tournament has also been conducted to identify the top leading four teams in SoccerBots. It has also been identified that the best of all 20 teams in SoccerBots is MattiHetero. Hence it can be concluded that MattiHetero is the best team due to the heterogeneous nature of the team, strategy behind the division of work to the players and the dynamic strategies adopted by the players. New teams with improved strategies can further be developed to out beat the team namely MattiHetero. This paper will aid the researchers to develop the soccer team with best player strategies.

## 10. References

- Anon. (2003) The RoboCup Soccer Simulator, <http://sserver.sourceforge.net/>.
- Balch, T., Ram, A. (1998) Integrating Robotic Technologies with JavaBots, Working Notes of the AAAI 1998 Spring Symposium, Georgia Institute of Technology.
- David Andre, Emiel Corten, Klaus Dorer, Pascal Gugenberger, Marius Joldos, Johan Kummeneje, Paul Arthur Navratil, Itsuki Noda, Patrick Riley, Peter Stone, Romoichi Takahashi, and Travlex Yeap(1998). *Soccer server manual*, version 4.0. Technical Report RoboCup-1998-001, RoboCup.
- James Helferty (2003), "Applying Division of Labour Principles to Robot Soccer", Honours Project, Carleton University.
- Kitano H., "RoboCup: The Robot World Cup Initiative." In *proceedings of The First Int. Conf. on Autonomous Agent (Agents-97)*, Marina del Ray, The ACM Press, 1997.
- Mackworth, A.(1992), "On Seeing Robots", Vision Interface.

Minoru Asada, Hiroaki Kitano, Itsuki Noda, Manuela Veloso(1999) "RoboCup: Today and tomorrow—What we have learned" *Artificial Intelligence* 110 (1999) 193–214.

Official website of Robotic soccer: [www.robocup.org](http://www.robocup.org).

Official website of Teambots: [www.teambots.org](http://www.teambots.org).

#### **Acknowledgement:**

*Special acknowledgement to AICTE, as this paper is a part of the work done under the sanctioned ongoing AICTE Research Promotion scheme (RPS) project titled Bioinspired behaviours in Robotic Soccer (Ref. No. 8023/BOR/RPS-100/2006-07 dated 26.2.2007).*