

A GAME THEORETIC MODEL OF COOPERATION AND NON-COOPERATION FOR SOCCER PLAYING ROBOTS

M. Bader-El-Den, and J. Rüdiger Y. Kotb,
E. Badreddin,

*Automation Laboratory, University of Mannheim,
68131 Mannheim, Germany.*

Abstract: This paper proposes a model which combines cooperative and non-cooperative behaviors among autonomous mobile robots. This problem is well demonstrated by the Soccer playing robots, which consists of two sub-games, namely a non-cooperative game between the teams, and a cooperative game among the players of the same team. Game theory is used for modeling these games. The model consists of two layers, the first for the non-cooperative game which feeds its output to the cooperative game in the second layer. Fuzzy logic is used to evaluate the utility functional. The model is implemented using a soccer playing robot simulator. *Copyright*©2005 IFAC.

Keywords: Game Theory, Co-operation , Autonomous Mobile Robots, Fuzzification, Agents.

1. INTRODUCTION

Autonomous mobile robots can be employed in a large number of practical applications to relieve human beings from routine or risky tasks, increase the productivity, save costs or time and for educational purposes. Many of these applications require the robots to cooperate, e.g. transportation of bulky work-pieces in a factory or determining of large land-mine fields. Other applications require the robots to behave in a non-cooperative manner, e.g., protection of buildings against intruders. Such cooperative and non-cooperative behaviors are also found in nature and are essential for the survival of many forms of life. In (Skrzypczyk, 2003; Badreddin, 2000; Rüdiger, 2001), game theory was used to model coordination among agents. In (Skrzypczyk, 2003) motion planning for N mobile robots sharing the same workspace inside of which there are M moving obstacles based on game theory was made, the action selection process was modeled as an N -person non-zero

sum non-cooperative game in a normal form. In (Badreddin, 2000) the author proposed the RNBC (Recursive Nested Behavior Control structure) for soccer playing robots. The architecture is composed of five levels. These levels are the Execution, Behavioral, Tactical, Strategical and Task & command levels. In the tactical level the author designed a hybrid automaton to represent the continuous nature of the system as well as the discrete events which may happen in the surrounding environment. The Strategical level was modeled as a Non-cooperative zero-Sum Game matrix. In (Rüdiger, 2001) game theory is used to model two cooperating robots, their objective is to put the ball in the goal. The game was modeled in both extensive and strategic forms.

Skeletons and Dooley graphs are used to represent coordination between agents (Singh, 1998). In (Jung and Zelinsky, 2000) a heterogeneous cooperative multi-robot system was implemented. The system comprises two autonomous mobile robots

that perform cooperative cleaning. A layered solution for the problem was proposed. The architecture is built from 4 layers. The first layer, involves all the basic behavior required to clean the floor. The second layer gives one of the robots an awareness of the other. The third one introduces explicit communication and the fourth involves communication of litter locations between the two robots. In (Castelpietra, et al., 2000) a communication based coordination technique was proposed. In this technique, the coordination among players is built on top of a layered communication structure. The coordination is composed from Low level coordination layer which includes two steps, formation selection and role assignment, and high level coordination layer which amounts to the choice of the formation that is most suitable in the current state of the environment.

Hybrid systems are widely used in modeling multi-robot cooperative system as shown in (Chaimowicz, et al., 2003), the author proposes a methodology that uses hybrid systems to model multiple robots in the execution of cooperative tasks. In (Spaan and Groen, 2002) an approach for coordinating a team of soccer playing robots based on the idea of dynamically distributing rolls among the team members was presented and the notion of global team strategy was added. Utility function to measure how much a robot is suitable for a certain action was proposed.

The paper is organized as follows: section 2 defines the problem, section 3 demonstrates the proposed model, section 4 and 5 discuss the non-cooperative and cooperative layers of the proposed game model respectively. Section 6 describes the simulation of the proposed model.

2. PROBLEM STATEMENT

Given two teams of autonomous mobile robots. Each consists of $N \geq 2$ robots. The teams are to play soccer game under the following basic assumptions:

- The position of each robot is known to every other robot player.
- The ball position is known to all robots. Which is obtained from the robot sensory system.
- The robots of each team can exchange information over a secure communication channel (not accessible to the players of the other teams).

It is required to solve the soccer game problem consisting of the non-cooperative game between the teams and cooperative game among the players of each team.

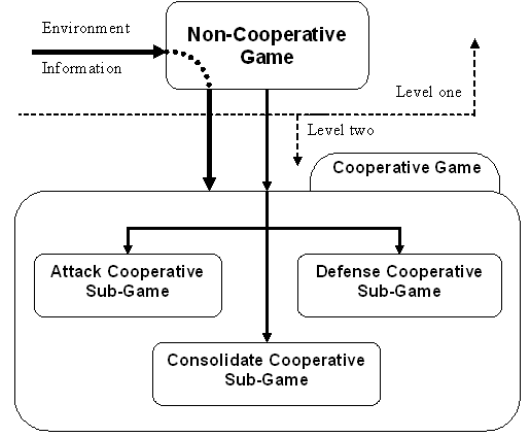


Fig. 1. multi-level approach for combining cooperative and non-cooperative games

3. PROPOSED MODEL

The soccer game is decomposed into an N -player, $N \geq 2$, cooperative sub-game among the players of the same team and a two-player non-cooperative game between both teams. The proposed decomposition is depicted in Figure 1. The non-cooperative game is solved first. Accordingly, the cooperative game is solved. Although this cascade structure neglects the fact that the solution of the co-operative game influences that of the non-cooperative game (feedback), it avoids instability issues and is faster to execute in real-time. Therefore, we opted to employ this -open-loop-structure despite its inferiority to the closed-loop.

4. NON-COOPERATIVE GAME

Non-cooperative games are concerned with the analysis of strategic choices. The strategic choices are those choices which depend on the reward of the combined actions of players in the game and not on the reward of a single action. The players are the two competing teams. The process of choosing a strategy depends on the team players position, opponent's players position as well as the ball position which are obtained from the sensory system. The sensed data are fed to a fuzzifier which is explained in section 4.2 actions rewards is calculated for the team and estimated for the opponent team. Strategies are formed by combining each action with all opponent's actions. Valid strategies rewards is based on the rewards of the combined actions, these rewards are then fed to the game theory engine in order to choose the best action for the team according to the estimated opponent's actions. Detailed explanation is provided in the following sections. The intentions and actions of the opposite team are unknown and consequently, the game is modeled in the extensive form with imperfect information

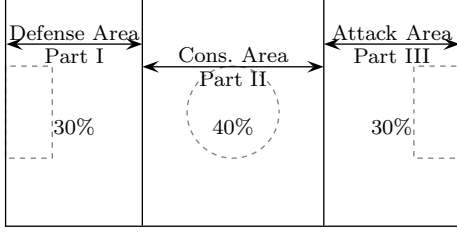


Fig. 2. multi level design for both cooperative and non-cooperative games

(Osborne and Rubinstein, 1997). To easily evaluate the robot position, the game environment is virtually divided into three parts as shown in Figure 2. For the left hand side team (team A), the left part is considered as part I (defense area), and considered as part III (attack area) for opponent team (team B). The lengths of the parts are 30%, 40% and 30% of the total length for parts (I, II, III) respectively

4.1 Actions

The set of actions available for each team are the following:

- Attack
- Defense
- Consolidate

The attack action opted when the team is in good situation to attack the opponent's goal, on the other hand, defense action means that the opponent's team has the chance to attack so the team should play in the defense mode. Consolidate action means that no team has the opportunity to attack, or no team controls the ball. These actions are considered as sub-games in our cooperative game modeling approach as shown and discussed later in details in section 5.

4.2 Action Reward Estimation

For the team to opt certain strategy, individual actions for each robot has to be evaluated and an action reward estimation for each one of them is calculated. To estimate the reward for the different actions, the attack utility AU_i , the defense utility DU_i , consolidate utility CU_i and ball status B_j are calculated as follow:

$$\begin{aligned}
 AU_i &= \sum_{i=1}^n FAtt_i(X_i) \\
 DU_i &= \sum_{i=1}^n FDef_i(X_i) \\
 CU_i &= \sum_{i=1}^n FCon_i(X_i)
 \end{aligned}$$

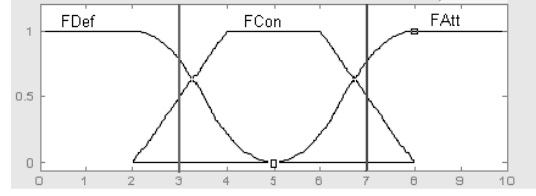


Fig. 3. shows the membership functions used in weighting the robot position

$$B_j = \begin{cases} 1 & \text{if the ball with team } j \\ 0 & \text{otherwise} \end{cases}$$

Where n is number of Robots in each team, i is player number $2 \leq i \leq n$, j is team number $1 \leq j \leq 2$ and x_i is the x position for robot i . $FAtt$, $FDef$ and $FCon$ are three Fuzzy membership functions used to weight the robot position as shown in Figure 3. These membership functions affect very much the behavior of the team. Moreover these membership functions can be easily changed to make the team behave in more attacking or defending manners. These functions were chosen in this work according to simulation experimental results. The horizontal scale shown in Figure 3 represents the distance of the player with respect to its goal. The vertical scale represents the membership values for this distance. It is clear from the figure that changing the membership function changes the behavior of the player and consequently of the team in terms of attack, defense or consolidate actions as mentioned above. The attack and defense membership functions are mathematically defined as follows:

$$\begin{aligned}
 FAtt(x) &= \begin{cases} 1 & 8 \leq x \leq 10 \\ 0.5 \left[1 + \sin\left((x - 6.5)\frac{\pi}{3}\right) \right] & 5 \leq x < 8 \\ 0 & \text{otherwise} \end{cases} \\
 FDef(x) &= \begin{cases} 1 & 0 \leq x < 2 \\ 0.5 \left[1 - \sin\left((x - 3.5)\frac{\pi}{3}\right) \right] & 2 \leq x \leq 5 \\ 0 & \text{otherwise} \end{cases}
 \end{aligned}$$

The consolidate membership function is defined as a trapezoidal function. For the team to attack it should control the ball, otherwise the outcome for this action is zero, because the team can not attack without the ball. This is described using the following equation:

$$Attack \text{ Action Reward} = AU_i \cdot B_1$$

For the team to defend, the other team should be in the ball position (controlling the ball), otherwise the reward from this action is zero and hence there is no need to defend. Defense action reward is calculated as follows:

$$Defense \text{ Action Reward} = DU_i \cdot B_2$$

Consolidation is the situation when most of the team players are in the consolidation area. The outcome for this action is calculated directly as follows:

$$Consolidate \text{ Action Reward} = CU_i$$

In this action the state of the ball is not taken into consideration, because this mode is designed so that the team members prepare themselves for either attack or defense cooperative sub-game.

4.3 Strategy reward

As mentioned above, combining actions together constitute different strategies. Weighting these strategies is essential to guide the process of decision making. The only two strategies which need to be weighted in this game are the [attack, consolidate] and [consolidate, attack] strategies. If one team is going to attack then the second will gain the half of the consolidate mode, and the attacking team will gain the double.

4.4 Game model

The game is modeled using the extensive form (Osborne and Rubinstein, 1997) as shown Figure 4, the dotted ellipse in Figure 4 shows that team B is not aware about what the action team A takes.

4.5 Game solution

The iterated dominance strategy is used to calculate the equilibrium for the non-cooperative game, if no equilibrium is found, Nash equilibrium is used (Osborne and Rubinstein, 1997).

5. COOPERATIVE GAME

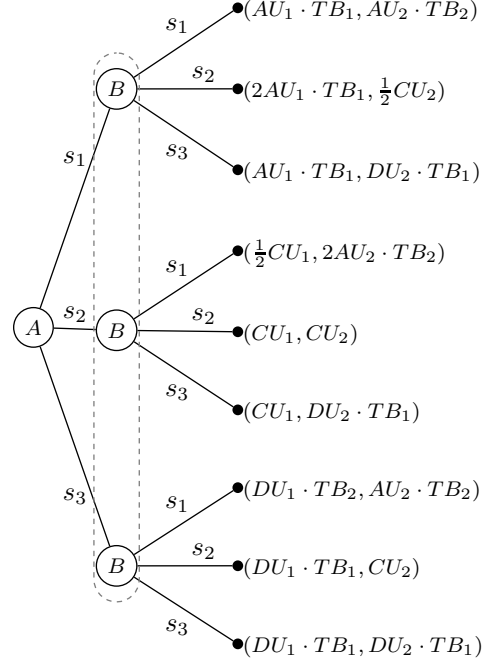
Cooperation is done among robots to achieve certain common goal. For the robots to cooperate, each one of them should add value to the desired goal solution. Cooperation is required when one robot can not achieve the goal when trying to achieve it without cooperation. Moreover, it is preferable when the cost of achieving the required goal while cooperating with other robots is less than the cost when trying to achieve it alone. The cooperative game is achieved among each team robots. As shown in Figure 1, the cooperative game is divided into three sub-games:

- Attack sub game
- Defense sub game
- consolidate sub game

5.1 Attack sub Game

5.1.1. Actions There are five actions in the attack cooperative sub-game.

- Kick (kick the ball into the opponent's goal)



s_1 : Attack s_2 : Consolidate s_3 : Def.
A: Team A B: Team B(opponent)

Fig. 4. Shows the extensive model for the non-cooperative game

- Dribble (move with the ball towards the opponent's goal)
- Pass (pass the ball to a partner robot).
- Wait (for pass from a partner robot).
- Move (move without the ball towards the opponent's goal).

5.1.2. Action Reward Estimation The action of kicking the ball to the goal depends on two main factors, the first is how far the robot is from the opponent's goal relative to the average distance the ball can reach when kicked. And the second is how large is the free angle the robot has to the opponent's goal relative to the minimal required for successful kicking. This is formalized in the following equation:

$$GK_i = [(\theta_{fi} - \theta_g)(K_{ball}/D_{gi})] Br_i$$

where, Br_i is 1 if the ball with robot i , and 0 otherwise, K_{ball} is the average distance the ball can move after it is kicked by any robot, and it depends on the kicker physical features, θ_{fi} is the maximum free angle which the robot can see the goal through any other robots. It is determined from the robot sensors i.e. camera, laser ranger, θ_g is a constant which is the minimum suitable angle for kicking the ball to the goal and D_{gi} is the distance between robot i and the opponent's goal. In the dribble action if the distance between the robot and the opponent's goal is relatively large to the distance the robot can kick and if the free angle the robot has to the opponent's goal relative

to the minimum required for a successful kicking is large then this will increase the reward gained from dribbling the ball. This is formalized in the following equation:

$$GD_i = [(\theta_{fi} - \theta_g)(D_{gi}/K_{ball})] \overline{Br_i}$$

For the pass action the gain from passing the ball to other partner will increase if the robot has small angle to the opponent's goal and the other robot is not far from the other cooperating player. This is formalized in the following equation:

$$GP_i = [(\theta_g - \theta_{fi})(K_{ball}/D_{ji})] \overline{Br_i}$$

where D_{ji} is distance between robot i and robot j from the same team. For the move action, the robot moves when it is in the same circumstances of dribbling the ball and it does not have the ball. This is formalized in the following equation:

$$GM_i = [(\theta_{fi} - \theta_g)(D_{gi}/K_{ball})] \overline{Br_i}$$

For the wait action, the robot waits when it is in the same circumstances of kicking the ball and it does not have the ball. This is formalized in the following equation:

$$GW_i = [(\theta_{fi} - \theta_g)(K_{ball}/D_{gi})] \overline{Br_i}$$

5.1.3. Strategy outcome The only two strategies need to be weighted in this game are [pass, wait] and [wait, pass]. If one robot is going to pass the ball then the second will gain more if it will wait for pass, the following constants are used in weighting the actions. $K_{wp} > 1$ constant to give more weight for (wait, pass) action, $K_{wm} < 1$ constant to give less weight for (wait, move) action,

5.1.4. Game Model In this part the cooperative attack sub-game is modeled in the extensive form, as shown in Figure 5. Player one is the robots controlling the ball, player two knows which action was taken by player one, so the cooperative attack sub game is considered as a game with perfect information (Osborne and Rubinstein, 1997).

5.1.5. Game solution The strategy is calculated using *follow the leader* where the leader is the robot controlling the ball.

5.2 Defense and Consolidate sub games

The models for defense and consolidate sub games are the same as the attack sub game, with different set of actions and different rewards calculation for each.

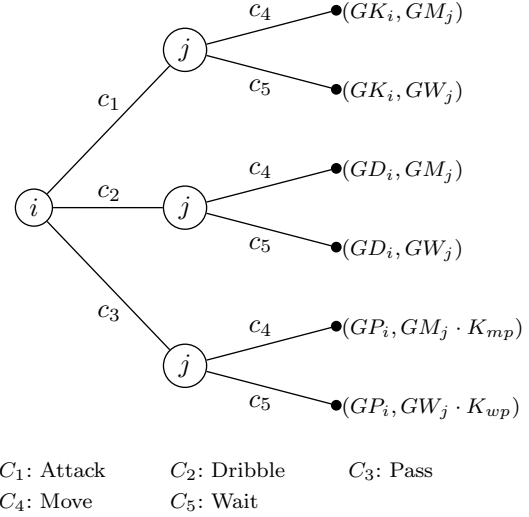


Fig. 5. Game tree (extensive form) for the attack cooperative sub-game

6. SIMULATION

The robot soccer simulator is fully described in (BaderElDen, 2003) which simulates a middle size robots team. Each simulated robot is modeled as an RNBC structure (Badreddin, 2000) and is used to simulate the proposed model. Two types of simulation have been done. In the first type, each of the cooperative and non-cooperative games have been simulated separately, Figure 6 shows snapshots of the simulator for two robots during the attack cooperative game trying to put the ball in the goal. External obstacles are added in the field. Snapshot (a) shows (robot 1) with small free angle to the goal, the action selected by this robot is to pass the ball to (robot 2). Both robots selected the (pass, wait) strategy. Snapshot (b) shows the simulator after (robot 2) has received the pass from (robot 1). The strategy selected in this situation is (move, dribble), so (robot 2) will dribble the ball until it reaches a position where the distance from the goal is short enough to kick the ball to the goal as shown in snapshot (c). Other simulation experiments for the non-cooperative, cooperative defense and cooperative consolidate have been done. In the second type of the simulation a complete game of two teams against each other has been simulated. Both teams are simulated using our proposed model. As shown in Figure 7, snapshot (a) shows both teams in the consolidate cooperative sub game and has the same strategy (go_to_ball, go_to_defense_area), trying to acquire the ball. Snapshot (b) shows the (team a) in the attack cooperative sum game and its selected strategy is (dribble, move). (team b) in the defines cooperative sub game and its selected strategy is (Go_between_ball_and_own_goal, Go_between_two_opponent_players).

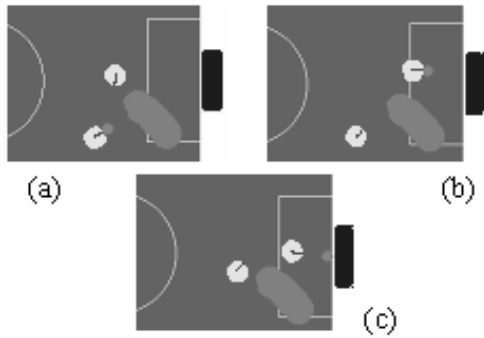


Fig. 6. Different snapshots of the simulator in the cooperative sub-game

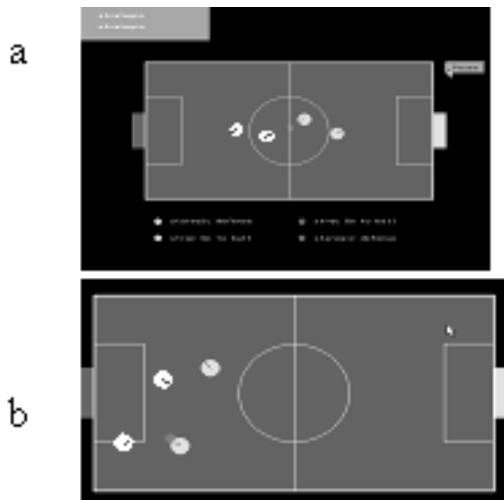


Fig. 7. Different snapshots of the simulator in complete game simulation

7. CONCLUSION

In this paper we modeled the soccer playing robots problem as a combination between a non-cooperative and cooperative games. The non-cooperative game between the two competing teams and the cooperative game is among the robots of the same team. Game theory is used to solve these games, in the non-cooperative, the game theory gave a good platform for each team to select its action according to what other team seems to be doing. And for cooperative game, the robots do not just evaluate their actions, but also evaluate the strategies they are going to use. The simulation shows different scenarios for the game and shows how game theory affected the robots choices in order to achieve the goal or sub-goal.

REFERENCES

BaderElDen, M.B. (2003). *Game Theoretic Schemes for the Strategical Cooperation among Soccer Playing Robots*, Master thesis. Department of Computer Engineering, Fac-

ulty of Engineering, Arab Academy for Science and Technology.
 Badreddin, E. (2000). A Hybrid Control Structure for a robot Soccer Player, In: *Proceedings of World Automation Congress*
 Castelpietra, C., L. Iocchi, D. Nardi and R. Rosati. (2000). Coordination in multi-agent autonomous cognitive robotic systems, In: *Proceedings of 2nd International Cognitive Robotics Workshop*.
 Chaimowicz, L., M.F.M Campos and V. Kumar. (2003) Hybrid Systems Modeling of Cooperative Robots, In: *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, pp. 4086-4091. Taipei, Taiwan.
 Jung, D. and A. Zelinsky. (2000) Grounded Symbolic Communication between Heterogeneous Cooperating Robots, In: *Autonomous Robots Journal*, special issue on Heterogeneous Multi-robot Systems, Kluwer Academic Publishers, Vol 8, No 3.
 Osborne. M.J. and O. Ariel Rubinstein. (1997) *A course in game Theory*, The MIT Press, Cambridge Massachusetts, London, England.
 Rüdiger, j. (2001) *Kooperative Spiele-Taktiken für Fußball spielende Roboter*, Diploma Thesis, Automation Laboratory, University of Mannheim, Germany. November 2001
 Singh, P. (1998) Developing Formal Specifications to Coordinate Heterogeneous Autonomous Agents. In: *3rd International Conference on Multiagent Systems (ICMAS)*, pages 261-268. IEEE Computer Society Press
 Skrzypczyk, K. (2003). Coordination in multi-agent system based on noncooperative games, In: *AI-METH 2003, Symposium on Methods of Artificial Intelligence*
 Spaan, M.T.J. and Groen, F.C.A. (2002) Team coordination among robotic soccer players, In: *RoboCup 2002*, pp. 409-416, Springer-Verlag, 2003.