The Benefits of Anticipation: An Experimental Study

Birger Johansson[1] and Christian Balkenius[1]

Lunds University Cognitive Science, Kungshuset Lundagård
222 22 Lund, Sweden
{birger.johansson, christian.balkenius}@lucs.lu.se

Abstract. This paper presents an experimental study using two robots. In the experiment, the robots navigated through an area with or without obstacles and had as their goal to shift places with each other. Four different approaches (random, reactive, planning, anticipation) were used during the experiment and the times to accomplish the task were compered. The results indicate that the ability to anticipate the behavior of the other robot could be to an advantage. However, the results also clearly show that anticipatory behavior is not always better than a purely reactive strategy.

1 Introduction

The ability to anticipate the behaviors of others is something we take more or less for granted and we often do not appreciate the complexity of this ability. When attempting to build robots with anticipatory abilities, it becomes clear that this is far from trivial. Not only does the robot need to control its own movement, it also needs to predict what other robots or possibly humans will do. Moreover, it needs to use the anticipated behaviors of others in a sensible way to change its own behavior.

Consider the following real life situation of the near future. You have sent your personal shopping robot to the “Autonomous supermarket” to get your favorite chocolate cake. To get the cake, located at the other end of shop, your robot cannot chose the straight path toward the cake because of the shelves and other obstacles, including all the other personal shopping robots in the store. Instead some alternative strategy must be used.

One possibility would be to move around randomly in the store until it finds the chocolate cake, but this would probably result in a long period of aimless wandering before it gets to its goal. This type of random behavior is very inefficient and is seldom used in robot navigation, although is rather common in robot exploration.

It is obvious that better methods can be used. Instead of moving at random, the robot may try to move in the direction of the goal. This is a reactive place approach method where the robot reacts to the position of the chocolate cake and selects actions accordingly [2]. The problem with this approach is that the robot cannot go straight to the goal because of the shelves and other robots in the store. It needs to apply an obstacle avoidance strategy when there is something in its way. For example, it may turn around and move in some other direction for a short while before turning toward the goal again. This type of reactive navigation has been widely used in robotics where the relation between the stimulus and response is often preprogrammed [6][12]. A number of rules are set up that must be fulfilled for an action to be executed. A problem with such a reactive approach to navigation is that the robot can easily get itself into situations where it becomes trapped.

Although the reactive strategy is more efficient than random movements it would be better to plan a path around the shelves based on knowledge of the layout of the store. This has traditionally been the most common way of dealing with robot navigation. This plan can use grids [20], potential fields [20][1][5], or some symbolic or geometric description of the environment. As long as the map of the shop is correct, the plan will also be correct and can be used to efficiently go to the cake.

Unfortunately, when the personal shopping robot reviews the map after a few seconds of moving according to the planned path, it realizes that the map is no longer accurate. The shelves are still where they are supposed to, but most of the other robots have moved and are not where the map indicates. As with the previous strategy, this makes it necessary to use some obstacle avoidance strategy to avoid colliding with the other robots which may limit the usefulness of the plan.

The solution to this problem is to include the movements of the other robots when the personal shopping robot makes its plan. This is however not trivial as it does not know where the other robots are heading. One reasonable assumption is that they will continue in the direction they have now, although this will only be true for a short while. Better predictions can be made if the robot knows the goals of the other robots. By anticipating the behaviors of others, it will be able to chose a better path and do not have to use the obstacle avoidance strategy as often. The better its ability to anticipate, the less it will need to use its alternative strategies.

Several different types of anticipatory behavior has been used in robotics and AI. First, it is possible to use an anticipatory mechanism to reduce the latency of a control system. For example, Behnke et al. [4] used neural network to reduce the control latency for the FU-fighter team in RoboCup. The control system had a delay of four frames (132 ms) and with a speed up to 2 m/s, this could result in an error between the actual position of the robot and the tracking of the robots of approximately 20 cm. By feeding a neural network with the position, orientation and motor commands from the last six frames to anticipate the current position, the influence of delay in the system was almost eliminated. A similar method has also been used to predict the location of a moving target for visual tracking [3].

A second type of anticipation concerns anticipation of the environment, for example the movement of other robots. Sharifi et al. [18] describe a system for the simulation league of RoboCup where the future state is used to anticipate which robot will posses the ball next, while Veloso et al. [21] anticipate the state of the whole team. This means that a seemingly passive agent is not passive at all. Instead it actively anticipate opportunities for collaboration.

The anticipation of robot movement can also be based on observation. For example, Stulp et al. [19] model the goal keeper in RoboCup to be able to anticipate its behavior. Ledezma et al. [14] used a similar method to model the behavior of the other players based on their observed input and outputs. Usually, some type of communication between agents are used in anticipation, either a complete knowledge of world or broadcasting of individual plans but there is also work on cooperation without sharing information between agents [17].

Human-robot interaction can also merit from using anticipatory behavior. Sabanovic et al. [16] used a stationary robotic receptionist that provided information to visitors and enhances interaction through story-telling to study human-robot interaction. In this study, the robot receptionist turns toward people passing by and tries to interact with them. To be able to interact in a efficient way, the robot receptionist anticipates the position of people passing by to produce timing and directions more suitable for interaction.

The importance of anticipation has also been studied in the domain of computer games [13]. In human activities, Saad [15] pointed out the close connection between driving and anticipation, even stating that "driving is anticipating".

Davidsson [8] used simulations to investiaget the benefits of anticipation. Two different types of experiments were conducted. The first investigated competition between agents and in the second experiment, the agents were cooperative. In the experiments, the task of the agents was to pick up targets in a two dimension grid world in a particular order. By using a linearly quasi-anticipatory agent architecture, one agent could realize that it would not reach the target before the other agent and would instead start to move toward the following target. In the second experiment, the agents cooperate which lead to a decreased total time for fetching all target objects.

Although simulations can be very valuable in testing different strategies, a simulation must necessarily include a perfect model of the simulated environment. It will thus always be possible to make perfect predictions in a simulation if this is desired. It is well known that this can easily lead to solutions that are not useful when applied to robots that have to operate in the real world [7].

To evaluate the benefit of anticipation in mobile robots, we tested a number of strategies in three different environments with two robots. We compared a random and a reactive strategy with control methods based on planing with or without anticipation of the behavior of the other robot. The goal of the experiments was to test under what conditions the ability to anticipate would help the robots in a simple task. In addition, we tested three different methods to use the anticipated behavior of the other robot.

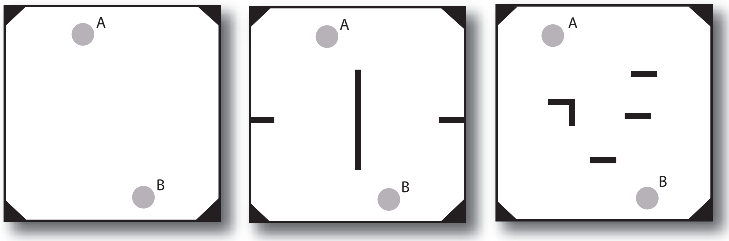

Fig. 1: *The three environments with different complexity. Left. The empty environment used in experiment 1. Middle. The environment with a central wall used in experiment 2. Right. The environment with random obstacles used in experiment 3. A and B: goal locations for the robots.*

2 Description of the Experimental System

2.1 Environment

The size of the experimental area was 2×2 m. Bricks marked with white color were placed in the area in two of the experiments to form obstacles (Fig. 1). Experiment 1 used the empty environment, Experiment 2 used an environment with walls and in Experiment 3, obstacles were placed at random in the environment.

2.2 Robots

The robots used were two modified BoeBots (Parallax Inc., Rocklin, California). These robots are approximately 14 cm long and use a differential steering. No sensors on the robots were used in these experiments. Instead, each robot was marked with two colors that could be detected by a camera mounted 3.5 m above the robot area. This camera transmitted images to a computer that calculated the position and orientation for the two robots four times per second. This computer also performed all the computations for the two robots and controlled the robots via wireless bluetooth communication. In addition, it stored tracking data and collected all statistics for the experiments.

2.3 Control Systems

The control systems of the robots were built using the Ikaros framework[1]. The interface components used included processing of the the video stream from the camera, color tracking to detect the position and orientation of the robots, and wireless communication. In addition, modules where built for reactive robot control, path planning, and anticipation.

[1] www.ikaros-project.org

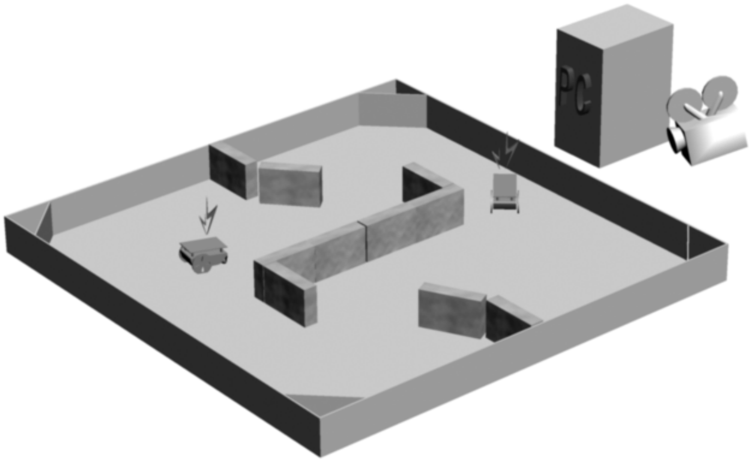

Fig. 2: *The computer is using the overhead camera to track the two robots and transmits motor commands via bluetooth.*

Random Control A random control system was the first tested in the experiments. This system simply transmits random motor commands to the robot until it has reached the goal. The robot is instructed to turn toward a random orientation and then travel in this direction until an obstacle activates an obstacle avoidance system, in which case a new random direction is set. This is repeated until both robots have reached their goals.

Reactive Approach The next control system performs reactive approach where the robot always tries to go directly toward the goal. The desired path is calculated as the straight line between the current location of the robot and the goal location. This strategy will obviously have problems when there are obstacles in the way and to handle this situation a reactive avoidance system was added.

Planning System The planning system is responsible for path finding within the environment. To accomplish this an A* based navigation algorithm is used [11]. This is a grid based navigation algorithm with full knowledge of the environment. It finds the shortest path to the goal by testing it in the grid-map. If it is unable to use the shortest path, the second shortest path is tested and so on, until a path has been found. Each robot uses the algorithm to find the best path through the robot area. The grid-map is divided into 32×32 elements with a status of either occupied or free. The planning system takes no account of where the other robots are located and only uses its own position, the desired position and the grid-map to find the path.

Anticipation System The anticipation system is similar to the planning system but also includes the movements of the other robots. If the other robots were stationary, the A* algorithm could register the other robots as obstacles. When the other robots are moving it becomes necessary to anticipate their position at each time-step in the future. To solve this, each robot has a model of the other robot. This model is built using each robots own planning system, for example, robot A assumes that robot B would use the path that robot A would have used if it were located at the position of robot B and heading for the goal of robot B. Before robot A tries to find its own path, it updates its model of the other robot and uses this to find the path for robot B by stepping forward in the planning and checking if there is any collision. If there is a collision, the robot chooses an alternative path and tests if this is a valid. This is repeated until a valid path is found. It should be stressed that the individual paths are not shared between the robots. Only the start and goal position is known by the other robot. With noise in the system this could lead to inaccurate models of the other robot and this could in turn lead to more activation of the reactive avoidance system. A similarly approach was presented by Guo [10].

An obvious problem arises with this approach. If both robots use the same method to find a valid path, it is possible for both robots to select the alternative path which will result in an collision. A way to avoid this problem is to assign a rank [9] to each robot where the robot with the highest rank always takes the shortest path. For example, let the robot with the longest distance to the goal have the higher rank and let the other robot replan its path around the more highly ranked robot. If the present robot has the lowest rank, we let A* see the other robots as a obstacle but only during that time step. This means that at just that time step there is an object at that position at some time steps later the obstacle has moved and the grid that was occupied in the first time step is free again. In the experiments, we tested three different ways to select the rank of each robot, (1) a fixed rank, (2) the robot closest to its goal would have the highest rank, and (3) the robot with the larger distance to its goal would receive the highest rank. Note that according to the last two strategies, the ranks of the robots may change when the robots move.

Reactive Avoidance A reactive avoidance system is placed on top of the other navigation systems and is activated if there is an obstacle too close to the robot. We divided the reactive area around the robot into 8 regions (Fig. 3). Three in front of the robot, one on each side of the robot and three behind the robot. The robot performs different types of avoidance behaviors depending on in which regions the obstacle was found. If an object is straight ahead, the robot turns on the spot until the obstacle has disappeared from the region and if an object is found to the left of the robot, it steers to the right to obtain a free path. Although the reactive avoidance system mainly helps the robot to reach its goal, it sometimes counteracts the control of the navigation system. For example, when the navigation system instructs the robot to turn right, the reactive avoidance system may detect an obstacle in that area and tell the robot to turn left instead.

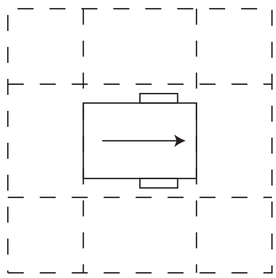

Fig. 3: *The robot with the reactive field around it. The reactive field divide the surroundings into eight regions and different avoidance behaviors are activated depending on the location of the obstacle.*

2.4 Experimental Procedure

The task for the robots was to switch places with each other. One robot started at position A and the other started at position B (Fig. 1). When the first robot had arrived at its goal position, it waited for the other robot to reach its goal. The goal locations were subsequently switched and same procedure was repeated. During this experiment, the time for each position switch was recorded together with the number of times the reactive avoidance system was used. Note that this was a cooperative task where it is the time for both robots to switch places that is recorded.

Six different strategies were tested: (RAND) random behavior, (APPR) reactive approach behavior, (PLAN) planning, (A-fixed) anticipation with fixed rank, (A-short) anticipation with higher rank for the robot closest to the goal, and (A-long) anticipation with lowest rank for the robot closet to the goal.

Each strategy was tested twice before the robots shifted to the next strategy. When all strategies had been tested two times, the procedure was repeated until in total 40 trials with each strategy had been run. In total, there were 240 trials in each experiment.

3 Results

The behaviors of the robots in the different conditions are illustrated in Fig. 4.

3.1 Experiment 1

The environment in experiment 1 did not contain any obstacles. As expected, the random behavior was significantly slower (389s) than all the other strategies (t-test, one-tailed, $p<0.0001$ for all test). The reactive approach behavior was significantly faster than all the other strategies (t-test, on-tailed, $p<0.01$ for all tests, Fig. 5 left). Two of the anticipatory strategies (A-fixed and A-long) were significantly faster than the planning strategy (t-test, one-tailed, $p<0.001$ and $p<0.05$, respectively).

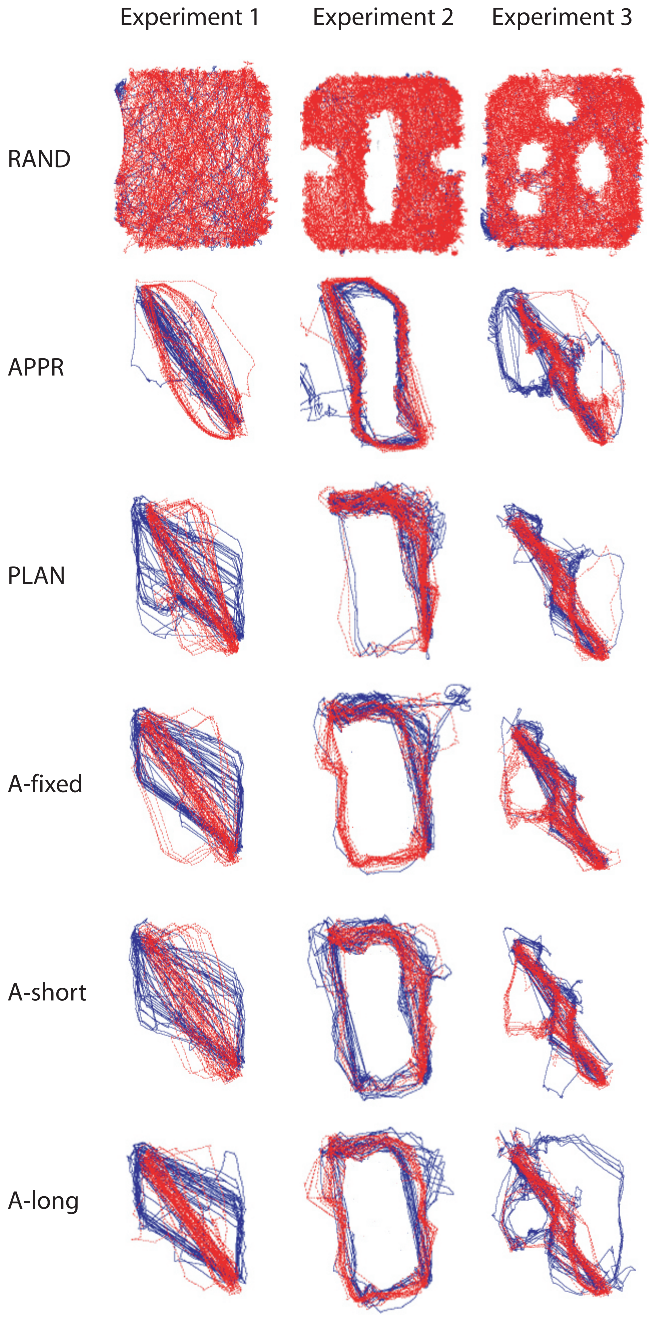

FIG. 4: *Illustration of all the movement of the robots. In the experiments with the random behavior, all the available area is covered. It is easy to discern the obstacle location in experiment 2 and 3. Using the reactive approach behavior, less of the area is covered. With this behavior, the required movement has been reduces in comparison to the random approach behavior. Using the planning behavior, the robots will often take the same path which will result in a possible collision and extensive use of the reactive avoidance system. This most clearly seen in experiment 2 where the robots often both select the top-right path. In the anticipation behaviors, the paths of the robots have more variation because the anticipation causes the robots to use different paths. Note that the robots balance the use of the two path between the two goal locations. In experiment 1, one robot uses the diagonal path while the other moves to the left or right. The same pattern can be seen in experiment 2 and 3, most clearly in A-long in experiment 3.*

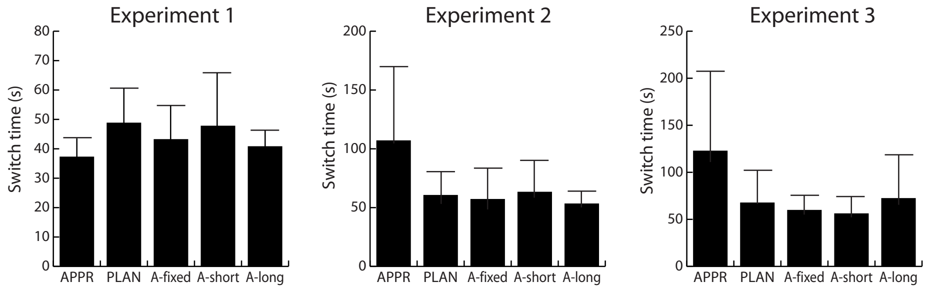

FIG. 5: *Left. Switch time for the different strategies in the empty environment in experiment 1. In the empty environment, the reactive approach behavior (APPR) performed best. The error bars show the standard deviation. Middle. Switch time for the different strategies in the environment with walls in experiment 2. The anticipatory strategy where the robot with longest distance to its goal had highest rank (A-long) was most efficient. Right. Switch time for the different strategies in the environment with random obstacles used in experiment 3. Strategy A-short was fastest in this environment.*

3.2 Experiment 2

Experiment 2 used an environment with walls. Again, the random behavior was significantly slower than all the other strategies (774s, t-test, one-tailed, $p<0.0001$). The reactive approach strategy was significantly slower than the planning and anticipatory strategies (t-tests, one-tailed, $p<0.0001$ for all tests, Fig. 5 middle). However, of the different anticipatory strategies, only A-long was significantly faster than the planning strategy (t-test, one-tailed, $p<0.05$).

3.3 Experiment 3

The environment in the last experiment contained randomly placed obstacles. Again, the random strategy was significantly slower than all the other strategies (817s, t-test, one-tailed, $p<0.0001$). The reactive approach strategy was again significantly slower than the planning and anticipatory strategies (t-test, one-tailed, $p<0.001$ for all tests, Fig. 5 right). Of the different anticipatory strategies, only A-short was significantly faster than the planning strategy (t-test, one-tailed, $p<0.05$).

4 Discussion

We have tested a number of behavioral strategies in robots in three different environment with varying complexity to investigate the usefulness of anticipatory abilities.

As expected, a reactive approach behavior that always tries to move in the direction of the goal performs well in an empty environment but is much worse when there are obstacles in the way. Also, all strategies were much better than a random behavior.

In some cases, some of the anticipatory strategies were more efficient than the planning strategy, but the merit of anticipation clearly depended on how anticipation was used and in what environment.

In the first experiment, the anticipatory strategies with fixed rank or with higher rank for the robot closest to the goal were significantly faster than the planning strategy. However, in experiment 2, it was instead the anticipatory strategy where the robot with the largest distance had highest rank that was significantly faster than the planning strategy. Finally, in the last experiment, it was only the anticipatory strategy where the robot closest to the goal that had highest rank that was significantly faster than the planning strategy.

The reason why anticipation is not always faster than planning without anticipation in these experiments is probably that there is too much noise in the system which interferes with the anticipatory behaviors. With anticipation, the robot will often take a longer path than with only planning and if something goes wrong during the avoidance of the anticipated obstacles, the robot will loose much time.

Had the robots been more accurate when they attempt to follow their planned paths, we expect that anticipation would have been better in most cases. It can clearly be seen that under optimal conditions, anticipatory behavior is very efficient but it is very sensitive to different disturbances. In the future, we want to increase the exactness of the control system to allow more precise movement control of the robots. This will probably lead to a greater advantage for the anticipatory strategies.

In the experiments, the robots had total knowledge of the environment as well as the position and goal of the other robot. It only had to anticipate the movement of the other robot. In such a situation it may be more advantageous to make a collective plan for all the robots. What we are aiming at in the future, however, is the situation where the robots do not have full access to the environment. In this case, the robots must explore the environment to learn about different paths and the positions of the other robot. As they will not know the goal of the other robot, it must be inferred from its observed movements. In this case, we will be able to explore different learning methods and different strategies for observing the behavior of the other robot.

In conclusion, we have presented experimental results with two robots in different environment that show that the ability to anticipate the behavior of the other robot can make the behavior of the robots more efficient. However, this is highly dependent on the complexity of the environment and the accuracy of the control of the robots.

5 Acknowledgement

We would like to thank Anna Balkenius for helpful comments on the manuscript and Henrik Johansson for the 3D drawing support. This work was supported by the EU project MindRaces, FP6-511931.

References

1. R. Arkin, "Motor schema-based mobile robot navigation." *International Journal of Robotics Research*, vol. 8, no. 4, pp. 92–112, August 1989.
2. C. Balkenius, *Natural Intelligence in Artificial Creatures.* Lund University Cognitive Studies, 1995, no. 37.
3. C. Balkenius, and B. Johansson, "Event Prediction and Object Motion Estimation in the Development of Visual Attention," In Berthouze, L., Kaplan, F., Kozima, H., Yano, H., Konczak, J., Metta, G., Nadel, J., Sandini, G., Stojanov, G. and Balkenius, C. Eds. *Proceedings of the Fifth International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems*, 2005, pp. 17–22.
4. S. Behnke, A. Egorova, A. Gloye, R. Rojas, and M. Simon, "Predicting away robot control latency." *RoboCup*, 2003, pp. 712–719.
5. R. Brooks, "A robust layered control system for a mobile robot," *IEEE Journal of Robotics and Automation*, vol. RA-2, no. 1, pp. 14–23, 1986.
6. R. A. Brooks, "Intelligence Without Reason," *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI-91)*, J. Myopoulos and R. Reiter, Eds. Sydney, Australia: Morgan Kaufmann publishers Inc.: San Mateo, CA, USA, 1991, pp. 569–595.
7. R. A. Brooks, *Cambrian Intelligence: the early history of the new AI.* MIT Press. 1999
8. P. Davidsson, "Learning by linear anticipation in multi-agent systems," *Distributed Artificial Intelligence Meets Machine Learning*, vol. 1221, pp. 62–72, 1996.
9. M. Erdmann and T. Lozano-Perez, "On multiple moving objects," *Algorithmica*, vol. 2, pp. 477–521, April 1987.
10. Y. Guo and L. Parker, "A distributed and optimal motion planning approach for multiple mobile robots," *Proceedings IEEE International Conference on Robotics and Automation*, 2002.
11. N. J. R. B. Hart, P. E. Nilsson, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on Systems Science and Cybernetics SSC4*, vol. 2, pp. 100–107, 1968.
12. J. U. Korein and J. Ish-Shalom, "Robotics," *IBM Systems Journal*, vol. 1, pp. 96–106, 1987.
13. J. E. Laird, "It knows what you're going to do: Adding anticipation to a quakebot," *Proceedings of the Fifth International Conference on Autonomous Agents.* Canada: ACM Press, May 2001, pp. 385–392.
14. A. Ledezma, R. Aler, A. Sanchs, D. Borrajo, "Predicting Opponent Actions by Observation," *RobuCup 2004*, pp. 286-296
15. F. Saad, "In-depth analysis of interactions between drivers and the road environment: contribution of on-board observation and subsequent verbal report," *Proceedings of the 4th Workshop of international cooperation on theories and concepts in traffic safety*, University of Lund, 1992.
16. M. M. S. R. Sabanovic, S., "Robots in the wild: Observing human-robot social interaction outside the lab." *International Workshop on Advanced Motion Control*, Istanbul Turkey, March 2006.
17. S. Sen, M. Sekaran, and J. Hale, "Learning to coordinate without sharing information," *Proceedings of the Twelfth National Conference on Artificial Intelligence*, 1997, pp. 509–514, (Reprinted from *Proceedings of the NationalConference on Artificial Intelligence, 1994*).

18. H. A. A. Sharifi, M. Mousavian, “Predicting the future state of the robocup simulation environment: heuristic and neural networks approaches,” *Systems, Man and Cybernetics*, vol. 1, pp. 32–27, 2003.
19. M. B. M. Stulp, F. Isik, “Implicit coordination in robotic teams using learned prediction models,” *IEEE International Conference on Robotics and Automation*, 2006.
20. S. Thrun, J.-S. Gutmann, D. Fox, W. Burgard, and B. Kuipers, “Integrating topological and metric maps for mobile robot navigation: A statistical approach,” *AAAI/IAAI*, 1998, pp. 989–995.
21. M. Veloso, P. Stone, and M. Bowling, “Anticipation as a key for collaboration in a team of agents: A case study in robotic soccer,” *Proceedings of SPIE Sensor Fusion and Decentralized Control in Robotic Systems II*, P. S. Schenker and G. T. McKee, Eds., vol. 3839, Bellingham, September 1999, pp. 134–143.