AIRG Sibiu

Ciprian Candea and Marius Staicu

Artificial Intelligence Research Group Sibiu
St. Reconstructiei Nr. 2A, 2400, Sibiu, Romania

1 Introduction

In this year the AIRG group prepares a new team for the RoboCup competitions based on new ideas and concepts. Based on the MAS(Multi Agent Systems) architecture implemented by us in 1999 for the Sibiu Team that took part at RoboCup 1999 competition hold in Stockholm, the paper describes a generic architecture for a RoboCup team with four holon types situated on three levels. This architecure tries to synergistically combine the characteristics of both domains: MAS and HMS(Holonic Manufacturing Systems).

2 Special Team Features

Our main research focus in this year was to promote a new approach able to increases the flexibility of decisional system. Accordingly with this new model we modified the decisional architecture used by us in 1999 keeping at the same time the functionality. First of all, our idea has many connections with the PROSA architecture developed at PMA/KLUeuven as a reference model for Holonic Manufacturing Systems. On the other hand, some concepts regarding the decision control and structure in team-based settings has been inspired from the studies that are usually approached in the field of GDSS (Group Decision Support Systems) research. The acronym PROSA came from Product-Resource-Order-Staff Architecture, the holon types used. The resource holon contains a physical part namely a production resource of the manufacturing system, and an information processing part that controls the resource. The product holon holds the process and production knowledge to assure the correct making of the product with sufficient quality. The order holon represents a task in the manufacturing system. It is responsible for performing the assigned work correctly and on time. The staff holon is implemented in the idea to assist the rest of three holons in performing their work. A holon is the fundamental part of any holarchy, and it can be referred to as an agent who has outspoken cooperatively and autonomy characteristics.

3 World Model

The World Model is one of the most important parts of any Robosoccer agent. The entire agent behavior is build around it and that is why the model must

P. Stone, T. Balch, and G. Kraetzschmar (Eds.): RoboCup 2000, LNAI 2019, pp. 409-412, 2001.

be as close as possible to the reality. The world is represented using an object-oriented approach. Each thing on the field is an object. The hierarchy is modeled subsequent to the object hierarchy of the robocup simulator. The data in the objects are the position, velocity, acceleration and other characteristics (shirt number, team side for players). The interesting part of our world model is that we have two copies of it. One is used by the input thread (the one that receives the simulator messages), and the other by the other holons. The idea behind this architecture is that while some holons access the world model data we cannot update it (if a SEE message is received). Using two copies we can process a SEE message as soon as possible. After the message is processed, on the next cycle, the other copy is updated and the holons will have the correct information. If after a cycle there is no see message, the world model will be updated using the current object position, their velocity and acceleration (if available) and the agent position, velocity and acceleration. Using this method the world model is kept as close to the reality as possible and it's updated without interfering with the agent operations. Even if the agent misses to receive messages for one or two cycles (because of a lengthy operation or bad skills), with this architecture the model will still be updated in background and will be available at the end of the operation. If some information for a player isn't know, depending on what information is missing (shirt number, team) the player will be used as a teammate or as an obstacle. The world model keeps a history for the two last see messages. Using this history it will update the data even for objects that weren't seen in the last 5 - 10 cycles. After 10 cycles, if the object isn't seen its position is marked as unknown. For the position of the objects that were computed using the history in the model there is a confidence factor that decrease in time. When the object is seen this factor is 95 and will decrease until 0 after 10 cycles. Further at each 100 cycles (mean value), each agent try to send to the teammates the data for the closest objects they see. This way the agents can update/improve their model.

4 Coach

Last year we used an online coach for next things. First of all we decided to implement inside an algorithm for identification in the opponent team of the correspondent player from our team i.e. if our defender from left has number 3 what is the corresponding opponent? This type of information is necessary because our neural network was trained using the correspondent opponent position and number (practically the opponent role). In the same time when it is possible our coach send to all team some information about all players position and the ball. All this function was kept in the current version of coach and now with our new architecture we have possibility to extend with other capabilities like player change or assign to a player a temporary role, really easy. To determine the pairs (our player - opponent player) we use a ray trace algorithm.

5 Skills

Because we try to propose a new architecture we implement our skills using mathematical analysis. A part from skills idea used in our implementation was taken from Magma Freiburg code. The positioning problems it is one of the most important problems in life generally. For RoboCup teams we resolved this task using a decision tree. In this tree we have a lot of possibilities and each possibilities is very easy defined based on predicates. The number of predicates is not so big (i.e. we use 15 predicates) and finally we can create in this way a simple methods to move our players on field. To pass we use a special predicate who try to predict based on the current field situation what teammate is in the best position to receive a pass. We used a probability algorithm for each teammate and this probability is calculated based on current position and how many opponent players are in area and what intention they have. The rest of skills implemented by us I think that are classical in this moment.

6 Strategy

As we already said for strategy we used the method described in [1] where each team member tries to learn from the corresponding human(or opponent team) player in a real game. Following a unified approach, strategic and tactical behavior is learned by training a feed-forward neural network (ANN) with a modified back-propagation algorithm. We implemented a feed-forward neural network made up by three layers. The ANN has 20 inputs (two for each of the 19 players and the ball - because the goalkeepers have no role to play in this simplified strategy). The player inputs are composed of two bits enabling to represent four different vectors describing the location of the players or ball, related to each agent (that means, four distinct "tactical" components of the overall picture). In this way our players can learn a strategy at the team level practically we clone the behavior from other opponent players. For example if we want to play with a CMU base strategy we have to take a game with CMU vs any other team and to train off-line our ANN. After the training period we have 10 strategy behaviors (one for each agent) based on CMU strategy. More we observe that in same situations our player applay some tactics unimplemented by us in the corresponding decision tree. The ideal model for our approach is to change the tactics for each opponent team in part but this think it is not possible in competition time, is necessary some time to train the ANN.

7 Team Development

Our team is part from a larger private research group situated in Sibiu, Romania. Last year we develop our team like a student program where we study some interesting idea about behavioral learning and we used the skills from HCII and the implementation was made in java. Because we had an interesting and new idea about how it is possible to put together two different domains like MAS and HMS we decide to participate again. This year we rebuild and rewrite all team because we proposed a new architecture and it was necessary to propose also a new world model. We use in our developing process some tools created by us like debugger or decision tree maker. The most important tool is the graphic debugger. With this debugger we can see everything what an agent do and more what he will do. Of course we can observe and analyze all logs off line and see why in a very clear situation he act maybe stupid. In this way the necessary time to develop our team was acceptable (2 months).

Team Leader: Ciprian Candea
Team Members:

Marius Staicu
- Artificial Intelligence Research Group Sibiu
- Romania
- Computer Science Enginer

Gabriela Iozon Simona
- Artificial Intelligence Research Group Sibiu
- Romania
- Computer Science and Matemathics Teacher

Web page http://airg.verena.ro

8 Conclusion

The results obtained in this year suggest that we are in a promising starting point for future work. This validates the approach of adding functionality stepwise (first of all flexibility) i.e. improving the 1999 architecture. Following the tests we intend to introduce a new level between the coach and the players - the team level. With this new level, we think that the architecture flexibility will increase substantially. For the game strategy the order holon is still insufficiently adapted. Our intention is to improve this team until next year and off course to win. We wold like to particularly mention the collaborators of the AIRG, namely Adrian Caraiman, Constantin Zamfirescu, Prof. Boldur Barbat and Prof. Daniel Volovici.

9 Bibliography

[1]Candea,C and Oancea,M and Volovici,D (1999). Emulating real soccer, in Proceedings of the International Conference Beyound 2000 Sibiu, pp. 35-38

[2] Giebels,M and Kals,H and Zijm,H (1999) Building Holarchies for Concurrent Manufacturing Planning and Control. Proceedings of the second International Workshop on Intelligent Manufacturing Systems, Leuven, Belgium, pp.49-56

[3] Hino,R. and Moriwaki,T (1999) Decentralized Scheduling in Holonic Manufacturing System. Proceedings of the second International Workshop on intelligent Manufacturing Systems, Leuven, Belgium, pp. 41-47

[4] Kitano,H and Asada,M and Kuniyoshi,Y and Noda,I and Osaw,E. Robocup: The Robot World Cup Initiative