Brainstormers 3D – Team Description 2006

Marc Halbrügge

AG Neuroinformatik, Universität Osnabrück, 49069 Osnabrück, Germany

Abstract. The main interest behind the Brainstormers' effort in the robocup soccer domain is to develop and to apply machine learning techniques in complex domains. Especially, we are interested in Reinforcement Learning methods, where the training signal is only given in terms of success or failure. Our final goal is a learning system, where we only plug in 'win the match' – and our agents learn to generate the appropriate behaviour. This paper focuses on the application of Reinforcement Learning in the field of soccer simulation.

1 Design Principles

The 3D team is based on the following principles:

- Two main modules: world module and decision making module
- Linear and nonlinear regression models are used to approximate future world states
- Input to the decision module is the approximate, complete world state
- The soccer environment is modelled as an Markovian Decision Process (MDP)
- Decision making is organized in complex and less complex behaviours
- A steadily growing part of the behaviours is learned by Reinforcement Learning methods
- Modern AI methods are applied wherever possible and useful

The different skills are layered corresponding to their complexity as shown in Figure 1. The idea behind this architecture is to divide the team tactic into subtasks that can be solved more easily (divide and conquer).

To avoid complicated nested `switch` or `if` statements, we created different classes for each type of player (the top box in Figure 1). The different players use the same medium level skills like GoalShot or Intersect. These depend on the low level skills like GoToPos.

Whenever we can come up with a simple analytical algorithm, we use it. Reinforcement Learning methods are used in the remaining cases.

An example for the former case is the GoToPos behavior: The agent drives at maximum speed towards the target position until the minimum break distance is reached. Then the drive vector is inverted.

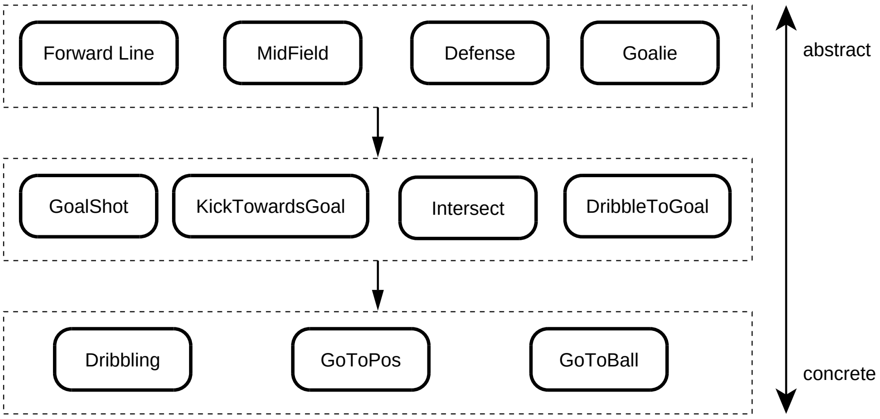


Fig. 1. The Behavior Architecture

2 Reinforcement Learning of Team Strategies

The more complex decisions, for example when to stop the positioning to perform the goalshot, are hard to program 'by hand'. Machine Learning provides algorithms that find good solutions to these problems.

The idea behind our approach is to find a value function $V(s, u)$ that describes how desirable a pair of situation and action is. V is a mapping from a state s and an action u to a value in $[0, 1]$. A value close to 1 indicates success, a value close to 0 failure.

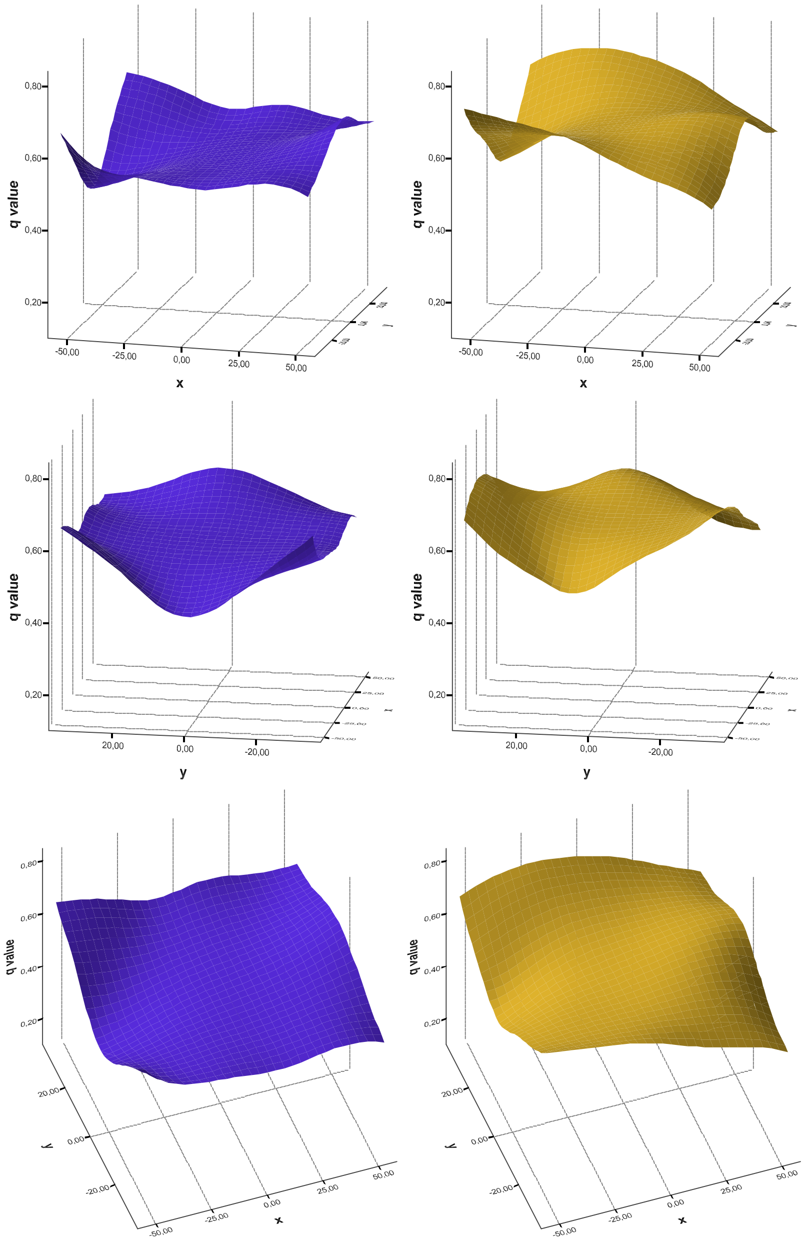
The value function is estimated using the $Q(\lambda)$ algorithm [1, chapter 7] with λ close to 1. Therefore it is referred to as q-function in the following. The state s consists (at least) of the position and velocity of the agent, the ball and the nearest opponent.

Depending on the complexity of the problem, we use neural networks or multivariate regression to approximate the value functions. Regression needs much less computation time but tends to be too biased for nonlinear problems. We use RPROP [2] for the training of neural networks as it is a fast and robust algorithm.

The strategy for the behavior of the agents is derived by evaluating the value function for the current situation s and all actions u. Then the maximum valued action is carried out.

We perform the training of the agents in epochs, that means that the strategies for all agents are fixed while the data for the estimation of the next set of strategies is gathered.

Figure 2 on page 3 shows the estimated q-function for the skills KickTowards-Goal (yellow) and DribbleToGoal (blue) depending on the position of the agent. The graphs are the result of one learning epoch (about 1.5 million single data points).



KickTowardsGoal

DribbleToGoal

Fig. 2. Average q-values for different skills depending on the position of the agent on the soccer field: Side, rear and top view. Position is displayed in meters with x along the sidelines and y parallel to the goals. The kickoff-point is centered at $(0, 0)$. See the text for further explanations.

The x-dimension is parallel to the sidelines of the soccer field, the goals are parallel to the y-dimension. The positions are displayed in meters, the soccer field's kick-off point is at $(0, 0)$, the opponent's goal at around $(50, 0)$. As one can see, the highest q-values are achieved when the agent is in front of the opponent's goal.

The comparison of the two functions reveals further interesting information: The dribbling works especially well on the flanks, but the q-value drops down heavily when the agent gets too near to one of the corner flags.

In such a situation, the evaluation of the value function will tell the agent to stop dribbling and kick the ball towards the opponent's goal. This simple example illustrates that complex decision problems can be solved elegantly using Reinforcement Learning techniques.

3 Future plans

Apart from the application of Reinforcement Learning to the behavior of the agents, two main issues will determine our work in this year.

First of all the success of the 2005 team was founded largely on the reliability of its self-localization [3]. As the 2006 server will abandon the Spades simulation environment [4], the agents will look the same, but they will not function as they used to do. Therefore we will have to retrain the complete world model. This will be a big part of our preparation for the 2006 competitions.

In the second place we are willing to make the strategies of our agents less stationary than they are now. We already use online learning to adapt to the current opponent, but only on a very limited scale. Especially the adoption of the offside rule in the new server forms an interesting new field for the application of non stationary strategies.

References

1. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press (1998)
2. Riedmiller, M., Braun, H.: RPROP: A fast and robust backpropagation learning strategy. In Jabri, M., ed.: Fourth Australian Conference on Neural Networks, Melbourne (1993) 169 – 172
3. Halbrügge, M., Voigtländer, A.: Brainstormers 3d – team description 2005. In Noda, I., Jacoff, A., Bredenfeld, A., Takahashi, Y., eds.: RoboCup 2005: Robot Soccer World Cup IX. Springer, Berlin (2005)
4. Riley, P.F., Riley, G.F.: Spades – a distributed agent simulation environment with software-in-the-loop execution. In Chick, S., Sanchez, P.J., Ferring, D., Morrice, D.J., eds.: Proceedings of the 2003 Winter Simulation Conference. (2003)