On-line robot learning using the interval estimation algorithm

Tijn van der Zant TIJN@AI.RUG.NL
Rijksuniversiteit Groningen, Artificial Intelligence

Marco Wiering MARCO@CS.UU.NL
Utrecht University, Intelligent Systems Group

Jürge van Eijck JURGE.VAN.EIJCK@PHILIPS.COM
Philips RoboCup Team, Mechatronics Research, Eindhoven

1. Introduction

To accomplish a certain goal with a robot many different solutions exist. Usually only one is implemented in a behavior-based architecture (Brooks, 1986; Arkin, 1998), but is it the best one? Since the amount of time for experimenting with robots is limited, especially during a RoboCup game, a strategy for choosing the best behavior from a set of human-programmed behaviors is desirable.

A lot of reinforcement learning algorithms are based on a full state space to learn from. In the RoboCup mid-size league this is impossible to do during the real games, due to the immense state space. This paper suggests a way to reduce the state space significantly by selecting among behaviors that are only triggered by few states. In fact to make the robot keeper learn very fast to select its best behavior with the purpose to defend the goal, we only used a single state in our experiments. For a behavior with a certain goal several implementations are made. From this behavior set the interval estimation algorithm chooses the behavior that has the highest probability to actually achieve the highest possible performance. This means fast learning, although the reduced state space also means that some solutions cannot be found.

The Interval Estimation algorithm is tested on the robot goalkeeper from the Philips mid-size league team. The results show fast convergence to the best performing behaviors.

2. Interval Estimation learning on the goal keeper

The algorithm in this article learns which behavior is the best one in a behavior-based architecture and can be extended to take state information into account. In the case of the goal keeper there is only one state s in the simplest case, which is to defend the goal (unconditioned on more game specific information). For this state s the best action a^* has to be chosen. By trying the behavior, the environment gives feedback about the reward $r_a(t)$ for the action a selected at time t. The optimal action corresponds to:

$$a^* = \arg\max_a E(r_a|a)$$

where E denotes the expectancy operator. However, we do not know the true expected reward, but only obtain samples around this average.

The IE algorithm stores an estimate of the expected reinforcement of an action and some information about how good the estimate is (Kaelbling, 1993). The IE algorithm estimates the confidence interval of the average of the data obtained when executing actions. The upper bound of the confidence interval can be calculated using the following standard statistics, with n as the number of trials a behavior has been selected and $\sum_{i=1}^{n} r_a(i)$ the total reinforcement a behavior a has received. The upper bound of a $100(1-\alpha)\%$ confidence interval for the mean of the distributions is calculated by

$$nub(n, \sum_{i=1}^{n} r_a(i), \sum_{i=1}^{n} r_a(i)^2) = Q(a) + t_{\alpha/2}^{(n-1)} \frac{s}{\sqrt{n}}$$

with $Q(a) = \frac{\sum_{i=1}^{n} r_a(i)}{n}$ as the sample mean, and

$$s = \sqrt{\frac{n \sum_{i=1}^{n} r_a(i)^2 - (\sum_{i=1}^{n} r_a(i))^2}{n(n-1)}}$$

being the standard deviation. $t_{\alpha/2}^{(n-1)}$ is the Student's T function with n-1 degrees of freedom at the $\alpha/2$ confidence level. The IE algorithm selects the actions with the highest upper bound and is therefore optimistic about the results. If the spread of the data points is

high, then the interval is large. As there are more data points collected through time, the interval shrinks because there is more information available.

In the case of the goal-keeper, the best action is the behavior that keeps the ball out of the goal for the longest time. The input of the algorithm is the amount of time there was no goal scored against. The action chosen is the one with the highest upper bound of the 95% confidence interval, because the longer no points are scored against our team, the better it is.

3. Experimental results

The experiments were performed with the Philips RoboCup robot. The IE algorithm is tested with different humans playing against the goal keeper. Every human opponent (5 in total over 9 experiments) had to start at the center of the field in the circle. The ball had to remain on the ground. In three of the experiments the human was handicapped. He had to dribble the ball with his hands using a plastic box. The keeper had five behaviors to choose from, ranging from 0 to 4, with 0 being the original behavior used during the competitions in Lisbon in 2004.

exp.	#init	conv.	best	avg. time
1	3	41	4	7.86
2	3	28	4	13.78
3	3	20	3	5.65
4	6	52	3	20.45
5	6	99	4	8.84
6	6	30	4	9.36
7	6	-	-	-
8	6	30	4	7.01
9	6	46	4	13.59

Table 1. results from robot experiments

In table 1 *exp.* stands for experiment number, *#init* for the amount of initializations per behavior before starting the IE algorithm, *conv.* for when convergence appeared (defined as choosing the same behavior for another 15 times in a row), *best* for which behavior is the best, *avg. time* for the average time the behavior defended the goal.

The results show that the IE algorithm most often converges to behavior 4. We noticed that the human opponents used the same strategy in these runs, which was full frontal attack. Behavior 4 is especially suited for defending a frontal attack. In the third and fourth experiment behavior 3 was the best. This was the result of a weakness in behavior 4, which was found very quickly by the human subject. The robot did not drive close enough to the goal and simply going around the keeper usually resulted in scoring a point. In two different runs (6 and 8) the initialization phase was enough to select the best behavior. On average it takes about $346/8 \approx 43$ (not counting the not converged run) executions for the algorithm to find the best solution. Since it took, on average, about 15 seconds to test one behavior (thus scoring against the goal keeper) and 15 seconds to set up the experiment anew, the entire behavior set was, on average, trained in less than 25 minutes!

4. Conclusions

In this paper we proposed a learning method for selecting behaviors in RoboCup soccer games with real robots. The system uses behaviors that are programmed by humans, which is a fast way of generating quite good behaviors. The system then learns to select between competing behaviors the ones which are promising to become the best performing one. The method is based on the Interval Estimation algorithm which works very well in resolving the exploration/exploitation dilemma (Thrun, 1992; Sutton & Barto, 1998) found in reinforcement learning; one wants to execute the best current behavior, but also try out different behaviors which may even be better.

Since the method does not rely on a huge number of experiences, it is a very interesting algorithm for quickly selecting behaviors that work well against a specific opponent. Future RoboCup competitions with the Philips RoboCup team will show whether our system is really effective, in fact in the RoboCup worlchampionship 2005 in Osaka, the Philips RoboCup team became 3^{rd}.

References

Arkin, R. (1998). *Behavior-based robotics.* The MIT Press.

Brooks, R. (1986). A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation, 2(1).*

Kaelbling, L. P. (1993). *Learning in embedded systems.* MIT Press.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction.* The MIT press, Cambridge MA, A Bradford Book.

Thrun, S. (1992). *Efficient exploration in reinforcement learning* (Technical Report CMU-CS-92-102). Carnegie-Mellon University.