ZJUNlict
Team Description for RoboCup 2011

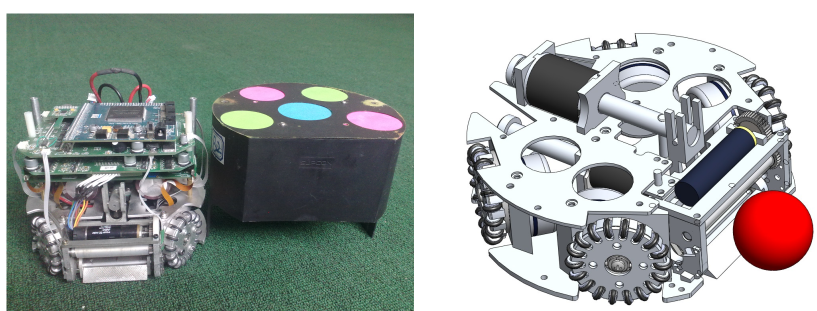
Yonghai Wu, Penghui Yin, Yue Ma, Yue Zhao, Qiangsheng Huang, Rong Xiong

National Laboratory of Industrial Control Technology
Zhejiang University
Zheda Road No.38, Hangzhou
Zhejiang Province, P.R.China
liunian@zju.edu.cn
http://www.nlict.zju.edu.cn/ssl/WelcomePage.html

Abstract. This paper describes the improvements of ZJUNlict we have made during last year. We make changes in the motor drive and communication. In this paper we will emphasize the main ideas of our AI software system. To strengthen the continuity of the strategy, we adopt a bayes-based method to evaluate game state. We also implement scirpt configuration in the agent-level, which is to be used in high play-level.

1 Introduction

Our team is an open project supported by the National Lab. of Industrial Control Technology in Zhejiang University, China. We have started since 2003 and participated in RoboCup 2004-2009. The competition and communication in RoboCup games benefit us a lot. In 2007-2008 Robocup, we were one of the top four teams of the world. We also won the first place in Robocup China Open from 2006 to 2008. Our Team members come from serveral different colleges, so each member can contribute more to our project and do more efficient job.



(a) Robot (b) Model

Fig. 1. Our Robots

2 Robot

2.1 Mechanical Design

Our new robots (Fig. 1(a)) mechanical information:

– Height: 145mm
– Diameter: 178mm
– Percentage of ball coverage: 18%

Our robots are equipped with 4 omni-directional wheels. Each is driven by a 30 watt brushless Maxon brushless motors which help our robot run with about $2.5m/s$ and $6.0m/s^2$. The reduction ratio of the gearbox with internal spur gear is 4:1. Besides there are three major machinery devices: a dribbling device, a shooting device and a chipping device.

We redisigned the omni-wheels to reduce the friction between the small passive wheels and the driving wheel. The friciton can cause the robot movement is not smooth. We spend a lot of time to test the dribber materail in order to choose a satisfactory one. The 3d view of robots mechanical system is shown in Figure 1(b).

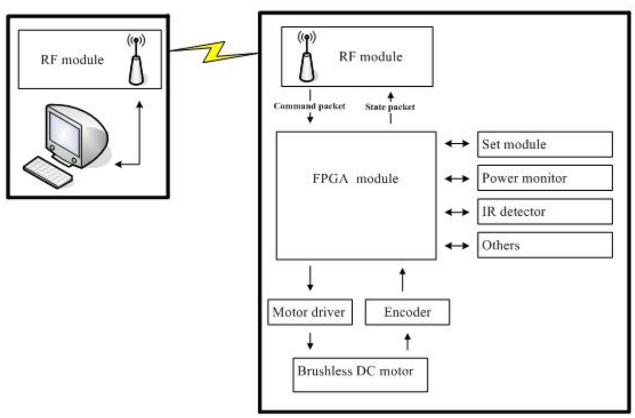


Fig. 2. Schematic diagram

2.2 ElectricalSystem

Our circuit architecture use FPGA-based all-in-one solution as the central pro-cessor module. Our motor driving part is a stable module based on MC33035, which has been developed to complete the four years since 2007 in Atlanta. There is an encoder module to form a local feedback control loop. A commercial wireless module based on nRF2401 is used on our robot. We choose two smaller capacitors with higher voltage, to achieve a better result. They will help us save more power and permit several shots in short intervals. In addition, we have set module, power monitor module, IR detector module, and so on, in order to complete the functions of our robots in Fig. 2.

3 AI System

3.1 Strategy Hierarchical Architecture

The AI module [2] for our off-board control system is shown in Fig. 3. It is the brain for planning strategy and coordination among robots in both attack mode and defense mode. The whole system is composed of world model and decision model. The former provide all the information in the match and also the decision subsystem will feedback to the predictor in world model. Thus, the close loop system adjusts all agents behavior according to the change of the environment in real time.

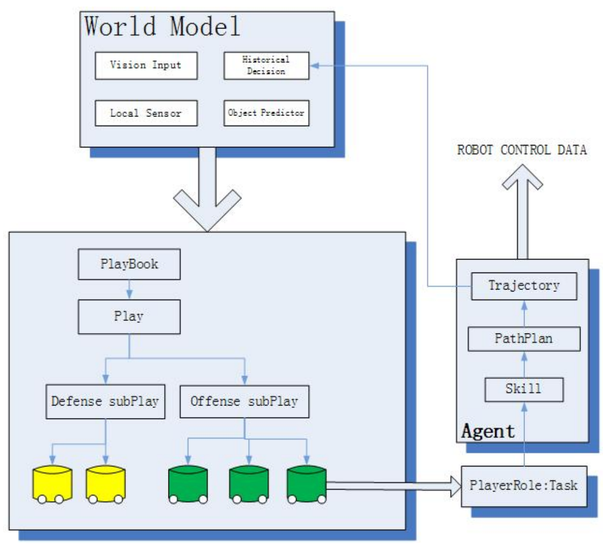


Fig. 3. Software Architecture

3.2 Play

Our AI module is implemented using a play-based approach. Each play represents a fixed team plan, in which each team group has a collaborating mission to perform and that group may have variable agents to execute. We can also consider the play a coach in the soccer game. The plays can transfer to each other, and the group for each agent can change too.

3.3 Agent

Agent is performed as robot behavior which is assigned by Play to control robot to performed a specific action such as manipulate ball to zone, passing the ball to a teammate, scramble ball from opponent, get ball. The Agent first select a best proper team member according to the assigned zone and task which receive form play, then selects a proper skill for the team member performing the assigned behavior in every execution cycle and finally generates the best target for robot action. For example, in the shooting task it will be the best shoot point on opponent goal line. This year, we mainly focus on script configuration to control robot behavior by a unified FSM-based mechanism, as see in Fig. 4.

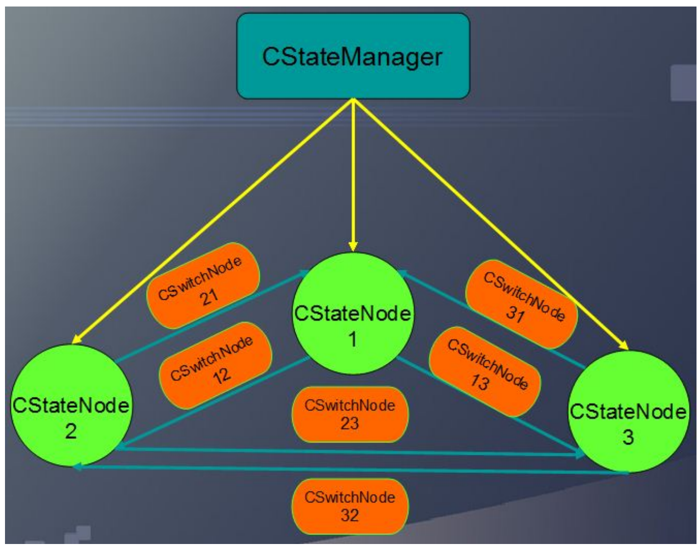


Fig. 4. FSM for Agent

Fig. 5 is a sample of our agent-level script:

3.4 Skill

Skill is a set of basic knowledge for every agent, such as how to move to a point,how to get the ball and kick. Some skill generates a next target point of the specific path which will be passed to the navigation module for finally generating a avoiding collision path. Some skill will direct generate the speed trajectory for some special behavior such as pulling the ball from a opponent front. Each of skills has different main idea of generating path for robot, intercept the ball skill is different from move to point skill in many ways. We can test each skill independently for the best performance for each of skills.

```
# Description: sample script for an agent

# Define num of inner states
num_of_states
3

# Give name to each state
names_of_states
state1  state2  state3

# Switch of inner states
switch_of_states
state1  2
        state2  normal  2       !a & b
                a       dist2ball<50
                b       canshoot
        state3  normal  2       !a & b
                a       dist2ball<50
                b       canpass
state2  2
        state3  fast    2       !a & b
                a       canshoot
                b       canpass
        state1  normal  2       !a & !b
                a       canshoot
                b       canpass
state3  2
        state2  fast    1       a
                a       canshoot
        state1  slow    2       !a & !b
                a       canshoot
                b       canpass

# Select corresponding skill
skills_of_states
beginning       startskill
state1          getball
state2          kickball
state3          passball
finished        finishedskill
```

Fig. 5. Sample script for agent-level

4 Game State Evaluation

A right evaluation of game state plays an important role in the match. For the complexity of game situation, it's really a troublesome work. Fig. 6 shows the method we used to adopt to evaluate game state which completely depends on the observation information.

Fig. 6. Previous Evaluation Method

It easily brings mistakes when there come momentary errors from the observation information, which lead to suffering in the strategy. Thus, we consider a comprehensive view of the observation information, the strength of the rivals as well as the strategy we take to realize the evaluation. Our new method is based on the Bayes Theory [1], which gives a creative combination between the observation information and the historical strategy feedback.

Fig. 7 depicts the basic Bayes Filter Algorithm in pseudo code form.

Algorithm Bayes filter p(x_k,u_k,z_k)
for all x_k **do**
  $\overline{p}(x_k) = \sum_{x_k} p\big(x_k|u_k, x_{k-1}\big)p\big(x_{k-1}\big)$
  $p(x_k) = \eta p\big(z_k|x_k,\big)\overline{p}\big(x_k\big)$
end for

Fig. 7. General Algorithm for Bayes-filter

Base on this filter, we have a new evaluator of game state, as see in Fig. 8.

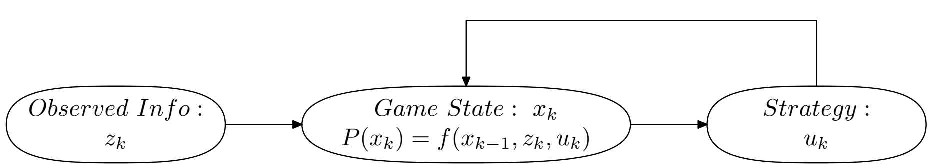


Fig. 8. Bayes-based Evaluation Method

Compared to previous one, our current method owns two main features:

- **More stable:** The evaluator gives more appropriate analysis of game state, even though there come momentary errors in observation information, and helps to reduce the perturbation of the strategy while strengthens the continuity of the strategy.
- **Well targeted:** Variable initial values of prior probabilities $p(x_k|u_k, x_{k-1})$ in the algorithm accustomed to different characteristic teams, making it more convenient to configure the attacking and defending strategy in a more flexible way.

5 Conclusion

Owing to our all team members' hard work, we've made some improvements in our system, both in hardware and software. If the above information is useful to some new participating teams, or can contribute to the small size league community, we will be very honor. We are also looking forward to share experiences with other great teams around the world.

References

1. Sebastian Thrun, Wolfram Burgard, Dieter Fox, Probabilistic Robotics, The MIT Press
2. Yonghai Wu,Xingzhong Qiu,Guo Yu, Jianjun Chen and Xuqing Rie: Extended TDP of ZjuNlict 2009 *Robocup 2009*