Strive(Simulation 2D) Team Description 2008

Zheng Yonglei , Zhu Peijun

Shanghai University, China
Box.3#, No. 149, Yanchang Road, Shanghai, China, 200072
wanmic@163.com
zhengyonglei@gmail.com
http://robocup.shu.edu.cn

Abstract. This paper describes the main features of Strive 2D soccer simulation team, which is now still under developing and preparing for the Robocup Soccer Simulation 2D competitions. The paper gives an overview of the architecture of the Strive agents, its skills, a simple statics description and the future research directions.

1   Introduction:

The team name "Strive" represents the motto of Shanghai University, which means never stop striving towards perfection.

The Strive 2D soccer simulation team of Shanghai University was established soon after the competition was introduced into China and attended the competitions of Soccer Simulation 2D League every year since then. From then on, the team have devoted their energy and creativity to the career and won their honor.

Current soccer simulation agent is being developed on the basis of Mersad-5.9.5 release source code. Thanks to Mersad team's kindness, we have access to their mature low level modules including world model and basic actions, even high level classes in this release and we are creating our own advanced actions and strategies to search for better solutions for the game.

2   Architecture:

The agent contains two important parts: player system and planning system.

2.1   Player

A player consists of 4 levels. A higher level relies on the lower levels which offer necessary functions.

2.1.1   SignalHandled

The lowest level, which handles the signals sent by the operation system.

2.1.2   BasicAgent

Above SignalHandled is the basic agent level. In this level, agent handles the basic information of the field and offer basic function to higher layers. World model is attached to this level, by which the agent can establish a reliable model of the field objects and the players. In this level, the agent initializes its first connection with server and guarantee the agent running in a stable status.

2.1.3   SyncedAgent

This level is responsible for the synchronization with the current environment, which is important for getting updated information.

2.1.4   AdvanedAgent

The highest level, which is called advanced agent, handles the information collected and makes plans and decisions.

Many classes offering various functions are attached to different levels according to their usage.
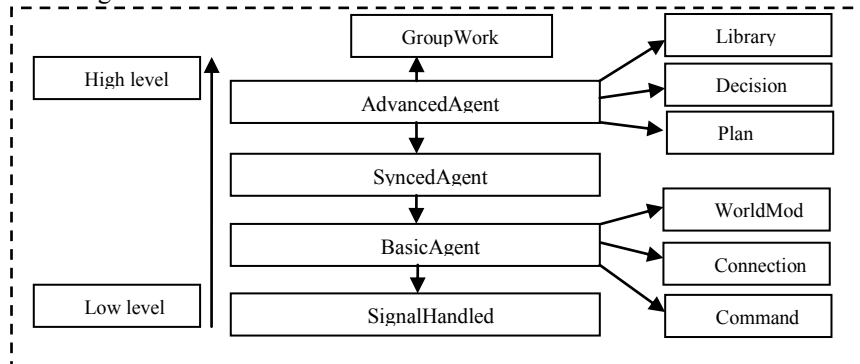

Fig. 1. An overview of the Architecture of Strive agent

2.2   Plan

2.2.1   HPS

The planning system is designed as a hierarchical planning system (HPS).This planning system allows programmers to apply different techniques in separate layers. The HPS is a very high level, higher than advanced actions.

2.2.2   Library

The class Library is useful for handling and calculating information that are needed to be operate only once in a single cycle. Unlike the entire HPS system which may be recalled in one cycle, the Library is updated once a cycle. This section contains all lengthy and static processing.

2.2.3   Form

The class Form is used as a standard template for some fundamental issues, especially like neck decision and say decision. A plan must "fill out" the form if it wants to change these behaviors.

2.2.4   Success Rate

Success rate represents the chance of the successful implementation of this plan. This factor play an important role in the HPS system on deciding whether to implement a certain plan.

3   Basic Skills

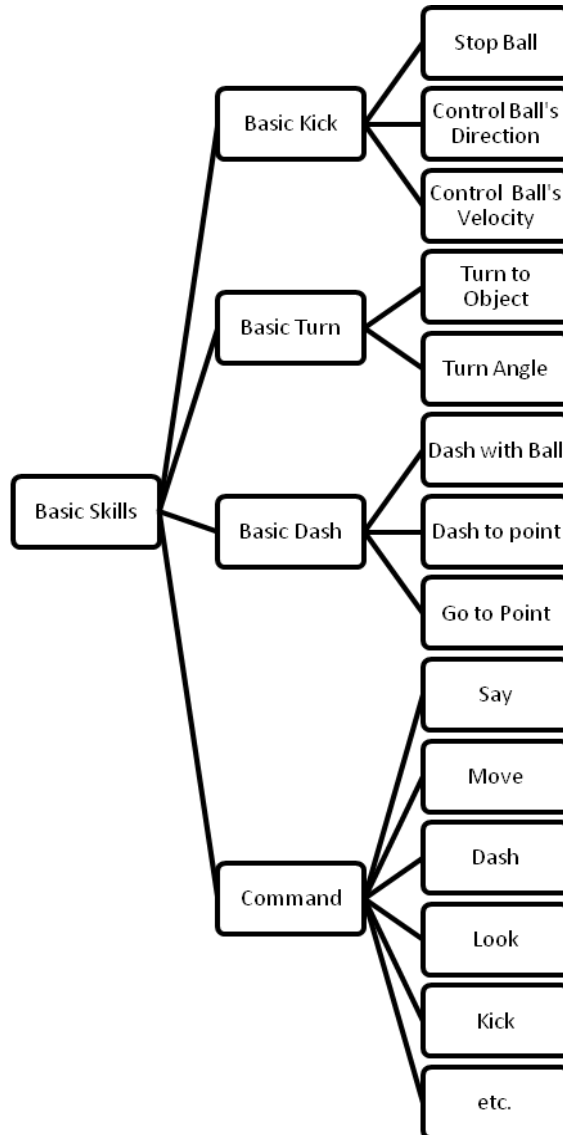The basic skills of Strive agent include Dash, Kick, View, Turn and Command.

Fig. 2. Basic Skills (Not all skills are listed)

4   Advanced Skills

The Advanced Skills of Strive agent include Block, Clear, Defense, Dribble, Intercept, Positioning, Pass and Tackle.
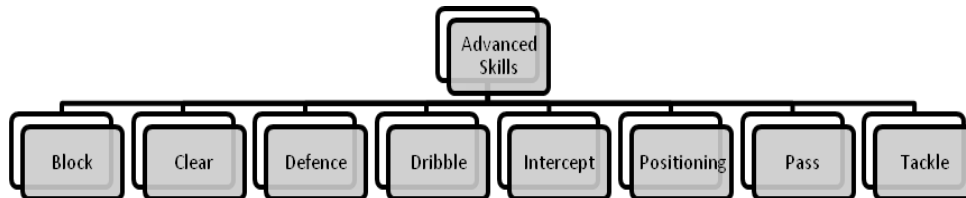
Figure 3 offers a clears overview.

Fig. 3. Advanced skills (Not all skills are listed)

5   Tactics Examples

5.1   Plan and Decision-making

Almost all the tasks are assigned by the players uniform number, each player has his own uniform number, players with different union number has different tactics functions. Strive 2D soccer simulation team's formation is "4-4-3". Set the midfield players as an example, when Strive team is offending, most of the time, the WF will dribble the ball along the sideline, the mid player decides what position to take in order to help the ball taker. On the other hand, when the defense happen, the midfield player decides whether to block the opponent from going ahead or go straight back directly. Of course, in some given conditions, some players have special assignment for the tactical needs.

A variable "weight" is used to estimate which decision to make because various schemes can be taken. Strive has certain algorithms to get the "successrate". If "successrate" is high enough, this plan will be implemented.

Figure 4 offers a brief look on one of the offending tactics of Strive team. In some conditions, when player A is offending towards the right side, player B is assigned to get to the opponent's goal as close as possible, so that if A could pass the ball through the front of the goalkeeper, B would get a chance to shoot. In most circumstances, the opponent's defenders will be attracted to get back close to the goal, and then the center may be a chance. The ball can be passed to the player C in the very good position and win a large chance to goal.
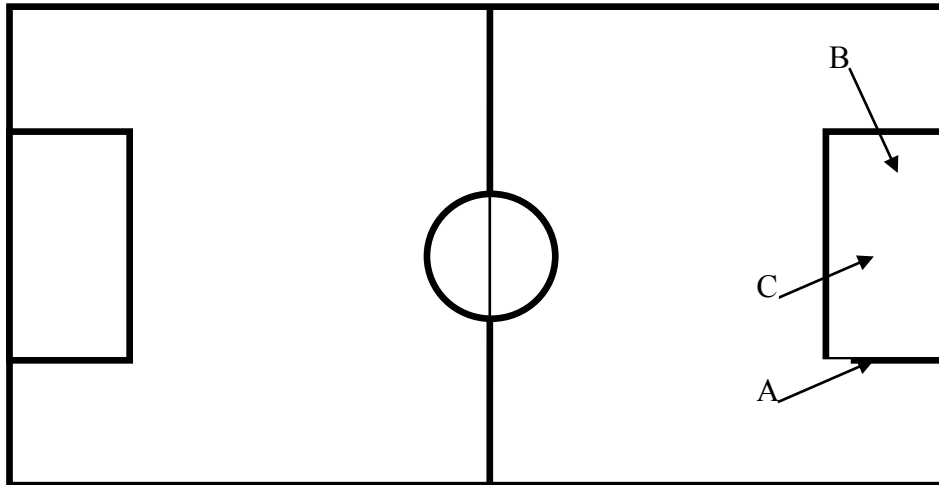
Fig. 4. A sample Tactics

5.2   Positioning

Most of the time, Strive team use a classic formation of "4-3-3". In fact, Strive team has a flexible positioning system which is design to optimize the offense and defense formation as well as saving the player's stamina. The default formation information is stored in the binary code while alternative options are available in configure files. This system offers a convenient way to adjust the formation according to different conditions. The formation may become "5-2-3" when defense need to be enhanced, and the formation may become more aggressive when it is easy to score.

6   Future Research Directions

In this paper, we have simply describe the main features of Strive team. Although the current agents have a relatively stable status, there are several problems must be corrected as soon as possible. In addition to debugging, we are planning to design new plans, tactics and advanced skills for Strive agents. We are also interested in studying reinforcement-learning techniques and hoping to apply these techniques to Strive team.

References

1. Peter Stone. Layered Learning in Multi-Agent Systems[PhD thesis]. Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, USA. December 1998.
2. Remco de Boer, Jelle Kok. The Incremental Development of a Synthetic Multi-Agent System: The UvA Trilearn 2001 Robotic Soccer Simulation Team[Master Thesis]. Artificial Intelligence and Computer Science, University of Amsterdam. February 2002.