Study on Teamwork in Robot Football Game Based on Multi-Agent System (MAS)

Deyu Chen and Shicheng Dong

School of Computer and Information Engineering
Harbin University of Commerce
cdy1974610@sohu.com

Abstract

The robot football game requires several robots to accomplish task in an real-time antagonism environment, which provides a good platform for studying multi-agent cooperation. This paper applied a real-time multi-agent cooperation model to achieve teamwork in robot football game, the concrete contents include: description of strategy, cooperation based on strategy, formation and termination of cooperation and execution of cooperation.

Keywords: *Robot football game, Agent unit, Multi-agent system (MAS), Cooperation model*

1. Introduction

The study of multi-agent system is an important issue and evolution direction in the nowadays artificial intelligence field [1-3, 8-10]. The principal issue of artificial intelligence is supposed to be "the solution to the problem of multi-entity in a dynamic and unpredictable environment." Being an environment with complexity, non-linearity, time-variance, uncertainty and alike while with a determined goal requiring cooperation of several agents to score as much as it can be to win the game, the robot football game, a typical multi-agent system in which every robot footballer is an agent, has became an experiment platform for agent control and artificial intelligence [5, 6].

Although being decisive to the performance of the system and thus making the designing of multi-agent system a challenging task, the decisive factors, interaction, coordination and cooperation among the Agents are not evidently found significant in the software and hardware designing processes until it comes to the cooperation among several agents [4]. How to realize effective cooperation and teamwork among each Agent member in a changeable environment presents itself as a principle problem to the study and research of Multi-agent system.

In complex systems, the research object is a unit which has self-determination, communication ability and certain adaptability. Agent which has commonsense in complex systems is able to be abstracted and defined as Agent unit in order to be described formally. According to dynamics character of Agent, it is defined as the following:

$$\text{Agent unit}=(T, I, O, S, IA, OA, R) \quad (1)$$

Satisfy:

$$T \in R_0^+$$
$$I = EI \cup II$$
$$R : \left(S \times I \times T \xrightarrow{\partial} S \times T\right) \cup \left(S \times I \times T \xrightarrow{\partial} O\right)$$
$$\cup \left(S \times I \times T \xrightarrow{\partial} II\right) \cup \left(S \times T \xrightarrow{\partial} II\right)$$
$$\forall i \in I \Rightarrow i.source \in IA$$
$$\forall o \in O \Rightarrow o.dest \in OA$$

Hereinto, T is an aggregate of separated time, usually logic time. I is an aggregate of input message and O is an aggregate of output message. EI is outside input message. II is inside message. S is an aggregate of the state of Agent unit and the state is finite dimensional real vector. IA is Agent aggregate of all possibility of input message to the Agent. OA is Agent aggregate of all possibility of input message from the Agent. R is regulation process of Agent unit.

According to the definition of Agent unit, Agent unit can be divided into several parts:

AgentID: the identification of Agent unit used to confirm each Agent in complex systems uniquely.

Message Process System (MPS): receiving and sending messages.

Inside State (IS): inside state of Agent unit

Regulation Process Systems (MPS): According to different application destinations, RPS is either a simple equation or a rather integrated regulation system including regulation-base, regulation making system, regulation selecting system and regulation evaluating system.

Kernel of Agent unit: selecting messages from MPS and processing messages with regulation from regulation system while modifying regulation system and IS to send output message to other Agents.

Logic Time of Agent: LT

The structure of Agent unit is shown in Figure 1

It is important to notice that in complex systems there exists some uncertain self-determination and passive units which are part of environment of goal complex system. These passive units are unified to environment Agent which is able to process messages related to such units.

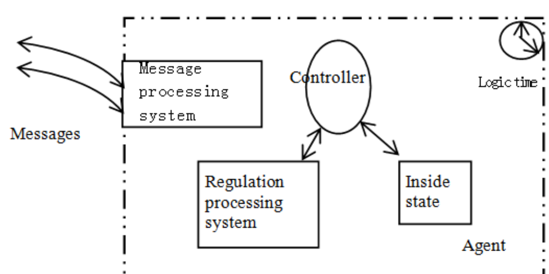

Figure 1. Structure of Agent Unit

Agent unit is defined as the smallest unit in the complex system so that all units could be made in the complex system by defining arrangement relationship between inheriting and Agent.

The inheriting relationship of Agent is defined as:

According to $Agent_i$ and $Agent_j$, if $I_i \subset I_j$, then $Agent_i$ and $Agent_j$ satisfy with inheriting relationship, that is, $Agent_i$ satisfy $Agent_j$. I_i is the aggregate of all examples of $Agent_i$ and I_j is the aggregate of all examples of $Agent_j$.

It forms the Agent which is a certain application by inheriting the property and interface of Agent unit.

Complex systems are characterized by multilayer in which high layer subsystem is composed of the connection of low layer subsystems. The lowest layer subsystem is unit subsystem.

The layer relationship is defined as the follow:

According to $Agent_i$ and $Agent_j$, if $Agent_i \in Agent_j$, then $Agent_i$ and $Agent_j$ satisfy with layer relationship, noted $Agent_j \propto Agent_i$.

Usually, the layer relationship of Agent is that one Agent corresponds to several Agents. The deduction is in (2).

$$
\begin{aligned}
T_{A_c} &= \left\{ \min \sum_{A_i \propto A_c} T_{A_i} \middle| A_c \in Agent \cap A_i \in Agent \right\} \\
I_{A_c} &= \left\{ \bigcup_{A_i \propto A_c} I_{A_i} \middle| A_c \in Agent \cap A_i \in Agent \right\} \\
O_{A_c} &= \left\{ \bigcup_{A_i \propto A_c} O_{A_i} \middle| A_c \in Agent \cap A_i \in Agent \right\} \\
S_{A_c} &= \left\{ \bigcup_{A_i \propto A_c} S_{A_i} \middle| A_c \in Agent \cap A_i \in Agent \right\} \\
IA_{A_c} &= \{ A_k | A_c \in Agent \cap A_i \in Agent \cap A_i \\
&\quad \propto A_c \cap (A_k \in IA_{A_i} \cap \neg (A_k \propto A_c)) \} \\
OA_{A_c} &= \{ A_k | A_c \in Agent \cap A_i \in Agent \cap A_i \\
&\quad \propto A_c \cap (A_k \in OA_{A_i} \cap \neg (A_k \propto A_c)) \} \\
R_{A_c} &= \left\{ \bigcup_{A_i \propto A_c} R_{A_i} \middle| A_c \in Agent \cap A_i \in Agent \right\}
\end{aligned} \tag{2}
$$

2. Multi-agent system

In the process of artificial intelligence research, it became dawn on researchers that they should integrate research outputs in each artificial research filed into a "human being" with intelligent behavior concept, what's more important, they come to realize that the essence of human intelligence is social intelligence. The predominating portion of human activity involves association form by several individual, the solution of a big and complicate problem takes several professionals or organized cooperation [7-11]. The most important and most portion of intelligence shall assert itself in the various activities proceeded in the society formed by numerous individuals. Cooperation, competition, negotiation and alike are the most common manifestations of human intelligent behavior. To research into social intelligence, the "Agent" as a counterpart of the basic element of society – "human", naturally qualifies itself to be the basic object for artificial intelligence research.

Agent as a term in distributed artificial intelligence is defined as an entity in a particular environment who is able to exercise its knowledge and ability to attain a particular goal formed based on corresponding reaction to the requirements of environment or itself. Agent is characterized with autonomy, social ability, reactability and spontaneous action [2], of which autonomy refers to that Agent has certain control over its behavior and the interior condition; social ability refer to that the Agent is able to exchange information with other Agent in a Agent communication language; reactibility refers to that the Agent is perceptive to the environment and able to change the environment by own action; spontaneous behavior refers to that the behavior of Agent is active, they sense the change of the environment and they take certain action to achieve its goal.

Despite the fact that individual Agent has the ability to accomplish some task by itself, along with the development of network and distributed computing, some reality system is unusually complicate and gigantic with the feature of distribution, multi-Agent system in rapid development has become a hot field in artificial intelligence research. With distributed data and resources, a multi-Agent system comprises several Agents which possess incomplete information and ability. As an effective solution to complicate system problems, Multi-Agent system divide the complicate system into relatively independent Agent subsystem and the cooperation and competition among Agents shall provide an answer to the complicated problem.

3. Model of cooperation among Agents in a robot football game

Once the robot football game is selected as the artificial intelligence test platform, most problems in a multi-Agent system will be embodied in the game for the following reasons:

1) The robot football team as a whole formed by several robots has a decided goal (to kick the ball into the rival’s net while keep the rival from scoring), of which each robot can be considered as Agent, and consequently the robot football game can be considered as a multi-Agent system.

2) The robot football game is a distributed multi-robot system where a group of robot contests with the other robot group, of which for the common goal each robot has not only to exercise individual skill, but only it has to work as port of the group to bring out the strength of a whole by cooperating and coordinating with each other.

3) On the game field, the situation of balance is ever changing, robot of each side has to make decision on the current situation based on the understanding of the dynamic changes in the formations of its own side and the rival’s side to regroup, and this is a dynamic organizing processing of a robot group.

4) Besides that each of both sides has to make real-time decision, it has to cooperate and coordinate with each robot of its own side to accomplish the common goal (to win the game). These are the problems about coordination mechanism and real-time information processing among robots.

The robot football game proceeds in an environment of real-time and dynamic changes and antagonism. To take each robot footballer as an Agent and organize them, each Agent not only requires more than independent ability to handle the situation, but also coordinative teamwork with other Agent to act as a group [5, 6]. The real-time Agent model adopted in this paper has the ability of making independent decision as well as cooperating with other Agent shown as Figure 2.

In the Agent model structure shown in Figure 2, information base is used to store decision making information transmitted from the decision making unit and field information, and game situation information provided by vision system. The decision making unit has two functions: the information amalgamation and movement selection. The information amalgamation function is responsible for collecting all information related to this Agent and evaluating the game situation, where as movement selection function for selecting right movement based on the result of information amalgamation. The Knowledge base includes related knowledge information to the decision making. Movement execution unit is responsible for the accomplishment of execution of specific movement. Blackboard Agent serves as a communication modular to bring cooperation among several Agents to reality by handling information transmission between an Agent and other teammate, reporting the position, direction and speed of ball and players of each side, behavior of teammate to prevent fall afoul with behavior of teammates.

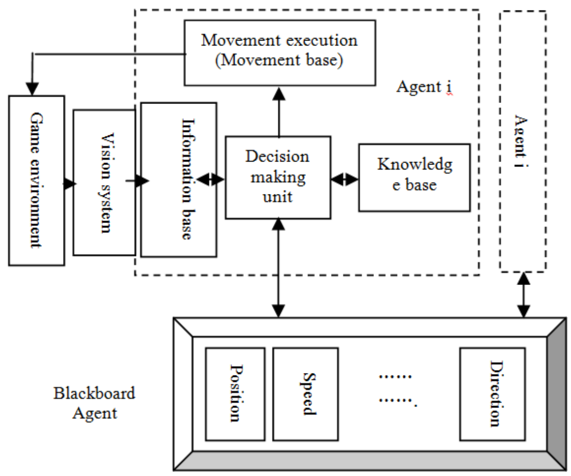

Figure 2. Cooperation model among multi-agent

4. Execution of Multi-Agent System

A. Description of Strategy

The cooperation of a multi-Agent system is based on the strategy comprising certain occasions. The description of strategy shall direct each Agent how to act on every occasion. All occasions in the strategy shall not appear but it requires the Agent to make selection among them accordingly. A description of strategy in XML language given by this paper embraces the name, code, priority, number of the member, number of the occasion, defeat condition of the strategy and each occasion included in the strategy.

B. Cooperation based on Strategy

We can read strategy as the cooperation among and a behavior plan for several Agents. When an individual Agent is unable to attain its goal and turn to other Agent for assistance, the action sequence in a certain period forms the cooperation the Agent calling for assistance and the one it turns to. In the cooperation, the actions of several Agents are closely interlinked, thus the cooperation can be viewed as sequence knowing interaction between several Agents performing a strategy.

The cooperation based on strategy is built on displayed communication, namely, a particular Agent decides a strategy then extends requests to other participants of the strategy, and other participants will grant the request or turn it down, the cooperation shall be formed only if all participants agree on the strategy. Upon formation of cooperation, each Agent shall select different action according to the description of strategy. The termination of cooperation takes two forms: accomplishment and abortion, of which the former means smooth execution of strategy and the organizer of the cooperation shall inform every participant of it; but the later is a result of the fact that a participant has selected some cooperation of higher priority or the current cooperation can not be successfully executed.

To guarantee the smooth execution of the strategy, every participant must be on a same occasion of the strategy at the same moment, so in the cooperation process when every occasion is over, there will be the participant to notify everyone to move on to next occasion, and on different occasion, there will be different participant will give this notification.

There might be a number of cooperation taking place at the same moment, but for each time the number of cooperation in which an Agent can take part is only one.

C. Formation and Termination of Cooperation

The content of communication of connection formation includes the following three parts:

- Sign of cooperation formation.
- Number of cooperator.
- Content of strategy.

We only need to convert the strategy into code to transmit in communication. The content of communication to form cooperation is arranged in Table 1.

Since the number of cooperation in which each Agent can take part is only one, when Agent receiving cooperation request, it will check if it has be in another cooperation, if not, it will grant the request, otherwise it will make a choice between the current cooperation and the one it is requested to join, if it takes the new request, it will quit the current cooperation and send out message of quitting in Table 2.

If it choice to continue the current cooperation as good as turn down the request of new cooperation, it has to send out a refusal message in Table 3.

The termination of cooperation takes two forms: abortion and accomplishment, of which the latter means the goal of the current cooperation has been attained, the cooperation organizer will send out message of accomplishment in Table 4.

Whereas the former is classified into two kinds:

1) Proactive Abortion: Organizer of cooperator or participant of the cooperation aborts the current cooperation because it finds the strategy can not be carried out successfully or the situation on the field is calling for a new strategy.

2) Forced Abortion: When the cooperation is botched it will be forced to abort.

For the first situation, it will fall upon the cooperation organizer or strategy former to send out abortion message, for the second situation, every participant is able to send the abortion message in Table 5.

D. Execution of Cooperation

Once the cooperation is formed, each Agent shall select different actions to execute on different occasions according to description of strategy. Cooperation is an activity only accomplished by several Agents, since the behavior of each Agent is influential upon each other, the coordination of behavior among Agents comes very important. To guarantee the smooth enforcement of strategy, coordination must be synchronized, namely, each participant of the cooperation shall be on the same occasion described in the strategy at the same time. In the process of cooperation, when every occasion comes to a close, that will a participant notify everyone of the termination of current occasion and what the next is., which Agent to act as the notifier shall depend on different occasions. The occasion shift message format is in Table 5.

Table 1. Forming Cooperation Message

A	Strategy code	Cooperation organizer	Cooperator 1	……	Cooperator n

Table 2. Quitting Cooperation Message

B	Code of current strategy	Cooperation organizer	Cooperator 1	……	Cooperator n

Table 3. Refusal Message in Cooperation

C	Code of new strategy	Cooperation organizer	Cooperator 1	……	Cooperator n

Table 4. Accomplishment massage of cooperation

D	Strategy code	Cooperation organizer	Cooperator 1	……	Cooperator n

Table 5. Abortion Massage of Cooperation

E	Strategy code	Cooperation organizer	Cooperator 1	……	Cooperator n

Table 6. Occasion Shift Message in Cooperation

F	Strategy code	Current occasion	Next occasion	Cooperation organizer	Cooperator 1	……	Cooperator n

The cooperation in a multi-Agent system in a robot football game shall go through the following stages:

1) A certain Agent forms a strategy.

2) Send out cooperation request, the cooperation shall be formed when no one turns the request down.

3) When the cooperation is not aborted or terminated, the participant is allowed to decide its own action according to strategy and the strategy stage.

4) Termination or abortion of cooperation.

The specific process is expressed as the flowchart in Figure 3:

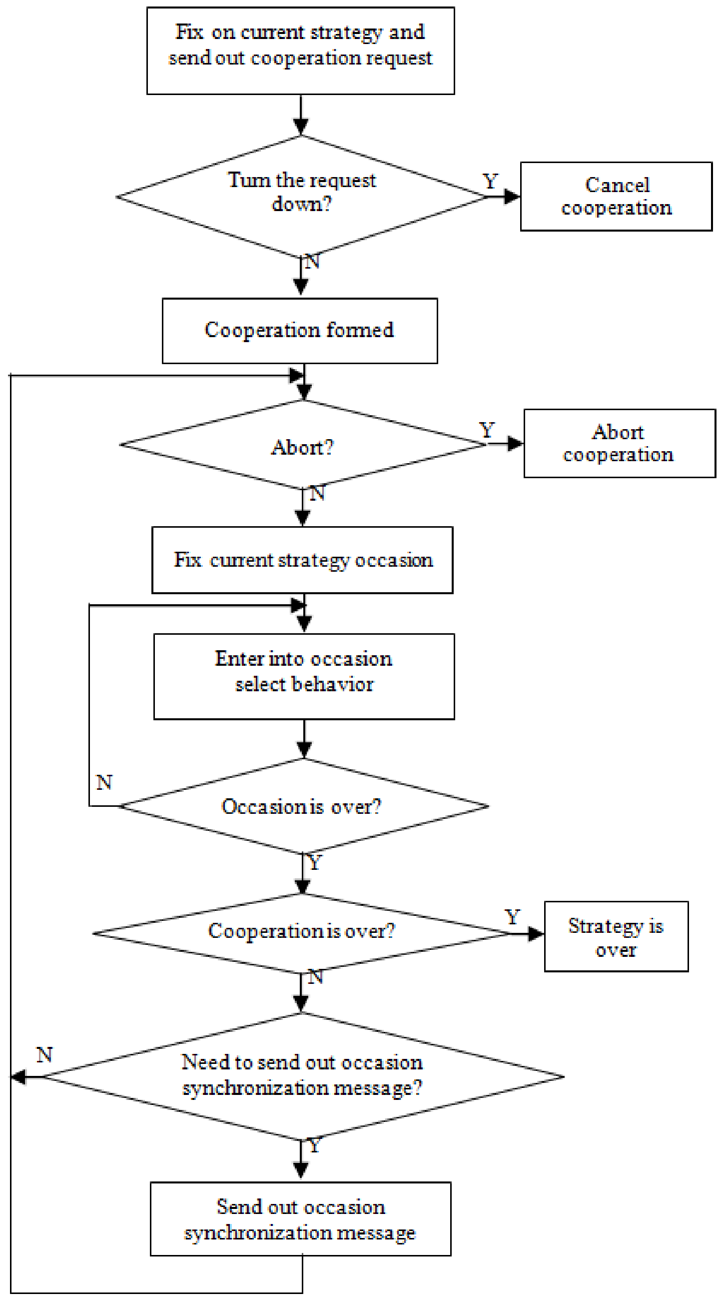

Figure 3. Flowchart of cooperation execution

In each specific robot football game, strategy is continually variational and robots' movement is also continually variational with cooperation execution. As a result, robots' movement determines that if the cooperation execution is successfully completed. Movement mainly comes down to the several aspects, which are teammate's position, shoot, movement and defence, *etc*.

The partial sound code of the "movement" is listed the following:

```
//StrategySystem.cpp
#include"Stdaft.h"
#include"StrgtegySystem.h"
int CStrategySystem:: Distance(CPoint which1,CPoint which2)
{
    int x;
    x=(int)sqrt((which1.x-which2.x)*(which1.x-which2.x)+(which1.y-
    which2.y)*(which1.y-which2.y));
    return(x);
}
int CStrategySystem::  Angle(CPoint which1,CPoint which2)
{
    int angle;
    angle=(int)(180.0/PI*atan2((double)abs(which1.y-which2.y),(double)abs(which1.x-
    which2.x)))
    if((which1.y>which2.y)&&(which1.x>=which2.x))
    angle=180+angle;
    else
        if((which1.y<=which2.y)&&(which1.x>which2.x))
            angle=180-angle;
        else
        if ((which1.y>which2.y)&&(which1.x<=which2.x))
            angle=360-angle;
        end if
        end if
    end if
return(angle)
}
```

5. Conclusion

In a complex and dynamic environment, the multi-Agent system provides communication among each distributed agent for the real-time accomplishment of a common task. The multi-Agent system, so to speak, is a new breakthrough in the development of artificial intelligence which has made the once closed and isolated knowledge system an open and distributed system and brought artificial intelligence into more fields of practical application.

Acknowledgements

This research is supported by PhD scientific research foundation of Harbin University of Commerce (12DL026).

References

[1] E. Oliveiro and K. Fischer, "Multi-Agent systems: which research for which applications", Robotics and Autonomous Systems, no. 27, **(1999)**, pp. 91-106.

[2] M. Wooldridge, “An introduction to MultiAgent systems” Publishing House of Electronics Industry, **(2003)**.
[3] R. Wang, “Intelligent Agent and its application in information network”, Beijing: Beijing University of Posts and Telecommunications Press, **(2006)**.
[4] J. Liu and S. Wan, “Design and realization of MultiAgent control system”, Control theory and applications, vol. 16, no. 4, **(1999)**, pp. 580-583.
[5] A. Moreno, A. Valls and M. Marin, “Multi-agent simulation of work teams”, Multi-agent Systems and Applications III, Heidelberg: [s.n.], **(2003)**.
[6] Y. Zhang, Y. Yang and B. Chen, “The application of Multi-Agent teamwork in Robot Soccer”, Microcomputer Development, vol. 14, no. 7, **(2004)**, pp. 112-114.
[7] N. Sun, “Study and application on negotiation mechanism in Multi-Agent system”, Beijing: Beijing Institute of Technology, **(2002)**.
[8] E. Al Maghayreh, S. Samarah and F. Alkhateeb, “A Framework for Monitoring the Execution of Distributed Multi-agent Programs”, International Journal of Advanced Science and Technology, vol. 38, **(2012)**, pp. 53-66.
[9] D. Blanes, E. Insfran and S. Abrahão, “A Requirements Engineering Approach for the Development of Multi-Agent Systems”, International Journal of Software Engineering and its Applications, vol. 4, no. 2, **(2010)**, pp. 1-14.
[10] A. Dardzinska, “Searching the Best Collaborating Agent for Query Answering Systems”, International Journal of Grid and Distributed Computing, vol. 4, no. 1, **(2011)**, pp. 67-80.
[11] H. Youn, S. Jang and E. Lee, “A Novel Approach for Performance Improvement of Multi-Agent based System Architecture”, International Journal of Software Engineering and its Applications, vol. 2, no. 1, **(2008)**, pp. 1-10.