Behavior Coordination in the Internet-based
Multi-robot Teleoperation System

Wei Zheng, Yuechao Wang and Ning Xi

1 Robotics Laboratory of Shenyang Institute of Automation, Chinese Academy of Sciences
2 Graduate School of Chinese Academy of Sciences
Shenyang, Liaoning Province, China
zhengwei@sia.cn

Abstract - The Internet-based teleoperation has become a popular research topic during the recent years. More and more fields are being involved and one of them is the multi-robot systems. Multi-robot systems have the ability to perform better in a complex environment; however, it is difficult for the robots to complete tasks autonomously, thus a human operator is needed to contribute his ability to direct the behaviors of the robots. Now the Internet can be used as the bridge between the human operator and the multi-robot system. In this paper the behavior coordination of the multiple robots directed by a human operator via the Internet is studied. The architecture of the human directed multi-robot system is presented and a hierarchical automaton model is proposed to describe the decision-making process for the behavior coordination of the Internet-based multi-robot system. An Internet-based tele-robot soccer system is built based on these methods and the details of the system are described.

Index Terms - Multi-robot, Internet, teleoperation, behavior coordination, tele-robot soccer

I. INTRODUCTION

Since the Internet is a general interactive and economical telecommunication entity, the Internet-based teleoperation has become a popular research topic during the recent years. More and more fields are being involved and one of them is the multi-robot systems. Multi-robot systems have the ability to perform better in a complex environment and thus the interest in them is arising in military, rescue, space and many other potential fields. However, it is very difficult for the multiple robots to complete tasks autonomously in these complex environments. Therefore, a human operator is needed to contribute his ability and intelligence to direct the behaviors of the robots [1]. Thus the ability and the performance of the multi-robot system can be significantly improved. Now the Internet, which is available all over the world, can be used as the bridge between the operator and the multi-robot system.

When the Internet is used as the communication channel between the human and the machine, the system will be more economical and universal than traditional teleoperation systems. However, the characteristic of the Internet also brings some problems. The random delay, data loss, disconnection and other problems will affect the performance of a teleoperation system greatly, such as the stability, transparency, synchronization, etc.

Theories have been proposed to analyze the Internet-based teleoperation systems. Anderson et al. introduces passivity theory into teleoperation using the conception of passive circuit system [2]. Niemeyer and Slotine develop Anderson’s approach to wave variable [3]. Xi et al. propose a kind of event based theory and methods [4] [5], in which a non-time reference is introduced to avoid the problems caused by using the time reference. All these approaches aim at avoiding the influence of the time delay and focus on the stability of the system. With the development of the robotics, the Internet-based intelligent systems have come to be studied [6]. In the Internet-based intelligent systems, the robots have some intelligence and can perform autonomously to some extent. Thus the influence of the time delay can be alleviated. However, the intelligence of the robot system and the interactivity between the robots and the operator must be lucubrated.

In this paper, the Internet-based multi-robot teleoperation system is studied. The architecture of the system is presented and the coordination of the robots’ behaviors in the system is discussed. A hierarchical automaton model is proposed to describe the decision-making process for the behavior coordination of the system. The model is applied to an Internet-based tele-robot soccer system, which is a typical Internet-based multi-robot system. The analysis of this system can illustrate our concepts on the similar systems.

The paper is organized as the following. Section II presents the architecture of the Internet-based multi-robot system. Section III introduces the concepts of using hierarchical automaton model to describe the decision-making process for the behavior coordination of the multi-robot system and Section IV explains the model in an Internet-based tele-robot soccer system. The setups of the experiment system are described and some results are given in Section V. Then it follows the conclusion.

II. ARCHITECTURE OF THE INTERNET-BASED MULTI-ROBOT SYSTEM

The control architecture plays a key role in integrating motor controller, motion algorithms and human inputs for an intelligent robot system [7]. Based on the features of the multi-robot and the Internet robot, the architecture of the Internet-based multi-robot system is built, as shown in Fig. 1. The system is divided into four layers from the bottom to the top. Each layer has its own responsibilities and provides functions to the upper layer.

The execution layer is composed by the multiple robots, various sensors and the environment. The concepts of the be-
behavior-based approach are introduced here. Each robot in the system has some basic behaviors and all the behaviors aggregate a set $B$. Different from the typical behavior-based scheme, in which the robots react only to the changes of the environment, here the behaviors should also react to the stimuli from the commands. Thus the set $B$ can be divided into two subsets $B_1$ and $B_2$. The basic behaviors in $B_2$, which are the reactions to the tasks to be performed, are called intentional behaviors while the behaviors in $B_2$, which react to the changing environment, are called reactive behaviors. The basic behaviors encapsulate the dynamics of the multi-robot system, which can be represented by the following equations [8]

$$\begin{align*}
\frac{dx_i}{dt} &= f(x_i, u_i, r_i) \\
y_i &= h(x_i)
\end{align*}$$

(1)

where $k$ is the total number of the robots, $x_i$ represents the state variable of the $i$-th robot and $u_i$ the control variable of the $i$-th robot. And $r_i$, which is the function of $(x_j, y_j)$ ($i\neq j$), is the coupling of the robots. The output function $h_i$ represents the performance. The robots execute the proper behaviors to work in the alternating environment and in the meanwhile, the sensors detect the state of the robots as well as the environment and feedback their data to the other three layers.

The coordination layer generates intentional behaviors according to the subtasks of each robot generated by the organization layer. It also detects the obstacle by checking the sensory information and insures the safety by generating the reactive behaviors. The intentional behaviors and the reactive behaviors are coordinated in this layer and thus proper behaviors can be generated to be executed by the execution layer.

The organization layer is responsible for the analysis of the performance situation according to the strategy the operator selected from its database and the processed sensor data transmitted from the execution layer. Then the organization layer will generate the subtasks of each robot and send them to the coordination layer via the Internet. In this paper, we introduce the concept of role. The robots can act as different roles during the process of task performance. When a robot act as a certain role, it can perform a number of behaviors, which constitute a subset of the basic behavior set $B$. The organization layer generates the subtasks by dynamically assign the roles among the multiple robots.

The interaction layer is designed for the operator to interact with multiple robots. There have been some researches about how an operator can command a multi-robot system [9] [10]. In this paper, we think of the operator as a commander and he can either plan and command the system form the high level or directly intervene in the behaviors of the robots in the low level. Meanwhile, real-time supermedia information is feedback [11]. Since the lower layers could understand the operator’s intention generate appropriate commands for the robots, thus the human intelligence and the intelligence of the multi-robot system are combined. We call this interaction pattern as comprehensive teleoperation.

The core parts in this architecture are the coordination layer and the organization layer, which built a map $D$ from the information space to the behavior space, namely:

$$D : I \rightarrow B$$

(2)

Where $B$ is the behavior space and $I$ is the information space which is composed of the human commands and the sensory information. The map determines the procedure of the behavior coordination in the system and we will describe it with a hierarchical automaton model in the next section.

III. HIERARCHICAL AUTOMATON MODEL FOR BEHAVIOR COORDINATION OF MULTI-ROBOT SYSTEM

In this section, we consider describing the procedure of the behavior coordination in the system with a hierarchical automaton model.

A system that can be modeled by automata has the following features:

1) It is in a stable state during a period;
2) Some events or inputs can stimulate the system;
3) The events or inputs will bring in a sequence of managing processes, including performing specific functions, producing corresponding output and etc.
4) After these processes, the system will transit to a new relative stable state.

In the Internet-based multi-robot teleoperation system, the roles and behaviors can be viewed as stable states. The events generated by the changing environment and by the operator can make the states of the robots change from one to another. The transitions of the states can manage the behavior coordination of the robots. However, the Internet-based multi-robot teleoperation system has the following features that must be considered when using the automaton model:

1) The multi-robot system is complicated in its structure and coordination;
2) The dynamics of the system can be represented by (1), which are continuous differential equations, while the behaviors or roles are discrete. Therefore, the system is a hybrid system in essence.

For the first feature, an automaton for role assignment is built for each robot and an automaton for behavior selection is built for each role. The multiple hierarchical automatons can
be integrated to handle the complexity of the system. For the second feature, a hybrid automaton model is used when describing the behavior selection for each role. So a hierarchical hybrid automaton model is built for the system, in which the high layer is a discrete finite state automaton for role assignment and the low layer is a hybrid automaton for behavior selection.

A. The automaton for role assignment

We considered that a role is a function that one or more robots perform during the execution of tasks. A robot with a role can execute a series of basic behaviors. Each robot will be performing a role when some internal or external conditions are satisfied and will change its role with the change of these conditions. Dynamically assuming and changing roles, the robots are able to perform the task more efficiently, adapting to unexpected events in the environment and improving individual performance in benefit of the robot team. In this paper, we model the role assignment procedure as a finite state automaton (FSA).

An FSA model can be represented by a collection with five components:

\[ M = (Q, \Sigma, \delta, q_0, F) \]  

where \( Q \) denotes the state set; \( \Sigma \) is the finite input alphabet of events and \( \Sigma^* \) is the set of all finite-length strings of events; \( \delta: Q \times \Sigma^* \rightarrow Q \) is the function of state transition; \( q_0 \) represents the initial state and \( F \) is a subset of \( Q \), which denotes the marked states.

In our system, each role is a state, and they constitute the state set \( Q \). If the number of the roles in the multi-robot system is \( n \), then \( |Q| = n \), namely, there are \( n \) states in the model. All the events changing the environment and the tasks are the inputs of the FSA, and they can be represented by finite-length strings and thus constitute \( \Sigma^* \). The changing of the inputs will drive the robots to change their state, and with the specific task and workspace, we can define the function of state transition \( \delta \). An initial state \( q_0 \) is needed in the model to drive the FSA, so we can design a special initial role in the system and it can transit to any role based on the first input when the system starts to work. Because the robots in the system can perform any role during the execution of the task, all the states except \( q_0 \) are marked states.

When a robot performs a role, it can execute a series of basic behaviors. This means a subtask is generated for the robot. The robot must select proper basic behaviors to achieve this subtask and we will describe this process in the following.

B. Hybrid automaton for behavior selection in a role

When a role is assigned to a robot, the robot has the ability to execute a series of basic behaviors to act the role. The basic behavior is an atomic operation that encapsulates a specific dynamics of a robot. The dynamics of the robot is continuous and the behavior is discrete, so the hybrid automaton model is considered to describe the behavior selection in a role. Here we model behaviors and behavior selection mechanism as discrete modes and mode switching in a hybrid automaton respectively.

A hybrid automaton is a finite automaton augmented with a finite number of real-valued variables that change continuously, as specified by differential equations and inequalities [12]. It is used to describe hybrid systems, i.e., systems that are composed by discrete and continuous states. A hybrid automaton can be represented by a collection with six components:

\[ H = (Q, X, \Sigma, E, f, G) \]  

where \( Q \) is the set of discrete states, also called control mode and \( X \) represents the set of continuous variables. \( \Sigma \) is the inputs of the automaton. The hybrid state and input are denoted by \((q, x) \in Q \times X \) and \( \sigma \in \Sigma \), respectively. Discrete transitions among control modes are specified by the control switches \( E \) while the continuous dynamics of the variables are determined by the flows \( f \), generally described as differential equations in each control mode. \( G \) assigns a guard to each edge of the control mode.

By using a hybrid automaton, we are able to describe the behavior selection in a role. The basic behaviors the robot can execute in this role can be represented by the discrete control mode. The internal states and sensory information can be represented by continuous variables and updated according to the dynamic equations within each mode. The behavior selection is represented by the discrete transitions and the set of guards defines when a new basic behavior should be executed.

C. Hierarchical hybrid automaton model

For each state in the role assignment FSA, there should be a hybrid automaton to describe the behavior selection in the role. And during the task performing, each robot in the multi-robot system should act as a role. Thus the decision-making process for behavior coordination in the Internet-based multi-robot teleoperation system could be thoroughly modeled by automaton. Considering the hierarchical structure and the hybrid feature of the low layer automaton, we called this model a hierarchical hybrid automaton model.

Cooperative execution can be represented by a parallel composition of several these hierarchical hybrid automaton, each for one robot. The hybrid feature can be handled by the low layer of hybrid automaton while the hierarchical structure can manage the complexity of the system. In the following section, we will further explain this model in an Internet-based tele-robot soccer system.

IV. HIERARCHICAL HYBRID AUTOMATON MODEL IN AN INTERNET-BASED TELE-ROBOT SOCCER SYSTEM

As a standard question of the artificial intelligence and robotics, the robot soccer system is studied for years [13]. In this section, the hierarchical hybrid automaton model is applied to a 3vs3 Internet-based tele-robot soccer system built based on a centralized robot soccer system, in which the global information of the match can be obtained by a camera over the match field [14].

A. Basic behaviors and roles

In order to make the robots have the ability to play simple actions, the basic behaviors are designed. The intentional be-
haviors react to the tasks and can be further divided into attack behaviors, defense behaviors and assistant behaviors. The obstacle avoiding behavior and bound avoiding behavior constitute the reactive behaviors, which react to the changing of the environment. Fig. 2 shows the basic behaviors of the system. The basic behaviors encapsulate the dynamics of the basic actions the robots able to take and thus we can only consider the discrete behaviors when designing the high level of decision-making.

Then with inputs of the match information, the robot will assume a role, which can be the attack role, the defense role or the assist role. According to the input of the match information and the transition function, the robot can change its role among these three roles. When an emergency appears in the match, like the robot will hit the wall, the robot will change its role to the react role. And when the robot completes its react role, it will transit to the initial static state. The sketch of the state transition is shown in Fig. 3.

The match state can be used as the input of the system. Since the ball is the core of the match, a dynamic reference frame is built, with the center of the ball as the origin and the velocity direction of the ball as the x axis. The coordinate conversion can be expressed by (5).

\[
\begin{align*}
\begin{bmatrix}
    x' \\
    y'
\end{bmatrix} &=
\begin{bmatrix}
    \cos \theta & -\sin \theta \\
    \sin \theta & \cos \theta
\end{bmatrix}
\begin{bmatrix}
    x - x_0 \\
    y - y_0
\end{bmatrix},
\end{align*}
\]

\[
\begin{align*}
\theta' &= \theta - \theta_0, \\
\bar{v} &= \bar{v} - \bar{v}_0
\end{align*}
\]

Where \((x', y')\), \(\theta'\), \(\bar{v}\) is the position, direction and velocity of the robot in the dynamic coordinate separately; While \((x, y)\), \((x, y)\) is the position of the ball and the robots in the global coordinate, \(\bar{v}, \bar{v}_0\) is the velocity of the ball and the robots in the global coordinate; \(\theta, \theta_0\) are the direction of the velocity of the ball and the direction of the robot in the global coordinate separately.

According to these data in the dynamic coordinate, the feature of the match can be described by a state vector \(d, \bar{I}, A, i, M > \), where

- \(d\) is the distance between the origin(position of the ball) and the opponent's goal;
- \(\bar{I}\) is the vector of the opponent's goal;
- \(A\) is the state to judge which side is more prone to control the ball. It can be measured by the difference between the minimum distances to the ball of each team.
- \(i\) is the number of the robot in the team and \(i=1, 2, 3\). The smaller \(i\) is, the better position and gesture the robot has to kick the ball. And \(i\) is determined by comparing the vector of the kick coefficient \(p\) of each robot which is decided by the position and orientation of the robot. Noticing that \(i\) is determined by comparing the kick coefficient \(p\) of each robot, the role of a robot is correlated with the others.
- \(M\) represents the tactic selected by the operator.

**TABLE I**

<table>
<thead>
<tr>
<th>(i)</th>
<th>(A=0)</th>
<th>(A=2)</th>
<th>(A=2&lt;) (A=1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>offense</td>
<td>offense</td>
<td>offense</td>
</tr>
<tr>
<td>2</td>
<td>offense</td>
<td>offense</td>
<td>offense</td>
</tr>
<tr>
<td>3</td>
<td>assist</td>
<td>defense</td>
<td>assist</td>
</tr>
</tbody>
</table>

\*A1, A2 are constants.
The change of the match state will change the role of the state, and Table I is an example. It shows the state transition of all the 3 robots with the change of $A$ when $d<d_1(d_1$ is a constant), $I$ intersects with $x+$, and $M =$ offense-defense balanced. All the transitions of the state can be described by such tables and the transition function is thus determined.

C. Behavior selection

For each role, the robot has the ability to execute some basic behaviors. Here we use the attack role to illustrate the behavior selection process described by the hybrid automaton.

When a robot assumes the attack role, it will execute the basic behaviors which will drive the ball into the opponent’s goal. These basic behaviors include chase, pass and shoot and they can be selected as the control modes of the hybrid automaton. An initial state is introduced here and when the robot comes into the role, it will first enter the initial state. And a final state is also introduced. When the robot changes its role, the robot will first enter this state and then enter another role’s initial state. So the control modes can be represented by $H = \{\text{initial, chase, pass, shoot, final}\}$. The basic behaviors encapsulate the dynamics of the robot, and for each mode in the automaton, there are a series of differential equations inside the mode and these equations determine the translation velocity $v$ and rotation velocity $\omega$ of the robots. So the continuous variables in the automaton is $(v, \omega)$, and a state of the hybrid system can be represented by $(q, v, \omega)$, where $q \in H$.

![Figure 4 Sketch of state transition in attack role](image)

When a robot takes the attack role, it starts from the initial mode and then with the input of the automaton, it will enter one of the three modes. When the robot is not near enough to the ball, it will chase the ball and when it gets near enough, it will pass or shoot the ball according to the situation of the match. When the robot completes the pass or shoot behavior, it will chase the ball unless it changes its role. So the input of the automaton is the position between the ball and the robot and the position of the opponents’ goal. The sketch of control mode transitions is shown as Fig. 4. The dynamics in each mode and the transition conditions are not marked for convenient.

V. IMPLEMENTATION AND EXPERIMENTS

In this section, we will described the system setups and present some experiments.

A. System setups

Fig. 5 is the sketch map of the setup for the system.

![Figure 5 System setup of the system](image)

The robots are two-wheel driven mobile robots and each is $12\text{cm} \times 12\text{cm} \times 8\text{cm}$ in size and $1200\text{g}$ in weight. The match field is $220\text{cm} \times 180\text{cm}$ and a tennis ball is used as the ball of the match. A SSC-131 camera is located on top of the field and the video it collects is processed by a MY VISION image card. Two computers are used as video server and control server separately. The video server sends the video captured by the camera to the client via the Internet in real time while the control server accepts the commands from the client and sends them to the robots for execution. The control server also process the image captured by the camera to get the data of the match and feedback the data to the client. At the remote site, the client computer sends the commands to the control server and a Logitech joystick can be used as part of the command generator. The client computer also receives the information from the match site and displays the information to the operator. The clients and the server are connected by the Internet. Each operator can direct a team of robots on the remote site by a client and 2 clients can be connected to the server at one time. The control software program used in the system is based on the client/server mode, and is realized by windows socket programming.

B. Experiments

Fig. 6 is a typical sequence of behaviors executed by one robot team in the system. Since no opponents in the match field, the match state $A$ is fixed to a big positive constant, which means the team is absolutely prone to control the ball. And $d_1$ mentioned in Table I is set to be $110\text{cm}$, which is the distance between the middle line of the field and the goal. In Fig. 6(a), according to the dynamical reference frame, the match state is just the same as shown in table I. In this situation, Robot 1 has the best kick coefficient and Robot 2 has the worst kick coefficient. So Robot 1 and 3 take the attack role and robot 2 takes the assist role. The two attack robots chase...
the ball for the ball is not near the robots. And robot 2 turns and moves towards the penalty area. In Fig. 6(b), Robot 1 is near enough to the ball, and it changes its behavior from chase to shoot. Fig. 6(c) shows the final match state when the ball is in the goal.

The operator can see the match information from the graphic interface and select strategy for the robots. Also, the operator can choose one of the robots to direct by a joystick at any moment in case for conflict. And a reset button is designed to reassign all the robots to the initial static state when the goal is in or when an emergency appears.

In the future, the system would be further improved by the hierarchical hybrid automaton model. More complicated case would be considered and the operator’s commands, which have been delayed in the system, will be lucubrated.

VI. CONCLUSION

This paper presents a model for behavior coordination of the multi-robot system directed by one human operator through the Internet. The system is behavior-based and has ability to understand the operator’s commands and cooperate to achieve tasks by executing proper behaviors. The architecture of the Internet-based multi-robot system is presented and analyzed and then a hierarchical hybrid automaton model is proposed to describe the process of task assignment and behavior execution. The model is applied to a 3v3 Internet-based tele-robot soccer system and experiments have been done to explain the model proposed in this system.

REFERENCES