Robot Behavior Planning by a Coordinator Robot in a Symbiotic Autonomous Human-Robot System

Tao Zhang, MD Hasanuzzaman, Vuthichai Ampornaramveth and Haruki Ueno

Intelligent Systems Research Division, National Institute of Informatics,
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan

Abstract. This paper addresses robot behavior planning by a coordinator robot in a symbiotic autonomous human-robot system according to human requests. A coordinator robot is constructed based on the knowledge representation of coordination which is a description world to describe a physical domain world on coordination, including interpreter for the human request domain, coordination interface for the problem domain, coordination policy for the solution domain, etc. The coordination policy in the model is adopted for planning robot behaviors by a coordinator robot. By means of a software platform and distributed intelligent agents, a coordinator robot can be implemented and the robot behavior planning can be carried out according to human requests. In this paper, a coordinator robot is actually constructed and it can autonomously plan robot behaviors in a symbiotic autonomous human-robot system. The experimental work demonstrates the effectiveness of the proposed method.

1 Introduction

In modern society, coordination is definitely essential for performing complex tasks. Making an effective plan is one of important aims of coordination. Originally, a human coordinator usually plans the activity of people. With the development of the robotic technology in recent decades of years, a human coordinator extends its plan to the activity between human and robots. Toward the future society, an intelligent robot is required not only to be able to perform tasks, but also can make high-level decision by means of knowledge like a human being. Such a robot is particularly required in the symbiotic autonomous human-robot system which is proposed recently [1]. In this system, various types of robots coexist with human at the places of everyday life, communicate with, and aim in helping human. A human person in this system is an ordinary member without rich knowledge on robots. But he/she can acquire services from different robots along his/her requests expressed to robots by human-robot interface. Each kind of tasks can be cooperatively performed between human and robots or among robots only. The robots in this system have different intelligences. Some robots called coordinator robots can interact with human through human-robot interface and understand human requests. Other robots called specific robots can only perform tasks without ability to interact with human. Therefore, the planning of robot behaviors for various tasks is entirely performed by a coordinator robot.

Zhang T., Hasanuzzaman M., Ampornaramveth V. and Ueno H. (2005).
Robot Behavior Planning by a Coordinator Robot in a Symbiotic Autonomous Human-Robot System.
In *Proceedings of the 1st International Workshop on Multi-Agent Robotic Systems*, pages 72-79
DOI: 10.5220/0001159500720079
Copyright © SciTePress

Many researches on the planning of robot actions have been carried out for many years. For instance, Ingrand [2] proposed extending procedural reasoning for robot actions planning. Lee [3] adopted learning momentum for planning the robot behavior in a robotic team. Learning momentum is a variation of parametric adjustment methods. Lopes [4] presented the architecture and algorithms developed for Dorn Dinis, a simple compass-guided robot. In this research, the robot is able to plan its tasks based on the (possibly incomplete) map. Although these methods are effective for planning robot actions under some certain conditions, they are lack of the ability to plan robot actions flexibly concerning various human requests. That is, when a person freely expresses his/her requests to robots using natural manners without any knowledge on the operation of robots, a coordinator robot can understand the scenes and autonomously plan robot actions for human requests.

In our research, a coordinator robot is constructed based on the knowledge representation of coordination. The coordination which would be carried out by a robot in a symbiotic autonomous human-robot system concretely represents the solution of problems that occurred when robots trying to coordinate the different kinds of services offered by human or robots according to human requests. The robot in charge of coordination should possess the functions of understanding human requests, clarifying problems in the coordination, providing the solution and maintaining the high belief degree for its decision. With the definition on coordination, a new knowledge representation of coordination called a description world is proposed for the physical domain world on coordination. Based on this model, a coordinator robot is implemented by means of a software platform and distributed intelligent agents. Since the model includes the important coordination policies for planning robot behaviors, a coordinator robot can use them to plan robot behaviors according to various human requests. In this paper, a coordinator robot is actually constructed and robot behaviors for different real tasks are planned by this robot. The experiment is carried out to demonstrate the effectiveness of the proposed method.

2 Knowledge Representation of Coordination

2.1 Description World on Coordination

Coordination is existed in such a kind of physical domain world, which consists of human request domain, problem domain, solution domain and sensor domain. In the human request domain, a human person can express his requests by communication. Based on human requests, problems will occur when coordinate the required services and be expressed in the problem domain. In the solution domain, the solution of the problem is given. Proper behaviors or activities of human and robots will be then carried out in the sensor domain. Actually, this process of coordination is easily fulfilled by human based on human knowledge and experience. However, if we expect a robot to implement this process in a complex system, it is very difficult not only due to the insufficient functions of hardware, but also short of strategies for robot to coordinate different tasks. In this paper, a knowledge representation for coordination is proposed in order to construct a coordinator robot, which can generate strategies for implementing coordination.

A description world is built for the physical domain world on coordination. Concerning human requests, in the proposed knowledge representation there is an interpreter to interpret human requests given by various patterns, such as language, writing, gesture, etc., and transform them into the pre-defined format. By interpreter, it should get the information on the object, required services, change of events and properties specification so that the coordination model can be correctly constructed. For the problem domain, various kinds of coordination interfaces are defined to describe various kinds of information related to the problem. Coordination policies, corresponding to the solution domain, are defined to provide effective policies for solving problems. Based on coordination policies, controllers for the sensor domain are developed to perform the tasks according to human requests. These items can form a hierarchy structure of the description world. The top is the interpreter for human requests and the bottom is the controller for sensor domain. This hierarchy can represent the process of coordination like the domain world. In addition, it can represent the relations among components of the description world. In the following sections, as the core of the description world, the coordination interface and the coordination policy are respectively defined in detail.

2.2 Coordination Interface

The definition of coordination interface is to describe the problem domain in the physical domain world. In order to identify and understand problems, declare what a human person is expecting in the way it has been designed to control it, so that to find out good solutions for the problems, coordination interface is defined for the problem domain. The coordination interface consists of concept definition, services and events assignment, and properties description. The meanings of them are explained as below.

- **Concepts**: represent the objects of the problem domain offered from human requests. It was indicated by the name of the coordination interface. It has many types, such as human, animal, plant, instrument, furniture, image, etc.
- **Services**: identify operations that the domain must provide for the human person to invoke. The types of services have action, speech, writing, etc.
- **Events**: identify state transitions that the human person needs to be able to detect in the problem domain, such as spatial position, temporal state, shape, component, etc.
- **Properties**: illustrate the kinds of assumptions that can be made and the implications that they have, such as logical relation, restriction, conflict, scope of problem, etc.

2.3 Coordination Policy

The definition of coordination policy is to describe the solution domain in the physical domain world. The components of coordination policy comprise of required coordination interface, attributes of the policy, arguments, and coordination rules. The meanings of them are explained as below.

- **Coordination interface**: identify and understand the problem, declare what the human person is expecting.
- **Attributes**: declare features that are local to the policy itself that facilitate the definition of the scheduling of the different rules, such as actions, status, results, tasks, etc.
- **Parameters**: declare some values which would be adopted for describing coordination rules, such as numerical, logical, variable, functional parameters, etc.
- **Rules**: describe the schedule of the behavior of the solution domain. The rules are given by three items: (1) "when": specify the trigger of operations; (2) "with": specify the condition defined over the internal state of the policy; (3) "Do": a set of operations, each of which is either a service or an update.

3 Robot Behavior Planning by a Coordinator Robot

3.1 Modeling for Robot Behavior Planning

Based on the knowledge representation of coordination, a coordinator robot can be constructed and robot behaviors can be planned for different tasks. In this paper, an example is used to explain how to model the coordination of tasks which including the coordination polices for planning robot behaviors. The example discussed here is about a case of activity between an elderly or disable person with robots.

In an elderly or disable person's daily life, several basic activities are expected to carry out by various types of robots, as give in table 1. For instance, a coordinator robot "CoRobot" can greet a person, recognize a person and get person's name. A mobile robot "MobRobot" can move to a specific place along person's requests and bring something for a person. An entertainment robot "EntRobot" can play with a person.

Table 1. List of tasks for human requests

No.	Tasks	Executors
1	Greet a person	CoRobot
2	Recognize a person	CoRobot
3	Get person's name	CoRobot
4	Move along person's request	MobRobot
5	Bring something for a person	MobRobot
6	Play with a person	EntRobot

These tasks are first interpreted by the interpreter defined in the coordination model and transformed into the format to express these tasks. Then, many coordination interfaces and coordination policies are defined, as listed in table 2. In table 3, the coordination interface "EntRobot" for an entertainment robot is defined. It includes services, events and properties. The services possibly provided by an entertainment robot consist of "onSit", "onStand", "onLie", "onWalkForward", "onWalkBackward", "onTurnLeft", "onTurnRight", etc. The events of robot include free and busy. The properties of robot consists of "(onStand before onWalkForward) and (Free)", "(onLie before onSit) and (Free)", etc. With using this interface for an entertainment robot, many tasks for the activity of playing can be possibly fulfilled by use of currently developed entertainment robot.

Table 2. List of coordination interfaces and policies

Coordination interfaces	Person, Environment, Timer, CoRobot, MobRobot, EntRobot …
Coordination Policies	Greeting, recognition, GetHumanName, Moving, BringThings, PlayingWith …

Table 3. Format of coordination interface

Items	Contents
Name	EntRobot
Services	onSit, onStand, onLie, onWalkForward, onWalkBackward, onTurnLeft, onTurnRight …
Events	Free, Busy
Properties	(onStand before onWalkForward) and (Free) (onLie before onSit) and (Free) …

In table 4, the coordination policy "Greeting" is defined. It includes interfaces, attribute, parameter and rules. The interfaces for this policy consist of "CoRobot" and "Person". The attribute of this policy is action. There is no parameter for this policy. The policy includes several rules, expressed by when-with-do format. With use of this policy, the tasks for greeting can be possibly fulfilled by the current developed communication robot as well as its additional functions for recognition. The behaviors of the communication robot can be planned based on this policy.

Table 4. Format of coordination policy

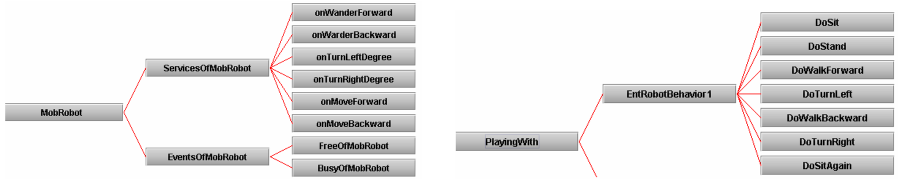
Items	Contents	
Name	Greeting	
Interfaces	CoRobot, Person	
Attribute	Action	
Parameter	None	
Rules	When	Person.onState(Free)
	With	CoRobot.onState(Free)
	Do	CoRobot.onFirstMeet and CoRobot.onGetHumanName …

3.2 Definition in Software Platform

For implementing a coordinator robot, the knowledge representation of coordination is defined in a software platform by which the coordinator robot can coordinate various tasks and plan robot behaviors. A software platform has been developed recently, called Software Platform for Agents and Knowledge Management (SPAK) [5]. This

SCITEPRESS Science and Technology Publications

platform provides a central module, which acts as blackboard, knowledge processing brain, memory, and do the judgment, task planning and execution. Frames and their slots in SPAK are adopted to define the knowledge representation of coordination. Connecting with the root of the model are frames of “HumanRequests”, “Interfaces”, “Policies”, etc.



(a) Interface frame module for “MobRobot”   (b) Policy frame module for “PlayingWith”

Fig. 1. Definition of coordination interface and policy in SPAK

For each coordination interface, an interface frame module is defined, respectively including frames for services and events, slots for properties, etc. Figure 1(a) illustrates an interface frame module for interface “MobRobot”. For each coordination policy, a policy frame module is defined, mainly including Do frames and condition slots, special slots, etc., corresponding to the rules using various services defined in interfaces, parameters, etc. Figure 1(b) illustrates a policy frame module for policy “PlayingWith”. Besides, the execution of these frames for coordination is by use of the inference engines defined in SPAK. The inference engines are for doing forward and backward chaining [5]. In SPAK, frames and their slots are written by use of XML format. The example of the definition of frames in SPAK by use of XML format can be found in our previous paper [6]. Corresponding to XML file, there is an interpreter to translate XML specification into relative commands.

4 Experiment

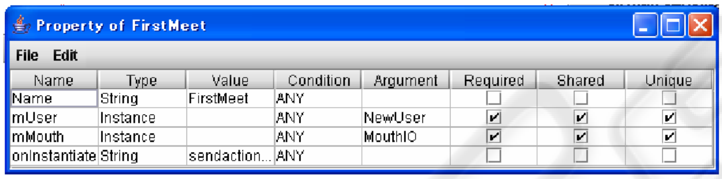
4.1 Experiment Setup

In order to demonstrate the effectiveness of the proposed method for planning robot behaviors by a coordinator robot, the experiment has been carried out by use of actual robots. The experimental system is mainly comprised of SPAK, intelligent agents and three types of robots: (1) Robovie: a humanoid robot to play the role of coordinator robot, called “CoRobot” in the example, (2) Scout: a mobile robot called “MobRobot”, (3) AIBO: an entertainment robot called “EntRobot”.

All robots are connected with SPAK via wireless TCP/IP network. Since the server computer of Robovie is embedded into the body of Robovie, but SPAK is a platform to manage the knowledge for coordination, SPAK must install in a stand-alone server computer with intelligent agents. The agents include a server agent, a user agent, three robots agents, a sentence parser agent, etc. A sentence parser was developed for interpreting human language. A user agent represents each user on the system, relays commands from the user to other agents, queries states of the robot agents, and provides the user with enough feedback information. A robot server represents each robot under control.

4.2 Results

According to the knowledge representation of coordination, a coordinator robot is defined in SPAK by use of XML format. Each frame is represented by a click-able button. Figure 2 is a slot editing table for the frame "FirstMeet". Each row represents a slot in this frame. For this frame, if two instances ("NewUser" and "MouthIO") are set up, this frame will be activated and execute the JavaScript codes written in "onInstantiate" slot. In this slot, special functions "sendaction()" for Robovie action and "sendmsg()" for Robovie speech are defined as the values of this slot.



Name	Type	Value	Condition	Argument	Required	Shared	Unique
Name	String	FirstMeet	ANY		□	□	□
mUser	Instance		ANY	NewUser	☑	☑	☑
mMouth	Instance		ANY	MouthIO	☑	☑	☑
onInstantiate	String	sendaction...	ANY		□	□	□

Fig. 2. Slot editing table for the frame "FirstMeet"

Through interaction between Robovie and users by means of speech and gestures, human requests can be obtained by SPAK. The relative frames for coordination interfaces and coordination policies in SPAK will be activated and the robot behaviors can be planned and conducted. For instance, the task of "Play with a person" can be planned as the following process. In the frame "Policies", the frame "PlayingWith" includes several behaviors of "EntRobot" AIBO for playing. If the coordination policy for behavior No.1 is selected, the sub-frames of frame "EntRobotBehavior1" are selected as the rules for planning behaviors of AIBO, as illustrated by Figure 1(b). Based on these Do frames, SPAK can create instances of these frames in which contain the command to AIBO actions. When creating these instances, a lot of information stored in SPAK can be used to adapt to the variable situations. For instance, when AIBO should move to somewhere, SPAK can plan the routes of movement. Finally, the instances of Do frames can form a robot action series so that AIBO can perform the task "Playing with a robot" by its actions.

With this approach, all tasks given in table 1 can be performed. In the future, more complex tasks in a symbiotic autonomous human-robot system can be implemented with the improvement of functions of intelligent robots.

5 Conclusions

A new robot behavior planning method by a coordinator robot is proposed in this paper for a symbiotic autonomous human-robot system. A coordinator robot is con-

SCITEPRESS
Science and Technology Publications

structed based on a description world used to model a physical domain world on coordination. According to coordination policies, robot behaviors can be planned according to human requests. By means of SPAK and distributed intelligent agents, a coordinator robot can be implemented and robot behaviors can be planned. The experiment demonstrates the effectiveness of the proposed method. The constructed coordinator robot and its function for planning robot behaviors will be great significant for realizing advanced symbiotic autonomous human-robot system in the future.

References

1. Ueno, H.: A knowledge-based information modeling for autonomous humanoid service robot. IEICE Transactions on Information and Systems, E85-D (2002) 657-665.
2. Ingrand, F., and Despouys, O.: Extending procedural reasoning toward robot actions planning. Proceedings of IEEE International Conference on Robotics and Automation. 1 (2001) 9-14.
3. Lee, J. B., and Arkin, R. C.: Adaptive multi-robot behavior via learning momentum. Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. 2 (2003) 2029-2036.
4. Lopes, S., Lau, L. N., and Reis, L. P.: Intelligent control and decision-making demonstrated on a simple compass-guided robot. Proceedings of the 2000 IEEE International Conference on Systems, Man and Cybernetics, Nashville, Tennessee, (2000) 2419-2424.
5. Ampornaramveth, V., Kiatisevi, P., and Ueno H.: SPAK: Software Platform for Agents and Knowledge Systems in Symbiotic Robots. IEICE Transactions on Information and System, E87-D (2004) 886-895.
6. Zhang, T., Hasanuzzaman, Md., Ampornaramveth, V., Kiatisevi, P., and Ueno, H.: Human-robot interaction control for industrial robot arm through software platform for agents and knowledge management. Proceeding of 2004 IEEE International Conference on Systems, Man and Cybernetics, Hague, Netherlands, (2004) 2865-2870.