Grid-Based Robot Control

S.D. Wu[1] and Y.F. Li[2]

[1] Dept. of Manufacturing Eng. and Eng. Management, City University of Hong Kong, Kowloon, Hong Kong s.d.wu@student.cityu.edu.hk
[2] Dept. of Manufacturing Eng. and Eng. Management, City University of Hong Kong, Kowloon, Hong Kong meyfli@cityu.edu.hk

Abstract: Grid, as the new generation Internet information infrastructure, presents a new solution for robot control. In this paper, we present our research on grid-based robot system which aims at organizing and integrating robot's control capabilities to construct a RCG(Robot-Control-Grid), which can provide control ability for robot tasks. Not only single robot can be connected to the RCG, but also multiple robots located in different places can collaborate in the RCG to perform distributed robot tasks, which can provide more powerful control ability. In the RCG, portals corresponding to various robot systems are designed to user for robot accessing and controlling. This paper primarily consists of the analysis of grid techniques, the key issues for constructing the RCG, and an experimental grid-based robot control instance. The experience shows that the RCG is a feasible and effective mode for robot's work and management.
Keywords: Grid, robot, OGSA, grid service, collaboration, security, real time

1 Introduction

The most important part of a robot system is its controller. Nowadays an obvious technical trend of robot system is to construct controller applying open PC software and hardware infrastructure, which can take full advantages of the relevant PC techniques to enhance robot controller's performance effectively. Grid, as the new generation Internet information infrastructure, has been researched widely with much attention in a variety of fields[FKT01],[DCL02]. In terms of robot control system, grid also emerges as a new solution with great potential.
Grid is an integrated computing and resource environment[FC98] and it has strong ability in soaking up various computing resources such as computer and other kinds of network devices, which are transformed to be easy-available, reliable and standard as well as economical computing abilities. A more complete grid definition can be found in the reference[BDG04]. DIS(Distributed

Instrumentation System) is a representative grid application field. DIS manages the expensive instruments located in different places, and presents the corresponding interfaces for remote control and accessing, which greatly promotes the using-efficiency and is convenient to users. There is a famous DIS project Xport[MBH00] which is developed for the shared instrumentation laboratory for macromolecular crystallography. Similarly, robot is also a kind of complex instrument, so that the idea of constructing a robot control grid is feasible and valuable.
Firstly, this paper introduces grid and the relevant essential techniques such as OGSA and Globus, especially, highlighting their technical foundation for constructing the RCG. Secondly, grid-based single robot work model and multi-robot collaboration model are proposed, and the grid service technique and its communication/collaboration mechanism are discussed. Thirdly, the user portal of the RCG for submitting tasks is designed. Fourthly, the security and real time performance of the RCG are studied and analyzed. Finally, taking PT series industrial robots as control objects, and applying famous grid platform-Globus, this paper constructs a grid-based robot control instance under the basic architecture and characteristics of the RCG.

2 Grid and Its Technical Foundation

The robot control has its own characteristics and flows. Hence, a basic precondition to construct the RCG is that grid should do be able to provide technical support and foundation according to robot system's specified demands.
On one hand, what user respected is to take robot to perform and complete certain works. On the other hand, for robot, user's all demands can be deemed as some different kind of tasks. So the robot controller, the core of robot system, should not only be able to provide a control interface to the user, but also contain related logical functions modules which are responsible for conducting the tasks submitted by user.
Under this circumstance, the key of the RCG is to construct robot controller based on grid infrastructure. An effective work mode of the RCG is service-based robot control architecture, that is to say, user submits tasks to service, and then service executes the tasks, which drives robots running. The OGSA(Open Grid Services Architecture), characterized by grid service, provides great support to build service-based robot control architecture for the RCG[FKN02].
OGSA is the latest grid system architecture, which is proposed by integrating Web Service technique. The service in OGSA is call grid service. The essential support technique of OGSA is Globus, which has been the grid technical solution accepted widely by science computing and engineering applications, which presents the GT(Globus Toolkit) to support developing grid-based applications. As an open and stand architecture, GT resolves well the key technical issues about toolkits, services and applications in grid environment. So

GT is one of the most effective developing toolkits for building grid-based applications.

In OGSA framework, anything is abstracted as grid service. Under this concept, a robot connected into grid is also a kind of grid service, which is very favorable to take grid service as a universal interface to manage and use robot. Hence, in the RCG, grid service plays a very important role, which acts as the interface image of grid capabilities, and all robots' control functions are implemented and deployed by grid services. Some characteristics of grid service possessed are of great help to meet the robot control's demands, for instance, grid service is stateful, which is useful to keep the medium information during robot running. Another important feature is that grid service is able to change dynamically such as expending or shrinking, this means that when executing some complex tasks, the required more advanced and abstract robot service can be formed by combining multiple simple and basic robot services, which is a great advantage for the collaborative control of multiple robots. In addition, the service data and notification are two key mechanisms of grid service, which support the communication and interaction of multiple robots or grid services.

3 Model of Robot-Control-Grid

According to the introduction above, in the support of grid service, this paper proposed grid service-based architecture model for the RCG. Since the core factor of grid service, the control to robot mostly is incarnated as the managing, scheduling and utilizing of robot grid services. The following is a simple analysis of single robot control model and the discussion of the collaboration mechanism of multiple robots.

3.1 Model for Single Robot

Generally, a robot control system is complex, including multiple modules for different functions. In order to give prominence to demonstrating the service-based architecture, here firstly simplify the robot control system moderately to care for only the robot motion control function, and then have this function mapped to a grid service to implement. The running of this grid service will drive robot motioning. In this way, a basic principle is that a robot is controlled only by one grid service. A robot is acting as grid resource, and user can submit task to the grid service through a user portal when desiring to control robot. Meanwhile, user can monitor the running of grid service and perform starting or stopping such kind of operations to control robot when necessary. The control model for single robot in the RCG is illustrated as Fig. 1.

In the model above, the solution of one robot being controlled by only one grid service shows the basic grid service driven robot control principle in the RCG. In fact, a robot system maybe consists of multiple grid services. For

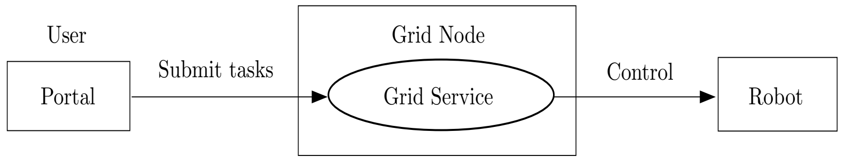

Fig. 1. Control model for single robot in the RCG

instance, besides the most important robot motion control service, the motion simulation grid service is also very important. Upon that, one grid service is responsible for motion control and another grid service is responsible for simulating and, the data used in simulating comes from the motion control grid service, which requires enabling communicating and interacting among gird services. Here the collaboration of grid services in the RCG is involved naturally.

3.2 Multi-Robot Collaboration Model

The collaboration of multiple robots is an important problem should be resolved, because there must be multiple robots in the RCG. Fortunately, the characteristics of grid environment present great advantages to robots' collaborative control. Although the control relations for multiple robots looks complicated, according to the basic principle introduced above, the collaboration of multiple robots can be translated naturally as the collaboration of multiple grid services in grid environment, which is similar in solution with the collaboration of multiple grid services for single robot control. The two kinds of collaboration solutions have their own means respectively. However, they are accordant in the technical way.

For complex robot tasks, it requires the collaboration of multiple grid services. There are two key issues for robot collaboration. One is how to decompose complex tasks and distribute the pieces of subtasks. The detailed guidelines depend on task's properties and robot's motion algorithms. For instance, for instruction-driven industrial robots, each robot's subtask can be separated relatively easily from the collaboration task. The other issue is the collaboration rules when multiple robots collaborate. Although there are no general rules, there exists a basic rule called "waiting-synchronizing": on the process of multiple gird services running, the starting of a service's instruction may require another service completing a specified instruction. Thus it has to wait for that grid service to synchronize. Usually, the user can custom the collaboration relations among grid services when submitting a task.

After resolving the collaboration mechanism, the grid service's communication, the base of the collaboration, is another important problem should be settled subsequently. Globus presents registration and notification based communication techniques for grid services. If a grid service wants to communicate with another grid service, it should register to that service beforehand, and

then it will be able to receive the notification message from that service. Because the PUSH technique is used in sending the notification message to the registered services, the notification communication mechanism shows a higher efficiency.
According to the introduction above, the grid services used for single or multiple robots control both require communication each other. For each grid service participating in robot control, according to its function and the role played in robot collaboration, it needs to register to the grid services from which it desires to get interested information. At the same time, it may accept other service's registration and send messages they required to them. Here two circumstances are taken to analyze the grid services' collaboration. One is for the communication of grid services for single robot control and, the other is for multiple robots control. For the former, for the motion control grid service and simulation grid service, when the motion control grid service has completed sending pulse to robot control card in an interpolation cycle, it should send current robot joints position data to the simulation grid service to conduct simulation. Obviously, in this situation the communication content is large. For the latter, the collaboration of multiple robots, when a grid service for a robot completes an instruction or a piece of subtask, it may be required to send notification message to another gird service which is responsible for another robot. When this message arrived, the grid service will decide what next step to do according to the information contained in message. For instance, if it knows that the partner robot is running ok, it also will execute the next instruction orderly according to the collaboration rule configured in advance. This kind of communication load is low in some extent because the transmission content mainly consists of specified commands and signals.
Besides the communication of grid services, the communication demands between grid service and portal client is also indispensable. Certain messages in the robot control process should be delivered to the portal end to provide states information to user. This kind of communication also can be conducted by registration and notification mechanism. Fig.2 is the collaboration relations illustration of multi-robot control.

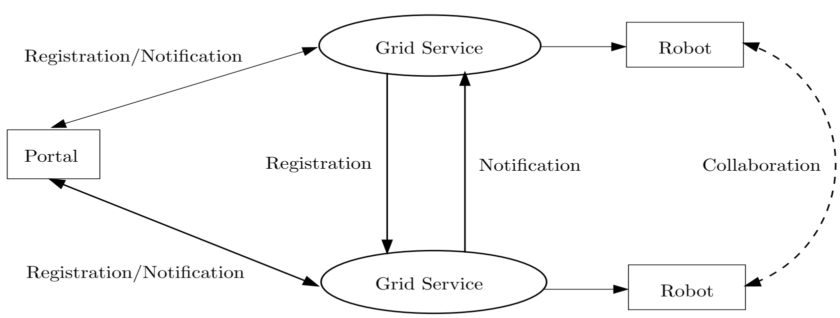

Fig. 2. Collaboration relations of multi-robot in the RCG

The other issue of communication is the format of communication message, which can be resolved finely with service data techniques. Service data is an important feature of grid service, which can not only organize the basic information of grid service, but also define data format for communication content according to the demands of concrete application. The registration of grid service, actually, is to subscribe the desirable SDE(Service Data Element). So SDE plays an important role in the communication of grid services. Here the content of SDE in different circumstances in the robot control is listed. For simulating service, the SDE contains mainly the position data of robot joints. For the grid services for robot collaboration, the SDE contains the following items: signals specified for collaboration, which are defined in common and can be understood by all the participated grid services, currently executed instruction, potential next instruction, position data in specified moment, the states of partner robots and any other information related to collaboration. The SDE that the user portal cares contains the global states of robot grid, the current states of each robot, the interaction flows and traces of grid services, the progress of task or subtasks and various error or warning information and so on.

4 User Portal Design

Portal is the user entry to control robot in the RCG, and it provides a universal robot control interface. But robot could be accessed by no other than being connected to the RCG. The connection procedure requires two steps in physical and logical ways. For the former, nowadays most robots controllers are based on PC platform, which ensures the PC used for controlling robot can be connected to the RCG conveniently to act as grid node, and robot is linked with PC through related hardware control structures.

Furthermore, the RCG utilizes a logical connection entry to register the corresponding grid services used for controlling robot, and add these grid services to index service. The feature data and basic information of a robot system is saved in grid service's service data. Hence, by index service, the user can query grid service and compare the feature data, then choose desirable robot and invoke the corresponding grid services to control robot. Here the service data describes the inherent features of robot and grid service, in general, contains the following items: robot's type, joint's physical characteristics, work modes, compatible task formats, efficiency and information related with collaboration, etc.

Because user's all control activities to robot are via the portal interface, the user portal of the RCG should present at least six function interfaces listed below.

- Querying and selecting of robot grid services. For a robot, maybe there are several suits of grid services by which different performance could be used. Thus the portal should give user an interface to query and select desirable grid services, and according to the characteristics of the selected grid service to prepare compatible robot task.

- Submitting robot task. In general, the robot task is in fact a piece of program, which is composed of robot instructions and relevant syntax sentences. Through a task submitting interface, user can submit task program to robots.
- Configuring of collaboration and work parameters. When the task is submitted, before starting control, some robot parameters such as running speed, times and simulating switch are required to setup. Furthermore, a more important step is to configure the collaboration relations among multiple robots and grid services. In some extent, this configuration may be complex. The robots will work under the collaboration rules configured by user.
- Management and monitoring of robot tasks. When robots are running, the user can manage task such as to start, stop, pause, or resume robot task, or turn off the robot controller and so on when interventions are needed.
- Information feedback to user. In the process of robot running or collaborating, the related detail information and key message should be feed back to portal client in time, which make user master the robots' states and provide support for determining whether to take an action.
- Interface for security setting. The detailed security solutions will be introduced in the next section. The portal should provide user with an interface to configuring security options.

5 Security Solutions

A robot is expensive and high-precision device. A robust robot grid environment has to present appropriate security solutions to guarantee robot working securely. Here are the solutions of two aspects of security problems in the RCG. One is the fundamental security demands of grid application such as privacy and integrity, etc. and the other is the security of robot task.

5.1 Security of Grid Application

The security is one of the most important parts of a grid application[WSF03]. For the RCG, the following demands should be considered for constructing secure robot control environment.

Privacy. In the interaction between user and grid services, what is transmitted is the robot task, a set of robot instructions and other codes. In general, there isn't special privacy demand for robot task, because the robot instruction is public. However, if user wishes the robot task keeps privacy, some encryption and decryption algorithms can be selected for use.

Integrity. The task user submitted represents user's specified control intention.

If the task is intercepted and tampered by malicious user in the transmission, the means of the task will change, which is not allowed to happen. So the robot task has to ensure its integrity, which can be guaranteed by digital signature technique.
Authentication. The authentication is a compulsory security requirement for robot grid application. For the robot, grid must ensure the users submitted task are who they claim to be. In other words, a robot should be protected from malicious users who try to impersonate one of the parties in the secure robot control. On the other hand, for robot user, grid must assure that the robot is what user desires to control, then user will really perform controlling actions. So in the RCG, the authentication demand is mutual between robot and user, which can be implemented by digital certificates technique.
Authorization. Based on the authentication, authorization is a higher level security setting. In fact, for the robot control, these authenticated legal users may be authorized different level security rights according to user's role or robot task's requirement. For instance, some advanced grid services may only be allowed to be invoked by specified high level user. The authorization setting can be implemented by the ACL(Access Control List) mechanism.
The four methods can ensure robot running securely in the RCG. For their implementation, Globus presents powerful GSI(Globus Security Infrastructure) to support the working of all the above security methods. GSI possesses several important features such as completed public-key system, mutual authentication through digital certificates, credential delegation and single sign-on, etc. which can effectively guarantee a secure robot control environment in the RCG.

5.2 Security of Robot Task

What user submitted to gird service is robot task code. Before a robot executes a task, whether the program code itself is safe must be considered.
The main composition of robot task is robot's instructions. A robot system's instructions have inherent rules such as limited legal instruction name, so for the task user submitted, robot grid service can prevent illegal robot instructions by syntax inspector via compiling it in advance.
However, the validity of instruction data may be impossible to be checked out by syntax inspector in some extent. If the instruction data are not in the valid range of robot joints' allowed, this potential error may result in damaging the robot hardware. This kind of security problem can be avoided by motion simulating. In the RCG, before really executing robot task, firstly conduct offline simulating with 3D animation to check the validity of instruction data.
The above methods mainly resolve the security problems of robot instruction system. However, robot task program also includes some extended functions, even including some codes which have "write" operation right to the grid service end. In this situation, robot task program is very similar to the program written by traditional high level languages. It may include virus or malicious

codes, which can result in unthinkable damage to robot control system. Mostly, this problem can be resolved in some extent through testing or preventing the "write" operation codes, but grid service can't guarantee it can remove all this kind of security obstacles completely and automatically by itself. Sometimes, it still needs user's help.

6 Real Time Performance Analysis

For robot control, real time performance is an important issue. In general, user requires being able to control robot real time, for instance, when occasional fault happened with robot, user has to turn off the robot controller immediately, otherwise it maybe result in an accident. However, spanning a long distance and the limitation of network bandwidth impacts the real time performance of robot control system[LMG03]. So due to the distributed structure of the RCG, it looks the real time performance may be unable to satisfy an ideal requirement. A simple example, when testing robot or debugging rough robot task in the RCG, a higher real time performance is required.

However, the RCG can still run well, which aims to those reliable and experienced robot tasks. When the rough tasks are running for the first time, they can be improved and trained through simulating, and then they become experienced tasks. During the simulation, even if some mistakes happened, it won't make any damaging to the robot body. When the experienced task is submitted to robot, there will not be anything wrong with the robot, and it can run smoothly and reliably, as long as the robot body is able to follow the commands and there is no mistake on the robot hardware. Actually, the user can get the robot's simulating views or monitoring video, by which user can take appropriate actions in emergency. Also in most cases there is operator in the local spot of robot working, so robot can be stopped in time when mistakes happened on robot hardware. In this situation, the real time performance of the RCG doesn't matter much to some extent.

7 A Simple Demo

According to the introduction above, taking PT series industrial robots as control objects, this paper constructs a grid-based robot control instance under the basic architecture of the RCG. PT series robot includes 32 instructions, including motion instructions such as PTP, MOV and CIRCLE, and program control syntax instruction such as FOR, WHILE and RETURN, and welding and conveying used instructions.

In our previous work, we have built the basic control framework for PT series robots which can run well locally[WCH01]. To construct grid-based robot control system, we take the robot instruction interpreter, which directly drives robot motioning, and motion simulating functions as samples, implement them applying grid service in Globus platform. By collaboration, these two grid services make up of a little-scale robot control system in grid environment. The interpreter is the kernel of PT series robot system, and the interpreter grid service is responsible for interpreting and executing the robot program submitted by user. Because the program is composed of instructions introduced above, the implementation of every instruction is the most important foundation for the interpreter services. The simulating grid service is to trace and visualize the robot's motion process via 3D animation with light, in which the simulating data, robot joints' coordinate position, comes from the sensors when instruction interpreter service is running. After completing and deploying these two grid services, the user can submit compatible tasks to the interpreter grid service to control robots. Fig. 3 is the simulating illustration of PT500 robot with different view angles and the collaboration of two robots. For more detailed control principle and methods related with the controller of PT series robots, please refer to the reference [WCH01]. The programming and implementation of grid service can be referred to the Globus technical documents.

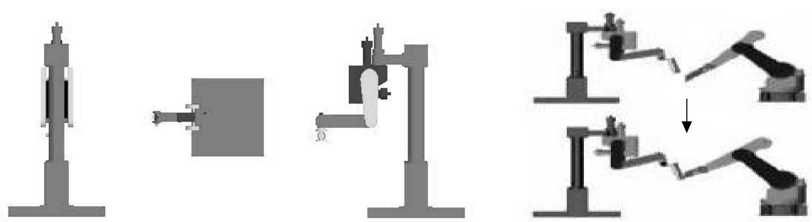

Fig. 3. Simulation views of PT500 and PT600 robots

The following is the introduction of the demo of client portal, as illustrated in Fig. 4. The functions of the buttons are indicated by their captions. Basically, the portal program presents three kinds of functions.

- Select and submit robot task. The user can select a task program and submit it to the interpreter grid service. The Select submenu in File menu will open a file operation dialog and display all the available task files.
- Task control. The user can control robot task's running by the buttons of Start, Stop, Pause, Resume, and set the work mode to be Step, Continue or Cycle.
- Hint of feedback information. The user can open the status feedback dialog to show control related information, by which the robot's working status will be in control of the user.

Fig. 4. User portal demonstration

8 Conclusion

This paper introduces grid and Globus techniques in detail, and discusses the key problems to construct the RCG. Considering the inherent characteristics of robot control, this paper studies how to apply grid techniques to resolve the problems related to robot control, and analyzes the factors influencing robot control system aiming at achieving good robot control performance. Finally a simple robot control system demo is implemented in Globus platform. The work in this paper has proven that the idea, control robot in grid environment, is feasible and valuable, and presents appropriate approaches and solutions for building robot control system in grid environment.

It is not easy to construct a sophisticated robot control grid. Our works are just at the beginning, and surely there are many problems requiring more widely and deeply research. At present, our more experiments and studies are still in progress. For the future work, we will pay more attention to the design of the physical and logical modes of grid robots, and propose stand rules for managing grid robots effectively as well as the practical and powerful collaboration mechanisms for multiple robots, etc.

References

[FC98] I. Foster, C. Kesselman: The Grid: Blueprint for a Future Computing Infrastructure. Morgan Kaufmann, San Francisco, CA (1998)

[MBH00] Donald McMullen, Randall Bramley, John C. Huffman et al.: The Xport Collaboratory for High-Brilliance X-ray Crystallography, tech. report, http://www.cs.indiana.edu/ngi/sc2000 (2000)

[FKT01] I. Foster, C. Kesselman, S. Tuecke: The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International J. Supercomputer Applications, 15(3) (2001)

[WCH01] Wu Shandong, Chen Yimin, He Yongyi: Development Technique of High Level Software System for Industrial Robot Controller Base on WIN 9X/NT, Journal of Shanghai University, Vol.7, No.6, 532-535 (2001)

[DCL02] Du zhihui, Chen Yu, Liu Peng: Grid Computing, Beijing: Tsinghua University Press (2002)

[FKN02] I. Foster, C. Kesselman, J. Nick et al.: Grid Services for Distributed System Integration. Computer. 35(6) (2002)

[WSF03] Welch, V., Siebenlist, F., Foster, I. et al.: Security for Grid services, Proceedings. 12th IEEE International Symposium on High Performance Distributed Computing, 48-57 (2003)

[LMG03] Peter Xiaoping Liu, Meng, M.Q.-H., Jason Gu et al.: Control and data transmission for Internet robots, Proceedings. ICRA '03. IEEE International Conference on Robotics and Automation, Vol 2, 1659-1664, (2003)

[BDG04] Bote Lorenzo, M.L., Dimitriadis, Y., Gmez Snchez, E: Grid characteristics and uses: a grid definition , Proceedings of the First European Across Grids Conference, Springer-Verlag, LNCS 2970, 291-298 (2004)