A Web Services Based Wide-area Corporation Multi-agent System Platform for Power System

Xiaoyang Tong
School of electrical engineering
Southwest Jiaotong University
Chengdu, Sichuan, 610031 China
E-mail: xytong@swjtu.cn

Abstract—**In order to meet the requirements of cooperation and information sharing with different layers among some heterogeneous subsystems in power system, a wide-area cooperation multi-agent platform is proposed based on web service and agent technology. The wide-area cooperation model is discussed firstly; the principles of the framework implementing the platform are expatiated. Some key technologies are studied, such as the design pattern of diagnostic agent, the implementation of web service components, and the negotiation mechanisms based on business event and publish/subscribe. The design and implementation of the prototype for wide-area backup protection in power system illustrate that the platform can provide bidirectional communication for wide-area cooperation agents and greatly improve the intelligence and flexibility of distributed system.**

Keywords—**wide-area cooperation, multi-agent, platform, web service, design pattern**

I. Introduction

In the recent years, the power blackouts in USA, Canada and Europe have made researchers study the cause of calamities with a view to global or regional system. Traditional hierarchical control architecture in power system doesn't fit for new situation. Multi-agent technology is believed as a new solution for distributed system with its autonomous, cooperative, pro-active features. Agent has been applied on the fault diagnosis of power grid, monitoring in substation, wide-area backup protection, secondary voltage control, integrating of information system for power enterprise in Ref [1]-[8].

In order to improve the integrating of distributed heterogeneous system, some traditional software technologies, including DCOM, CORBA and RMI, are used in the operation of the enterprise. But DCOM only works in windows operation system, CORBA needs the support of ORB (Object Request Broker), RMI must run in JVM (Java Virtual Machine). The big problem is that they can't call each other and run in LAN (Local Area Network). Web Service (WS), as a new-generation web technology, uses the standards of HTTP (Hypertext Transfer Protocol) and XML (Extensible Markup Language) to transfer messages. As a result, all kinds of the applications can visit backup service with the common interface of WS through the firewall.

The multi-agent system is proposed to provide preventive and corrective self-healing strategies to avoid catastrophic failures of a power system in Ref [9]. A concept of distributed autonomous real-time system for power system operations based on WS is proposed in Ref [10]; the agents are expected to coordinate local and global information to improve the reliability, stability, and efficiency of the power system. But It wasn't studied how to implement of the system in the paper. Some studies have focused on the integration of WS and CORBA or DCOM. The clients call CORBA object or DCOM component via WS method to visit internal functions of enterprise. Agent has been used to find WSDL file URL in the pointed page to help to filter the invalid registry information in Ref [12].

Web service has its shortages and limits. WS is only a remote object in the internet, its intelligence is limited. The initiative control of the application is in the calling program. WS lacks semantic description and flexible cooperation strategies. Although web client can find the publisher of WS by UDDI, it can't know whether the provider is running and which information the provider can share. The agent has flexible feature based on autonomous and pro-active actions. The multiple agents can work in fundamental or group form to share information and resources with related cooperators to get effective negotiation results.

In this paper, the combination of web service and agent is studied to give a novel solution for cooperative requirement in power system, Web services are designed to transfer messages to solve interoperating among heterogeneous applications across enterprises. Software agents can build high-level cooperating with flexible interaction patterns. A multi-agent wide-area cooperating platform for power system is studied. All kinds of business or programs that run in various operation systems and in different areas can be encapsulated to intelligent agents. Decouple and effective cooperation among the agents can be implemented by WS-based point-to-point communication cross networks.

The wide-area cooperation model is described firstly. The framework of the platform is constructed. Some key technologies are studied, such as the design pattern of business agent, the implementation of web service agent, negotiation mechanisms based on business event and publish/subscribe. The design and implementation of the prototype for wide-area backup protection in power system is given to illustrate the advantages of the platform that greatly improves the intelligence and flexibility of distributed system.

978-1-4244-1674-5/08 /$25.00 ©2008 IEEE

II. THE ARCHITECTURE OF WIDE-AREA COOPERTING PLATFORM

A. The formalization of wide-area cooperating model

In the paper, a wide-area cooperating is defined as an interactive process undertaken by some loosely-coupled agents, which may decentralize in working groups in wide-area ranges. The cooperating model can be described with the functions and the architecture of whole system, the logical relevancies among agents, and the construction of each agent.

A wide-area cooperating C is defined by a 2-tuple C = (A, R), where A is the set of the agents, $A = \{A_i, i = 1, 2\ldots n\}$, R is the set of the revelations among agents, $R = \{R_{ij}, i, j \in N\}$, where R_{ij} represents interacting relation between agent i and j.

The agent is a 5-tuple A= (O, B, F, R_s, I), where:

O means the objectives of the agent;

B represents the believes of the agent, including information variables of the environment and the judgments on them;

F is the functions of the agent that represent its abilities;

R_s is the set of the resources that belong to the agent;

I means the interfaces of the agent, provided for the outer world to invoke.

The dependence between agent i and j is a 5-tuple R = (D, M, P, Q, T), where:

D means the dependences between two agents, controller/ slaver, requester/responder, competitive or consistent;

M is the message list transferred between two agents;

P represents the obeyed protocol during the interacting;

Q means the requirement of communication quality;

T means the trustiness to another agent under special situation.

B. The wide-area cooperating multi-agent platform

A web service based wide-area cooperating multi-agent platform is constructed for the distributed system that may run in the environment of Internet, shown in Figure 1. The agents running in different working groups can asynchronously exchange information by the web service components with bidirectional coupled mode. The cooperating can overcome the limiting of the network, operating system, program language with the superior characteristics of Web service technology. The initiating and interacting of negotiating processes are decided by the intelligent agents.

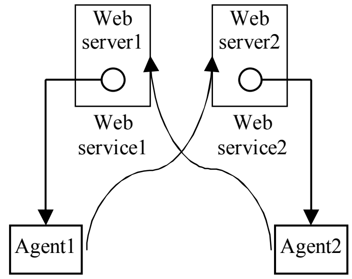

Figure 1. web service based wide-area cooperation for multi-agent system

The implementing architecture of wide-area cooperation platform is given in Figure 2. It's composed of web server, web service components, business agent (BA), login-managing agent (LMA), UDDI center, authored client programs.

Web server is responsible of transferring the request from internet to corresponding web service components. XML, SOAP, WSDL and UDDI technologies are used to insure the messages to transmit through internet and firewall. The security of WS is ensured by the web server. The provider of WS publishes its information to UDDI center. The requestor finds the information from UDDI center, and binds to it. That's working process of WS at present.

The interfaces of web service component provide the methods for outer world; they are designed as the message entrances for the interacting of different programs. In this wide-area cooperating platform, the web interface is separated from its implementation. Some implementation can be coded and executed in the web server, but most of them are finished by

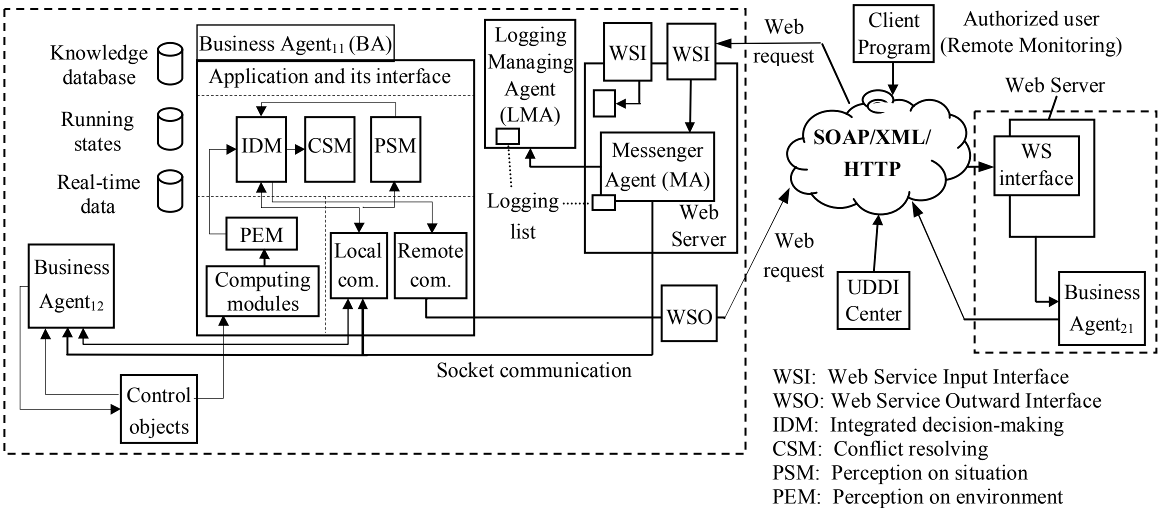

Figure 2. the architecture of wide-area cooperative platform for multi-agent system based on web service

background business agents, designed to improve the intelligence of enterprise's business or control system.

In order to effectively forward and parse the messages from outer agents, a special web service component, named as messenger agent (MA), is designed. It belongs to reactive agent. The functions of MA are described as below:

1) A bridge connecting outer program to the business agent. The main task of MA is to receive the message from Internet.

2) Parsing of the content of message. The information about objective agent is obtained from the message.

3) An adapter. MA can transfer the format of the message to that suitable for the objective agent.

4) LMA running in the web server is responsible for receiving the logging-in requests from BAs while they starts, and records the information into logging-list. MA receives and parses a message to get the address of objective agent, then searches its local logging-list, which is updated from LMA, to check whether the objective agent has started.

5) Some simple tasks can be implemented for the interfaces that are called directly, such as querying data from database.

The business can be processed in web server, or be implemented by DCOM component or CORBA object. But DCOM and CORBA have a limiting of the operation systems and developing languages, so true business is implemented by the agent in the paper. MA transfers the message to BA with socket. MA becomes a client of the BA. In local area network, socket is also adopted for communication among Bas.

III. THE KEY TECHNOLOGIES OF THE PLATFORM

A. *The design pattern for cooperative agents*

Most business agents in power system belong to deliberative type. There are some models in the agent theory, such as BDI by Rao, and ARCHON by Jennings in [14]. On the other hand, in order to improve the reusability and flexibility in object-oriented software, some design patterns that include the roles of the objects and the cooperating relations are proposed for some special problems, such as adapter, mediator, observer, and so on.

The business agents in power system normally use specialistic rules to infer and get the diagnostic results. The programs are hard to design. In the paper, the diagnostic process in agents can be divided into some correlated components or modules, such as perceptron on the environment, decision-making, perceptron on the abnormal situation, action-planning and coordination, basic computing. A design pattern of layered perceiving and decision-making for fault diagnosis agent is studied, shown in Figure 3. The perceptive event based on the basic computing is obtained to start the negotiation among agents. According to local events and the messages from cooperators, the agents get the negotiating results and resolve the conflicts with multiple times under common goals. The real-time data is used to produce the judgments, as well as knowledge and rules from database. The collaborative strategies are adopted to deal with the changes of environment and conflicts among agents. BA plays the roles of client and server at the same time.

In the design pattern, the components of the agent are independent, and can be updated respectively. As a result, the expanding and maintenance of the system will be improved.

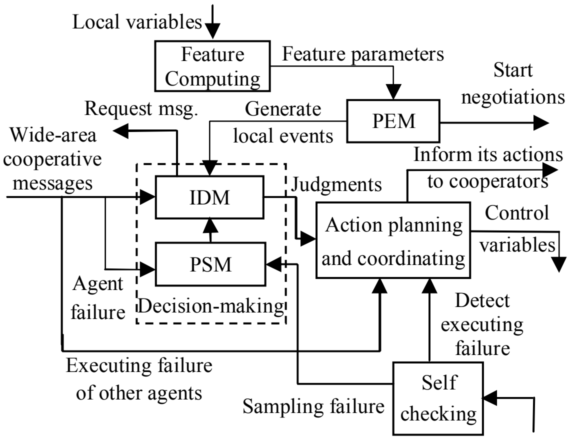

Figure 3. a design pattern of layered perceiving and decision-making for fault diagnosis agent

B. *The design of WS components and its communication*

WS components in the platform include messenger component and other business method. MA, who is a messenger component, is viewed as an adapter for transferring messages to business agents. The key codes of the method recv_msg in MA are shown below:

```
String recv_msg(string msg)
{  String objAgent = Parse(msg);   // Parse the message to get the
                                   // name of objective agent
   // Get its IP address from the agent list
   Struct objIPadr = findobjIP(objAgent);
   // Set up TCP connect to this cooperator (agent)
   IPAddress ip = IPAddress.Parse(objIPadr.ip);
   TcpClient tcpc = new TcpClient( );
   tcpc.Connect(ip, objIPadr.port);    // Connect to objective agent
   NetworkStream ns = tcpc.GetStream();//Definite network stream
   if (ns.CanWrite)     //Check if can write data to this TCP connect
   {   // Change the format of request to bytes
       byte[] wbyte=Encoding.UTF8.GetBytes(msg);
        ns.Write(wbyte, 0, wbyte.Length); // Transmit WS request
   }
   // Read the return result from this TCP connect
   if (ns.CanRead) //Check if can read data from this TCP connect
   {  byte[] rbyte = new byte[tcpc.ReceiveBufferSize];
      ns.Read(rbyte, 0, tcpc.ReceiveBufferSize);  // Read the result
      String r= Encoding.UTF8.GetString (rbyte);//Change to string
   }
   tcpc.Close();  // Close this TCP connect
   return r;         // Return agent's result to client
}
```

When WS method recv_msg is called, the parameter msg is sent. Firstly, the method parses the content of message, gets the name of objective agent, and then sets up a TCP connect to latter. After the business agent deals with the request, the result or a sample response that indicates it has received the message is sent back to MA, and returns back to the client.

The calling of WS triggered by client has two modes, synchronous and asynchronous. For those sample requests to business agents, the client or calling agent can wait for the returning of the result in synchronous way. If the business is complex, or the waiting time is not certain, the asynchronous way must be adopted. In this way, the calling agent only gives a calling to remote agent, and continues to do other thing. The called agent deals with the request, and sends the result via calling agent's WS interface.

C. The negotiation mechanisms of business event and publish/subscribe

The negotiation among business agents is triggered by business events. In the normal situation, the device where agent lives in gets signals from environment and makes feature computing, then gives the feature parameters to PEM of the agent. After some business events are perceived by an agent, the starting message is sent to its cooperators to invoke the negotiations. Each agent receives the messages from cooperators, and calls PSM module. The special events, such as exception, communication or sampling failure, are generated and sent to IDM which does the decision-making.

In order to reduce the communication traffic and make effective negotiation, publish/subscribe mechanism is used. According to pre-define business events, the agents temporarily subscribe interested data or service to the publishers. Once the data is updated in the publisher, it can be sent to the subscribers without their requests. The agility of multi-agent system on the changing of environment is enhanced.

IV. THE DESIGN AND IMPLEMENT OF A PROTOTYPE SYSTEM

A. business flow of wide-area backup multi-agent system

In the recent years, wide-area backup protection (WABP) algorithms in power system based on agent and wide-area communication network become a hot to study in [1][3][4]. The main idea of WABP algorithm in Ref [3] is that the distance protection actions can be exchanged among the agents living in IED to detect the line-fault, the rejection of main protection or breaker, misoperation of the protection. Once one of distance protection Zone1, Zone2 or Zone3 is active, the negotiations among agents are started. When receiving distance protection actions from opposite agent, the agent uses information-integrating strategy, in which the minority obeys to the majority, to find whether there is a fault in this power line. If the decision can't be made, current differential is used to find the fault in the power line, so the network traffic is reduced. The WABP algorithm can accurately find the line-fault, shorten the delay than traditional backup protection, and reduce the range of fault.

The WABP multi-agent systems are constructed in the wide-area cooperating platform. Each agent works in IED of substation; they can negotiate and send the diagnosis result of power-line fault to the host in control center or substation. The cooperative flows of wide-area backup protection multi-agent in the platform are shown in Figure 4. The information is: (1) sampling signal; (2) local event and protection actions; (3) starting of negotiation; (4) messages transferred from messenger agent to objective agent; (5)judgment of this power line fault; (6) simile to (4); (7) sampling failure; (8) exception events; (9) tripping order on breaker; (10)(11)(12) the diagnosis result about power-line fault to cooperators, host and control center; (13) control order from control center. The messages exchanged among WABP agents are shown in Figure 5.

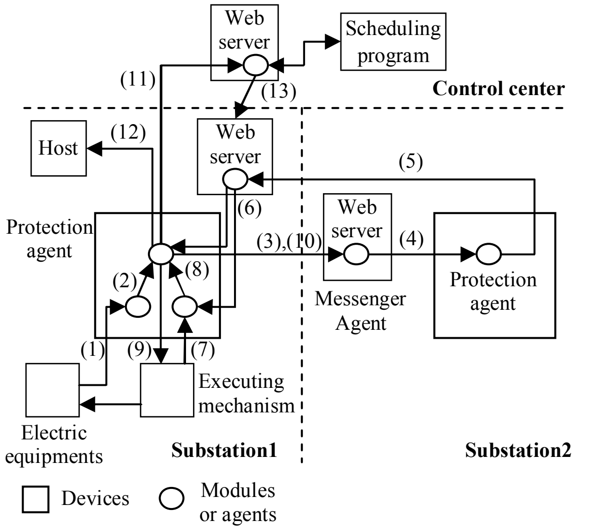

Figure 4. the cooperative flows of wide-area backup protection multiple agents in the platform

The structure of wide-area protection agent class is shown as below:

```
class TestAgent : public Agent
{
 protected:
   double get_time;                    //timestamp of signal
   double PhaseA, PhaseB, PhaseC;  //phases of current
   double Ia, Ib, Ic;                  //amplitude of current signal
   int primary_relay_signal;          //Zone1 protection
   int second_relay_signal;           //Zone2 protection
   int backup_relay_signal;           //Zone3 protection
   int *relay_send_list;              //Cooperators
   bool *relay_line_fault;            //power fault of adjacent lines
   const int path0;             // path0 cooperative statue
   const int path1;             // path1 cooperative statue
   const int path2;             // path2 cooperative statue
   bool flag_line_fault;        // Fault flag of the power line
   bool Curr_Diff_use;          // Flag of making current differential
   bool Sample_Fault;           // Sampling failure of the agent
   bool Trip_Fault;             // Tripping failure of the breaker
   bool Comm_Fault;             // Communication failure
   ……
 public:
   TestAgent(…);
   virtual void request();
   virtual void action();        // Scheduling entrance
   CIDM  o_ IDM;                // the instance of object CIDM
   CPEM  o_PEM;                 // the instance of object CPEM
   CPSM  o_PSM;                 // the instance of object CPSM
```

```
    void recv_comm_msg(string msg);          // Receiving messages
    CmdStruct *parse_command(char *raw_data); //Parse message
    void update_agent();    //Update running variables
    ……
};
```

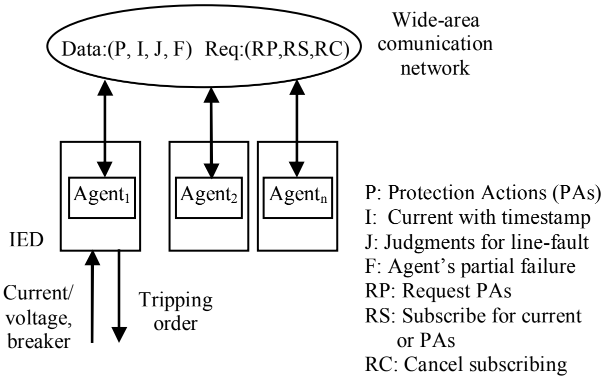
Data:(P, I, J, F) Req:(RP,RS,RC)
Wide-area comunication network
Agent$_1$
Agent$_2$
Agent$_n$
IED
Current/ voltage, breaker
Tripping order
P: Protection Actions (PAs)
I: Current with timestamp
J: Judgments for line-fault
F: Agent's partial failure
RP: Request PAs
RS: Subscribe for current or PAs
RC: Cancel subscribing

Figure 5. the messages exchanged among WABP agents

According to the design principles and pattern above, the WABP agent is implemented by C++ language, and simulated in EPOCHS platform studied by Kenneth in Ref [15]. The WABP multi-agent systems can find the power-line fault currently. The simulation testing of illustrate that the design of agent is effective.

B. The design and deploy of prototype multi-agent system

The programs of WABP multi-agent system include protection agents and monitoring program in substations and scheduling program in control center. The agents can push their data to the cooperators by calling the interface of MA in web server. The communications among agents are bidirectional. Some methods of WS components are designed to provide data-query function or remote control for client programs, such as the query for protection actions, the positions of the breakers or recorded data of fault.

The operation system of the platform is Windows 2003, web server uses IIS 6.0. The programming tool is Visual Studio.NET 2005.

After the web services of the substations and control center are built, they are deployed to the directory of web server C:\Inetpub\wwwroot. All protection agents can exchange distance protection actions, current signals, judgments of power-line fault each other by visiting the web services. Authored users can query running states and recorded data in substations with remote client program.

The prototype of wide-area cooperating protection simulation system based on web service has been implemented. The agents in different substations can exchange their protection actions and make a joint decision about the fault of power line. The system demonstrates the feasibility and effectiveness of the wide-area cooperating platform based on web service. But the wide-area backup protection requires short communication about 100ms, the real-time performance of the platform needs to be studied.

V. CONCLUSION

Aiming at the requirement of wide-area cooperating in power system, a wide-area cooperating platform is studied with web service and multi-agent technology. The business agents are loosely coupled and transparently transfer messages via web service interfaces. This platform has some features as below:

(1) Strong encapsulating capability. The traditional programs or new modules can be encapsulated to the agents with autonomous and cooperating ability. The modularization of the system is enhanced.

(2) Remote bidirectional communication ability. The business agents can communicate each other across networks with the help of web service messenger agent. The agent can find objective agent who provides wide-area cooperating service and shared resources by UDDI. The messenger agent as a special web service component can ensure reliable communication among agents across the firewall of networks.

(3) Intelligence and robustness of wide-area negotiation. Domain-dependent negotiation principles that include high-level decision-making and actions planning are finished by agent groups. The information and states of cooperators in each agent are set up. During the negotiation process, the agents exchange the messages including judgments, self-states and real-time data. The multi-agent system can tolerate the exiting or partial failure of some or other agents with joint strategies. Under the circumstances of communication failure, self-failure or other exception, whole system can continue to work and get good results.

(4) Platform and language independent. SOAP and XML are adapted for transfer the messages, the platform and language used by business agent, web server, web service components, client programs may be diversiform, which fits for the development and integrating of the systems.

(5) The agents mainly use messages to communicate each other, seldom to directly call other agents' methods. New functions can be implemented by changing the contents of messages. New modules may be easily added to the system. The running and negotiating process of agents is separated from the communication part, in other words, the agent needn't care about the implementation of communication. This design may bring good upgrading and reconstructing for the system.

The platform wide-area cooperating may be used in the applications with the requirement of wide-area cooperating, such as the information integrating in enterprise, distributed resource managing, etc. In the future, the real-time performance and routing of messages need to be studied deeply. In some control system, the real-time is very important to the control quality.

REFERENCES

[1] J. C. Tan, P. A. Crossley and D. Kirschen et al, "An expert system for the back-up protection of a transmission networks," IEEE Trans. Power Delivery, vol. 15, no.2, pp. 508-514, 2000.

[2] S.J. Lee,B.W. Min,K.H.Chung,et al, "An adaptive optimal protection of a distribution system using a multi-agent system," *in Proc. 2004 Eighth IEE International Conference on Developments in Power System Protection*, pp. 611-614.

[3] Xiaoru Wang,K.M.Hopkinson,J.S.Thorp, et al, "Developing a novel backup protection system for the electric power grid using agents," Automation of Electric Power Systems, vol. 29, no.21,pp.50-54, 2005.

[4] Renan Giovanini,Kenneth Hopkinson,Denis V Coury, et al, "A primary and backup cooperative protection system based on wide area agents, " IEEE Trans. on Power Delivery,vol.21,no.3,pp. 1222-1230, 2006.

[5] S.D.J. McArthur, E. M. Davidson, J.A. Hossack, et al, ”Automating power system fault diagnosis through multi-agent system technology,” in Proc. 2004 the 37th Hawaii International Conference on System Sciences, pp.1-4.

[6] WU Kecheng; LIN Xiangning; LU Wenjun; et al,"Principle and realization of the hierarchitdmcal region protective system for power systems," Automation of Electric Power Systems, vol.31, no.3, pp. 72-78, 2007.

[7] J.Q.Feng,D.P.Buse,Q.H.Wu, et al, "A multi-agent based intelligent monitoring system for power transformers in distributed substations," Power System Technology, vol.3, pp. 1962-1965, 2002.

[8] D.P. Buse, P.Sun, Q.H.Wu, et al, "Agent-based substation automation," IEEE power & energy magazine, vol.1, no.2, pp.50-55, 2003.

[9] Juhwan Jung, Chen-Ching Liu,"Multi-agent system technologies and an application for power system vulnerability," in Proc.2004 Power Engineering Society General Meeting, vol.1, pp.61-64.

[10] Moslehi K, Kumar A, Dehdashti, E, et al, "Distributed autonomous real-time system for power system operations-a conceptual overview," in Proc. 2004 Power Systems Conference and Exposition, pp.27-32.

[11] Chhabra, Manish; Lu, Hongen, "Towards agent based web service," in Proc. 2007 Computer and Information Science 2007, 6th IEEE/ACIS International Conference, pp.93-99.

[12] Felix F. Wu, Khosrow Moslehi, Anjan Bose, "Power system control centers: past, present, and future,” in 2005 Proceedings of the IEEE, vol.93, no.11, pp.1890-1908.

[13] Qizhi Chen, Hamada Ghenniwa, Weiming Shen, "Web services-based wide-area protection system design and simulation," in Proc.2006 Proceedings of the 10th International Conference on Computer Supported Cooperative Work in Design, pp. 201-209.

[14] M. Wooldridge and N. R. Jennings, "Intelligent Agents: Theory and practice, "The Knowledge Engineering Review, vol.10, pp. 115-152, 1995.

[15] Kenneth Hopkinson, Xiaoru Wang, Renan Giovanini, et al, "EPOCHS: A platform for agent-based electric power and communication simulation built from commercial off-the-shelf components," IEEE Trans. on Power Systems, vol.21, no.2, pp.548-558, 2006.