Mobile Agent Software Applied in Maintenance Scheduling

X. Xu M. Kezunovic
Department of Electrical Engineering
Texas A&M University
College Station, TX 77840

Abstract: The deregulation in power systems brings some new issues in equipment maintenance scheduling where more coordination and communication among different entities is required. In this paper, the mobile agent software is applied in the facility maintenance scheduling in deregulated power systems environment. In the given example, the problem is de-coupled into a master problem and a sub-problem using Benders decomposition. The two problems are solved at the GENCO and the ISO respectively, and the mobile agent takes care of the communication and coordination between these two problem-solving processes.

Keywords: maintenance scheduling, mobile agent, Benders decomposition, power systems

I. INTRODUCTION

The power system equipment maintenance involves scheduling, both long-term and short-term, and executing the actual maintenance work. In a traditional maintenance scheduling process, all the steps are usually done in one utility. But it is no longer true in a deregulated environment where the situations become more complicated due to the diversities of the ownership of the equipment and the fact that only limited information is available. Inevitably, equipment maintenance in the deregulated environment involves the need for more coordination and communication among different entities. This leads to the requirement of an information exchange mechanism among market participants for the purpose of maintenance scheduling thus the need for development of new tools for maintenance scheduling.

A typical maintenance scheduling procedure in deregulated environments involves the planning, submitting and approving steps [1]. At first, an asset owner makes a tentative schedule/plan for the maintenance, and then the initial schedule is submitted to another entity, usually ISO, for approving. The ISO checks the submitted schedule against the system constraints to decide its feasibility. If it is not feasible, the entity will be asked to modify its maintenance schedule. The whole process may be tedious and very complicated when more than two parties are involved.

The authors in [1] applied Benders decomposition method to deal with the new maintenance scheduling issues in the deregulated environments. The basic idea is to de-couple the whole scheduling problem into one master problem and several sub-problems. The master problem is solved at the assets owner's side, whereas each sub-problem represents a set of constraints imposed by a third entity. An iterative process is usually needed until the solution converges or no solution conclusion is reached. In [1], how the actual iteration proceeds is not discussed. This may be a problem in the real world, since different information exchange formats may exist, if an automated scheduling system is to be built. Exchanging information on paper and involving human beings to create the input and output may be time consuming and error-prone.

The Concordia mobile agent software provides a flexible framework for mobile agent applications [2]. An agent application program can travel through the internet/intranet to the computers where the Concordia server or transporter is running. Concordia also supports Distributed Events, Agent Collaboration and Service Bridge. Commercial versions of Concordia add encryption and proxy supports. Compared with distributed database systems, an agent can process the data locally and thus reduce the network traffic. Besides, the Java platform encapsulates the network layer from the agent, which makes the programming easier. The Concordia mobile agent may fit very well in the above-mentioned iterative process of making the maintenance schedules.

The agent is a natural concept for representing an entity. By integrating some gaming strategies into an agent, it can represent the client doing the time-consuming negotiation work, especially when some conflicts need resolving or other types of negotiation are needed. On the other hand, the exact system models are difficult or too complicated to establish in some cases, so that the conventional "hard" computing cannot easily be implemented. Intelligent agents with heuristic experience captured from human experts may become very useful. In [3], the agent technology has been considered as a promising approach to construct the third generation EMS systems.

In this paper, we considered how Concordia mobile agent software might be applied in the maintenance scheduling process. Without losing generality, a generation long-term maintenance-scheduling problem is used to illustrate the concepts. Section II formulates the problems and the solution method is discussed in Section III. At last an example of how to apply the concept to the simple three-bus system is given in Section IV.

II. PROBLEM FORMULATION

In general the maintenance-scheduling problem is determined by the adopted maintenance strategy. The current commonly used maintenance strategy relies on preventive maintenance with fixed time intervals. The basic rational behind this is: "devices should be overhauled regularly". A more systematic solution is to perform Reliability Centered Mainteannce (RCM) analysis, and determine what is the most proper maintenance strategy for each device. The analyses performed in RCM include function and failure definition, failure modes and effects analysis, and the selection of maintenance actions at last [4].

Even if the practice of periodically performing preventive maintenance is adopted, the beginning time of the maintenance may still change in a given range, therefore there will be a problem of optimization. The optimization problem can be modeled in two ways [1]. The first one is based on fictitious costs to penalize deviations from a preferred maintenance schedule. The second approach is based on identifying maintenance windows (time intervals) in which the preferred maintenance schedule is represented within windows and the objective or goal is to minimize the real maintenance cost instead of a fictitious cost. The maintenance window is shown in Figure 1. e_i and l_i represent the beginning and the ending of the window respectively. The maintenance begins at the time of s_i and lasts for a time length d_i. s_i must fall into e_i and l_i.

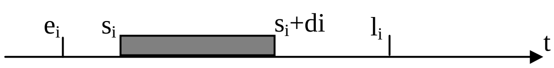

Figure 1. Maintenance Window

The ultimate goal of the optimization is to maximize the income, which is the revenue minus the production cost and maintenance cost. From the viewpoint of maintenance, the objective can be equalized to minimize the maintenance cost plus the loss of revenue due to the maintenance.

The constraints may include maintenance constraints (maintenance policy, maintenance crew availability, etc), and system constraints.

Mathematically, long-term generation maintenance scheduling problem can be formulated as follows (1-4):

$$Min \quad \sum_t \sum_i \{C_{it}(1-x_{it}) + c_{it} g_{it}\} \qquad (1)$$

Subject to:

$x_{it} = 1$ for $t \le e_i$ or $t \ge l_i + d_i$ (2)

$x_{it} = 0$ for $s_i \le t \le s_i + d_i$

Maintenance constraints (3)

System constraints (4)

Where:

C_{it} - maintenance cost of unit *i* at time *t*

x_{it} - status of unit *i* at time *t*, 0 stands for maintenance

c_{it} - lost of revenue of unit *i* at time *t* due to maintenance

g_{it} - generation output of unit *i* at time *t*

s_i - maintenance variable of unit *i*, start of maintenance

e_i, l_i - maintenance window for s_i

d_i – the duration of maintenance of unit *i*

Constraints in (2) represent the maintenance window, which reflects the fixed time-interval preventive maintenance strategy. The maintenance constraints may include the crew and resource availability, seasonal limitations, and other constraints such as fuel and emission constraints. The system constraints represent the peak load balance, transmission flow limits and energy reserve requirements, which are usually not available to a GENCO who is making the schedules.

There are several simplicities in the about formulation worthy of being pointed out. First, the maintenance strategy may be different than the assumed fixed time-interval preventive one, so the maintenance window method is no longer applicable. Second, in a deregulated environment, the price of the electricity will not be fixed for an extended period any more, and the generation companies need to consider the price fluctuation when making their maintenance schedules. The actual maintenance-scheduling problem is more complicated, which may prefer some rule-based heuristic solutions, since the mathematical model becomes too complicated to build. Third, there are usually more than one GENCOs submitting maintenance schedules to the ISO in the real world. The ISO must have a strategy to resolve conflicts among GENCOs. The mobile agent architecture and intelligent agent technique may be very suitable to handle all these varieties.

III. SOLUTION METHOD

The problem given in (1) is a Mixed-Integer Programming (MIP) problem. In general, the number of generation units is usually not very big, so the problem generally can be solved efficiently. But, the system constraints are usually not available to a GENCO in a deregulated environment, which means a trial schedule without system constraints must be defined at first and then be submitted to the ISO for approbation. The ISO will check the feasibility of the schedule, and either approve it or deny it. When a proposed schedule is denied, the GENCO would prefer if the ISO could feed back some information on how to modify the initial schedule to meet the system constraints.

For the generation unit maintenance problem given in (1), Benders decomposition method can be used to decompose it into a GENCO master problem and an ISO sub-problem. The master problem is solved without system constraints at first to get the decision x_{it} variables. Once those variables are fixed, the ISO sub-problem can be solved with system constraints. If the initial trail schedule is not feasible, a Benders cut is generated. The cut is added into the master problem in the next iteration. The whole process is depicted in Figure 2.

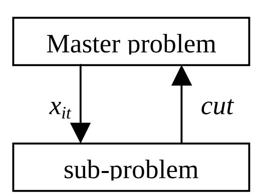

Figure 2. Iteration of solution process

Benders Decomposition: [1]

To make the illustration more clear, the mixed-integer problem of (1) is rewritten as the following form:

$$\begin{aligned} Min \quad & c^T x + f(y) \\ S.T. \quad & Ax + F(y) \geq b \\ & x \geq 0 \\ & y \in Z \end{aligned} \tag{5}$$

Where:

A: constraint matrix

x: continuous variable vector

y: integer variable vector

If values of ***y*** are fixed at first by choosing:

$$y \in R = \{y \mid Ax \geq b - F(y), x \geq 0\} \tag{6}$$

Then (5) is linear in ***x***, and it can be written as:

$$\min\{f(y) + \min\{c^T x \mid Ax \geq b - F(y), x \geq 0\}\} \tag{7}$$

Based on the duality theory, the set R in (6) may be rewritten as:

$$R = \{y \mid (b - F(y))^T u_i{}^r, i = 1 \ldots n_r\} \tag{8}$$

Where u_i^r is the extreme point vector which belongs to cone C = {$\boldsymbol{u}$ | $\mathbf{A}^T\boldsymbol{u} \leq 0$, $\boldsymbol{u} \geq 0$}, and n_r is number of extreme points of core C. The inner minimization in (7) can be rewritten as follows:

$$\begin{aligned} Min \quad & c^T x \\ S.T. \quad & Ax \geq b - F(y) \\ & x \geq 0 \end{aligned} \tag{9}$$

Its dual problem is:

$$\begin{aligned} Max \quad & (b - F(y))^T u \\ S.T. \quad & A^T u \leq c \\ & u \geq 0 \end{aligned} \tag{10}$$

Substituting (10) into (7) yields a new form of (5):

$$\min\{f(y) + \max\{(b - F(y)))^T u \mid A^T u \leq c, u \geq 0\}\} \tag{11}$$

(11) is equivalent to the following program:

$$\begin{aligned} Min \quad & z \\ S.T. \quad & z \geq f(y) + (b - F(y))^T u_i{}^p, \quad i = 1 \ldots n_p \\ & (b - F(y))^T u_i{}^r \leq 0, \quad i = 1 \ldots n_r \end{aligned} \tag{12}$$

Where u_i^p is an extreme point of P = {$\boldsymbol{u}$ | $\mathbf{A}^T\boldsymbol{u} \leq$ c, $\boldsymbol{u} \geq 0$}. Problem (12) is equivalent to problem (5) with integer variable y. Problem (12) has one constraint for each extreme point that translates into an enormous number of constraints even in a problem with moderate dimensions. However, only a small fraction of constraints will be binding at an optimal solution. Therefore, we begin with a few constraints and solve (12), which is the master problem. And sub-problem (9) or (10) is used to see if this solution satisfies the remaining constraints.

Requirements for Implementation:

The solution method mentioned above requires an implementation with the following features:

- It shall work on heterogeneous hardware platforms and operating systems. The iteration process involves different entities, such as the GENCO and the ISO, which may have different hardware and software environments.
- It shall be able to authenticate and authorize the users.
- It shall provide secure communicate channels in the cases that the public network is used to connect the GENCO and the ISO.
- It shall integrate with the existing systems well.

IV. IMPLEMENTATION

The maintenance scheduling system is implemented using Concordia mobile agent software [2]. Two computers have been setup to represent the GENCO and the ISO respectively as shown in Figure 3. Both computers have the Concordia server running.

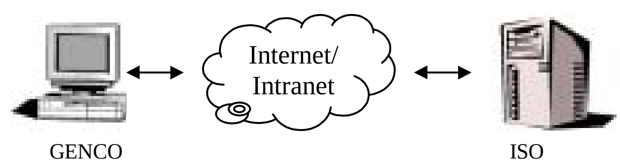

Figure 3. Computer setup

The mobile agent travels between those two servers and calls some local services provided by the servers. The services may be provided by other Java software, or exported from legacy systems using Java Native Interface (JNI) or Common Object Request Broker Architecture (CORBA) [6].

When solving maintenance scheduling problems, the mobile agent architecture provides the following advantages:

- The mobile agents can be authenticated when they arrive at an entity (GENCO or ISO) to determine their identification.
- An entity can control the information the mobile agents can access according to their identification.
- An entity can expose its services to mobile agents and the entity can control the accessibility to the services.

- The software structure becomes more flexible by decoupling the overall negotiation process and the local problem solving processes (at GENCO or ISO). For example, the GENCO may use totally different methods to generate the maintenance schedule, and may use different data format of the schedule. As long as the service interfaces are not changed, the whole system structure will not be effected.

Security Consideration

Two apparent security problems arise when applying mobile agents. First, the mobile agents need to be authenticated and authorized at the servers. Second, to ensure the integrity of the data, it must be transmitted in secure communication channels.

Every mobile agent must be authenticated at first to identify whom it represents. The Concorida SecureAgent supports user authentication by using the username/password pairs. Once identified, mobile agents can be checked against the security policy to see whether they are authorized to do certain things at a server. The Concordia Administer as shown in Figure 4 provides a user-friendly interface for server, security and service management.

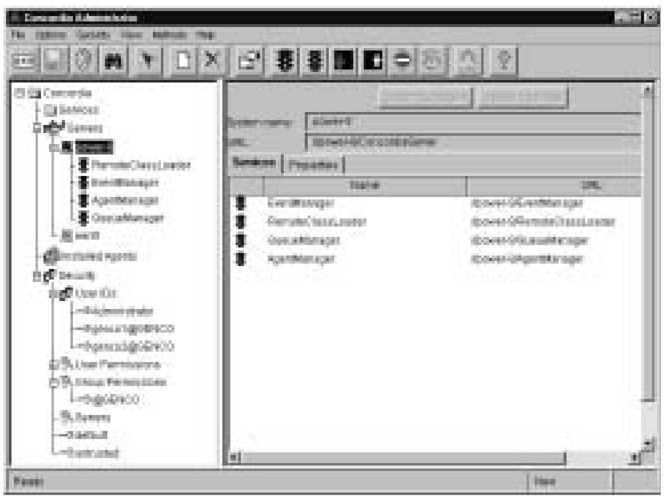

Figure 4. Concordia Administer

The Concordia server can control the agent's access to resources depending on both the user identification and the server permits. A user can be created for each GENCO. Alternatively, a GENCO group can be used to represent all the GENCOs. In Figure 4, a GENCO group has been created, which has two members: genco1 and genco2. The GENCO group has the permissions to access maintenance-related services.

The user interface to assign different types of permissions to users or groups is shown in Figure 5. The permissions are divided into different groups for agent, class, event, file, etc. For example, the permissions in agent group decide if an agent with the user/group's identity can arrive or be launched to/from this server. The file group permissions determine whether a user or group can access the local files.

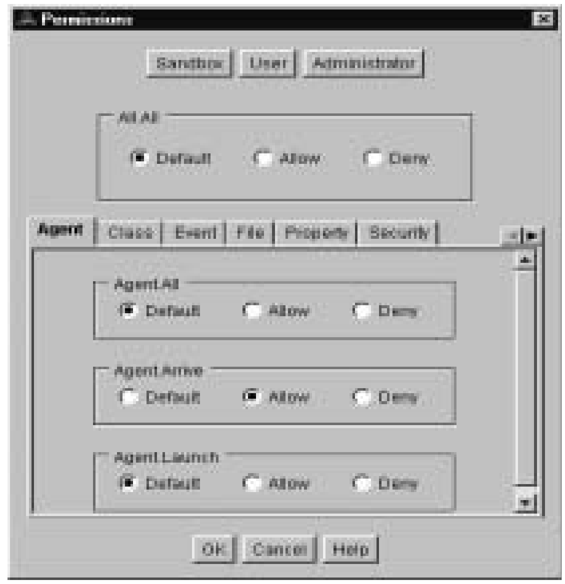

Figure 5. Set user permissions

Secure communication channels among the mobile agent servers may become important, especially when the data passes through the public network or wireless channels. Concordia mobile agent software provides options to encrypt the data when it is in transit, and thus prevents others tampering with the data. A digital envelope will be used to protect the Concordia SealedAgent when travelling.

All the above security measures are supported by Concordia software directly and thus greatly simplify the programming work. There are some other security-related features provided in Concordia. As an example, Concordia mobile agents can work with firewalls, which is important when access to the company Intranet from outside public Internet is needed.

Interfacing with Other Systems

Mobile agents travel to Concordia servers and call the exposed services to accomplish their tasks. The recommended way to expose services in Concordia is to use service bridges, which are application specific services written in Java and installed at the servers. Service bridges can be thought of as the non-mobile component of a mobile agent-based application. They represent server-specific resources that can be accessed by mobile agents using the exposed interfaces. Service bridges may also act as interfaces for accessing existing systems via RMI, JNI, CORBA, or DCOM.

When the mobile agents need to access the functions of some existing systems not implemented using Java, two possible interfacing techniques are JNI (Java Native Interface) and CORBA (Common Object Request Broker Architecture) [6]. As an example, in the maintenance-scheduling problem, the mobile agent needs to call the Linear Programming (LP) solver and the Mixed Integer Programming (MIP) solver of some optimization packages. Both JNI and CORBA approaches have been implemented to call the optimization library.

In the JNI approach, the optimization routines provided at both the GENCO computer and the ISO computer are stubs to Matlab optimization routines. The linear programming routine of Matlab optimization toolbox is used to solve the ISO sub-problem, and the master problem is solved using the enumeration method. As shown in Figure 6, the mobile agent calling the optimization routines through the JMatlink[5] module, which uses Java Native Interface (JNI) to invoke the Matlab engine. In this way, a virtual link between the mobile agent and the optimization routines is established.

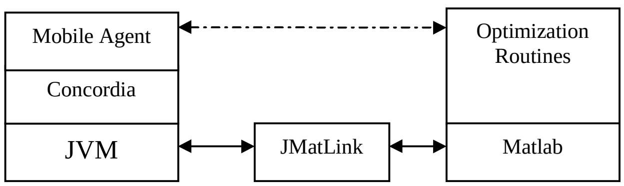

Figure 6. Links between JVM and Matlab

In the CORBA approach, the commercial solver CPLEX 6.5 has been encapsulated using CORBA. CPLEX can solve large-scale LP problem and MIP problems. Using CPLEX, larger systems can be solved since it provides a better MIP solver and can handle larger scale problems. This approach is shown in Figure 7.

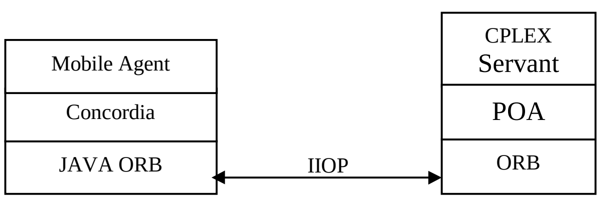

Figure 7. Calling CPLEX using IIOP

CORBA is a platform and language neutral standard for component programming [6]. CORBA is a standard, and the actual software is called ORB (Object Request Broker), which is in charge of locating the remote objects and doing the remote method invocation. The standard protocol IIOP (Internet Inter-ORB protocol) in CORBA specification ensures the interoperability among the ORBs implemented by different venders. OmniORB [7] is a free ORB and has been used to encapsulate and export the CPLEX library.

Java 2 comes with a Java ORB and supports RMI-over-IIOP directly. The current version Concordia doesn't work with Java 2 (JDK 1.2.x, 1.3.x) very well, we used another free Java ORB – JacORB[8], which can work with Java 1 (JDK 1.1.x). According to the vendor, the next version of Concordia will work with Java 2.

Exposing the functions of the optimization package (CPLEX in this specific case) using CORBA makes it accessible to other systems, which may be programmed with different types of languages than the optimization package. Also the location of the services becomes transparent. Unlike the JNI approach, where the exposed functions can only be assessed by local Java applications, the CORBA approach makes the functions assessable to more applications with different languages and even from remote computers. For example, in our case, the CPLEX library doesn't need to be at the same computer as the Concordia server. Since the CPLEX license limits it usage only at the computer where it had been installed, having a CORBA server running at that computer will make it accessible to more applications and thus improves its utilization rate. In this way, fewer licenses need to be purchased.

Using CORBA to call CPLEX library also provides an example on how to integrate legacy system using mobile agents. Legacy systems may be implemented using different languages (COBOL, Fortran, ADA, C, etc.) and running on different platforms (Mainframe, minicomputer, etc.). As long as the functions of the legacy systems can be exported using CORBA, those functions can be accessible from Java and thus mobile agents.

V. EXAMPLE

A three-bus system is used to demonstrate the main concepts, and the system is shown in Figure 8. A GENCO owns all the three generators and an ISO is in charge of scheduling the transactions over the three lines. Only one maintenance period is considered, and in that period at least one generator will be maintained. Transport model is used to check the feasibility of a schedule in the ISO, so the data of line capacity is sufficient, which is shown in Figure 8. The generator data is given in Table 1. To further simplify the problem, only the maintenance cost part is considered in the objective. So, at last, the maintenance-scheduling problem of (1) is simplified to:

$$Min \quad \sum_{i=1,2,3} \left\{ C_i (1 - x_i) \right\} \tag{13}$$

S.T.

(1) Maintenance constraints: (14)

$x_1 + x_2 + x_3 \le 2$

$x_i = 0$ or 1, $i = 1,2,3$

(2) System constraints (15)

$-f_{12} - f_{13} + g_1 = d_1$

$-f_{23} - f_{12} + g_2 = d_2$

$f_{13} + f_{23} + g_3 = d_3$

$0.5 \le g_1 \le 2.5$

$0.6 \le g_2 \le 2.5$

$0.6 \le g_3 \le 3.0$

$-0.5 \le f_{12} \le 0.5$

$-0.5 \le f_{13} \le 0.5$

$-1.0 \le f_{23} \le 1.0$

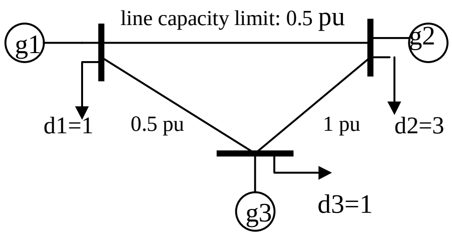

Figure 8. Three-bus System

Table 1. Generator Data

Unit	Min cap. (p.u.)	Max cap. (p.u.)	Maint. cost *Ci* ($)
1	0.5	2.5	300
2	0.6	2.5	200
3	0.6	3.0	100

The above problem can be decomposed into a master problem solved at the GENCO, and a sub-problem checked at the ISO. The master problem tries to minimize the maintenance cost and consists of the objective and the maintenance constraints part, while the sub-problem is only responsible for the system constraints (only considering the overload of the transmission lines in this example).

The computer setup has been shown in Figure 3 and the mobile agent will travel between those two computers. The mobile agent calls some service functions provided by the GENCO computer at first to solve the master problem. After a trial schedule is obtained, the agent travels to the ISO computer with the schedule and submits it. The services provided at the ISO computer would check the schedule and generate infeasible cut if it is infeasible. The agent brings the cut back to the GENCO computer and adds it into the set of constraints and starts the next iteration. The process continues until no cut condition is returned from the ISO computer.

At the initial step, the mobile agent's itinerary contains three destinations: the GENCO computer, the ISO computer and then back to the GENCO computer. According to the status of problem, the mobile agent will modify its itinerary dynamically. When the ISO sub-problem returns a cut, the mobile agent resets its itinerary and begins the next iteration.

The results returned by the mobile agent at each iteration are given in Table 2.

Table 2. Results of each iteration

Iteration	**Problem**	**Results**
1	Master	x = [1, 1, 0], z=100
	Sub	Benders cut: c*x ≤ -2.9, c = [0, -2.499, -2.999]
2	Master	x = [1, 0, 1], z= 200
	Sub	Benders cut: c*x ≤ -1.4, c = [0, -2.499, 0]
3	Master	x = [0, 1, 1], z= 300
	Sub	No Benders cut, so the above schedule is feasible and approved

From the above table, we can see the total cost is 100 and unit #3 will be taken out for maintenance in the first iteration. Since this case doesn't consider the system constraints, the solution will not be feasible when it is submitted to the ISO computer. An infeasible cut is returned, which is –2.499*x2 – 2.499*x3 ≤ -2.9 as shown in the table. By adding this constraint into the master problem, the scheduling problem is solved again at the GENCO computer, and the result is that unit #2 will be maintained this time. Again, this result is submitted to the ISO computer and with a new cut returned. After adding this new cut, the solution turns out that unit #1will be maintained and the total cost is 300. This time, ISO computer doesn't return any cut, which means this schedule is feasible and thus approved.

Both approaches proposed in Section IV have been implemented and compared. The results are the same and no difference in speed has been observed. The main reason is that the system is very small. When bigger systems are used, it can be expected that using CPLEX library instead of calling Matlab will give much better performance.

The example given here just illustrates one possibility of applying mobile agent software in the maintenance-scheduling problem. Actually, the GENCO may use very complicated method or system to schedule the maintenance, and also the ISO may consider much more factors when check the feasibility of a submitted schedule. But the iteration process among different entities (the GENCO and the ISO in this case) is unchanged.

VI. CONCLUSION

A new software structure using mobile agents to solve the maintenance-scheduling problem is proposed. This new structure is more flexible and platform independent and thus very suitable for the scheduling process in the deregulated environment, where multiple entities are involved.

Further developments may include: integrating with RCM systems, utilizing robust optimizing package for large-scale systems, using soft-computing methods (fuzzy expert system, neural network, etc.) in the cases that the exact mathematical models (hard-computing) are difficult to get, exploring agent negotiation methods when more than two entities are involved.

VII. ACKNOWLEDGE

The project is support by PSerc, an NSF I/UCRC consortium. The authors would like to acknowledge Mitsubishi Electric Research Lab. for providing access to the Concordia mobile agent software.

VIII. REFERENCE

[1] M. Shahidehpour, M. Marwali, "Maintenance Scheduling In Restructured Power Systems", Kluwer Academic Publishers, 2000
[2] Concordia white paper, http://www.meitca.com/HSL/Projects/Concordia/MobileAgentsWhitePaper.html
[3] Gilberto P. Azevedo, et.al. "Control Centers Evolve with Agent Technology", IEEE Computer Applications in Power, July 2000
[4] John Moubray, "Reliability-centered Maintenance", 2nd edition, Industrial Press Inc. 1997
[5] JmatLink, http://www.held-mueller.de/JMatLink/
[6] CORBA, http://www.omg.org/
[7] OmniORB, http://www.uk.research.att.com/omniORB/
[8] JacORB, http://www.jacorb.org/

IX. BIOGRAPHIES

Xiangjun Xu (S'99) received his B.E and M.E. degrees from Southeast University and Shanghai Jiaotong University, all in electrical engineering, in 1992 and 1995 respectively. After that, he worked as a teacher/researcher in Shanghai Jiaotong University. Since Sep. 1998, he has been with Texas A&M University pursuing his Ph.D. degree.

Mladen Kezunovic (S'77, M'80, SM'85, F'99) received his Dipl. Ing. Degree from the University of Sarajevo, the M.S. and Ph.D. degrees from the University of Kansas, all in electrical engineering, in 1974, 1977 and 1980, respectively. Dr. Kezunovic's industrial experience is with Westinghouse Electric Corporation in the USA, and the Energoinvest Company in Sarajevo. He also worked at the University of Sarajevo. He was a Visiting Associate Professor at Washington State University in 1986-1987. He has been with Texas A&M University since 1987 where he is the Eugene E. Webb Professor and Director of Electric Power and Power Electronics Institute. His main research interests are digital simulators and simulation methods for equipment evaluation and testing as well as application of intelligent methods to control, protection and power quality monitoring. Dr. Kezunovic is a registered professional engineer in Texas, and a Fellow of IEEE.