Web-Based Security Cost Analysis in Electricity Markets

Hong Chen, *Member, IEEE*, Claudio A. Cañizares, *Senior Member, IEEE*, and Ajit Singh

***Abstract*—Security cost analysis is important in electricity markets to address the correlation between market operation and power system operation. This paper proposes an efficient security cost analysis method and describes its implementation using a three-tier client/server architecture and up-to-date web technologies. The proposed security cost analysis is based on a system security index and its sensitivities with respect to certain system parameters. The web implementation allows easy and effective access by all market participants to make competitive decisions based on the security costs and sensitivity information obtained through the proposed security cost analysis techniques. The web implementation of the proposed method is successfully tested on a six-bus system and a 129-bus system.**

***Index Terms*—Client/server, electricity markets, Java, power system security, transmission congestion, web.**

I. INTRODUCTION

IN competitive electricity markets, efficient market operation must be coordinated with secure system operation. Because of transmission security constraints, transactions determined by market forces are feasible only when they are within the system's security limits. Thus, transactions in electricity markets need to be evaluated and analyzed based on system security to make sure of their operational feasibility.

Transmission congestion happens when the dispatching of transactions violates system security constraints. Given the complexity of power transactions, congestion management and pricing are key issues in the daily operation of electricity markets. The correct cost information for congestion relief, which ensures the operational feasibility of transactions, and is referred to as "security cost" in this paper, can help independent system operators (ISOs) determine congestion prices, coordinate and manage transactions, and also help market participants make profitable market decisions and help to maintain system security by properly responding to the given price signals [1].

Determining the costs associated with system security has been of great interest in power systems, especially under the framework of competitive electricity markets [2]–[9]. In these papers, optimization techniques are extensively used, and the Lagrange multipliers associated with the given security constraints are used to represent security costs. Rather conservative limits determined through offline studies on the transmission system power flows and bus voltage magnitudes are usually used to represent system security. A series of engineering approaches not based on optimization techniques to analyze security costs are reported in [1] and [10]–[14]; these articles concentrate on accurately representing system security on electricity markets.

Market participants are usually geographically widely distributed and have different software and hardware platforms. It becomes a suitable choice to implement security cost analysis in a web-based platform, so that all market participants can easily and widely access it through heterogeneous hardware and software platforms without any configuration and installation or update operations, to assess their transaction costs for competitive decision making.

Web-based computing, based on the Internet Protocol, distributed processing, and web browsers, permits data sharing and computing over a large range of heterogeneous hardware and software platforms, thus facilitating systems that are highly available, expandable, and relatively easy to maintain and update. However, the efficiency is highly influenced by the communication network reliability and communication speeds; fast and reliable networks are needed. Security is also an important issue that has to be considered for web-based applications.

Web-based applications have been proposed for a variety of uses in electricity markets [15]–[20]; most of these implementations use web techniques for information access and exchange. Using web-based computing to implement electricity market decision support systems is still a challenge [1], [17], [19], [21]. Reference [22] presents the congestion management in a bilateral market using Java, based on the dc load flow model. In [23] and [24], the authors introduce market simulators implemented using Java, with the difference that JNI is used in [23] to access the load flow calculation from MATLAB, while Tcl is used in [24] for the communication between Java applets and market clearing price application written in MATLAB. A web-based load flow simulation using combined J2EE-CORBA environment is presented in [25], based on multi-tier architecture, and CORBA is used as the bridge for Java platform to access legacy load flow analysis routine.

Compared to the existing research works in this area, this paper proposes a novel three-tier client/server design of a web-based security cost analysis system. The efficient and transparent security cost analysis is implemented based on a system security index (SSI) and associated sensitivities with respect to power transactions. A prototype implementation of the design is discussed in detail as well.

The paper is structured as follows: The SSI and its sensitivities are defined in Section II. A rescheduling method and asso-

---

Manuscript received March 25, 2003; revised November 9, 2004. This work was supported by NSERC, Canada, and the Electrical and Computer Engineering Department at the University of Waterloo. Paper no. TPWRS-00129-2003.

H. Chen is with ISO New England, Holyoke, MA 01089, USA (e-mail: hchen@iso-ne.com).

C. A. Cañizares and A. Singh are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, N2l-3G1 Canada (e-mail: ccaniza@uwaterloo.ca).

Digital Object Identifier 10.1109/TPWRS.2005.846099

ciated security cost analysis are proposed and described in Section III. In Section IV, the web-based architecture design and prototype implementation of the security cost analysis are discussed in detail, presenting the results of its application to two test systems. Finally, Section V summarizes the main contributions of this paper.

II. SYSTEM SECURITY INDEX AND SENSITIVITY COMPUTATIONS

The security cost analysis proposed in this paper is based on SSI computations and sensitivity analyzes, which consider voltage stability constraints as well as thermal and bus voltage limits.

A. SSI Calculations

A power system can be represented in steady state with the following set of nonlinear equations:

$$f(x,\lambda,p)=0 \tag{1}$$

where $x \in \Re^n$ stands for the state and algebraic system variables, such as bus voltage magnitudes and angles. The scalar $\lambda \in \Re$ is a loading parameter used to represent the system loading margin, as the load powers are modeled as

$$P_L = P_{L_o} + \lambda P_D \tag{2}$$

with P_{L_o} representing the power levels of loads that *do not* directly bid in the market (e.g., "nondispatchable" loads in the Ontario market [26]) and, hence, define the initial loading conditions and P_D corresponding to the demand power bids; all loads are assumed to have constant power factors. In this analysis, generator powers are modeled as

$$P_G = P_{G_o} + (\lambda + k_G)P_S \tag{3}$$

where P_{G_o} stands for the generator power levels that *do not* directly bid in the market (e.g., "must-run" generators in the Ontario market [26]), and k_G is a variable used to represent a distributed slack bus. The parameters $p \in \Re^m$ represent "controllable" market or system parameters, such as the supply and demand power bids P_S and P_D, respectively. Equation (1) typically corresponds to a set of "modified" power flow equations, which basically result from modeling system controls and limits in greater detail than in the typical power flow equations [27].

The voltage stability limits for the system represented by (1) are basically associated with saddle-node and limit-induced bifurcations (SNBs and LIBs) of the corresponding set of nonlinear equations [27]; at these bifurcation points, the system collapses. Thermal and voltage limits, on the other hand, can be treated mathematically, for the purpose of sensitivity analyzes, in a similar way as limit-induced bifurcations, although the system does not collapse when these limits are reached. Hence, the SSI is defined as

$$\mathrm{SSI} = \lambda_c \tag{4}$$

where λ_c represents the "critical" (maximum) loading parameter at which the system is at a limit condition due to voltage stability, thermal, or bus voltage constraints for a worst-contingency scenario. This index is defined in this way to try to capture the main characteristics of transmission congestion.

The SSI may be basically viewed as a "system-wide" available transmission capability (ATC), i.e., an ATC value not directly associated with particular transmission corridors, which is typically how ATC is currently being used in electricity markets but rather a system ATC for an overall market transaction.

B. Sensitivity Formulas

Since voltage stability constraints as well as thermal and voltage limits are used to determine the SSI value, one can also readily determine the sensitivities of the SSI with respect to various system parameters, especially with respect to the participants' bids, i.e.,

$$\frac{d\mathrm{SSI}}{dp} = \frac{d\lambda_c}{dp}. \tag{5}$$

The required sensitivity formulas can be obtained from the use of basic voltage stability concepts [27].

1) SNBs: SNBs are characterized by a pair of equilibrium points coalescing and disappearing as the parameter λ "slowly" changes. Mathematically, the SNB point is an equilibrium point (x_c,λ_c,p_c) with a singular Jacobian $D_x f|_c$ and associated unique right and left "singular" eigenvectors v and w, respectively, i.e., $D_x f|_c v = D_x^T f|_c w = 0$.

By taking the derivatives of (1), one has at the SNB point that

$$\begin{aligned} & D_x f|_c dx + D_\lambda f|_c d\lambda + D_p f|_c dp = 0 \\ \Rightarrow\quad & w^T D_x f|_c dx + w^T D_\lambda f|_c d\lambda + w^T D_p f|_c dp = 0. \end{aligned}$$

Hence, from these equations and as proposed in [28], one can conclude that the sensitivities of SSI with respect to changes in the parameters p at the SNB point can be determined by using

$$\frac{d\mathrm{SSI}}{dp} = -\frac{1}{w^T D_\lambda f|_c} w^T D_p f|_c. \tag{6}$$

2) Limits: LIBs are equilibrium points where a system control limit is reached, which in *some* cases may lead to a system collapse characterized by a pair equilibrium points coalescing and disappearing for slow changes of the parameter λ. At an LIB, as opposed to an SNB, the system Jacobian is not singular at the bifurcation point (x_c,λ_c,p_c); hence, (6) does not directly apply at this point. Furthermore, sensitivities of system limits that are not necessarily associated with stability problems but are rather the result of equipment limitations, such as thermal limits on transmission lines, cannot be studied using (6) either.

In general, a system reaching any particular limit at an equilibrium point (x_c,λ_c,p_c), such as a bus voltage, thermal, or reactive power limit, can be characterized by two different sets of equations, i.e.,

$$\begin{aligned} f_1(x_c,\lambda_c,p_c) &= 0 \\ f_2(x_c,\lambda_c,p_c) &= 0 \end{aligned} \tag{7}$$

where the first set $f_1(\cdot)$ corresponds to the "original" system equations, whereas the second set $f_2(\cdot)$ corresponds to a modified set of equations where the limit is active. For example, when a reactive power generator limit is reached at a bus i, a generator voltage control equation, say $V_i - V_{i_c} = 0$, may be replaced by $Q_{G_i} - Q_{G_{lim}} = 0$ at the limit condition. Hence, taking the derivatives of (7) at the equilibrium point where the limit becomes active

$$D_x f_1|_c dx + D_\lambda f_1|_c d\lambda + D_p f_1|_c dp = 0 \quad (8)$$
$$D_x f_2|_c dx + D_\lambda f_2|_c d\lambda + D_p f_2|_c dp = 0 \quad (9)$$

Eliminating dx from these equations leads to

$$\frac{d\text{SSI}}{dp} = \frac{1}{\mu^T \mu}\mu^T \left(D_x f_2|_c D_x f_1|_c^{-1} D_p f_1|_c - D_p f_2|_c\right) \quad (10)$$

where

$$\mu = D_\lambda f_2|_c - D_x f_2|_c D_x f_1|_c^{-1} D_\lambda f_1|_c.$$

Observe that the sensitivity formula (10), which can be shown to be equivalent to a formula proposed in [29], applies to any limit condition, independent of whether it corresponds to an LIB or a thermal or voltage limit. Hence, (10) together with (6) are used to determine the sensitivity of SSI with respect to the system parameters p.

In this paper, all SSI and required sensitivity values are computed based on the results generated by UWPFLOW [30], which is a continuation power flow program capable of representing various power system elements using "detailed" steady-state models.

III. RESCHEDULING TECHNIQUE

The rescheduling technique proposed in this paper considers both generation redispatch and load curtailment, where rescheduling is used to address the issue of a simple auction mechanism that yields a market clearing condition that violates certain security criteria. The costs resulting from dispatching a participating unit, which might be more expensive than the market clearing price (MCP), or curtailing loads to solve the congestion problem are then "distributed" among the different participants. The idea here is to redispatch units or curtail load based only on the effect that these have on transmission system congestion, without too much regard for costs, as at this point, system security takes precedence over economic considerations, especially in view that, in most markets, bidding results can be rejected based on security criteria. Observe that by choosing the units or loads based on this criterion, there will be minimum impact on the desired transactions.

The proposed rescheduling technique is based on rescheduling methodologies currently used in various electricity markets, presenting the following advantages with respect to the existent techniques.

- Thermal, bus voltage, *and* voltage stability limits are all accurately considered in the computation of the SSI, which is used to represent system security and, thus, evaluate transmission congestion for a given market transaction, without relying on power limits on transmission lines computed off line and typically used to represent stability limits in most electricity markets.
- The necessary rescheduling of generators and/or loads to solve transmission congestion problems are based on the impact that each one of the market participants have on the system security, which is reflected on the SSI sensitivity value for a given market transaction; this is typically not the case in existent markets of the type discussed in this paper (e.g., Ontario [26]).
- The security costs associated with the rescheduling process are distributed among the market participants based on the actual impact that each one of them has on the system security for a given transaction. This is not typically the case in most markets, as in general, these rescheduling costs are "socialized" (e.g., in Ontario, both "constrained-off" and "constrained-on" generators are paid when rescheduling is needed to solve a congestion problem, and the cost is then distributed among market participants based on their power levels [26]).

The suggested rescheduling technique to address the problem of market clearing conditions that do not meet the actual SSI requirements is summarized in the flow-diagram shown in Fig. 1. The depicted methodology is based on a series of linearizations; however, the SSI does change nonlinearly as the system parameters change due to the highly nonlinear behavior of the system. The latter is the main reason for using an iterative process, representing an improvement over typical rescheduling techniques currently in use. The proposed technique determines the security costs for the different market participants as follows.

A. MCP Computation

Using a simple auction mechanism, the MCP and associated total transaction power level T are determined, together with the load and generator power levels that clear the market, namely, P_D and P_S for all loads and generators, respectively. The values of P_D and P_S are used in the determination of the SSI, as these define the load and generator direction used for the computation of λ_c in UWPFLOW [30].

B. Transaction Impact and Rescheduling

If the SSI requirement is not met, i.e., if the load cannot be increased up to the bid level P_D $(\text{SSI} < 1)$ due to security constraints, the impact of each possible system transaction is determined using (6) or (10), depending on the limiting factor that defines the SSI. Thus, the generator i with the most positive impact on the SSI that has not been fully dispatched in the bidding process and the generator j dispatched in the market clearing process with the most negative or least positive impact on the SSI are chosen for rescheduling. Thus, the corresponding increase and decrease in generation is defined as

$$\Delta P_{S_i}^{(k)} = -\Delta P_{S_j}^{(k)} = \Delta P_S^{(k)}$$

where k is the number of iteration in the redispatch process, and $\Delta P_S^{(k)}$ is chosen depending on the value that one wants for the SSI, since the new value of the SSI may be approximated using

$$\text{SSI}^{(k+1)} \approx \text{SSI}^{(k)} + \frac{d\text{SSI}}{dP_{S_i}}^{(k)} \Delta P_{S_i}^{(k)} + \frac{d\text{SSI}}{dP_{S_j}}^{(k)} \Delta P_{S_j}^{(k)}. \quad (11)$$

Observe that since $d\text{SSI}/dP_{S_i}^{(k)} > d\text{SSI}/dP_{S_j}^{(k)}$, this particular generation redispatch clearly results in an SSI improvement.

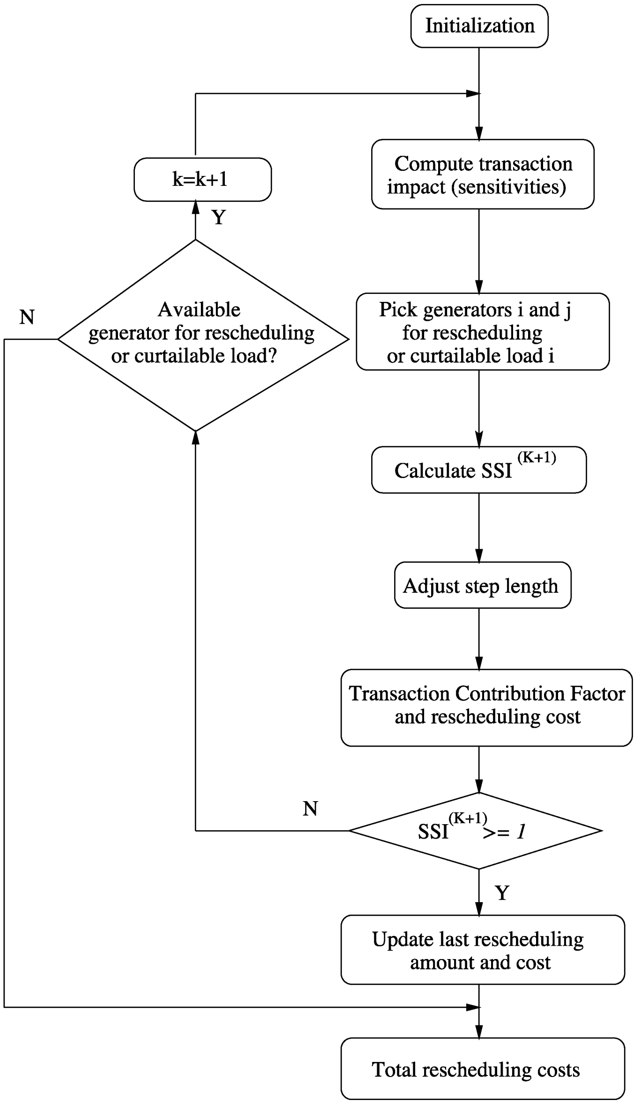


Fig. 1. Rescheduling technique for a simple auction system.

Since the whole process is based on a linearization, one cannot make large changes in generated power; otherwise, this might have a large effect on the actual SSI value, which changes nonlinearly as the parameters change, hence, the need for an iterative process. The amount of generation chosen for redispatch $\Delta P_S^{(k)}$ may be readjusted when determining the actual value of $\mathrm{SSI}^{(k+1)}$ using the full nonlinear system.

Only if there are no adequate generators available for redispatch is the load considered for curtailment. This approach is to be expected when the load is inelastic, as these types of loads require that the forecasted load be dispatched, given the high "costs" of load curtailment. In the case of elastic demand associated with demand-side bidding or loads with curtailment bids, however, the load could be considered for rescheduling in the same way as the generators, i.e., the load with the most negative impact on the SSI, say i, may be reduced by an amount that has a "significant" impact on the SSI value, as per the following approximation:

$$\mathrm{SSI}^{(k+1)} \approx \mathrm{SSI}^{(k)} - \left.\frac{d\mathrm{SSI}}{dP_{D_i}}\right|_c^{(k)} \Delta P_{D_i}^{(k)}. \tag{12}$$

When loads are rescheduled, the transaction level T is affected by the load reduction; thus, in this case

$$T^{(k+1)} = T^{(k)} - \Delta P_{D_i}^{(k)}.$$

Furthermore, generator power bids must be reduced in this case to compensate for the reduction in demand. This reduction of excess power generation will depend on the particular market rules. Here, we assume a reduction that considers the original generator's power bid, the amount to be rescheduled from previous iterations, as well as the transaction level; thus

$$\Delta P_{S_i}^{(k)} = \Delta P_{D_i}^{(k)} \frac{P_{S_i}^{(0)} - \sum_{j=1}^{k-1} \Delta P_{S_i}^{(j)}}{T^{(k)}}.$$

This could be considered a reasonable and fair mechanism to share the load curtailment; however, other mechanisms could be readily implemented, such as distributing the load reduction among the participating generators based on their SSI sensitivities.

C. Rescheduling Adjustment

The step changes in generation or load are readjusted by computing the actual value of $\mathrm{SSI}^{(k+1)}$ and comparing it to the approximated value computed using (11) or (12). If the difference is greater than a chosen tolerance, the previous step and this one are repeated with smaller changes in the supply and demand until a desired tolerance is met. This step is required to account for the system nonlinearities.

D. Rescheduling Cost

Based on the definitions of P_{G_o} and P_{L_o}, we will assume that the security cost incurred by potential transactions will not be distributed among these "must-run" generators and "must-serve" loads, as their prices are determined by different market mechanisms (e.g., must-run contracts and averaging market prices in the Ontario market [26]). Under this assumption, the rescheduling costs for the given iteration k are determined based on a transaction contribution factor (TCF), as defined by

$$\mathrm{TCF}_i^{(k)} = \frac{\frac{dSSI}{dp_i}^{(k)} p_i^{(k)}}{\sum_j \frac{dSSI}{dp_j}^{(k)} p_j^{(k)}} \tag{13}$$

where i stands for the bus number, and $p_i^{(k)}$ corresponds to the value of the corresponding parameter, i.e., the value of $P_{S_i}^{(k)}$ or $P_{D_i}^{(k)}$. Only buses with negative impact on the SSI, i.e., buses with $dSSI/dp_i^{(k)} < 0$, are considered in this computation; buses with positive impact are given a zero TCF value. The parameter values $p_i^{(k)}$ are included in this "normalization" process to account for the "size" of the corresponding transactions in the security cost.

Observe that these TCFs can be readily adjusted to account for the "base" generation or load P_{G_o} and P_{L_o}, by considering the sensitivities $d\mathrm{SSI}/dP_{G_o}$ and $d\mathrm{SSI}/dP_{L_o}$, which can be readily computed using (6) and (10) in (13).

The generation redispatch security cost of the kth iteration may be defined as

$$\mathrm{SC}_k = (C_{S_i} - \mathrm{MCP})\,\Delta P_{S_i}^{(k)} \tag{14}$$

where i is the generator chosen, as per Section III-B, with bid C_{S_i}. In the case of noncurtailable loads or inelastic loads, one can assume that there is a cost associated with curtailing the load (e.g., demand response incentives) that should be considered as part of the cost of keeping the system secure; thus, one would have, in this case, that

$$\mathrm{SC}_k = A_{D_i}\Delta P_{D_i}^{(k)} \tag{15}$$

where A_{D_i} is the "cost" of curtailing the load at the chosen bus i, which could be negotiated (e.g., bidding, contacts) or "imposed."

E. Convergence Check and Final Rescheduling Adjustment

If the SSI requirements are met, i.e., if $\mathrm{SSI}^{(k+1)} > 1$, then the iterative process stops, say, at $k = m$. At this point, the final generator or load reschedules are adjusted based on (11) or (12), respectively. Thus

$$\Delta P_{S_i}^{(m)} = -\Delta P_{S_j}^{(m)} = \frac{1 - \mathrm{SSI}^{(m)}}{\frac{d\mathrm{SSI}}{dP_{S_i}}^{(m)} - \frac{d\mathrm{SSI}}{dP_{S_j}}^{(m)}}$$

or

$$\Delta P_{D_i}^{(m)} = \frac{1 - \mathrm{SSI}^{(m)}}{\frac{d\mathrm{SSI}}{dP_{D_i}}^{(m)}}.$$

F. Transaction Security Cost (TSC)

The final transaction levels and security cost for each i node are readily determined as follows:

- Generators:

$$P_{S_i} = P_{S_i}^{(0)} + \sum_{k=1}^{m}\Delta P_{S_i}^{(k)}$$
$$\mathrm{TSC}_i = \sum_{k=1}^{m}\mathrm{TCF}_i^{(k)}\mathrm{SC}_k \tag{16}$$

- Loads:

$$P_{D_i} = P_{D_i}^{(0)} - \sum_{k=1}^{m}\Delta P_{D_i}^{(k)}$$
$$\mathrm{TSC}_i = \sum_{k=1}^{m}\mathrm{TCF}_i^{(k)}\mathrm{SC}_k$$

IV. WEB-BASED DESIGN AND IMPLEMENTATION

A. Architecture

The web-based application is designed using the three-tier client/server architecture, as shown in Fig. 2. The application logic is separated from the GUI and the backend database, so that changing any one of the layers can be done without the need to change other layers. Java is chosen as the programming language and software platform [31], so that the application can run on any hardware and software platform, given the presence of Java Virtual Machine (JVM); Java servlets are used for implementing secure, protocol-independent, and platform-independent server-side web-enabled computation modules; and Java Data Base Connectivity (JDBC), which is cross-platform and nondatabase product specific, is used for accessing and manipulating the relational database.

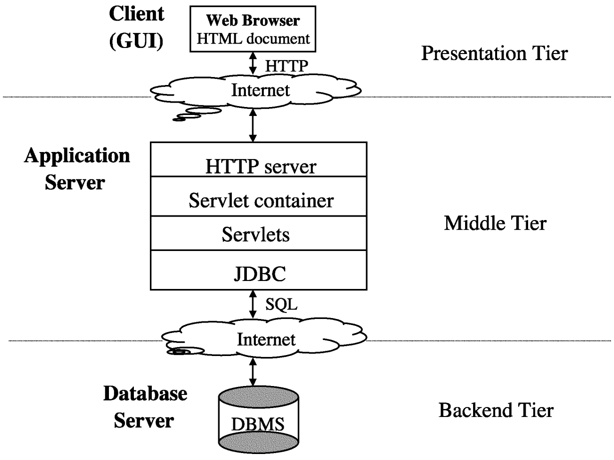


Fig. 2. Three-tier client/server architecture.

The top layer is the presentation layer or GUI, which is located on the client machine. Data are presented by a thin client solution, using a web browser and the Hyper Text Mark-up Language (HTML) standard file format. The parameters of the application are controlled by the client side using HTML forms. The user sets the parameters and then activates the process on the server side via the network. What the client sees is an abstract operation request that involves certain input and output parameters. Since web browsers are available for almost all platforms, using them as GUI eliminates the need for designing different application interfaces across different platforms and also allows to adequately present the application to the user without too much coding effort.

The middle layer is the application server, which implements the application logic to process data requests. An Hyper Text Transfer Protocol (HTTP) web server is running on the machine to receive HTTP request from clients and send HTTP response back to clients. A servlet container and servlet API must be installed to allow the servlets to run. JDBC API and a JDBC driver associated with the database should be installed to communicate with Data Base Management System (DBMS) using SQL query language statements. The main control logic is encapsulated on the server side through Java servlets, which represent different modules of the application. Using modular design enables modifications and enhanced features to be added easily to the system to properly respond to specific user requirements.

The backend layer is the database server, which uses a typical relational database that stores and manipulates the data at the backend. The connectivity to the database from the middle layer is facilitated through the JDBC drivers; this allows the application logic to be written with little dependency on the type of database use.

The way the system works is as follows: The user fills in a HTML form and clicks the Submit button, which posts the request to a Java servlet. The servlet reads the input parameters and performs the business logic and at the same time uses JDBC to communicate with a database to obtain necessary data; a response is then generated and given back to the client for display, depending on the user inputs.

B. Implementation

The prototype of the web-based application was implemented on Sun machines, using the Sun Solaris 2.8 operating system. A Sun Enterprise 450 machine is used in the middle tier for hosting the application server and web server, whereas a Sun Ultra 10 machine is used in the backend tier for hosting the database server. The client can be any machine that has a web browser installed with access to the Internet. The prototype was tested using a PC running Windows 2000 with either Internet Explorer or Netscape browser as a client.

In the middle tier, an Apache HTTP Server 1.3 is installed as the web server. Java Web Services Developer Pack 1-0-01 is installed for the support of the Java Servlet API. Java JDK-1.3.1 is installed as the Java runtime environment. MySQL Connector/J 2.0.14 is used as the JDBC Driver to access MySQL relational database. MySQL 3.23.52 resides on the backend tier.

Five major modules, implemented as five servlets, run on the application server.

1) *basecon*: Receives base system conditions from the client and calculates the base power flow.
2) *bidMatching*: Receives supply and demand bids from the client and performs a high–low bid matching.
3) *SSI*: Calculates SSI using the method described in Section II-A.
4) *impact*: Computes the impact of potential transactions on the SSI using the method described in Section II-B.
5) *rescheduling*: Analyzes security costs by means of the rescheduling technique described in Section III.

The pseudocode of servlets *basecon* and *SSI* are shown in Figs. 3 and 4 for illustration purposes.

UWPFLOW provides the computing engine for power flow and loading margin calculations [30]. It is a well-tested legacy software package that can be called by the Java servlets. Thus, the engine used for SSI calculations can be readily changed to any other package designed for these purposes. Thus, this package is encapsulated in servlets *basecon*, *SSI*, *impact*, and *rescheduling*, using Java *Runtime.exec*, which creates a native process and returns an instance of a subclass of *Process* that can be used to control the process and obtain information about it, as shown in Figs. 3 and 4.

A database interface is included in each module to retrieve and update information from the database. Its main logic is illustrated in Fig. 5.

```
Public class basecon extends HTTPServlet
{
    ...
    Public void doPost(HttpServletRequest request,
                       HttpServletResponse response)
          throws ServletException, IOException
    {
        PrintWriter out=response.getWriter( );
        response.setContentType("text/html");
        ...

        //parse the parameters of HTML form and
        //save the uploaded file

        //encapsulate UWPFLOW to generate base
        //power flow using Java Runtime.exec( )

        //Database Interface: update tables

        //generate response web page

    }
}
```

Fig. 3. Pseudocode of *basecon* servlet.

```
Public class SSI extends HTTPServlet
{
    ...
    Public void doPost(HttpServletRequest request,
                       HttpServletResponse response)
          throws ServletException, IOException
    {
        PrintWriter out=response.getWriter( );
        response.setContentType("text/html");
        ...

        //Database Interface: retrieve data from tables

        //use UWPFLOW to calculate loading margins
        //under normal condition and N-1 contingencies

        //Database Interface: update tables

        //generate response web page

    }
}
```

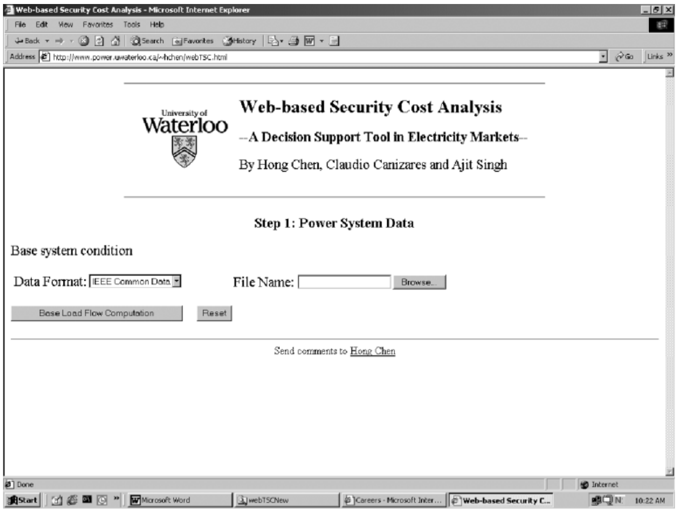
Fig. 4. Pseudocode of *SSI* servlet.

C. Results

The first input GUI of the application is depicted in Fig. 6. Figs. 7–9 depict the results for a six-bus test system, and Fig. 10 shows the results for a 129-bus system. A detailed explanation of the test systems and the results presented here can be found in [1] and [12]–[14]. The loading margin results for the six-bus system under normal system condition, and most critical contingencies are shown in Fig. 7. The most critical contingency, i.e., Line 2 to 4 outage, determines the SSI value as 0.8496. Since it is less than 1, transactions are not feasible, and a security cost occurs. The generation impact and load impact at initial point are shown in Fig. 8; generator 1 has the most positive impact, and generator 2 has the negative impact. When rescheduling strategy is used to facilitate potential transactions, the results are shown in Fig. 9; generator 1 is chosen as the constrained-on generator and generator 2 as the constrained-off generator, based on the impact at each iteration. The rescheduling results for

```
//loading driver
try{
    class.forName("org.gjt.mm.mysql.Driver");
}catch(ClassNotFoundException e){
    System.err.print("ClassNotFoundException:");
    System.err.println(e.getMessage( ));
}

try{
    //making the connection
    Connection con=DriverManager.getConnection(
                                Dburl, "login", "password");

    //creating JDBC statements
    Statement stmt=con.createStatement( );

    //create or update database
    stmt.executeUpdate("....");

    //retrieve information from database
    ResultSet rs=stmt.executeQuery("...");
    //retrieve values from ResultSet
    while(rs.next( )){
        float p=rs.getFloat("Pg");
        ...
    }
}catch(SQLException e){
     System.err.println("SQLException:" + e.getMessage( ));
}
```

Fig. 5. Database interface.



Web-based Security Cost Analysis

--A Decision Support Tool in Electricity Markets--

By Hong Chen, Claudio Canizares and Ajit Singh

Step 1: Power System Data

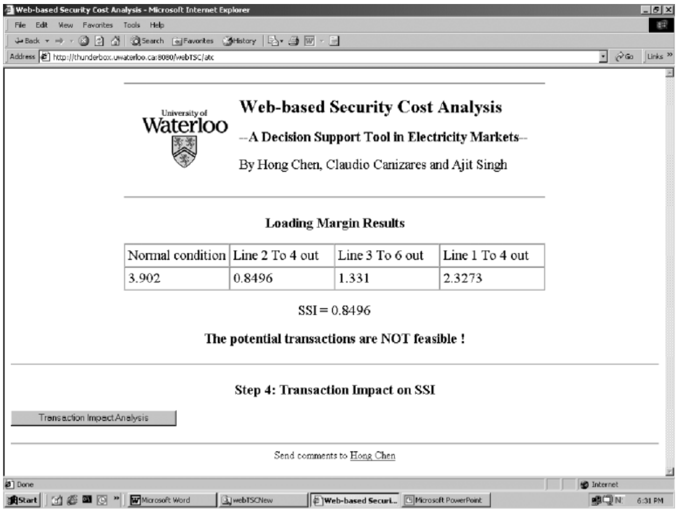
Base system condition

Data Format: IEEE Common Data File Name:

Send comments to Hong Chen

Fig. 6. Part of the input GUI.

the 129-bus system are shown in Fig. 10. Generator 12, 6, and 8 are chosen as the constrained-on generators at different iterations, while generator 13 is the constrained-off generator. After rescheduling a 60.44-MW generation, the potential transactions become feasible. The latest version of the application and the data for the six-bus test system can be accessed at the website http://www.power.uwaterloo.ca under the "downloads" menu (TSC analysis).



Web-based Security Cost Analysis

--A Decision Support Tool in Electricity Markets--

By Hong Chen, Claudio Canizares and Ajit Singh

Loading Margin Results

Normal condition	Line 2 To 4 out	Line 3 To 6 out	Line 1 To 4 out
3.902	0.8496	1.331	2.3273

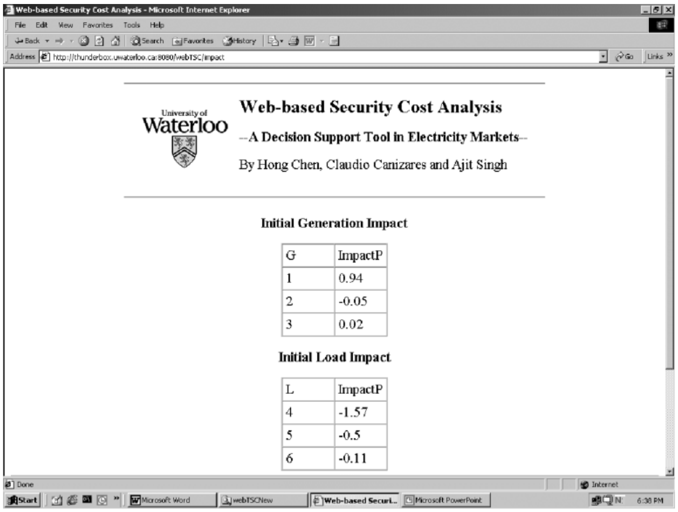
SSI = 0.8496

The potential transactions are NOT feasible !

Step 4: Transaction Impact on SSI

Send comments to Hong Chen

Fig. 7. SSI results for six-bus test system.



Web-based Security Cost Analysis

--A Decision Support Tool in Electricity Markets--

By Hong Chen, Claudio Canizares and Ajit Singh

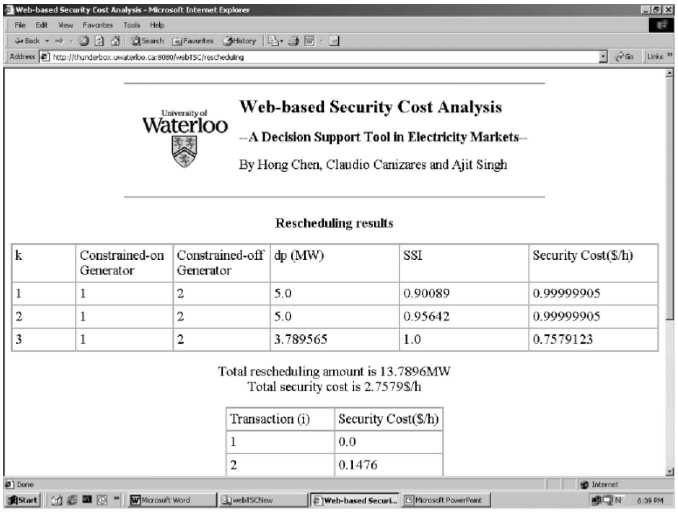
Initial Generation Impact

G	ImpactP
1	0.94
2	-0.05
3	0.02

Initial Load Impact

L	ImpactP
4	-1.57
5	-0.5
6	-0.11

Fig. 8. SSI sensitivity at initial point for six-bus system.



Web-based Security Cost Analysis

--A Decision Support Tool in Electricity Markets--

By Hong Chen, Claudio Canizares and Ajit Singh

Rescheduling results

k	Constrained-on Generator	Constrained-off Generator	dp (MW)	SSI	Security Cost($/h)
1	1	2	5.0	0.90089	0.99999905
2	1	2	5.0	0.95642	0.99999905
3	1	2	3.789565	1.0	0.7579123

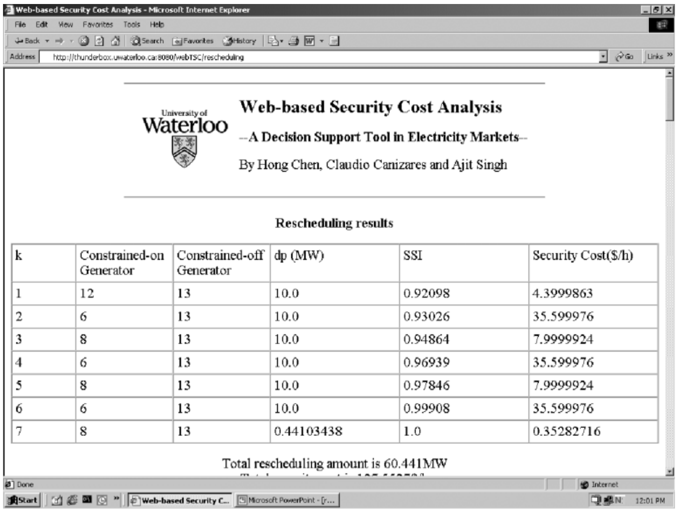
Total rescheduling amount is 13.7896MW
Total security cost is 2.7579$/h

Transaction (i)	Security Cost($/h)
1	0.0
2	0.1476

Fig. 9. Rescheduling results for six-bus system.

V. CONCLUSION

A novel design for a web-based security cost analysis system, based on the three-tier client/server paradigm, is presented with the prototype implemented using up-to-date web technologies. The underlying security cost analysis is based on an SSI and its sensitivities with respect to the power transactions, using a legacy continuation power flow program as the computation



Web-based Security Cost Analysis

--A Decision Support Tool in Electricity Markets--

By Hong Chen, Claudio Canizares and Ajit Singh

Rescheduling results

k	Constrained-on Generator	Constrained-off Generator	dp (MW)	SSI	Security Cost($/h)
1	12	13	10.0	0.92098	4.3999863
2	6	13	10.0	0.93026	35.599976
3	8	13	10.0	0.94864	7.9999924
4	6	13	10.0	0.96939	35.599976
5	8	13	10.0	0.97846	7.9999924
6	6	13	10.0	0.99908	35.599976
7	8	13	0.44103438	1.0	0.35282716

Total rescheduling amount is 60.441MW

Fig. 10. Rescheduling results for 129-bus system.

engine. Test results of six-bus test system and 129-bus system examples show the feasibility and simplicity of the proposed web-based security cost analysis implementation.

Using the modularized design, new functions, such as transient stability limits computation, could be easily added later on. The current web-based prototype could also be improved by increasing computation speed (e.g., by using more powerful hardware necessary for handling the anticipated level of user requests). The data security of the system can be enhanced by using the secure HTTPS protocol while keeping the rest of the software architecture essentially the same.

REFERENCES

[1] H. Chen, "Security Cost Analysis in Electricity Markets Based on Voltage Security Criteria and Web-Based Implementation," Ph.D. dissertation, Univ. of Waterloo, Waterloo, ON, Canada, 2002.

[2] F. C. Schweppe, W. C. Caramanis, R. D. Tabors, and R. E. Bohn, *Spot Pricing of Electricity*. Norwell, MA: Kluwer, 1988.

[3] R. J. Kaye, F. F. Wu, and P. Varaiya, "Pricing for system security," *IEEE Trans. Power Syst.*, vol. 10, no. 2, pp. 575–583, May 1995.

[4] L. Willis, J. Finney, and G. Ramon, "Computing the cost of unbundled services," *IEEE Computer Appl. Power*, vol. 9, no. 4, pp. 16–21, Oct. 1996.

[5] G. Strbac, S. Ahmed, D. Kirschen, and R. Allan, "A method for computing the value of corrective security," *IEEE Trans. Power Syst.*, vol. 13, no. 3, pp. 1096–1102, Aug. 1998.

[6] O. Moya, "Model for security of service costing in electric transmission systems," *Proc. Inst. Elect. Eng. Gen., Transm., Distrib.*, vol. 144, no. 6, pp. 521–524, Nov. 1997.

[7] A. Jayantilal and G. Strbac, "Load control services in the management of power system security costs," *Proc. Inst. Elect. Eng. Gen., Transm., Distrib.*, vol. 146, no. 3, pp. 269–275, May 1999.

[8] T. J. Overbye, D. R. Hale, T. Leckey, and J. D. Weber, "Assessment of transmission constraint costs: northeast U.S. case study," in *Proc. IEEE Power Eng. Soc. Winter Meeting*, Jan. 2000, pp. 903–908.

[9] B. Corniere, L. Martin, S. Vitet, N. Hadjsaid, and A. G. Phadke, "Assessment of the congestion cost and the risk of curtailment associated with Available Transfer Capability (ATC)," in *Proc. IEEE Power Eng. Soc. 2000 Winter Meeting*, Jan. 2000, pp. 891–896.

[10] F. Alvarado, Y. Hu, D. Ray, R. Stevenson, and E. Cashman, "Engineering foundations for the determination of security costs," *IEEE Trans. Power Syst.*, vol. 6, no. 3, pp. 1175–1182, Aug. 1991.

[11] R. Rajaraman, J. V. Sarlashkar, and F. L. Alvarado, "The effect of demand elasticity on security prices for the poolco and multi-lateral contract models," *IEEE Trans. Power Syst.*, vol. 12, no. 3, pp. 1177–1184, Aug. 1997.

[12] H. Chen, C. A. Cañizares, and A. Singh, "Transaction security cost analysis by take-risk strategy," in *Proc. Power Syst. Comput. Conf.*, Jun. 2002, pp. 1–6.

[13] C. A. Cañizares, H. Chen, and W. Rosehart, "Pricing system security in electricity markets," in *Proc. Bulk Power Syst. Dynam. Control-V*, Onomichi, Japan, Aug. 2001, pp. 199–209.

[14] C. A. Cañizares, H. Chen, F. Milano, and A. Singh, "Transmission congestion management and pricing in simple auction electricity markets," *Int. J. Emerging Elect. Power Syst.*, vol. 1, no. 1, pp. 1–30, May 2004.

[15] I. Slutsker, K. Nodehi, S. Mokhtari, K. Burns, D. Szymanski, and P. Clapp, "Market participants gain energy trading tools," *IEEE Computer Appl. Power*, vol. 11, no. 2, pp. 47–52, Apr. 1998.

[16] F. Shahzad and A. Z. Khan, "An application of Java for power system stability," *IEEE Power Eng. Rev.*, vol. 18, no. 4, pp. 45–46, Apr. 1998.

[17] Q. Zhao, G. Huang, X. Luo, and X. Wu, "A software architecture style for deregulated power markets," in *Proc. IEEE Power Eng. Soc. Winter Meeting*, Jan. 2001, pp. 1497–1502.

[18] B. Qiu and H. B. Gooi, "Web-based SCADA Display Systems (WSDS) for access via internet," *IEEE Trans. Power Syst.*, vol. 15, no. 2, pp. 681–686, May 2000.

[19] M. Marmiroli and H. Suzuki, "Web-based framework for electricity markets," in *Int. Conf. Elect. Utility Dereg. Restruct. Power Technol.*, Apr. 2000, pp. 471–475.

[20] W. L. Chan, A. T. P. So, and L. L. Lai, "Internet based transmission substation monitoring," *IEEE Trans. Power Syst.*, vol. 14, no. 1, pp. 293–298, Feb. 1999.

[21] H. Chen, C. A. Canizares, and A. Singh, "Web-based transaction operational feasibility analysis in electricity markets," in *Proc. IEEE Power Eng. Soc. Summer Meeting*, Jul. 2002, pp. 1528–1533.

[22] Y. Yuen and K. Lo, "Congestion management using Java technologies," in *Proc. Power Syst. Technol.*, Oct. 2002, pp. 1322–1326.

[23] Y. Lam and F. Wu, "Simulating electricity markets with Java," in *Proc. IEEE Power Eng. Soc. Winter Meeting*, Jan. 1999, pp. 406–410.

[24] J. Contreras, A. Losi, and M. Russo, "A JAVA/MATLAB simulator for power exchange markets," in *Proc. IEEE Power Eng. Soc. Power Ind. Comput. Appl.*, Jan. 2001, pp. 106–111.

[25] S. Chen and F. Lu, "Web-based simulations of power systems," *IEEE Comput. Appl. Power*, vol. 15, no. 1, pp. 35–40, Jan. 2002.

[26] "Market Rules of the Ontario Electricity Market," Independent Electricity Market Operator, Tech. Rep., [Online] Available: http://www.theIMO.com.

[27] C. A. Cañizares, Ed., "Voltage Stability Assessment: Concepts, Practices and Tools," IEEE-PES Power Systems Stability Subcommittee Special Publication, Tech. Rep. SPS101PSS, Aug. 2002.

[28] S. Greene, I. Dobson, and F. L. Alvarado, "Sensitivity of the loading margin to voltage collapse with respect to arbitrary parameters," *IEEE Trans. Power Syst.*, vol. 12, no. 1, pp. 262–272, Feb. 1997.

[29] ——, "Sensitivity of transfer capability margins with a fast formula," *IEEE Trans. Power Syst.*, vol. 17, no. 1, pp. 34–40, Feb. 2002.

[30] C. A. Cañizares *et al.*. (2000) UWPFLOW Program. Univ. of Waterloo. [Online] Available: http://www.power.uwaterloo.ca

[31] (1998) The Java Tutorial. Sun Microsystems. [Online] Available: http://www.javasoft.com/docs/books/tutorial

Hong Chen (M'98) received the bachelor's degree in 1992 and the master's degree in 1995, both in electrical engineering, from Southeast University, Nanjing, China, and the Ph.D. degree in 2002 from the University of Waterloo, Waterloo, ON, Canada.

She worked in Nanjing Automation Research Institute (NARI), Nanjing, China, from 1995 to 1998 and was engaged in developing EMS power application software. She is currently a Senior Engineer at ISO New England, Holyoke, MA, working on real-time market applications.

Claudio A. Cañizares (SM'00) received the Electrical Engineer diploma in 1984 from the Escuela Politécnica Nacional (EPN), Quito, Ecuador. He received the M.S. degree in 1988 and the Ph.D. in 1991, both in electrical engineering, from the University of Wisconsin-Madison.

He was with EPN from 1983 to 1993 , where he held different teaching and administrative positions. He is currently a Professor with the Electrical and Computer Engineering Department, University of Waterloo, Waterloo, ON, Canada, and his research activities are mostly concentrated in studying stability, modeling, and computational issues in ac/dc/FACTS power systems.

Ajit Singh received the B.Sc. degree in electronics and communication engineering from the Bihar Institute of Technology (BIT), Sindri, India, in 1979 and the M.Sc. and Ph.D. degrees from the University of Alberta, Edmonton, AB, Canada, in 1986 and 1991, respectively, both in computing science.

From 1980 to 1983, he worked at the R & D Department of Operations Research Group (the representative company for Sperry Univac Computers in India). From 1990 to 1992, he was involved with the design of telecommunication systems at Bell-Northern Research, Ottawa, ON, Canada. He is currently an Associate Professor at Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. He has published several research papers in the areas of software engineering, network computing, database systems, and artificial intelligence.