A Web based Computational Architecture for Power Systems Analysis

Quirino Morante, Alfredo Vaccaro, Domenico Villacci, Eugenio Zimeo
Dep. of Engineering - University of Sannio, 82100 Benevento – ITALY
villacci@unisannio.it

There are many phenomena that can compromise power systems operation, and they all need to be carefully analyzed in order to evaluate their impact on the security and reliability levels of the electrical networks. This process, known in technical literature as power system contingencies analysis, requires the estimation of the actual network state and the solution of the system state equations in order to classify critical events that could lead to system vulnerabilities.
This analysis requires large computational efforts in terms of both the number of case study to analyze, several hundreds for medium complexity networks, and the data sifting needed for checking operational limits violations. The entire computation requires, moreover, computation times typically less than few minutes for the information to be useful.
To address this problem, in the paper a distributed architecture based on Web and Grid computing is proposed. The architecture integrates a network of TCP/IP based remotely controlled units distributed in the most critical sections of the electrical network for fields data acquisition and dynamic loading of power equipments, a grid computing based solution engine for on-line contingencies analysis, and a web-based interface for graphical synoptic and reporting development. Experimental studies demonstrate the effectiveness of the proposed solution.

Introduction

Recent advent of the deregulated electricity market will lead to a sensible revision of the targets, the principles and the guidelines that should be adopted as reference in the management and control of electric systems.
This trend will give rise a natural evolution in the planning and modernization of plants that involve both the producers, who should activate proper services development processes in order to improve their trade activities and to attract new users, and the distributors, who are oriented to increase their plants effectiveness by a revision and a radical optimization of the existing infrastructure management.
In these contexts the recent technological innovations and the sensible growth of Information Technology could play a strategic role in the attainment of the new targets induced by the mutated market scenario. In particular, as concern plants updating, the development of Hi-Tech based solutions could be a useful tool to support the full exploitation of the existing infrastructures without requiring the employment of new complex and expensive realizations [1].
This trend has fueled the need for faster and more accurate contingencies analysis.
Traditional approaches proposed to address this problem refers, typically, to worst case scenarios and, consequently, give operational limits often too restrictive or, in the case when the real time conditions differ to the assumed reference values, highly conservative [1,3,4]. These approaches appears therefore to be inadequate in the new competitive scenario.
This has forced utilities to adopt an on-line based approach in power system contingencies analysis. With this approach a real time estimation of the system state variables is continuously updated by distributed data measurements and adopted as reference for the system state equations solution. This analysis if integrated with advanced tool for dynamic loadability assessment of power equipments, as proposed by several research works [1,4,5,6], could lead to a sensible improvement of the infrastructures exploitation at adequate reliability levels allowing system operators more realistic operational guidance in planning preventive and corrective actions aimed to mitigate the effect of critical contingencies [7].
These benefits could nevertheless be compromised for the large computational efforts required by the on-line computations in terms of the number of power flow solutions (several hundreds for medium complexity networks), dynamic components loadability calculations and the data sifting needed for checking equipment thermal limits violations [3]. The entire computation requires, moreover, times typically less than few minutes for the information to be useful. Thus much of the development of on-line contingencies analysis in recent years has concentrated on making this computation more efficient.
In this regard considerable research efforts have been oriented to develop dedicated computer architectures based on supercomputers or network of workstations for the fast solution of the power system state equations [9-13]. These platforms are designed to meet the computational requirements of security analysis for a reference electrical network, in both static and dynamic scenarios, given the maximum number of contingencies that should be analyzed, the interval time width of the dynamic simulations, the maximum computation time etc. However, the constant growth of the electrical networks complexity and the need of system operators to continuously improve the plant infrastructures exploitation rising the reliability levels, by more detailed and timely control actions and more advanced protective units, make these values subject to large variations. Consequently, the computational efforts required by on-line contingencies analysis may rise dynamically and so more scalable processing systems are required.
In addressing this problem Grid computing, a new paradigm for distributed computing, could play a strategic role. A computational Grid is a hardware/software infrastructure that provides coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations [14].
In agreement with these argumentation in the paper a distributed architecture, based on the Web and Grid computing, for power system contingencies analysis is designed and prototyped. The

proposed architecture integrates a network of remotely controlled units distributed in the most critical sections of the electrical network for field data acquisition and dynamic components loadability assessment, a grid computing based solution engine for the on-line contingencies analysis, and a web based interface for graphical synoptic and reporting development. Remote units are fully managed by advanced TCP/IP based services and include a prototype version of an Intelligent Electronic Device (IED) implementing advanced protective functionalities as far as standard industrial products, as energy and power quality meters. The employed IEDs are based on hardware microcontrollers equipped with web based functionalities and implement adaptive components thermal models, integrating physical modeling with adaptive learning techniques (e.g. semi-physical modeling), to dynamically update the components capability curves and to automatically detect if the contingency would case the system to violate the real equipment loading limits.
The proposed architecture has been applied to develop the contingencies analysis for the IEEE 118 nodes network.
The outline of the paper is as follows. A formalization of the contingencies analysis problem, in both static and dynamic regime, is given in Section 2. The role of distributed architectures in power system contingencies analysis is discussed in Section 3. In Section 4 a Grid-based distributed architecture for contingencies analysis is proposed. Section 5 contains the description of the experiments and a discussion of the results. Conclusions and future work are summarized in Section 6.

Problem formulation

The on-line security assessment can be divided in three sequential activities: (1) contingency screening; (2) static and dynamic contingencies analysis; (3) preventive and corrective control.
In this paper, the second activity is mainly analyzed since it is known to be the bottleneck in the on-line computations and it represents an essential tool in supporting the optimal identification of the preventive and corrective measures aimed to mitigate the effects of critical contingencies.
On-line static contingencies analysis requires the calculation of a power flow solution by using the real time data measurements from the power system to obtain the best estimate of the system state variables. This real time power flow solution, updated every few minutes, is adopted as reference in the automatic assessment of the static security of the system. During this process the solution engine automatically studies hundreds of possible contingencies that could happen on the power system determining how well the system can withstand them.
The real time power flow solution is also employed in the on-line dynamic contingencies analysis as the initial state vector of the power system state equations. Also, these equations should be solved for the entire set of contingencies and within a maximum interval time. In particular, since the system dynamics can be highly non linear, especially for large disturbances, the corresponding dynamic solutions are computed by time domain numerical integration, by using both explicit integration such as Euler or Runge Kutta methods, or implicit methods like Trapezoidal integration.
According to these argumentations, the main functional steps and the corresponding logical information flows required to address the on-line contingencies analysis in both static and dynamic formulation can be described by the UML [17] activity diagram depicted in fig.1. This procedure, that should be periodically activated for each control period, is characterized by some steps.
Initially, the field data from remote meters are acquired. Then a software routine that solves the static power flow problem is invoked to obtain the estimation of the state variables not directly acquired on the field. This state estimation is then adopted in both static and dynamic contingencies analysis as base case study and initial state respectively. In particular, for each contingency, the procedure generates an input file containing the network data modified by the effect of the considered contingency. This file is then used by dedicated software routines to solve the corresponding power flow problem in both static and dynamic scenario according to fixed setting parameters (as the stopping criteria adopted in the iterative state equations solution, the simulation time and the fault proprieties in the dynamic analysis, and so on).

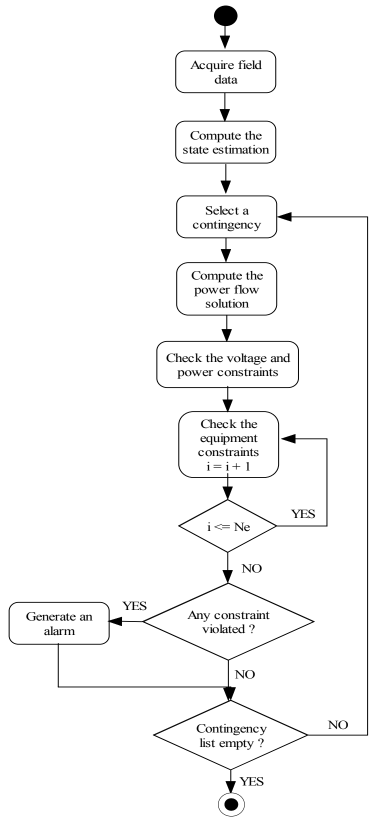

Figure 1: Activity diagram of a solution to the on-line security

Once the power flow problems are solved, the system behavior resulting from the considered contingency can be analyzed in order to check if the network technical limits, in terms of nodes voltages, nodes active and reactive powers and equipments loading, are violated. This process is performed by invoking dedicated routines that, starting from the computed power flow solution and the maximum allowable values, verify the limits violations and generate alarms in the presence of an expected system malfunctioning.
Unfortunately, the computational requirements for on-line simulating power systems dynamics represent the main impediment to the use of a single processor based architecture. Parallel processing based on distributed systems seems to be the only viable solution to speed up the simulations in order to obtain results in useful time.

Related Work

In recent years, distributed systems composed of pools of computers interconnected by commodity networks have demonstrated their effectiveness for solving computationally intensive problems, in the past solved only by supercomputers, such as state estimation, power system simulation, optimal power flow studies, security analysis [9].
The application of these solution engines is particularly gainful in on-line power system security assessment, where system operators have to predict in few minutes the impact on the electrical network of the set of credible contingencies determining corrective and preventive actions aimed to mitigate the effects of critical events.
In [10] a distributed processing environment based on a network of ten workstations has been proposed to solve the so called security constrained optimal power flow problem, i.e., to determine a minimum cost operation point which will not lead to overload if any contingency out of a given list occurs. In [11] the various functions of security analysis are mapped onto a network of workstations (LAN with SUN SPARC stations 1 and 2) which work as a pipeline with a continuous flow of base case conditions. This mapping is realized by the PVM (Parallel Virtual Machine) [18] distributed programming environment that allows programmers to exploit distributed computing in a local network of heterogeneous computers.
In [12] a PVM based on an homogeneous cluster of eight nodes (both DEC ALPHA and IBM SP2 machines) has been applied for the analysis of dynamic stability contingencies in on-line dynamic security assessment.
In [13] a parallel architecture based on eight PCs is designed to address the problem of dynamic security risk of voltage collapse for a reference network of 700 buses and 1100 circuits by analyzing 20 test cases selected from a list of 60 contingencies. These platforms are designed to address the problem of security assessment for a reference scenario described by a fixed network complexity, maximum number of contingencies to analyze, minimum control time period, maximum analysis time etc. However, this static design approach represents an intrinsic limitation of these architectures since an unexpected evolution of the assumed reference scenario could affect the overall processing performances requiring an upgrade of the hardware facility with further computational resources [19].
This technological evolution is not infrequent in power systems where the constant growth of the electrical networks complexity and the need for larger security and reliability levels of plant infrastructures lead to more detailed simulation analysis, lower control times, more advanced protective functionalities etc [3].
The simulation time in dynamic analysis typically focuses on the first few seconds after a disturbance, when most instabilities occur; however there are slower phenomena from seconds to several minutes that may give rise to instabilities. Such instabilities should be accurately analyzed in order to obtain higher accuracy levels in dynamic security assessment [3].
All these factors make the computational efforts required by on line security analysis to rise dynamically and, consequently, make necessary the adoption of powerful and easily scalable processing systems.
A further critical element emerging in security analysis computations deals with the safeguard of power equipments. This topic is assuming a major role in the deregulated market of electricity, where a malfunctioning power system could be responsible for serious damages to a large number of system operators having access to the shared energy resource[4,5]. This has forced electrical industries to improve power equipments protection accuracy by decentralizing the main control and monitoring functions according to the distributed computing paradigm. In this connection, modern trends are oriented toward the updating of Energy Management Systems with networks of intelligent protective units, equipped with suitable computational resources [8], and reliable control architectures able to support effectively intensive data exchanges [20]. As supporting tools in developing these activities, the application of TCP/IP based communication services and web based control architectures have been recently proposed in power systems' literature [20].
The integration of advanced remote protective units and web based processing represents therefore a further key element to address in developing a suitable architecture for power system security analysis.

A distributed architecture for Contingencies Analysis

The solution algorithm
To exploit the potentialities of distributed computing, the domain decomposition to parallelize the on-line contingency analysis has been adopted. This choice has been justified since functional decomposition does not guarantee good performances [12].
With this approach, the whole job is divided in similar tasks, each one assigned to a different processor. Since each task coincides with the sequential execution of the analysis of a single contingency.
With the proposed algorithm a state estimation of the electrical grid is firstly computed. Then a number N of contingencies are analyzed to evaluate the security of the electrical grid. For each contingency an independent slave task is created, which takes charge of (1) computing the power flow solution for the electrical grid, (2) checking voltages, power and equipments loading constraints, (4) eventually notifying alarms (see Fig. 2).

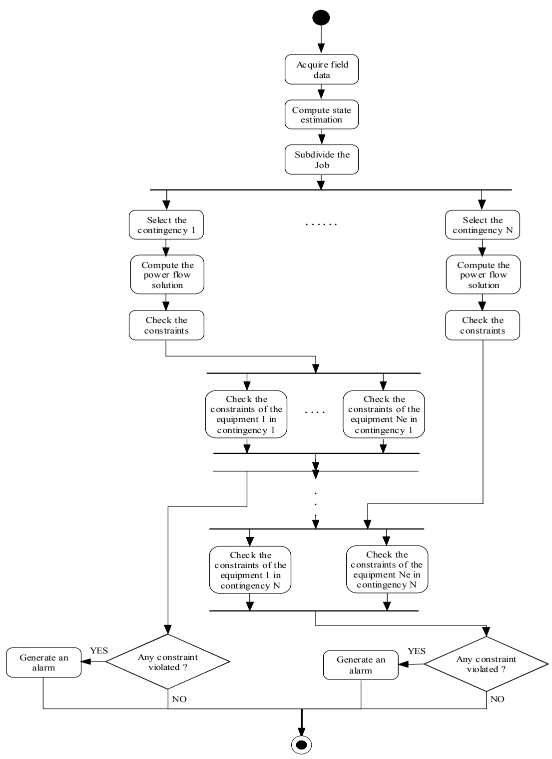

Figure 2: Activity diagram of the parallel algorithm for the power systems security analysis

As far as the check of the equipment loading constraints, it is developed by comparing, for each component, the calculated operating point with the corresponding load capability curve that gives the maximum allowable time duration for each hypothetical load. This curve, according to the principle of dynamic loading of power components, should be periodically updated in function of the real component thermal state and the expected environmental conditions.

The proposed architecture

To execute the on-line contingency analysis by adopting the parallel algorithm described in the previous section, a complex distributed architecture (fig. 3) has been used. It is composed of (1) Web-based components, (2) a network of distributed fields power meters (FEM), (3) a network of distributed intelligent electronic devices (IEDs), (4) a computational engine that may be based on a computational Grid and (5) a storage system (Real time Data Base Management System, RDBMS).

A Web-based architecture is used as a substrate on which the overall distributed system is deployed. So, the system is organized according to a complex three tier architecture where (1) the presentation tier is implemented by Web browsers used by power system operators, (2) the middle tier is composed of Web components for handling the presentation at server-side, for acquiring information from the electrical grid and for performing high performance calculations by using a computational engine, (3) the storage tier is composed of a remotely accessed RDBMS.

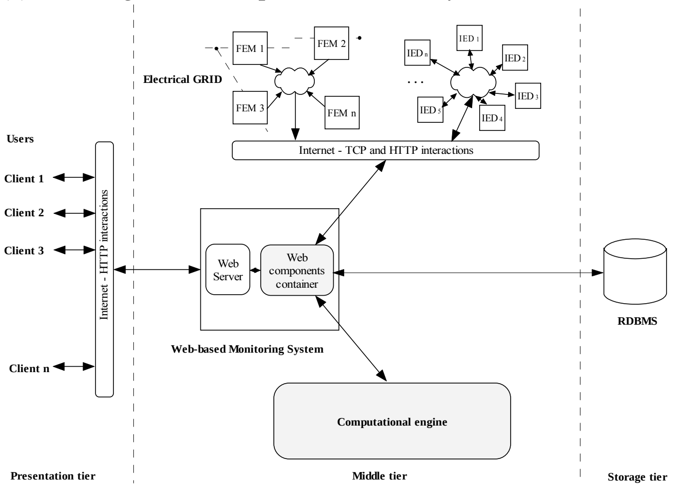

Figure 3: The proposed distributed architecture for power systems contingency analysis

Clients can monitor the electrical grid by interacting with a Web server connected to a container of web components. These are organized according to the Model View Controller (MVC) pattern [15] in order to assure flexibility both at client side and at server side. This pattern, in fact, subdivides an application in three kinds of components: views, used for presenting data; controllers, used for handling inputs; models, which represents the business logic of the application. This way, modifications either to the business logic or to the presentation logic have no impact on the other components of the system.

The schema has been useful exploited to integrate different computational engines in the distributed architecture without modifying the other components of the system.

The grid based computational engine

Data acquired from field power meters are used as input for performing power flow analysis. This activity is a compute-intensive task especially when a high number of contingencies should be analysed. To decrease the execution time for performing the power flow analysis, a concurrent algorithm based on the domain decomposition can be used. In fact the analysis of each contingency can be performed as a function of the overall input, independently of the analysis of the other ones. In particular, to usefully analyse the security of modern electrical grids, characterized by a more and more growing and variable structure, the computational engine has to assure high reliability, flexibility, and scalability. To achieve these goals, Grid computing represents a viable solution. In fact, this paradigm was put in practice in order to exploit the enormous amount of computing power available across the Internet to solve large-scale problems. However, to achieve this goal a special software infrastructure, called middleware, is to use. Such infrastructures extend the concept, introduced by PVM, of a "parallel virtual machine" restricted and controlled by a single user, making it possible to build large and dynamically extensible computing systems, based on network contexts characterized by different administrative domains, physical networks and communication protocols. Among the middleware platforms and toolkits developed in recent years [14], we chose to use a Hierarchical Metacomputer Middleware (HiMM), which has been developed at University of Sannio [16].

HiMM is suitable for solving the contingency analysis problem in the architecture shown in fig. 3 thanks to: (1) the adoption of a hierarchical topology; (2) the support for the object-oriented programming paradigm; (3) the use of the Java language for application development; (4) an easy integration with the Web technologies. In particular, the hierarchical topology is important for Grid-based applications as it allows (1) remote resource owners to enforce their own policies on external users [14], (2) applications to exploit the performances of dedicated networks which are not directly accessible by the Internet and (3) the system to be more scalable. In addition HiMM supports a brokering service, which allows a user to access grid resources in a simplified way, hiding the complexity of the underlying system. The broker delivers a specific service to easily submit a master/slave application by using its sequential version and some information necessary for the deployment. This approach guarantees a seamless sequential, concurrent and distributed programming so improving simplicity and code reuse in the parallel programming on a variety of hardware systems, such as SMPs, local area networks of PCs and the Internet. It supports the separation of concerns related to *functional* (algorithmic issues, such as the creation of high-level domain-dependent abstractions in the form of objects and classes) and *system* (lower-level tasks such as object distribution, mapping and load balancing) aspects of programming.

Since the decomposition of the contingencies analysis problem considered above can be exploited to implement a distributed and parallel solution following the master/slave programming model, the brokering service offered by HiMM is particularly suitable. A web component of the architecture shown in figure 4 submits the job to the broker. This uses information collected from the *Information System* in order to find resources and select those that satisfy the QoS requirements. A resource, selected by the broker, is used as master whereas the others are used as slaves.

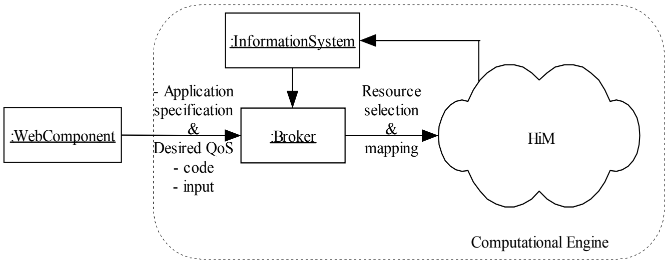

Figure 4: Computational engine based on a computational grid architecture

Experimental results

To evaluate the effectiveness of the proposed architecture in addressing the problem of contingencies analysis for complex electrical networks, some experimental tests was developed on the standard IEEE 118-nodes test network. For performing the experiments the following system was used: each remote IEDs was based on a ZWORLD-BL2100 unit, an advanced single-board computer that incorporates the Rabbit 2000™ microprocessor (operating at 22.1 MHz), 128KB static RAM and 256KB flash memory, 40 digital I/O channels, eleven 12-bit A/D converter inputs, four 12-bit D/A converter outputs, RS-232/RS-485 serial ports. The control software, including the gray box thermal modeling architecture and the TCP/IP based functionalities, has been developed by using the Dynamic C™ premier development suite. The overall middleware infrastructure to manage the distributed architecture was implemented by using the Java language and related technologies, provided by the JDK 1.3.x and the J2EE® 1.3.x platform. HiMM 1.0 was adopted to implement the computational engine. An Apache Web server was connected to the Tomcat/4.1.18 Web container. So, Java Servlets®, Java Server Pages® (JSPs) and Java Beans® was used as web components to respectively implement controllers, views and the model of the MVC pattern.

Tab. I

Resource Name	CPU type and frequency	Memory capacity	OS	PSSABenchmarck	Price ($ forPSSABenchmarck)
R0	Pentium III, 500 MHz	256 MB	Windows 2000	75 sec	4
R1	Pentium II, 500 MHz	500 MHz	Windows 2000	77 sec	4
R2	Pentium III, 500 MHz	64 MB	Windows 98	70 sec	5
R3	Pentium II, 350 MHz	128 MB	Windows 2000	9.56 sec	10

The resources that we made available to build a HiMM and their performance characteristics are described in table 1 and schematically depicted in fig.5. As far as the computational price, we have considered an hypothetical value (expressed in USA dollar) for each resources, varying with the computational power of the resource.

The resource R3 in particular is a cluster composed of eight homogeneous computers described in the table. These resources are accessed by a front-end, which interfaces with the cluster and with the outer. It is a Pentium II, 400 MHz, with 256 MB of RAM and equipped with Windows NT 4.0.

The servlet container is hosted on another computer which is of the same kind of R1 and the client is executed on a notebook with Pentium III, 650 MHz, 256 MB of RAM and equipped with Windows 2000. In particular in this experimentation the brokering service of HiMM was executed on the same machine of the client. All the described resources are interconnected by a Fast Ethernet LAN and their topology can be seen in figure 5. The PSSABenchmark consists of the evaluation of the mean execution time of a single contingency using a local request.

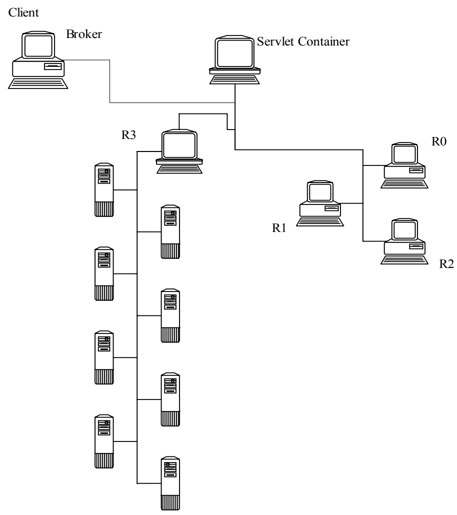

Figure 5: The experimental testbed

The experiments are referred to 186 credible contingencies related to the break of one of the 186 transmission lines of the considered power grid.

The following experimental results were obtained by using the test-bed described above, varying the deadline (max execution time) and the maximum budget allowable.

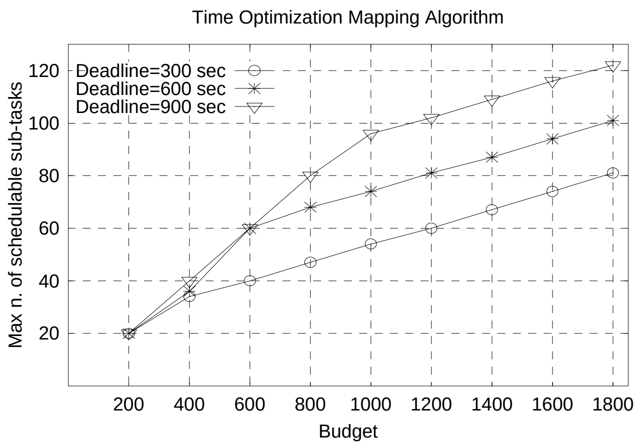

Figure 6: Max # of contingencies for different budget and deadline values

Fig. 6 shows a provision of the maximum number of contingencies that could be analyzed with the adopted resources, considering three different deadlines (300, 600 and 900 sec), and various budgets. At limit conditions, i.e. with execution time that reaches the deadline and the expense that reaches the budget, the time optimization and cost optimization algorithm show the same results.

In particular, the figure states that given a deadline, there is a maximum number of tasks that can not be exceeded even with a greater budget. This behavior is due to the enforcement of a maximum deadline, the set of available resources and the limitation of their performances. For example, to execute 50 contingencies within a time of 300 sec, more resources or more powerful ones would be necessary.

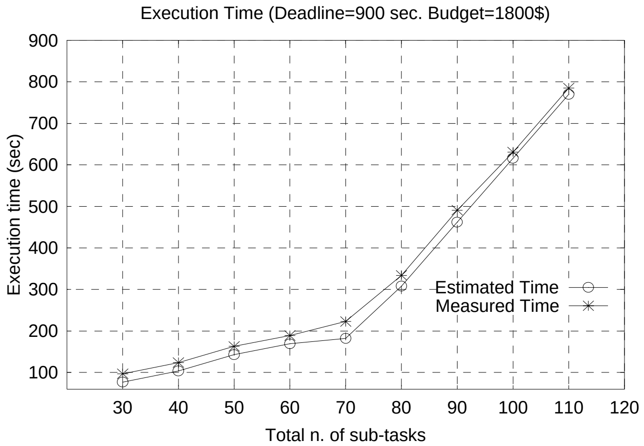

Figure 7: Predicted and measured execution times with a fixed budget

Fig. 7 shows the execution time actually measured by executing the application on the test-bed, varying the number of contingencies, fixing the deadline and the budget and adopting the time optimization algorithm. The figure shows also the difference between the measured execution time and the execution time estimated by the broker mapping algorithm. The difference between each couple of values is due to two reasons: (1) the mapping algorithm currently implemented does not take into account overheads and communication tasks; (2) the resources, both computers and networks, are supposed to be dedicated, but in a more realistic scenario these are shared.

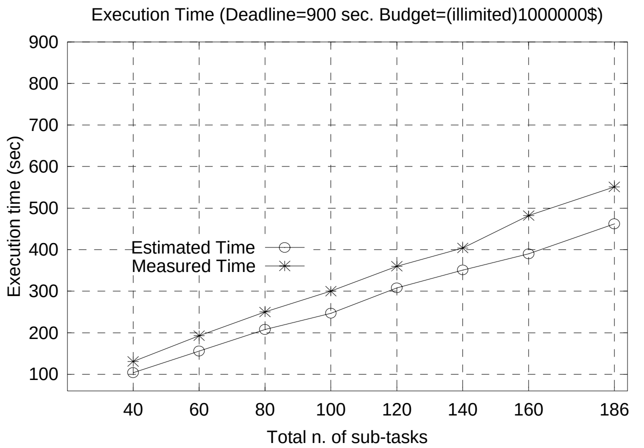

Figure 8: Predicted and measured estimation times with an unlimited budget

Finally, in order to estimate the execution times obtained with HiMM, a further experiment was conducted with a deadline of 900 sec. and an unlimited budget. This way, only the fastest resources were used and the execution time was minimized. Fig. 8 shows that for simulating 186 contingences the execution time is 551 sec.

Conclusion

The changing scenario in the energy market asks the system operators for faster and more accurate methodologies of security assessment. In addressing this problem, a distributed architecture for power system security assessment is proposed. The architecture integrates a network of remotely controlled units distributed in the most critical sections of the electrical network for field data acquisition and advanced protective functionalities, a solution engine based on distributed computing for the on line analysis of the system security, and a web based interface for graphical synoptic and reporting development. The paper shows that Grid computing could play a strategic role to guarantee better accuracy in power system security analysis since it is able to furnish on-demand a dynamically variable computational power for solving large scale problems. By using a computational grid, energy operators can buy at low-cost high performance computational resources only when their monitoring systems have to perform intensive real-time computations.

The Grid middleware platform adopted uses a broker system for reserving on-demand computational resources and for automatically splitting the security analysis task in sub-tasks to allocate onto reserved resources according to the master/slave computing model. The encouraging experimental results confirm the usefulness of computational Grids for performing an effective on-line security analysis of large electrical grids.

Further works will address the integration of the proposed architecture with other power systems applications by designing a web services based framework for electric utilities that, thanks to the adoption of an open standard, allows any piece of software to communicate with each other with a standard XML messaging system. The availability of sophisticated integration technologies, including complete web based functionalities opens the ways to interesting future scenarios like the improvement of interoperability of heterogeneous applications, the promotion of the interoperability in energy control centers, the development of an integrated utility information system.

References

[1] G.D'Alessio, S.Corgnati, A. Vaccaro, D.Villacci, "A novel architecture for the optimal control and management of distribution system" European Transactions on Electrical Power (ETEP) - **15**(5), March 2005

[2] C.W. Taylor, "The future in on-line security assessment and wide area stability control", *IEEE/PES Winter Meeting*, 2000, pp.78-83.

[3] A. Bose, K. Tomsovic, "*Power system Security*" in Electrical Engineering Handbook

[4] D.Villacci, G. Bontempi, A. Vaccaro, M. Birattari, "The role of learning methods in the dynamic assessment of power components loading capability" IEEE trans. On Industrial Electronics – **52**(1), February 2005.

[5] Study Committee 23-CIGRE', "Dynamic loading of transmission equipment", *Electra, Vol. 202*, 2002, pp. 63-73.

[6] D.Villacci, A. Vaccaro, G. Bontempi, "A semi-physical modelling architecture for dynamic assessment of power components loading capability". *IEE generation transmission and distribution*, **151**(04) July 2004.

[7] P. Daponte, D. Grimaldi, A. Piccolo, D. Villacci, "A neural diagnostic system for the monitoring of transformer heating", *Elsevier Measurement*, **18**(1), 1996, pp 35-46.

[8] IEEE committee report, "Parallel processing in power system computation", *IEEE trans. On power systems*, **7**(2),1992, pp.629-638.

[9] O.R. Saavedra, "Solving the security constrained optimal power flow problem in a distributed computing environment" *IEE proc. of Gener., Transm. Distrib.*, **143**(6), November 1996, pp. 593-598.

[10] A.B. Alves, A. Monticelli, "Static security analysis using pipeline decomposition", *IEE proc. of Gener., Transm. Distrib.*, **145**(2), March 1998, pp. 105-110.

[11] G. Aloisio, M. la Scala, R. Sbrizzai, "A distributed computing approach for real time transient stability analysis", *IEEE Trans. On Power Systems*, **12**(2), May 1996, pp. 981-987.

[12] A.M. Leite de Silva, J.L. Jardim, A.M. Rei, J.C.O. Mello, "Dynamic security risk assessment", proc. IEEE/PES Summer meeting,1999, pp.198-205.

[13] I. Foster, C. Kesselman and S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", *Intl Journal of Supercomputer Applications*, **15**(3) 2001.

[14] F. Bushmann et al., "*Pattern-Oriented Software Architecture: A System of Patterns*". J. Wiley and Sons, 1996.

[15] M. Di Santo, F. Frattolillo, W. Russo and E. Zimeo, "A component-based approach to build a portable and flexible middleware for metacomputing", *Parallel Computing* (North-Holland 2002).

[16] OMG, "Unified Modeling Language Specification" (N. formal/01-09-67), v. 1.4, Sep. 2001.

[17] V. Sunderam, J. Dongarra, A. Geist and R. Manchek, "The PVM Concurrent Computing System: Evolution, Experiences, and Trends", *Parallel Computing*, 20(4) (1994) 531–547.

[18] K. Morison,"Power System Security in the New Market Environment: Future Directions" proc. of *IEEE Power Engineering Society Winter Meeting,* 2000, pp.: 78 –83

[19] B. Qiu, H. B. Gooi, Y. Liu, E. K. Chan," Internet-based SCADA Display System" *IEEE Computer applications in Power 15(1)*, 14-19, Jan. 2002.