Multipurpose power system simulator: implementation based on modern principles

Ivan Naumkin, Mikhail Balabin, Nikolay Lavrushenko, Roman Naumkin

Abstract--**This paper presents the modern principles of implementation of a multipurpose software MAES for transient processes simulation in electric power systems.**

Keywords: **power system simulator, offline simulating, remote access, real-time, cross-platform programming, heterogeneous system, Software as a Service, overcurrent and overvoltage protection, power system accident analysis.**

I. INTRODUCTION

Digital simulators of electric power systems have found wide application in power industry and allow to solve a wide range of important tasks, such as:

1. Producing of the computational experiments when
   - doing research work in developing new technical measures that provide the desired reliability for consumer's power supply;
   - implementing projects concerned with the choice of the main electrical equipment and its parameters, the choice of overcurrent and overvoltage protection devices;
   - analyzing power system accidents and developing measures aimed at preventing similar accidents in power system's operation.
2. Performing simulations in hard real time for:
   - testing of existing or experimental new electrical devices (digital devices of relay protection and emergency automation, registrators of emergency events, monitoring devices, etc.);
   - designing, testing and implementation of new software products (simulators, SCADA, etc.).

The currently popular electromagnetic transient simulation software is actively expanding, such as EMTP [1]-[3], PSCAD/EMTDC [4], etc. The following real-time hardware/software systems are being developed worldwide: Real Time Digital Simulation for the Power Industry (RTDS Technologies Inc. Winnipeg, Manitoba, Canada) [5], RT-LAB, Distributed Real-Time Power (OPAL-RT Technologies Inc. Montréal, Canada) [6], HYPERSIM, Real-time Digital Power System Simulation (Silicon Graphics Inc. (SGI) and Hydro-Quebec TransEnergie Technologies) [7], etc.

Many of the software and hardware products stated above are the result of many years' development started at the beginning of 70s-90s years of the XX century. Methodological principles of implementing products were based on IT-technology of that time. In particular, the software was intended for local installation; programs were designed for single processor computers running under particular operating system; network solutions were designed to work at local area networks. Later power systems software modernization was carried out mostly without fundamental review of these principles.

The rapid development of computer technology at present days led to the necessity of the revision of digital power systems simulators design concept. The following requirements should be taken into account in the development of advanced software:

1. Due to the existence of different platforms the user should be able to work in any operating system on a broad range of 32- and 64-bit platforms, thus a software product must be **cross-platform**.
2. The mass changeover to multicore processor requires application of **parallel programming**.
3. The open systems and Open Source systems are growing in popularity, so it's necessary to focus on the use of **open protocols and standards** when creating the software.
4. Users' desire to **reduce their efforts and financial costs** of supporting current versions of the product must be taken into consideration.
5. Mobility of users and wide spread of distance learning require the use of **network technologies** for access to the services of the software product from all over the world.
6. The popularity of the «thin client» techology and increasingly inexpensive computers like netbook, should not restrict the user to use all the features of the product, regardless of the **technical characteristics of a working PC**.

The paper presents the results of a new version of MAES (Modeling and Analysis of Electric power Systems), implemented considering the requirements stated above.

I. Naumkin is with Siberian Electric Power Research Institute (SibEPRI), Novosibirsk, 630126 Russia (e-mail of corresponding author: ie-nau@cn.ru).
M. Balabin is with SibEPRI, Novosibirsk, 630126 Russia (e-mail: mbalabin@googlemail.com).
N. Lavrushenko is with SibEPRI, Novosibirsk, 630126 Russia (e-mail: nlavrush@gmail.com).
R. Naumkin is with SibEPRI, Novosibirsk, 630126 Russia (e-mail: naroiv@gmail.com).
Paper submitted to the International Conference on Power Systems Transients (IPST2009) in Kyoto, Japan June 3-6, 2009

II. THE DESCRIPTION OF PROGRAM MAES

A. *History of creation MAES*

In 70s of the last century program VOLNA for analysis of electromagnetic transients has been developed in the Siberian Electric Power Research Institute (SibEPRI). A Bergeron's model was used for symmetrical three-phase transmission lines, element's differential equations were integrated using modified explicit 4th-order Runge-Kutta method. The program allowed to make research in the field of half-wave and EHV electric power transmission lines (500-1150 kV).

In the 1980-81 biennium the first version of MAES had been developed based on VOLNA program. However, its design used quite different approach of constructing a total system of algebraic equations based on the Kirchhoff's Current Law. Time quantization was made with a implicit Euler's method and trapezoidal rule, the solution of linear algebraic equations system was made with Jordan elimination method. The overshooting problem was solved by recalculation in the respective points. To eliminate the numerical oscillation caused by switching events occurring when applying the trapezoidal rule, the local switching to the implicit Euler method was used.

The program MAES had the same evolutionary path of development that other similar systems had: the later stages involved the development of the graphical user interface (GUI), then the parallel implementation of the program was created for multiprocessor systems (clusters and SMP-computers), and finally, a software and hardware real-time complex was developed.

The Siberian Electric Power Research Institute has been using MAES to make decisions on power equipment (switches, surge arrestors), on overcurrent and overvoltage protection, and on power system accident analysis. Simulation reliability and accuracy are confirmed by comparison with power system accidents data.

New version of MAES has been developed since 2006. The simulator is a unified multipurpose system capable to work in three modes: stand-alone application for offline simulation system with remote access, real-time system. MAES's architecture is based on several principles for developing advanced software systems that meet the stated above requirements: modular multi-tiered architecture, cross-platform implementation, adoption of the Software as a Service (SaaS) concept [8], parallel programming.

B. *The structure of MAES*

Creation of a system with remote access requires a modular architecture of software. The modern approach to the development of multi-tier software implies the usage of two-or three-tier architecture. The de facto standard in the SaaS-systems design is a three-tier architecture (see Fig. 1), chosen in the MAES program design.

At the first level (the level of the user interface) there is a Schematic Editor module, as well as visualization and analysis of the simulation results module. This level offers a multilingual user-friendly graphical interface that allows to visualize the creation and editing of the scheme of electrical grids.

The second level (middleware level) provides access to the computational core to remote users and performs the allocation of shared resources. Middleware acts as an proxy between the client software, and computation center software, providing end-users with the following services:

- Authorization and billing of remote users. Each user is given its own system account containing a: user profile, balance information, the list of the used services.
- Simulation results storage.
- The implementation of the task queue for the tasks

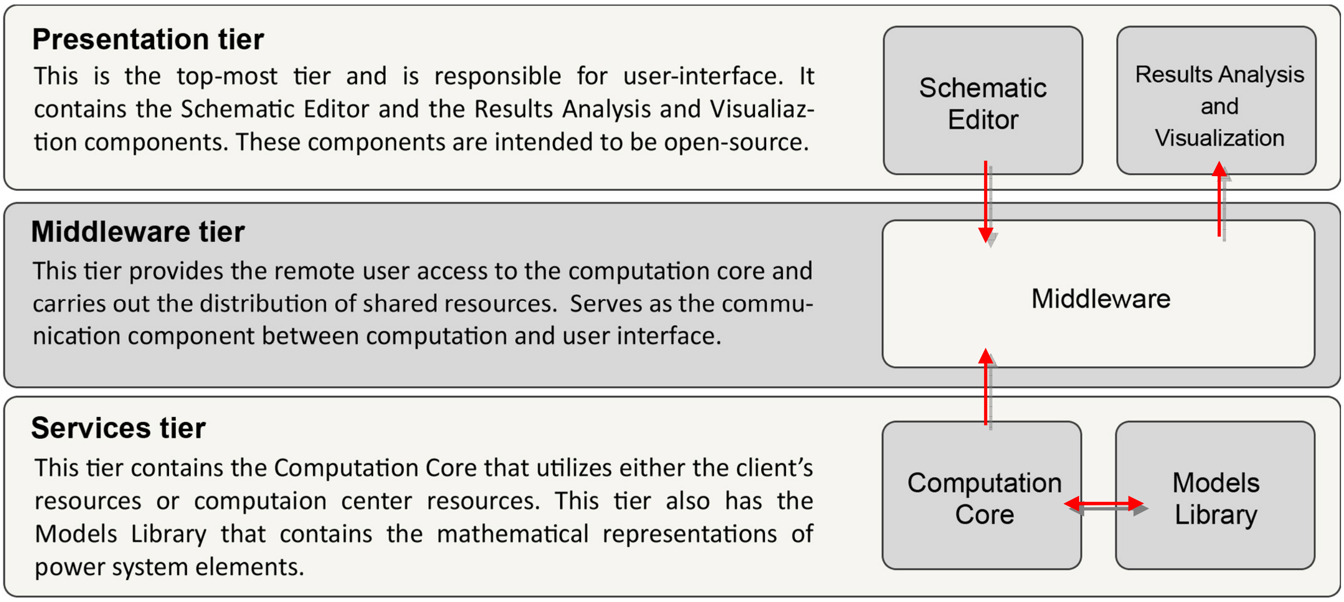


Fig. 1. MAES architecture.

allocation and execution using processing power, monitoring and control of queued computational tasks or computations themselves, the user can remove a task from the queue, stop the calculation or launch it again.
- Opportunity to communicate with the technical support and expert teams to interpret the simulation results.

The middleware level is used only in the one of three modes of the MAES program operations, namely in the remote access mode. Concerning standalone-mode and realtime-mode, the representative level and the level of services implementation interact directly without middleware participation.

At the third level (the level of services implementation) there are the Computational Core of the system and the library of mathematical models of power system elements. MAES's Computational Core is directly responsible for the simulation process and can run on parallel computing systems as distributed memory (cluster) systems, as shared memory systems (multiprocessor and multicore). The basic elements of modeled power systems used for constructing circuits, are contained in the simulator's library. Each element in the system is defined by its mathematical model, information about its parameters and pins location and types, as well as element's graphical representation. There is a facility to add elements constructed from existing ones to the library and the API (Application Programming Interface) to develop and add new elements to the system. The sets of element parameters can be stored in Models Library to ease the following use of them as a part of another scheme.

Intercommunication between levels is implemented by network protocol based on open XML standard. XML-based protocols are platform-independent, thus relatively immune to changes in technology. The strict syntax and parsing requirements make the necessary parsing algorithms extremely simple, efficient, and consistent. Also forward and backward compatibility are relatively easy to maintain. The protocol handles transfer of user-constructed scheme to be calculated with a given computational parameters of a simulation and obtaining results of the simulation back. Also it provides: interactivity during the simulation, data exchange while simulating for redrawing of commutator elements in the scheme, management of simulating process (cancel, pause, resume simulation).

C. *Cross-platform implementation*

Cross-platform implementation makes it possible to execute different MAES tiers on various platforms providing a number of advantages. Different operating systems may be used as in case of computational center as for an end-user allowing flexible approach of utilizing one center (with no dependence of the end-user OS) by users with different operating systems. Hence the migration from the 32-bit to 64-bit platforms and to different operating systems is painlessly acquired.

Nowadays there are several approaches to the development of platform-independent software: the creation of web applications, the use of virtual machines, the use of cross-platform application frameworks. The common technologies of web applications are either proprietary (Adobe Flash, Adobe Air, Microsoft Silverlight), or lack functionality (AJAX) for the implementation of the client software. Virtual machines (Java, Python) make high demands on computational resources, providing a poor performance on mobile computers, such as netbooks. Application frameworks (GTK, QT, wxWidgets, etc.) are not exposed to the above-listed flaws of other technologies. Therefore the cross-platform framework Nokia Qt4 is chosen as a basis of program MAES. Qt4 is an object-oriented C++ library which provides not only the facilities for GUI creation (compared to many other cross-platform C++ libraries) but methods to support network communications, databases and other services.

D. *Software as a Service Concept*

One of the main methods of MAES usage is remote access mode. Thus, MAES is not just a computational program for simulating transient processes, but it is a remote service which may be accessed by users via global, corporate and local networks. Access from the user side to the MAES services is committed by client software. User registers in the system and obtains an account which enables him to gain access to computational process management and to get technical and expert support. Client Software allows to make a scheme of electrical grid, to set up parameters of its calculation, to connect to server, to pass authorization procedure and to request for scheme calculation. It is possible to watch the state of accepted request and keep trace of calculations. When the calculations are finished client software loads results to user's computer for visualization and analysis.

Remote access to the computational services has a number of features. User is not tied to a workplace where the software is installed. He may access MAES while working at home or out of office. Being anywhere in the world and having a mobile computer (for example, netbook), the user can gain access to the resources of high performance computational center for solving large-scale problems.

Computational core modernization takes place transparently to users and update of the client software is carried out in automatic mode without requiring any user's actions.

Considering the fact the client software is updated automatically, just a single installation is needed, after which there is no need to manually update it regularly, which reduces the cost of servicing the system as a whole. Rental of computational resources from a remote computer center saves users from many of the concerns and costs associated with the organization and support of their own computer center. The organization of such center involves the purchase of expensive equipment, professional tuning of the system and regular maintenance. In addition, currently a large part of the cost of maintaining the high performance computer systems accounts for electricity. Thus, outsourcing of the computational center services leads to a significant reduction in costs of IT technology. Also, it makes possible to try the system in the real-life environment without significant costs and, if necessary, to begin to use it quickly. On the other hand, a large organization, for example, the University could afford a unified computer center available to their employees.

Internet has become one of the main means of distance education. Modern simulating tools are essential for students studying remotely or raising one's professional level in the field of power industry. Remote access capability of MAES can be used by universities as a part of the process of distance learning. The student gets a task over the Internet. Then he makes the necessary calculations to perform the task and saves the results on the server using remote access, and finally the professor verifies student's work stored on the server.

The network structure of the system facilitates the co-ordination of the team working on a common project, enabling co-operation of several geographically separated scientific or engineering communities working on the problem, remote testing of expensive and unique equipment resided in remote computing centers.

III. MAES Algorithms

A. *Mathematical basis of program MAES*

The complex technical systems may include interacting subsystems with various physical nature. Such technical systems are called heterogeneous systems. For example, the power system may contain such subsystems as the electrical subsystem, which includes transformers and power transmission lines; mechanical subsystem, consisting of turbines and other mechanical components; thermodynamic subsystem, including boilers, superheaters and other thermal components and control subsystems. Sometimes there are problems that require simultaneous detailed simulation of several parts of heterogeneous technical systems to study their interactions. Examples of heterogeneous systems, simulating which is of interest, are: the study of behavior of the emergency automation of power unit during the accident on the connected power lines, the study of interlocked gasdynamic processes in the pipeline and electro-mechanical processes in the gascompressor units in conjunction with supplying electric power system.

MAES architecture supports the simulation of the broad class of technical systems defined by the set of the multipole components, whose outlets (pins) are connected in nodes. Multipole components (see Fig. 2) can have an arbitrary number of pins. There are one or more state variables corresponding to each of the pins. Such state variables are called coupling variables.

A mathematical model of a multipole component is a system of equations on its coupling variables and internal variables. The system has an arbitrary number of nodes, in which the pins of multipole components of relevant physical nature are connected, and it forms a hypergraph. Pins of the multipole components belong to one of two types: energy type

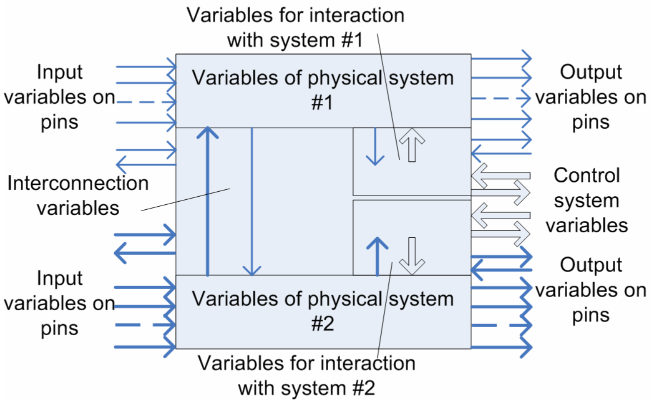


Fig. 2 Multipole component of heterogeneous system.

pins, each of which is corresponded by 2 variables (called flow variable and potential variable) and information (signal) pins, which are corresponded by a connection variable with a arbitrary physical meaning.

The combination of equations for all multipole components on a graph is supplemented by the Kirchhoff's Current Law equations for the flow variables. Point of reference is chosen for potential variables which is assumed as a potential of zero-node (basis node). Kirchhoff's laws apply to all nodes, except of basis node. Thus, the mathematical model of heterogeneous physical system occurs in the form of the following algebraic-differential system of equations:

$$\mathbf{U}_0 = 0;$$
$$[\mathbf{A}_\gamma] \cdot \mathbf{I}_{M\gamma} = 0;$$
$$\frac{d\mathbf{X}}{dt} = \mathbf{F}(\mathbf{X}, t) = \mathbf{W},$$

where $\mathbf{A}_\gamma$ is where is the incident matrix for the physical system γ,

$\mathbf{X} = (\mathbf{I}_M, \mathbf{U}_N, \mathbf{Z}_K)$ is a set of variables of heterogeneous system,

$\mathbf{I}_M = (I_{m_1}, I_{m_2}, ..., I_{m_s})$ are the flow variables on the element outlets,

$\mathbf{U}_N = (U_{n_1}, U_{n_2}, ..., U_{n_s})$ are the potential variables on the outlets of the elements,

$\mathbf{Z}_K = (Z_{k_1}, Z_{k_2}, ..., Z_{k_s})$ are the additional internal variables.

The first equation represents a basis node, the second is formed under the Kirchhoff's Current Law and the third is obtained from the totality of multipole equations.

Differential equations of the obtained equation system then undergo time quantization. It was shown [9] that numerical solution of stiff systems requires application of implicit multistep methods. Three linear multistep methods are implemented in MAES simulator: trapezoidal rule and backward Euler method in the Liniger-Willoughby form and the second order BDF method in the parametric form.

Adams-Moulton method in parametric form (Liniger-

Willoughby form) is expressed by the formula:

$$\frac{x_{n+1} - x_n}{h_{n+1}} = (1-\xi) f(x_n, t_n) + \xi \cdot f(x_{n+1}, t_{n+1});$$

or in the form of the two-step formula:

$$x_{n+1} = x_n + \xi h_{n+1} f(x_{n+1}, t_{n+1}) + (1-\xi) h_{n+1} f(x_n, t_n).$$

If $\xi = 1$ the formula coincides with backward Euler formula. If $\xi = 1/2$ the formula coincides with trapezoidal rule. Liniger-Willoughby method is A-stable if $\xi \geq 1/2$.

For the backward differentiation method the following parametric form was proposed by the authors of this paper:

$$\frac{x_{n+1} - x_n}{h_{n+1}} - (1-\xi)\frac{x_n - x_{n-1}}{h_n} = \xi \cdot f(x_{n+1}, t_{n+1});$$

or in the form of the three-step formula :

$$x_{n+1} = \left[1 + \frac{h_{n+1}}{h_n}(1-\xi)\right] x_n - \frac{h_{n+1}}{h_n}(1-\xi) x_{n-1} + \xi h_{n+1} f(x_{n+1}, t_{n+1}).$$

If $\xi = 1$ the formula coincides with backward Euler formula; if $\xi = \frac{h_n + h_{n+1}}{h_n + 2h_{n+1}}$ the formula coincides with Schichman's method with variable time step [10]. Parametric BDF method is A-stable if

$$1 + \frac{h_n}{h_{n+1}} \geq \xi \geq 1 - \frac{h_n}{h_{n+1}}.$$

With $\xi = \frac{h_n + h_{n+1}}{h_n + 2h_{n+1}}$ we can obtain a stability constraint on the step increase on two sequential steps: $h_{n+1} \leq (1+\sqrt{2}) h_n$.

A multiple change in the value of the step is commonly used. The last expression shows that in order to keep solution stability the increase of time step size should be not more than two-fold.

In case of the time step is constant (i.e. $h_n = h_{n+1} = h$) and $\xi = 2/3$ the parametric BDF formula coincides with the second order Gear's method formula [9].

As a result of time quantization using Liniger-Willoughby method or parametric BDF method followed by linearization of all nonlinear elements' equations using Newton-Raphson method:

$$\left[\mathbf{J}\left(\overset{(s)}{\mathbf{f}}\right)\right]\left(\overset{(s+1)}{\mathbf{X}^{j+1}} - \overset{(s)}{\mathbf{X}^{j+1}}\right) = -\overset{(s)}{\mathbf{f}},$$

where Jacobian $[\mathbf{J}(\mathbf{f})] = \left|\frac{\partial \mathbf{f}}{\partial \mathbf{X}^{j+1}}\right|$, s is a Newton-Raphson method iteration number, we obtain a following system of linear equations:

$$[\mathbf{F}] \cdot |\mathbf{V}| = |\mathbf{W}|.$$

This system of linear equations is then solved using a sparse direct method. To avoid numeric oscillations occurring due to switching processes, the simulator switches to backward Euler method at the moments of commutations, i.e. the assignment $\xi = 1$ is made.

B. Principles of the use of the multiprocessor systems

From the very beginning the current version of MAES was designed to run on multiprocessor systems. Computational core of MAES can be executed on parallel computation systems with two architectures: shared memory multiprocessor (SMP) systems and distributed memory systems (clusters). However, the real-time variant of the simulator supports only shared memory architecture due to the high performance requirements or real-time tasks. Parallel architecture of the simulator allows either simultaneous execution of tasks of several users on one parallel system or execution of one big task (or a set of variants of one task, for example, for the purpose of statistical analysis) of a single user using several processors.

The estimate of the speedup of parallel computations compared to sequential computations can be made [11].

During the run of sequential variant of program the algorithm shown in Fig. 3 is executed:

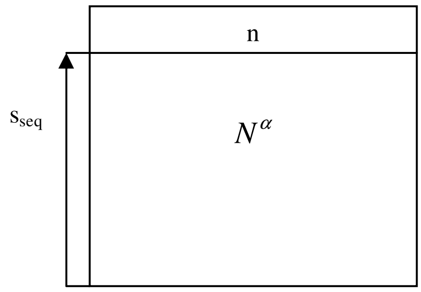


Fig. 3 Sequential algorithm execution.

where n is a number of operations that must be executed sequentially, N is a size of a matrix of linear equation system that is to be solved in order to find a solution. It is assumed that the number of operations required to solve linear system equals to N^{α} ($0 \leq \alpha \leq 3$). S_{seq} is a number of iterations required to perform on a given time step. Thus, sequential algorithm execution requires $n + s_{seq} \cdot N^{\alpha}$ operations per time step.

It is known than the matrices of linear equation systems modeling power grids have block diagonal structure. Such a structure is caused by delay of signal transmission though long distance transmission lines [12], [13]. Subsets of equations corresponding to blocks of matrix can be solved independently of each other and in parallel. Accordingly, the algorithm used in parallel variant of the program is shown in Fig. 4.

During the run of sequential variant of the program on k processors at most $n + \max_{i=1,\dots,k}(s_i \cdot N_i^{\alpha})$, operations are required on any of processors. Assuming that

$s_{seq} = \max_{i=1,\dots,k} s_i = s_m$, the computation speedup R equals

$$R = \frac{n + s_{seq} \cdot N^{\alpha}}{n + \max_{i=1,\dots,k}(N_i^{\alpha} \cdot s_i)} = \frac{n + \max_{i=1,\dots,k} s_i \cdot N^{\alpha}}{n + \max_{i=1,\dots,k}(N_i^{\alpha} \cdot s_i)} =$$

$$= \frac{\beta + s_m}{\beta + \max_{i=1,\dots,k}\left(\left(\frac{N_i}{N}\right)^{\alpha} \cdot s_i\right)},$$

where $\beta = \frac{n}{N^{\alpha}}$ is a ratio of sequential to parallel operations.

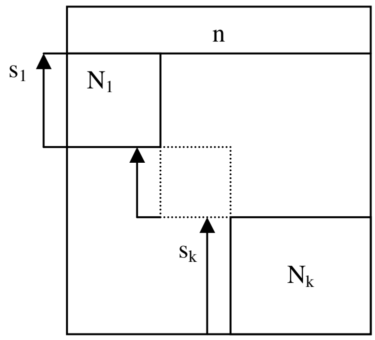


Fig.4. Parallel algorithm execution.

If the initial NxN is split uniformly into k sub-matrices and operations with each sub-matrix are carried out by a separate processor then $\frac{N}{N_i} = k$ and

$$R = \frac{\beta + s_m}{\beta + \frac{1}{k^{\alpha}} \max_{i=1,\dots,k} s_i} = \frac{\beta + s_m}{\beta + \frac{s_m}{k^{\alpha}}}.$$

Maximum speedup is obtained if $\beta \to 0$. In that case $R_{\max} = k^{\alpha}$ and $R_{\max} > k$ if α >1. Thus, the speedup can be greater than k times if using k processors because of algorithm change.

There is a number of tools for creation of parallel programs. Among them the main tools are:

- using of MPI message passing library;
- using of Unix programming facilities (processes and interprocess communication mechanisms);
- using of OpenMP.

Parallel programs can be created for several kinds of multiprocessor systems, the most widespread are cluster systems and SMP computers. Cluster-type multiprocessor systems are quite universal and cost-effective. However the problem of selection of communication equipment arises: the cost of communication network components rises steeply as the data transfer speed increases. Effectiveness of parallel program substantially depends on the ratio of time spent on computations to time spent on interprocess communication. The less is a fraction of time spent on communications the greater is program's effectiveness. From this point of view the use of SMP computers appears more preferable. A significant increase in the cost with an increase in the number of processors is a considerable limitation on the use of SMP systems. However, the development of multi-core architectures and their relatively low price weaken this limitation.

MPI communication protocol was used to create parallel version of MAES for cluster systems. UNIX programming facilities were used to create MAES version for SMP computers.

Parallel program created with the use of MPI is a set of processes executing in parallel and interacting with each other using special communication procedures to exchange data. These communication procedures form MPI library. Thus, in the most generic case MPI program implements the MPMD (Multiple Program Multiple Data) programming model, although in most cases SPMD (Single Program Multiple Data) model is used.

Power system's scheme is split into sub-schemes prior to simulation. Each sub-scheme or a set of sub-schemes is simulated by its own process. Two processes are called neighbor processes (adjacent processes) if their respective sub-schemes are connected with long transmission lines.

At the end of every time step data exchange between every pair of adjacent processes occurs. If the process has more than one process adjacent to it, then the interaction with them is carried out by turns. A process once started interaction with any of its neighbors can not stop data exchange (even temporarily) to communicate with another of its neighbor processes. This circumstance can lead to a situation called a deadlock.

Deadlocks occur in a parallel program as a result of a circular wait. To avoid the circular wait situation all processes are assigned unique numbers and each process sorts the list of it's neighbors according to their numbers in the ascending order. Then each process communicates with its adjacent processes by turns in order determined by its neighbors list, starting with the process with the least number.

In parallel computations the uniform distribution of a computational load among processes is quite important. In case of non-uniform distribution of the load much of the time during simulation is wasted due to idle wait by all processes for the one most busy process.

Parallelization of computations on SMP systems is carried out using Unix processes. This approach is more flexible compared to previous one. In this case there are means to efficiently create and destroy processes and a number of facilities for interprocess communication (shared memory, named and unnamed pipes, message queues, signals). Also, there is no need to use MPI loader and library, thus communication overheads are lower.

IV. EXAMPLES OF MAES USAGE

In this section an example of MAES usage for analysis of transients during a power system accident is given. The good fit of simulation results and oscillograms recorded during the accident (see Fig. 5) allowed to develop a set of recommendations to prevent such accidents in future. The cause of accident was the failure of the circuit breaker on the phase A during transmission line automatic reclosing. The following event sequence happened:

t_1 – one-phase short circuit on phase B;

t_2 – transmission line tripping;

t_3 – transmission line autoreclosing. The short-circuit still was not cleared at the moment of autoreclosing;

t_4 – tripping of the phase B;

t_5 – tripping of phase C;

t_6 – the breakdown of circuit-breaker on phase C;

t_7 – tripping of phase C;

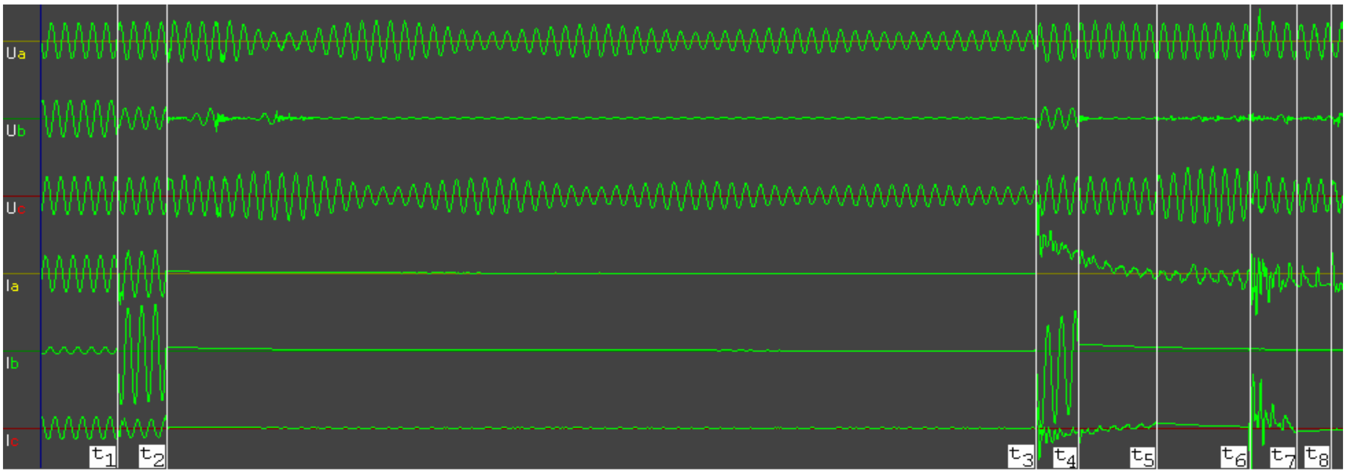
t_8 – unidentified commutation event.

V. CONCLUSIONS

One of the primary features of the MAES software suite is a SaaS concept implementation. Applying this concept in the area of power engineering was an innovative decision had been made during creation of MAES. Taking into consideration the success of similar systems in other sectors of industry, there is a hope that this approach can be introduced in the field of power engineering. At the present time a computation center for experimental operation of MAES is functioning in in-house testing mode in SibEPRI. After the end of in-house testing phase, public access to the computational center will be available.

Perspective lines of development of MAES are based on new technologies. Thus, there is an interesting new application of semantic networks for creation of user societies. Automated search for users with close interests based on the properties of schemes used allows them to share experience of solving similar problems.

(A) Oscillogram recorded by registrator



(B) Simulation

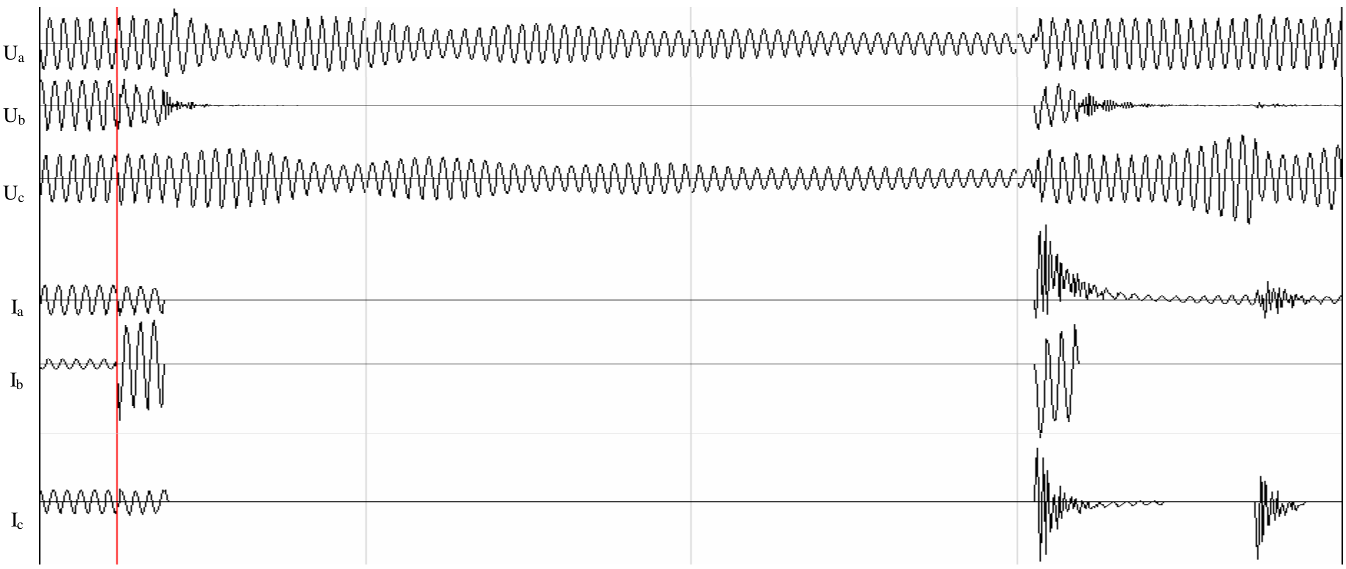


Fig. 5. Oscillograms and simulation results of the failed autoreclosing of the 500kV transmission line.

Another line of development of MAES is use of dynamic load balancing, that is dynamical redistribution of resources among computational nodes of multiprocessor computation system to increase its effectiveness.

Application of modern IT-technologies allows to create next generation software products with qualities and capabilities unattainable before.

VI. References

[1] Hermann W. Dommel, *EMTP Theory Book*. Second Edition. Microtran Power System Analysis Corporation. Vancouver, British Columbia. May, 1992. Latest update: April 1996.

[2] *MicroTran. Reference Manual. Transients Analysis Program for Power and Power Electronic Circuits*. Microtran Power System Analysis Corporation. Last revision January 2002. Canada. Available: http://www.microtran.com/goodies1/mt.pdf

[3] *Alternative Transient Program (ATP). Rule Book*, Portland, OR, 2001. Available: http://www.eeug.org/files/secret/ATP_RuleBook

[4] *Applications of PSCAD/EMTDC*. Manitoba HVDC Research Centre Inc. Winnipeg, Manitoba, Canada. Available: http://pscad.com/resource/File/Library/Application_Guide_2008.pdf

[5] *Real Time Digital Simulation for the Power Industry*. RTDS Technologies Inc. Winnipeg, Manitoba, Canada. Available: http://www.rtds.com/RTDS_Corporate_Profile.pdf

[6] *RT-LAB. Distributed Real-Time Power*. OPAL-RT Technologies Inc. Montréal, Canada. Available:http://www.opal-rt.com/opalbin/pds/rtlab_brochure_en.pdf

[7] *HYPERSIM. Real-time Digital Power System Simulation*. Silicon Graphics Inc. (SGI) and Hydro-Quebec TransEnergie Technologies. Available:http://www.sgi.com/pdfs/2320.pdf

[8] D. Fishteyn, "Deploying Software as a Service (SaaS)," WebApps, Inc. [Online]. Available: http:/www.saas.com

[9] Leon O. Chua and Pen-Min Lin. *Computer aided analysis of electronic circuits, algorithms and computational techniques*. Prentice-Hall, Englewood Cliffs, N.J., 1975.

[10] H.Schichman, "Integration System of a Nonlinear Network-Analysis Program," *IEEE Trans. on Circuit Theory*, pp.378-386, Aug.1970.

[11] V.Malyshkin, I.Naumkin, N.Malyshkin, V.Korneev, M.Ostapkevich, "Digital Electromagnetic Model of the Power System: Parallel Implementation," in Proc. PARELEC 2004 International Conference on Parallell Computing in Electrical Engineering, pp. 380-385.

[12] I.E. Naumkin, N.V. Malyshkin, M.B. Ostapkevich, V.D. Korneev, "Program of Analysis of Transient Electromagnetic Processes on Complex Electrical Power Schemes for Multiprocessor Calculating Systems," in Proc. 2003 International Scientific and Technical Conference on AC Power Delivery at Long and Very Long Distances (Novosibirsk, Russia), v. 2, pp. 16-22 (in Russian).

[13] I.E. Naumkin, "Digital Electromagnetic Model of Electrical System for Investigation of EHV and UHV Transmission Systems," in Proc. 2003 International Scientific and Technical Conference on AC Power Delivery at Long and Very Long Distances (Novosibirsk, Russia), v.2, pp. 23-35 (in Russian).