

Simulation and Evaluation of Optimization Problem Solutions in Distributed Energy Management Systems

Javier Contreras, *Member, IEEE*, Arturo Losi, *Member, IEEE*, Mario Russo, *Member, IEEE*, and Felix F. Wu, *Fellow, IEEE*

Abstract—Deregulation in electricity markets requires fast and robust optimization tools for a secure and efficient operation of the electric power system. In addition, there is the need of integrating and coordinating operational decisions taken by different utilities acting in the same market. Distributed Energy Management Systems (DEMS) may help to fulfill these requirements. The design of a DEMS requires detailed simulation results for the evaluation of its performance. To simulate the operation of a DEMS from the optimization standpoint, a general purpose distributed optimization software tool, DistOpt, is used, and its capabilities are extended to handle power system problems. The application to the optimal power flow problem is presented.

Index Terms—Deregulation, distributed energy management systems (DEMS), distributed optimization.

I. INTRODUCTION

THE deregulation and opening of competitive forces in the market are causing a deep restructuring of the electric power industry, affecting its organization and the operation of power systems.

From the point of view of the organization, many structures are possible, and the choice strongly depends on historical, political, and economical issues [1]. Nevertheless, some common elements can be found in all the restructuring models being proposed, namely, the Independent System Operator (ISO) granting for a nondiscriminatory access to the transmission network and system reliability, power brokers, generation companies, and distribution companies, just to name some of them.

The operation of power systems is strongly affected by restructuring [2], [3]. In a vertically integrated utility all technical and economical issues of the power system operation are considered together, and all the operational decisions are centrally assumed. It is not so in a deregulated environment, where the operational decisions can be classified into two categories [3]. Ones taken by the marketplace parties (such

as the energy trading agreements between producers and consumers), and others taken by a central authority, such as the curtailment of the proposed trades or the rescheduling of generation [1], which are typical control actions of an ISO. The ISO's decisions are supported by software application tools of the Energy Management System (EMS).

Two considerations can be made about the requirements for such EMS tools in a deregulated environment.

Firstly, central to the EMS is the functionality that allows for the congestion management, the scheduling of the system resources and the state estimation [1], [3]–[5]; optimization tools are required to accomplish these functions. It can be foreseen that deregulation will push the transmission system to operate closer than before to security limits [6]. In addition, the marketplace parties will be more sensitive to security concerns that affect the trades they can set [7], [8]. Thus, in a deregulated environment the need for fast and robust optimization software tools in the EMS is even more demanding than before.

The second issue concerns the transmission systems of several interconnected utilities [1], each one with its network jurisdiction and its own criterion and EMS. It is also possible that the participants in a poolco retain some of their EMS functions, while the ISO's EMS is intended to supervise the overall operation [9]. In such cases, there is a need to integrate the different EMSs by coordinating the solutions to achieve the overall optimal solution.

Fast and robust optimization software tools, and the coordination of different EMSs can be obtained by adopting the concept of a distributed computing environment. Such an environment is a viable way to obtain fast software tools based on the splitting of an optimization problem into subproblems that are easier to be solved and solvable in parallel. In distributed computing systems the processors can be of various types, can be located far from each other, and can be connected through high-speed communication links that can have an irregular topology. Thus, the concept of distributed computing helps to integrate the existing EMSs of different utilities and the EMS of the ISO. A distributed computing environment is robust; in case of some difficulty to solve an optimization subproblem in a subsystem or a communication failure, it is still possible to solve the other subproblems, and find a feasible solution for them, albeit not the overall optimal one. Therefore, Distributed EMSs (DEMS) are the right candidates for supporting the system operation and control in a deregulated electrical industry.

The design of a DEMS has to face an increasing number of options, in terms of architectures, algorithms, hardware, software, and methods for solving the various optimization

Manuscript received May 26, 2000; revised May 21, 2001. J. Contreras acknowledges the support of Ministerio de Educación y Ciencia, Spain, under a Predoctoral Fulbright USA/MEC Grant and a Doctors and Technologists Research Grant. A. Losi and M. Russo acknowledge the financial support of the CNR and MURST of Italy. This work was supported by the Hong Kong Research Grants Council under Project HKU 564/96E.

J. Contreras is with the E.T.S. de Ingenieros Industriales, Universidad de Castilla—La Mancha, Ciudad Real, Spain (e-mail: javier@ind-cr.uclm.es).

A. Losi and M. Russo are with the Dipartimento di Ingegneria Industriale, Università degli Studi di Cassino, Cassino, Italy (e-mail: losi@unicas.it; russo@unicas.it).

F. F. Wu is with the Department of Electrical and Electronic Engineering, University of Hong Kong, Hong Kong (e-mail: ffwu@eee.hku.hk).

Publisher Item Identifier S 0885-8950(02)00716-2.

problems. Performance evaluation is a key issue in the design process, and only the simulation of the operation of a DEMS can give detailed performance results [10].

The authors of this paper have previously developed a software framework, called DistOpt, to help to simulate the solution of general kind optimization problems in multicomputer systems. The capability of the earlier version of DistOpt was described in [11], which also includes a discussion of the theoretical aspect of the convergence issue of distributed optimization algorithms used in DistOpt. In this paper, the capability of DistOpt is extended specifically to handle power system problems. This is because the power system must be appropriately modeled to account for the peculiarities that arise when a power system is split into subsystems. A software module has been implemented and included into DistOpt that takes care of this modeling. The interface to available power system software packages is made easy with standard format input files. The aim of this paper is to present a friendly software environment useful to model and evaluate the performance of the solution of power system optimization problems in distributed computing environments.

In the paper, after a brief recall of the structure of DistOpt, the way to split a system/problem into subsystems/subproblems and the features of the developed software module are presented. Then, the performance of a DEMS when solving a particular optimization problem, the optimal power flow, is studied for the IEEE-14 bus and the IEEE-57 bus networks. The effects of splitting the optimization problem into different subproblems, of using different algorithms for solving the various subproblems, and of the synchronous versus asynchronous execution of the computation tasks are investigated.

II. DISTOPT SOFTWARE TOOL

DistOpt is based on the Ptolemy software environment, an extensible object-oriented general framework programmed in C++, in which system specification, simulation and design are possible [12]. DistOpt libraries of objects have been created to model the process of solving optimization problems in distributed environments, where the number of subproblems, the way they interact, and the solution algorithms can be selected [11]. The objects are based on the event-driven model of computation: the basic units of computation, the blocks, receive particles from the outside, process them, and generate output events after some latency.

DistOpt is made of three modules, corresponding to three levels of abstraction: the definition module, the mathematical model building module, and the optimization subproblem solution module. In the first module, the problem is described, and the splitting of the whole problem into subproblems is defined; the user provides the information describing the optimization problem and the splitting into subproblems in an input file. The second module automatically carries out the mathematical formulation of the problem and of the subproblems. This module is implemented as a library of objects for different numbers of subproblems in which the problem is split. The decomposition-coordination technique is based on the duplication of variables and on the Auxiliary Problem Principle (APP) [13],

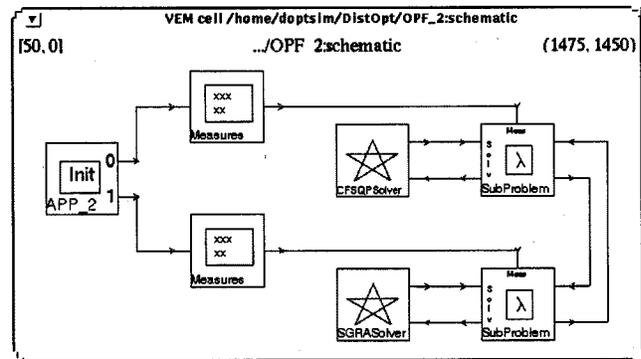


Fig. 1. Universe for the splitting into two subproblems.

[14], a very general theoretical decomposition-coordination framework; DistOpt allows the user to choose among various two-level decomposition-coordination methods. Within the third module, the optimization subproblems are solved in the framework established in the previous module. This module is made of a library of objects, each corresponding to a particular solution method, implementing either the code of the method or the interface to an external code.

A simulation case of DistOpt is run through a universe (the runnable entity in Ptolemy) which is built starting with the specification of the number of subproblems, by selecting the appropriate block from the mathematical model building palette. Then, the choice of the algorithm for solving each subproblem is made by grabbing the icon representing the chosen algorithm and dragging it to the appropriate ports of the block for that subproblem. According to the needs of information exchange of the subproblems, input/output (I/O) ports of the icons representing subproblems have to be graphically connected. An example of a universe for the case of a problem split into two subproblems is shown in Fig. 1.

III. DISTOPT AND POWER SYSTEM OPTIMIZATION PROBLEMS

In power systems, the decomposition of an optimization problem into subproblems, separately solved and appropriately coordinated, can be triggered off for many reasons. One reason is to split the constraints that represent different aspects of the operation of the system into subsets. The subproblems obtained in this way refer to the same physical system, and each subproblem cares for different constraints. It is the case, for example, of the unit commitment/optimal power flow problem [15]. Another reason is to divide the whole system into physical subsystems which refer to different control areas in the overall system. Each subsystem is described by its own constraints and its own part of the objective function (if any): optimization subproblems thus correspond to physical subsystems. It is the case, for example, of the state estimation problem split into state estimation subproblems [5], [16], or the optimal power flow problem divided into optimal power flow subproblems [17].

DistOpt directly treats the first kind of decomposition. Once the user has provided the right information in the input file (see Section II), then the analytical formulation of the subproblems is

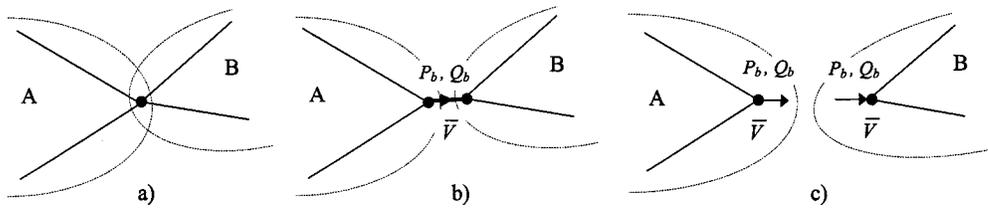


Fig. 2. Boundary node belonging to two subsystems.

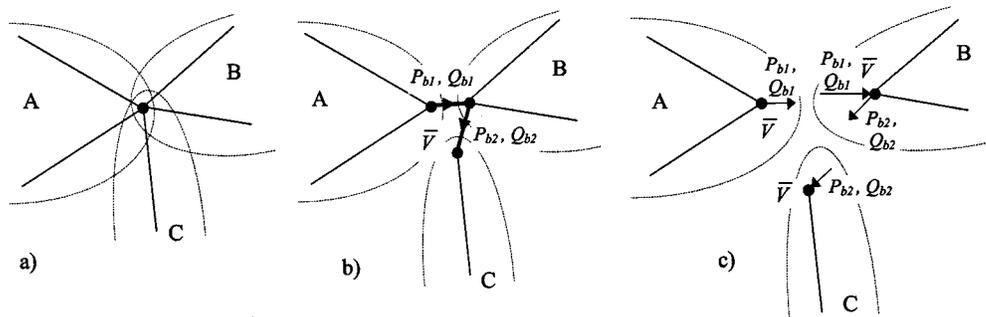


Fig. 3. Boundary node belonging to more than two subsystems.

carried out within the software environment without any further user's intervention. The second kind of decomposition, namely splitting a power system into subsystems, requires an appropriate modeling of the physical boundary constraints that each subsystem has to fulfill.

A. Power System Modeling and Splitting

When splitting a power system into subsystems, a special kind of node has to be considered, the boundary node, a node that is shared by contiguous subsystems. A boundary node can be a physical node or a dummy node, added to the border of two neighboring subsystems in order to divide the transmission line across them into two lines [17]. A boundary node can be shared by two or more subsystems. Following is the examination, first of the case of a boundary node shared by only two subsystems, and then the general case.

A boundary node shared by two subsystems [Fig. 2(a)] can be duplicated into two nodes connected by a zero impedance branch modeled with two variables: the active and reactive powers exchanged along it, P_b and Q_b [Fig. 2(b)]. The voltage of the two nodes representing the same physical node \bar{V} is obviously the same. Two separated subsystems are obtained by cutting the zero-impedance branch and by duplicating the voltage of the duplicated node and the active and reactive powers exchanged along the cut branch [Fig. 2(c)].

In the general case of a boundary node shared by more than two subsystems [Fig. 3(a)], the boundary node is represented by as many nodes as the number of subsystems sharing it, connected by zero impedance branches. The number of these branches is smaller than the number of subsystems by one [Fig. 3(b)]. Once again, separated subsystems are obtained by cutting the zero-impedance branches and by duplicating the appropriate variables [Fig. 3(c)].

The duplicated variables, which belong to different subsystems, are forced to assume the same value by means of coupling constraints, whose general form is

$$u_i^r - u_j^s = 0. \quad (1)$$

Equation (1) indicates that the r th component of the variables pertaining to the i th subsystem and the s th component of the variables pertaining to the j th subsystem are coupled because of the duplication. In the following, constraints like (1) are called consistency constraints, whereas all the other constraints of the optimization problem are called feasibility constraints.

While the duplication of voltage magnitudes and active and reactive powers along the zero-impedance branches is easily obtained, the duplication of voltage phase angles deserves more attention than it has usually received. Once the boundary nodes have been duplicated, the zero-impedance branches have been cut, and the appropriate variables have been duplicated, all the subsystems are obtained, separated from each other and coupled only by the consistency constraints (1). If the feasibility constraints are treated separately from the consistency constraints, it is necessary for the feasibility constraints to define the subproblems on their own. In particular, when the splitting refers to subsystems, it means that for each subsystem a phase reference must be assumed. As many slack buses are generated as the number of subproblems. Their voltage phase angles are a reference for the buses of the subsystems they belong to, and, at the same time, they have to assume the right value against the global reference.

To handle these situations, the concept of Δ -variables is introduced; they are quantities defined only if one of them is assumed as a reference. When splitting the problem requires the duplication of some Δ -variable, in each subproblem one Δ -variable is taken as the local reference. For the i th subsystem, the value of

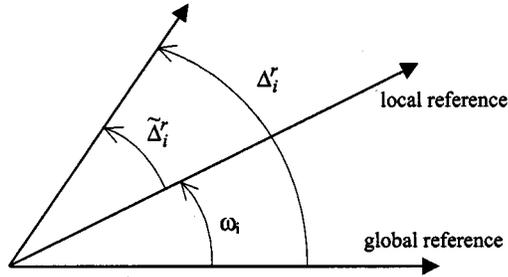


Fig. 4. Global and local phase references.

its Δ -variables against the global reference Δ_i is different from the value of the same Δ -variables against the local reference, $\tilde{\Delta}_i$. It is

$$\tilde{\Delta}_i = \Delta_i - \nu\omega_i \quad (2)$$

where ν is a unitary vector of appropriate dimension, and ω_i is the value against the global reference of the Δ -variable assumed as reference in the i th subsystem. Each subproblem is formulated in terms of local Δ -variables $\tilde{\Delta}_i$ and the consistency constraints (1) for the duplicated Δ -variables assume the form

$$\left(\tilde{\Delta}_i^r + \omega_i\right) - \left(\tilde{\Delta}_j^s + \omega_j\right) = 0 \quad (3)$$

so that the variables ω do appear in the consistency constraints for the duplicated Δ -variables (3), and do not in the feasibility constraints.

Fig. 4 depicts the modeling of local and global Δ -variables, where they are voltage phase angles. This modeling of Δ -variables has many relevant features: there is no limitation to the number of times a node can be duplicated (it can belong to more than two subsystems), and two subsystems can share more than one boundary node. In addition, the modeling (3) of the consistency of duplicated Δ -variables is completely and rigorously framed into the APP. It is even possible to specialize the treatment of the variables ω within APP; information about it can be found in [18].

B. Power System Software Module in DistOpt

With a specifically designed software package to be included into DistOpt the modeling of boundary nodes has been automated. The user has to provide an input file containing the power system data in a standard format and information about the geographical partition into subsystems; the package cares for modeling the appropriate subproblems to be solved within DistOpt.

Also the treatment of all the issues regarding the duplication of Δ -variables has been automated. In particular, the formulation of the feasibility constraints of each subproblem is made in terms of local variables: the voltage phase angles in each subsystem are referred to the local slack node, and the constraints for each subproblem are formulated as in usual power system modeling. Variables ω are kept hidden from the subproblems; they are automatically treated within the software package, which cares for their use to assure that the consistency constraints as (3) are satisfied.

Thanks to this package, the user is not involved either in building the analytical model of the subproblems, or in the intri-

TABLE I
TEST CASE DATA FOR IEEE-57 BUS NETWORK

Test case	No. of subsystems	No. of buses per area	No. of boundary buses	Computing times [p.u.]
A	2	9 / 53	5	0.03 / 1
B	3	9 / 42 / 14	5 / 3	0.03 / 0.41 / 0.06

cacies of the Δ -variables. The input file can be generated by existing power system packages, or by a user-friendly graphical interface. In addition, thanks to the characteristics of DistOpt (see Section II), it is easy to interface existing optimization codes to get the solution of each subproblem obtained after the splitting.

IV. DEMS OPTIMAL POWER FLOW

The Optimal Power Flow (OPF) problem is considered as an example of optimization problems within a DEMS, for which a full-fledged model is adopted (no model simplifications) and solved without any active-reactive (P-Q) decomposition.

The package described in Section III is used to easily study different power systems and different splittings into subsystems.

It is worth reminding that DistOpt allows to easily simulate the computing capacity of the processors/algorithms that solve the various subproblems, as well as the communication system that cares for the information exchange among the processors (indeed, a small amount in the case of distributed optimizations where the power system is split into subsystems). In fact, the simulation is event-driven and the information exchange is based on message passing. It is easy to use Ptolemy-based libraries for the simulation of communication systems, such as [19].

Thanks to the flexibility of DistOpt, many iterative schemes can be chosen and easily compared. The computation tasks which any iterative scheme is made of can be performed synchronously or asynchronously: in the first case, a new step of the iterative scheme is started only when the previous step is completed by all the subproblems; in the second case, each subproblem starts a new step as soon as it has completed the previous step, irrespective of the completion of the other subproblems. DistOpt allows to easily choose the synchronization strategy for each subproblem.

A. Case Studies

Two DEMS-OPF cases, the IEEE-14 bus and the IEEE-57 bus networks are studied; the objective function is the fuel cost, whose coefficients are taken from [20].

First order and quasi-Newton algorithms, namely SGRA [21] and CFSQP [22], are used and compared for solving the various optimization subproblems. Adequate interfaces are provided in the Solver palette. Both algorithms behave well in finding the solution. But, because of their nature, the former is much faster than the latter, as far as computation time is considered. Due to this fact, after testing with the IEEE-14 bus network, it was decided to use only SGRA for the IEEE-57 bus network. For the sake of space, only results of the bigger network are presented.

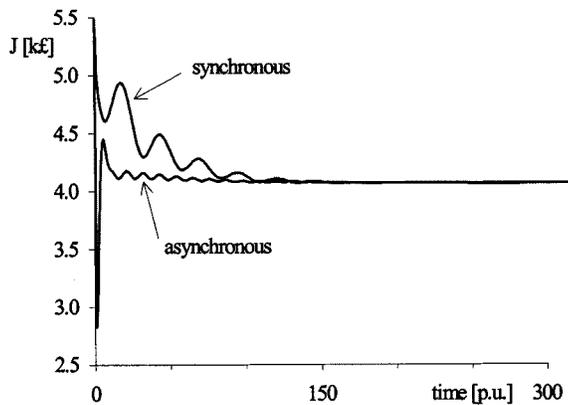


Fig. 5. Objective function for two subproblems.

The splitting of the system into two and three subsystems is considered, following natural boundaries that can be found by looking at the one-line diagram of the network; the splitting into subsystems is easily obtained by providing splitting information in the input file. In Table I the decomposition data for each test case are reported. As regards computing times, only the ones necessary to solve the subproblems are considered. An estimation of their values is obtained with a previous evaluation of the average time necessary to solve the subproblems. The computing times used in the case studies, and assigned as the latency of the blocks that care for the computations (see Section II), are reported in Table I: they are referred to the maximum one, which is the average computing time necessary to solve the OPF for the subsystem composed of 53 buses in test case A. So, for example, a computing time of 0.03 pu indicates that the mean time necessary to solve the OPF for the corresponding subsystem is only 3% of the previously indicated maximum computing time.

Concerning the iterative scheme, for the sake of conciseness only the results obtained with the one corresponding to the Arrow-Hurwicz algorithm [14] are shown. Both the synchronous and the asynchronous execution of the tasks which the iterative process is made of are considered.

For evaluating the performance of the DEMS in terms of convergence achievement and rate, the time evolution of the following quantities is considered

- overall objective function, J ;
- maximum absolute value of the error in the consistency constraints (1), E .

Figs. 5 and 6 show, respectively, the value of J and of E versus the computation time for test case A and both the synchronous and the asynchronous executions of the subproblems. For test case B, Figs. 7 and 8 report the same quantities as Figs. 5 and 6, respectively, (note the different time scales).

B. Discussion of Results

From Figs. 5–8, it is apparent that

- the iterative process with the APP always converges for any splitting of the system into subsystems;
- the convergence time in the two subsystems case is higher than the one in the three subsystems case. In general, convergence conditions are more restrictive in the three subsystems case than in the two subsystems case, and

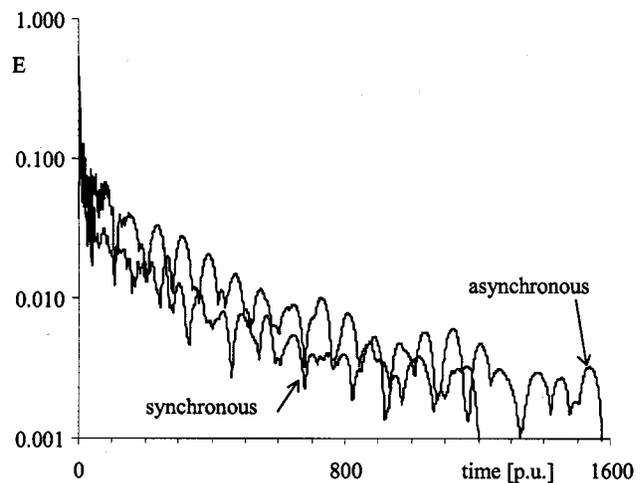


Fig. 6. Maximum discrepancy for two subproblems.

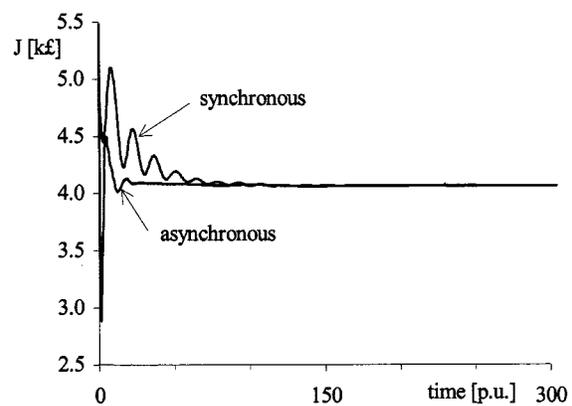


Fig. 7. Objective function for three subproblems.

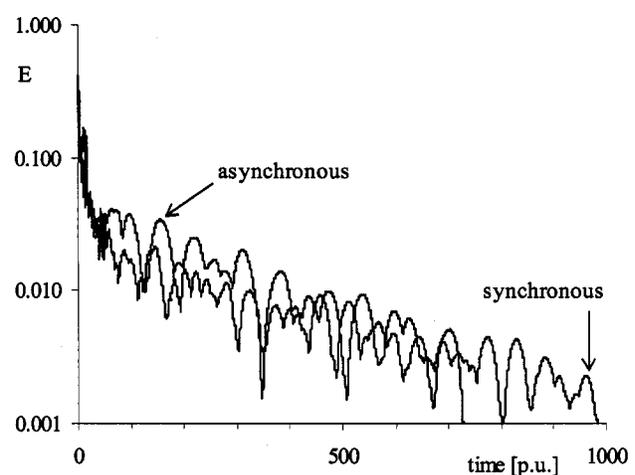


Fig. 8. Maximum discrepancy for three subproblems.

- more iterations are required for convergence. However, the biggest subproblem is smaller and takes less time to be solved. In this case, splitting into three subproblems requires less overall computing time;
- whenever using asynchronous execution, the objective function approaches its final value faster than with synchronous execution.

The first and second considerations seem to be quite general to the authors. The numerical experiments for the synchronous execution have shown that the convergence conditions of the APP [13], [14], [18] do hold even if the OPF problem is not convex and, although sufficient, they are not over-restrictive. On the other hand, the updating of the Lagrange multipliers associated with the consistency constraints has to be kept significantly slow to guarantee convergence in the asynchronous execution.

V. CONCLUSION

A software environment useful to model and evaluate the performance of the solution of power system optimization problems in distributed computing environments, e.g., OPF in DEMS, has been presented. The software is built on a general purpose distributed optimization simulation tool, called DistOpt, that was developed by the authors previously.

A serious modeling difficulty arises when one attempts to apply distributed computation to power system problems, that is the issue of reference angles. When the system is split into subsystems by the standard technique of duplicating variables, the boundary nodes are duplicated. The voltage phase angles of the boundary nodes impose a problem; the original voltage angle is really a relative value with respect to a reference. After a two-way splitting, for example, there are two references, one for each subsystem. A specific technique (called Δ -variable) is introduced in this paper to handle this specific problem.

Using OPF as an example, it is demonstrated that the extended DistOpt for power system applications can be conveniently and effectively used to experiment and study different ways of decomposition in a distributed optimization problem. In details, the software can be used to study different ways of splitting a system into subsystems, different algorithms for solving the various subproblems, the simulation of the computing capability and the selection of synchronous versus asynchronous schemes, etc.

REFERENCES

- [1] I. J. Pérez-Arriaga, H. Rudnick, and W. O. Stadlin, "International power system transmission open access experience," *IEEE Trans. Power Syst.*, vol. 10, pp. 554–564, Feb. 1995.
- [2] D. Shirmohammadi, B. Wollenberg, A. Vojdani, P. Sandrin, M. Pereira, F. Rahimi, T. Schneider, and B. Stott, "Transmission dispatch and congestion management in the emerging market structures," *IEEE Trans. Power Syst.*, vol. 13, pp. 1466–1474, Nov. 1998.
- [3] A. Moser, G. C. Ejebe, and J. G. Frame, "Network and power applications for EMS within a competitive environment," in *Proc. Power Eng. Society Transmission and Distribution Conf.*, vol. 1, pp. 280–285.
- [4] A. J. Wood and B. F. Wollenberg, *Power Generation Operation and Control*, 2nd ed. New York: Wiley, 1996.
- [5] R. Ebrahimi and R. Baldick, "State estimation distributed processing," *IEEE Trans. Power Syst.*, vol. 15, pp. 1240–1246, Nov. 2000.
- [6] O. Hirsch and S. Lee, "Security applications and architecture for an open market," *IEEE Comput. Applicat. Power*, pp. 26–31, July 1999.
- [7] F. F. Wu and P. Varaiya, "Coordinated multilateral trades for electric power networks: Theory and implementation," *Int. J. Elect. Power Energy Syst.*, vol. 21, no. 2, pp. 75–102, Feb. 1999.
- [8] E. Allen, M. Ilic, and Z. Younes, "Providing for transmission in times of scarcity: An ISO cannot do it all," *Int. J. Elect. Power Energy Syst.*, vol. 21, no. 2, pp. 147–163, Feb. 1999.
- [9] F. Albuyeh and Z. Alaywan, "Implementation of the California independent system operator," in *Proc. 21st Int. Conf. Power Industry Comput. Applicat.*, 1999, pp. 233–238.
- [10] K. Kato and H. R. Fudeh, "Performance simulation of distributed energy management systems," *IEEE Trans. Power Syst.*, vol. 3, pp. 820–827, May 1992.

- [11] J. Contreras, A. Losi, M. Russo, and F. F. Wu, "DistOpt: A software framework for modeling and evaluating optimization problem solution in distributed environments," *J. Parallel and Distributed Comput.*, vol. 60, pp. 741–763, Jun. 2000.
- [12] E.A. Lee, *Prolemy Reference Manual: V. 0.7*, EECS Dept., Univ. California, Berkeley, 1997.
- [13] G. Cohen, "Auxiliary problem principle and decomposition of optimization problems," *J. Optim. Theory Applicat.*, vol. 32, pp. 277–305, Nov. 1980.
- [14] G. Cohen and D. Li Zhu, "Decomposition coordination methods in large scale optimization problems: The nondifferentiable case and the use of augmented Lagrangians," *Adv. Large Scale Syst.*, vol. 1, pp. 203–266, 1984.
- [15] J. Batut and A. Renaud, "Daily generation scheduling optimization with transmission constraints: A new class of algorithms," *IEEE Trans. Power Syst.*, vol. 7, pp. 982–998, Aug. 1992.
- [16] D. M. Falcao, F. F. Wu, and L. Murphy, "Parallel and distributed state estimation," *IEEE Trans. Power Syst.*, vol. 10, pp. 724–730, May 1995.
- [17] B. H. Kim and R. Baldick, "Coarse-grained distributed optimal power flow," *IEEE Trans. Power Syst.*, vol. 12, pp. 932–939, May 1997.
- [18] A. Losi and M. Russo, "A note on the application of the auxiliary problem principle," *J. Optim. Theory Applicat.*, to be published.
- [19] S. M. Lun, F. F. Wu, N. Xiao, and P. Varaiya, "NetPlan: An integrated network planning environment," in *Proc. Int. Workshop on Modeling, Anal., and Simulation of Comput. and Telecomm. Syst.*, Jan. 1993.
- [20] O. Alsac and B. Stott, "Optimal load flow with steady-state security," *IEEE Trans. Power Apparatus Syst.*, vol. PAS-93, pp. 745–751, May/June 1974.
- [21] A. Losi, *Users' Guide for SGRA – Version 0.2*. Cassino, Italy: Dipartimento di Ingegneria Industriale, Università degli Studi di Cassino, 1999.
- [22] C. Lawrence, J. L. Zhou, and A. L. Tits, *Users' Guide for CFSQP Version 2.5d*. College Park, MD: Univ. of Maryland Press, 1998.

Javier Contreras (M'98) was born in Zaragoza, Spain, in 1965. He received the B.S. degree in electrical engineering from the University of Zaragoza, Zaragoza, Spain, the M.Sc. degree in electrical engineering from the University of Southern California, Los Angeles, and the Ph.D. degree in electrical engineering and computer sciences from the University of California, Berkeley, in 1989, 1992, and 1997, respectively.

From 1997 to 1998, he was a Researcher at the Instituto de Investigación Tecnológica, Universidad Pontificia Comillas (UPCo), Madrid, Spain. Currently, he is an Assistant Professor at the Universidad de Castilla—La Mancha, Ciudad Real, Spain. His research interests include power systems planning, operations and economics, and electricity markets.

Arturo Losi (M'97) received the Laurea and Ph.D. degrees in electrical engineering from Università degli Studi di Napoli, Italy, in 1980 and 1987, respectively.

Currently, he is Professor of electrical power systems, Università degli Studi di Cassino, Italy.

Dr. Losi is a member of AEI.

Mario Russo (M'92) received the Laurea degree in electrical engineering from Università degli Studi di Napoli, Italy, in 1988.

Currently, he is Associate Professor of electrical power systems, Università degli Studi di Cassino, Italy.

Prof. Russo is a member of AEI and IFAC.

Felix F. Wu (M'72–F'89) received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, R.O.C., the M.S. degree in electrical engineering from the University of Pittsburgh, Pittsburgh, PA, and the Ph.D. degree in electrical engineering and computer science from the University of California, Berkeley, in 1965, 1969, and 1972, respectively.

From 1976 to 1977, he worked for Pacific Gas and Electric as a Computer Applications Engineer. He held a TEPCO (Tokyo Electric Power Company) Professorship at the University of Tokyo, Japan, in 1991. He was a Professor of electrical engineering and computer sciences at the University of California, Berkeley, from 1976 to 1982. Currently, he is Professor of electrical engineering at the University of Hong Kong.