Parallel and Distributed State Estimation

D.M. Falcão*   Felix F. Wu   Liam Murphy

Department of Electrical Engineering and Computer Sciences
University of California at Berkeley
Berkeley, CA 94720

Abstract—The need in recent years for higher frequency in state estimation execution covering larger supervised networks has led to the investigation of faster and numerically more stable state estimation algorithms. However, technical developments in distributed Energy Management Systems, based on fast data communication networks, open up the possibility of parallel or distributed state estimation implementation. In this paper, this possibility is exploited to derive a solution methodology based on conventional state estimation algorithms and a coupling constraints optimization technique. Numerical experiments show suitable performance of the proposed method with regard to estimation accuracy, convergence robustness and computational efficiency. The results of these experiments also indicate the decoupled nature of the state estimation problem.

I. INTRODUCTION

State estimation is a basic module in the advanced application software (network analysis) of modern Energy Management Systems (EMS). Its main function is to provide reliable estimates of the quantities required for monitoring and control of the electric power system [1]-[3]. In almost all state estimation implementations, a set of measurements obtained by a SCADA system at approximately the same time instant throughout the whole supervised network is centrally processed by a static state estimator at regular intervals or by operator request. Modern high speed data acquisition equipment is able to obtain new sets of measurements every 1-10 seconds, but the present

---

*On sabbatical leave from the Federal University of Rio de Janeiro, Brazil.

EMS hardware and software only allow the processing of state estimations every few minutes. It has been argued (see the discussion by Slutsker et al in [4], [5]) that a more useful state estimation operational scheme would be achieved by shortening the time interval between consecutive state estimations to allow a closer monitoring of the system evolution, particularly in emergency situations in which the system state changes more rapidly. Another industry trend is to enlarge the supervised network by extending state estimation to low voltage subnetworks. These trends pose the challenge of performing state estimation in a few seconds for networks with thousands of nodes.

The requirement for higher frequency in state estimation execution has led to the development of faster state estimation algorithms. The desire for larger supervised networks increases the demands on the development of numerically stable algorithms [6]. Conventional centralized state estimation methods have reached a stage in which substantial improvements in either speed or numerical robustness are unlikely. On the other hand, technical developments in distributed EMS [7], based on fast data communication networks [8], open up the possibility of parallel or distributed algorithms [9, 10] for state estimation.

Early approaches to the multiprocessor state estimation problem concentrated on hierarchical methods [11]. These methods assume a *star-like* functional scheme and communication network, in which several remote processors perform local state estimation in network areas and the results are send to a central computer that refines the calculation. This hierarchical approach suffers from the bottleneck and reliability problems inherent in a system with one central controlling node. Recently, several new approaches to the problem have been proposed, most of which are based on parallel solution schemes for the linear system of equations associated with nonlinear state estimation algorithms [12]-[15] or linear programming decomposition techniques [16]. A recent project introduced a distributed algorithm based on dual recursive quadratic programming, but the reported results indicate limited computational performance [17].

In this paper, the possibility of parallel and distributed state estimation implementation is exploited to derive a solution methodology based on conventional state estima-

tion algorithms and a coupling constraints optimization technique. Conditions under which distributed state estimation provides accurate estimates of the state variables are determined. An extension of the algorithm which allows new measurements to be quickly incorporated into the state estimation process is investigated, leading to a distributed tracking algorithm. Numerical experiments are reported to assess the performance of the proposed method with regard to estimation accuracy, convergence robustness and computational efficiency, both in static and tracking mode operation. An evaluation of the degree of natural decoupling in the state estimation problem is also performed.

II. MATHEMATICAL MODELING AND FORMULATION

A. Measurement Models

The information model used in power system state estimation is represented by the equation

$$z = h(x) + \omega \tag{1}$$

where z is a $(m \times 1)$ measurement vector, x is a $(n \times 1)$ true state vector, $h(.)$ is a $(m \times 1)$ vector of nonlinear functions, ω is a $(m \times 1)$ measurement error vector, m is the number of measurements, and n is the number of state variables. The usual choice for state variables are the bus voltage phase angles and magnitudes, while the measurements are active and reactive power flows, node injections and voltage magnitudes.

As in loadflow calculations, it has been found that state estimation algorithms based on decoupled versions of (1) behave adequately for the usual power networks [1, 2]. Therefore, the following decoupled model is usually adopted:

$$z_p = h_p(\theta, v) + \omega_p \tag{2}$$
$$z_q = h_q(\theta, v) + \omega_q \tag{3}$$

where θ $(n_\theta \times 1)$ and v $(n_v \times 1)$ are the vectors of true voltage magnitudes and phase angles; p and q are subscripts indicating partitions of vectors and matrices corresponding to active and reactive measurements, respectively; $n_\theta = N - 1$, $n_v = N$, and N is the number of network nodes.

B. Problem Formulation and Optimality Conditions

Assume that the power network is decomposed into M areas. The areas are connected through boundary buses which belong simultaneously to both adjacent areas. Therefore, there are overlapping areas which may be observed using both adjacent measurement sets. The number of boundary buses may be kept to a minimum, or may incorporate a few extra buses in order to facilitate

some estimation functions such as bad data processing. A further assumption is that there are no injection measurements in the overlapping area buses. This assumption does not necessarily represent a practical limitation to the proposed methodology, as actual injection measurement buses in overlapping areas can be replaced by fictitious buses with no injection measurement connected to the actual buses (now placed outside the overlapping area) by zero impedance lines [18].

Under the above assumptions, the state estimation problem introduced in (2) and (3) can be decomposed as follows

$$z_p^k = h_p^k(\theta^k, v^k) + \omega_p^k, \quad k = 1, ..., M \tag{4}$$
$$z_q^k = h_q^k(\theta^k, v^k) + \omega_q^k, \quad k = 1, ..., M \tag{5}$$

where

z_p^k and z_q^k are vectors of active and reactive measurements in area k; dimensions: $(m_p^k \times 1)$ and $(m_q^k \times 1)$, respectively.

θ^k and v^k are vectors of voltage phase angles and magnitudes in area k, including the ones corresponding to the boundary buses; dimensions: $(n_\theta^k \times 1)$ and $(n_v^k \times 1)$, respectively.

The WLS state estimate for the distributed estimation problem defined in (4) and (5) can be obtained by solving a constrained optimization problem with a separable objective function and a set of linear constraints introduced to force state variables in overlapping areas to assume the same values :

$$\min_{\theta^k, v^k} \sum_{k=1}^{M} 1/2\{ [z_p^k - h_p^k(.)]^T [R_p^k]^{-1}[z_p^k - h_p^k(.)] + $$
$$+ [z_q^k - h_q^k(.)]^T [R_q^k]^{-1}[z_q^k - h_q^k(.)] \} \tag{6}$$

$$\text{s. t.} \sum_{k=1}^{M} A_p^k \theta^k = 0, \quad \sum_{k=1}^{M} A_p^k v^k = 0 \tag{7}$$

where A_p^k and A_q^k are $(l \times n_\theta^k)$ and $(l \times n_v^k)$ matrices, respectively (l is the number of boundary buses), whose nonzero elements are either 1 or -1.

The necessary conditions for the solution of the above problem, derived from the corresponding Lagrangian function L, are:

$$\frac{\partial L}{\partial \theta^k} = -[H_p^k]^T [R_p^k]^{-1}[z_p^k - h_p^k(\theta^k, v^k)] + $$
$$+ [A_p^k]^T \gamma_p = 0, \quad k = 1, ..., M \tag{8}$$

$$\frac{\partial L}{\partial v^k} = -[H_q^k]^T [R_q^k]^{-1}[z_q^k - h_q^k(\theta^k, v^k)] + $$
$$+ [A_q^k]^T \gamma_q = 0, \quad k = 1, ..., M \tag{9}$$

$$\frac{\partial L}{\partial \gamma_p} = \sum_{k=1}^{M} A_p^k \theta^k = 0, \quad \frac{\partial L}{\partial \gamma_q} = \sum_{k=1}^{M} A_q^k v^k = 0 \tag{10}$$

where γ_p and γ_q are $(l \times 1)$ vectors of Lagrange multipliers and

$$H_p^k = \frac{\partial h_p^k(\theta^k, v^k)}{\partial \theta^k} \quad , \quad H_q^k = \frac{\partial h_q^k(\theta^k, v^k)}{\partial v^k}$$

calculated at the solution point.

C. Basic Algorithm

As in the integrated state estimation approach [1, 2], the Gauss-Newton method combined with the usual *Fast Decoupled Loadflow* assumptions can be used to solve (8)-(10). This leads to the following algorithm :

$$\tilde{\theta}^k(i+1) = \theta^k(i) + [C_p^k]^{-1}\Delta b_p^k(i), \quad k = 1, ..., M \qquad (11)$$

$$\gamma_p(i+1) = N_p^{-1}\sum_{k=1}^{M} A_p^k \tilde{\theta}^k(i+1) \qquad (12)$$

$$\theta^k(i+1) = \tilde{\theta}^k(i+1) - [C_p^k]^{-1}[A_p^k]^T \gamma_p(i+1), \quad k = 1, ..., M \qquad (13)$$

and

$$\tilde{v}^k(i+1) = v^k(i) + [C_q^k]^{-1}\Delta b_q^k(i), \quad k = 1, ..., M \qquad (14)$$

$$\gamma_q(i+1) = N_q^{-1}\sum_{k=1}^{M} A_q^k \tilde{v}^k(i+1) \qquad (15)$$

$$v^k(i+1) = \tilde{v}^k(i+1) - [C_q^k]^{-1}[A_q^k]^T \gamma_q(i+1), \quad k = 1, ..., M \qquad (16)$$

where

$$\Delta b_p^k(i) = [H_p^k]^T[R_p^k]^{-1}[z_p^k - h_p^k(\theta_k(i), v_k(i))] \quad (17)$$

$$\Delta b_q^k(i) = [H_q^k]^T[R_q^k]^{-1}[z_q^k \\ - h_q^k(\theta_k(i+1), v_k(i))] \quad (18)$$

$$C_p^k = [H_p^k]^T[R_p^k]^{-1}H_p^k \qquad (19)$$

$$C_q^k = [H_q^k]^T[R_q^k]^{-1}H_q^k \qquad (20)$$

$$N_p = \sum_{k=1}^{M} A_p^k[C_p^k]^{-1}[A_p^k]^T \qquad (21)$$

$$N_q = \sum_{k=1}^{M} A_q^k[C_q^k]^{-1}[A_q^k]^T \qquad (22)$$

and H_p and H_q are calculated at nominal conditions and kept constant in the iterative process.

D. Distributed Algorithm

A straightforward implementation of the algorithm given in (11)-(22) leads to a *hierarchical estimator*. As explained earlier, this kind of estimator is less preferable for parallel or distributed processing. A version of the algorithm more suitable to this type of processing can be obtained by neglecting the off-diagonal elements in matrices C_p^k and C_q^k in equations (12,13) and (15,16), as in [19]. In this case, these equations can be merged and rewritten as

$$\theta^k(i+1) = \tilde{\theta}^k(i+1) + \Delta\tilde{\theta}^k(i+1) \qquad (23)$$

$$v^k(i+1) = \tilde{v}^k(i+1) + \Delta\tilde{v}^k(i+1) \qquad (24)$$

where all the elements of $\Delta\tilde{\theta}^k(i+1)$ are zero except the ones corresponding to boundary buses, which are given by

$$\Delta\tilde{\theta}_r^k(i+1) = \pm\frac{g_{rr}^k}{g_{rr}^k + g_{rr}^j}[\tilde{\theta}_r^k(i+1) - \tilde{\theta}_r^j(i+1)] \quad (25)$$

where g_{rr}^k and g_{rr}^j are diagonal elements corresponding to bus r of the inverse gain matrices of the neighboring areas k and j, respectively, and the sign is set to $+$ or $-$ according to A_p^k. The elements of $\Delta\tilde{v}^k(i+1)$ are calculated similarly.

E. Distributed Asynchronous Algorithm

The algorithm introduced in the previous section calculates the θ and v updates at every iteration in a synchronous way : it has to wait until the state vector is updated in all areas before it starts a new iteration. This approach presents drawbacks for both parallel and distributed processing. In parallel processing, synchronous algorithms usually fail to achieve high efficiency due to processor idle times, unless a perfect load balancing is obtained. In distributed processing, this approach requires difficult synchronizing operations in geographically distributed systems. Asynchronous algorithms, on the other hand, are more suitable for both parallel and distributed processing implementations.

In the case of the distributed algorithm described above, simulated experiments have shown that the computation can continue even in the absence of information from other areas. This fact can be understood by realizing that equations (11) and (14) are in fact a local decoupled state estimator. This property makes the algorithm suitable to asynchronous implementation. Obviously, in this case matching of boundary bus state variables would not be achieved.

III. COMPUTATIONAL EXPERIMENTS

The distributed state estimation methodology introduced in this paper was tested using a simulated environment running in a conventional uniprocessor computer. The objective of the simulation study was to assess the performance of the proposed algorithm with regard to estimation accuracy, convergence robustness and numerical

Table 1: Test Case Data

Test Case	No. of buses	Global Redundancy	Decomposition		
			No. of bus/area	Bound. buses	Area Redundancy
A1	14	1.7	5/7/6	4	1.4/1.3/1.4
A2	14	2.7	5/7/6	4	2.7/2.1/2.2
B1	30	1.4	14/19	3	1.5/1.3
B2	30	2.4	14/19	3	2.3/2.1
C1	118	1.5	47/39/38	6	1.5/1.3/1.6
C2	118	2.5	47/39/38	6	2.4/2.2/ 2.5

Table 2: Iterative Scheme Results

Iterative Scheme	Intermediate Tolerance	No. of Iterations Local/Constraints	J
S1	-	7/7	128.3
S2	-	5/2	0.48
S3	0.1	1.9/2,2/2,1.6/2	27.3
	0.01	2.7/2,2.3/2	0.46
	0.001	3.7/2,3.3/2	0.47

efficiency. Also, some tests were performed to evaluate the local nature of the state estimation process. No attempt was made to simulate communication delays as these are highly dependent on the parallel or distributed computer system used in practice. The simulation program is coded in Fortran 77 and the tests reported here were carried out in a DEC-Station 5000/200 computer under the Ultrix V4.2 operating system.

A. Simulation of a Distributed Environment

The simulation program was developed around a basic state estimation routine designed to be the node routine in an actual parallel or distributed implementation. This routine consists of three components:

- A fast decoupled state estimator as defined in method number 12 of [20].

- A routine to perform the coupling constraints enforcement as defined in (23,24).

- Routines to perform pseudo-communication functions consisting of the transference to and from a common memory buffer of data that would be interchanged between processors in an actual distributed implementation.

B. Test Cases

The IEEE 14-, 30-, and 118-bus standard networks were used to perform the tests reported in this section. The measurement systems were chosen to produce a relatively evenly distributed local redundancy ratio over the network. Measurement injections and line flows were always specified in active and reactive pairs. Two levels of global redundancy were specified for each measurement system: normal level and low level. Measurement errors were simulated adding normally distributed random numbers to loadflow results with parameters defined as in [20]. The network decomposition followed natural boundaries, identified by examining the one-line diagram of the test systems. Table 1 shows the data for each test case.

C. Iterative Schemes

Several iterative schemes can be used to implement the state estimation algorithm introduced in this paper. The following three were tested:

- S1: Apply the coupling constraint terms defined in (23,24) after every iteration of the local estimators given in (11,14).

- S2: First allow the local estimators to converge to the desired tolerance, and then apply the coupling constraint corrections without any further local estimation iterations.

- S3: Allow the local estimators to converge to a tolerance relatively close to the desired one, and from then on, use an alternating iterative scheme between the local estimations and the coupling constraints corrections.

Our computational experiments have shown that scheme S1 severely disturbs the local estimators iterative processes producing unacceptably inaccurate final results. Scheme S2 gives acceptable results, but slightly better results were achieved with scheme S3. For example, both schemes give similar values of J, but S3 takes fewer iterations to converge. In Table 2 some results obtained with the three iterative schemes above are shown for test case A2. In this Table, the number of iterations is the average number required in all area state estimations and J is the average sum of the squared normalized errors in the estimates of measured variables.

The distributed state estimator used for the remainder of this paper uses scheme S3 with the two tolerance thresholds set to 10^{-2} and 10^{-4}, respectively. The convergence of this iterative scheme appears to be robust, since similar results were found in all other test cases.

Iterative scheme S1 is a synchronous process requiring information exchange after every iteration of the local estimators. S2 can be implemented asynchronously as the values of the state variables in neighbor areas do not influence the local estimation iterative process and can be incorporated any time after the convergence of the local processes. S3 can be classified as a loosely synchronous process in which the external information is incorporated

in the local iterative process at a later stage, without requiring precise synchronism.

D. Estimation Accuracy Assessment

The accuracy of the algorithm introduced in this paper was assessed through several performance indices (PI) defined as follows:

1) Overall Estimation Accuracy: comparison of estimated and true values of state and measured variables. Two sets of indices were used: a) maximum and average percentage error in phase angles[1] and voltage magnitudes $(e_\theta^m, e_\theta^a, e_v^m, e_v^a)$; b) J as defined above.

2) Discrepancy in Boundary Variables: maximum and average discrepancy between the estimated values of phase angles and voltage magnitudes (pu) of boundary buses modeled as different buses in adjacent areas $(d_\theta^m, d_\theta^a, d_v^m, d_v^a)$.

The results of the distributed state estimation (DSE) algorithm are compared in Table 3 with those of the estimates obtained with the conventional integrated state estimator (ISE) and a completely independent area state estimation procedure (ASE). In almost all the cases studied, the DSE was an improvement over the ASE in terms of the overall estimation accuracy measures.[2] The discrepancies in boundary variables for the DSE are negligible and are not shown in this Table.

E. Efficiency

The efficiency of the distributed estimator is defined by the following formula :

$$E = 100\ T_{ise}/\ M\ T_{dse} \qquad (26)$$

where T_{ise} and T_{dse} are the elapsed time for the integrated and distributed state estimation calculations, respectively, and M is the number of areas.

Although no communication delays were simulated in the tests reported in this section, the calculated efficiencies give a first approximation to those expected in an actual parallel or distributed implementation. It should be noted that the network decomposition used in the examples was not optimized to produce maximum efficiency. The state estimation algorithms, both in the integrated and distributed form, involves two stages: a) noniterative calculations consisting of the calculation and factorization

---

[1]For this distributed state estimation, a reference angle was taken in each area, and matching at area boundaries was used to derive a system reference angle.

[2]For the ASE, the discrepancies in voltage magnitudes are higher than those in phase angles, although the voltage magnitude estimation accuracies are smaller. This is possible since estimation accuracies are percentage errors from the solution, and a more accurate set may still contain larger discrepancies.

Table 3: Estimation Accuracy Results

| PI	Case	A1	A2	B1	B2	C1	C2
	ISE	0.64	0.38	0.83	0.34	0.72	0.39
J	ASE	0.71	0.48	0.86	0.38	0.74	0.55
	DSE	0.74	0.46	0.85	0.35	0.73	0.52
	ISE	2.64	1.15	0.57	0.76	2.76	0.52
e_θ^m	ASE	3.83	2.25	3.57	0.77	9.07	1.76
	DSE	3.83	2.24	2.52	0.76	9.09	1.23
	ISE	2.18	0.80	0.27	0.30	0.22	0.18
e_θ^a	ASE	0.99	1.22	1.73	0.26	0.69	0.23
	DSE	2.72	1.13	1.69	0.25	0.68	0.24
	ISE	1.10	0.48	0.22	0.19	0.39	0.24
e_v^m	ASE	2.77	1.63	1.30	0.19	0.95	0.36
	DSE	1.59	1.51	1.29	0.19	0.94	0.35
	ISE	1.00	0.39	0.18	0.10	0.15	0.12
e_v^a	ASE	0.99	1.01	0.92	0.13	0.38	0.15
	DSE	0.99	0.70	0.90	0.12	0.37	0.15
d_θ^m	ASE	0.003	0.003	0.002	0.001	0.008	0.003
d_θ^a	ASE	0.001	0.002	0.001	0.001	0.004	0.001
d_v^m	ASE	0.015	0.023	0.021	0.003	0.024	0.003
d_v^a	ASE	0.010	0.017	0.009	0.002	0.017	0.001
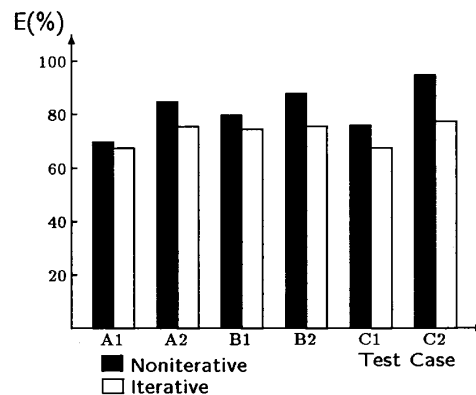
Figure 1: Distributed Estimation Efficiency

of the gain matrices; b) iterative calculations. The efficiency for both stages in the calculations were obtained for the test cases and are shown in Figure 1. The program elapsed times used in these calculations are average values obtained in a time sharing machine. In all cases there is reasonable speedup in the calculations due to the use of multiple processors. Similar results (slightly better for the iterative calculations) were obtained for the ASE.

F. Tracking State Estimation

The ability of the proposed distributed state estimation algorithm to track the state vector time evolution through incremental updates [21] was assessed in an experiment performed with the 14-bus system. The objective was to simulate a smooth but steep load evolution
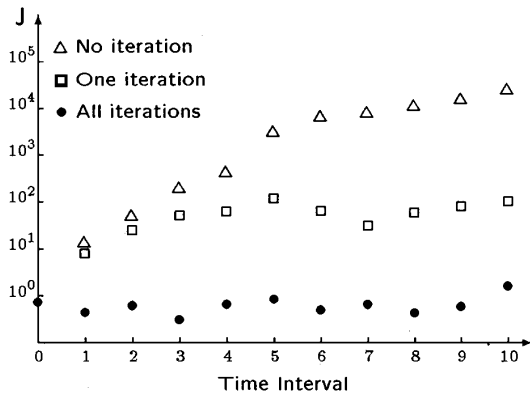
Figure 2: Tracking Estimation Performance

pattern, like for example in morning load pick-up. A series of 11 consecutive measurement sets for test case A2 were generated with an approximate 1% increase in total load from one time instant to the other. Three different estimation schemes were tested:

- Full iterative scheme (S1) in all intervals.

- Only one iteration in all intervals except interval 0, in which the full iterative scheme was used.

- The same as in the previous item except that in area 3 there was no processing of new measurement information in time intervals 1-10. Only the information obtained from the coupling constraints enforcement (23,24) is used.

The results obtained for area 3 are presented in Figure 2. These results show the ability of the algorithm to easily follow the system state time evolution (only one iteration per interval), with an acceptable and stable accuracy level. They also show its inability to transfer information from one area to the other through the coupling constraints.

IV. DISCUSSION

We make the following observations :

- The state estimation problem exhibits a naturally decoupled characteristic well suited for parallel and distributed processing. As long as observability is preserved, independent area state estimation (state estimation in each area without information exchange between areas) provides adequate results in terms of accuracy, convergence robustness and computational efficiency. The only drawback of this approach is the discrepancy in the values of boundary bus state variables estimated using different sets of measurements. In most cases, particularly the ones with higher redundancy levels, the values of the discrepancies are acceptable.

- The proposed distributed approach improves the accuracy of the overall estimation and eliminates the discrepancy in the boundary bus state estimates with minimal effect on the computational efficiency. This approach can be implemented using conventional algorithms for integrated state estimation.

- The iterative scheme used in the distributed state estimation should take into consideration the local nature of the estimation problem by allowing a first approximation of the local solutions before starting the information exchange and coupling constraints enforcement.

- Preliminary tests showed an adequate performance of the distributed algorithm in tracking mode operation. This indicates that the proposed method can be used to implement very fast incremental state estimation schemes.

- The proposed methodology can be implemented both in a distributed and parallel environment. In the distributed case, a distributed state estimation process would allow a better use of computer resources available throughout the geographically and functionally distributed power network, and the minimization of communication requirements. In the parallel case, a parallel implementation of an incremental state estimation process may produce state estimates at a rate comparable to the ones of modern data acquisition systems.

- The presence of bad data in state estimation is known to cause the accuracy of the results to deteriorate unless it is detected and removed. The proposed distributed scheme S3 has the advantage that bad data can be detected and removed during the local estimation stage using a standard approach (for example, the largest normalized residue test), so that the problem does not even come up in the coupling constraint correction stage.

Bad data close to the boundary, if not removed, may cause large discrepancies in boundary variables and may seriously affect convergence of distributed state estimation. The problem may be alleviated by selecting boundaries to ensure sufficient local redundancy to minimize the impact of bad data.

V. CONCLUSIONS

This paper has introduced a methodology for parallel and distributed state estimation based on conventional al-

gorithms associated with a coupling constraints optimization technique. The results of computational experiments indicate that the proposed algorithm is accurate, robust and efficient in terms of reducing the required computation time. An important by-product of the work reported in this paper is the indication that state estimation is a naturally decoupled problem. Important parallel and distributed processing issues, such as data base access and communication between processors, were not addressed here. In addition, bad data processing and observability analysis, in the distributed context, were not simulated and require further research. A strong practical advantage of the methodology introduced in this paper is the use of standard state estimation algorithms. However, this should not limit the investigation of other algorithmic approaches more tailored to the parallel or distributed aspects of the state estimation problem.

REFERENCES

[1] A. Bose and K.E. Clements, "Real-Time Modeling of Power Networks", *Proceedings of the IEEE*, vol. 75, no. 12, pp. 1607-1622, December 1987.

[2] F.F. Wu, "Real-Time Network Security Monitoring, Assessment and Optimization", *Int. J. of Electrical Power and Energy Systems*, vol. 10, no. 2, pp. 83-100, April 1988.

[3] M.B. Do Coutto Filho, A.M. Leite da Silva, and D.M. Falcão, "Bibliography on Power System State Estimation (1968-1989)", *IEEE Transactions on Power Systems*, vol. 5, no. 3, pp. 950-961, August 1990.

[4] L. Mili, V. Phaniraj, and P.J. Rousseeuw, "Least Median of Squares Estimation in Power Systems", *IEEE Transactions on Power Systems*, vol. 6, no. 2, pp. 511-513, May 1991.

[5] E. Liu, "Implementation Issues in Power System State Estimation", *Proceedings of the EPRI/NSF Workshop on Application of Advanced Mathematics to Power Systems*, Redwood City, CA, September 1991.

[6] F. F. Wu, "Power System State Estimation : A Survey", *Electrical Power and Energy Systems*, vol. 12, April 1990, pp 80-87.

[7] L. Murphy and F.F. Wu, "An Open Design Approach for Distributed Energy Management Systems", *IEEE Transactions on Power Systems*, vol. 8, no. 3, pp. 1172-1179, August 1993.

[8] S.M. Lun, T. Lo, F. Wu, L. Murphy, and A. Sen, "LANSIM and its Applications to Distributed EMS", *Proceedings of the 18th PICA Conference*, Scottsdale, AZ, May 1993.

[9] "Parallel Processing in Power System Computation", An IEEE Comittee Report by a Task Force of the Computer and Analytical Methods Subcommittee of the Power Systems Engineering Comittee, *IEEE Transactions on Power Systems*, vol. 7, no. 2, pp. 629-637, May 1992.

[10] D.P. Bertsekas and J.N. Tsitsikilis, *Parallel and Distributed Computation*, Prentice Hall, 1989.

[11] Th. Van Cutsem and M. Ribbens-Pavella, "Critical Survey of Hierarchical Methods for State Estimation of Electric Power Systems", *IEEE Transactions on Power Apparatus and Systems*, vol. 102, no. 10, pp. 3415-3424, October 1983.

[12] K. Seidu and H. Mukai, "Parallel Multi-Area State Estimation", *IEEE Transactions on Power Apparatus and Systems*, vol. 104, no. 5, pp. 1026-1034, May 1985.

[13] H. Sasaki, K. Aoki, and R. Yokoyama, "A Parallel Computation Algorithm for Static State Estimation by Means of Matrix Inversion Lemma", *IEEE Transactions on Power Systems*, vol. 2, no. 3, pp. 624-632, August 1987.

[14] S. Iwamoto, M. Kusano, and V.H. Quintana, "Hierarchical State Estimation Using a Fast Rectangular Coordinate Method", *IEEE Transactions on Power Systems*, vol. 4, no. 3, pp. 870-880, August 1989.

[15] A. Abur and P. Tapadiya, "Parallel State Estimation Using Multiprocessors", *Electrical Power Systems Research*, vol. 18, pp. 67-73, January 1990.

[16] A.A. El-Keib, J. Nieplocha, H. Singh, and D.J. Maratukulam, "A Decoupled State Estimation Technique Suitable for Parallel Processor Implementation", *IEEE Transactions on Power Systems*, vol. 7, no. 3, pp. 1088-1097, August 1992.

[17] S.Y. Lin, "A Distributed State Estimator for Electric Power Systems", *IEEE Transactions on Power Systems*, vol. 7, no. 2, pp. 551-557, May 1992.

[18] A. Monticelli and A. Garcia, "Modeling Zero Impedance Branches in Power System State Estimation", *IEEE Transactions on Power Systems*, vol. 6, no. 4, pp. 1561-1566, November 1991.

[19] F.F. Wu and A.F. Neyer, "Asynchronous Distributed State Estimation for Power Distribution System", *Proceedings of the 10th Power System Computation Conference*, Graz, Austria, August 1990.

[20] J.J. Allemong, L. Radu, and A.M. Sasson, "A Fast and Reliable State Estimation Algorithm for AEP's New Control Center", *IEEE Transactions on Power Apparatus and Systems*, vol. 101, no. 4, pp. 933-944, April 1982.

[21] D.M. Falcão, P.A. Cooke, and A. Brameller, "Power System Tracking State Estimation and Bad Data Processing", *IEEE Transactions on Power Apparatus and Systems*, vol. 101, no. 4, pp. 844-853, April 1982.

BIOGRAPHIES

Djalma M. Falcão (M'75) obtained his B.Sc. (1971) and M.Sc. (1974) from the Federal Universities of Paraná and Rio de Janeiro, Brazil, respectively, and the Ph.D. (1981) from UMIST, England. He is currently a professor of Electrical Engineering at the Federal University of Rio de Janeiro. His general research interest is in the area of computer methods for power system simulation and control.

Felix F. Wu received his Ph.D. from UC Berkeley, where he is currently Professor and Vice Chairman of the Department of Electrical Engineering and Computer Sciences. Dr. Wu is a Fellow of the IEEE.

Liam Murphy received his Ph.D. from UC Berkeley, where he is currently a Lecturer in the EECS Department. Dr. Murphy is a Member of the IEEE.