A Parallel and Distributed Algorithm to Power Systems State Estimation

José Beleza Carvalho
Instituto Superior de Engenharia do Porto
R. de S. Tomé, 4200 Porto, Portugal
e.mail: jabc@dee.isep.ipp.pt

F. Maciel Barbosa
Faculdade de Engenharia da Universidade do Porto
Rua dos Bragas, 4099 PORTO Codex, Portugal
Fax +351-2-319280 ; e.mail: fmb@fe.up.pt

Abstract: The fast development in electrical power networks and its consequent complexity, impose a more useful state estimation operational scheme. Its important to short 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. Technical developments in distributed Energy Management Systems (EMS), based on fast data communication networks, open up the possibility of parallel and distributed implementation of the state estimation function.

This paper introduces and discusses some ideas for the implementation of parallel and distributed state estimation in electrical power systems. Will be boarding a solution methodology based on conventional state estimation algorithms applicable in local networks. Some results obtained from IEEE test system are analysed, so that we can appreciate the performances of the proposed method. An evaluation of the degree of natural decoupling in the state estimation problem is also performed, and compared with the results of computational experiments with standard WLS methods, in distributed version.

Key Words: Power Systems, Static State Estimation, Parallel Processing, Distributed Processing.

1. INTRODUCTION

The implementation of robust methods for power system state estimation, which maintain performance suitable to the large models encountered in modern control centres is a topic that has received significant attention. Basically, we can define the state estimator as a calculation program carried in real time with the purpose of providing a complete, coherent and reliable database, which can describe the electrical state of the network [1]-[2].

To accomplish such purpose, the estimator processes a group of measurements and other information collected from the network at a certain moment, thus getting an estimate for the respective state vector (vector of voltages modules and phases on different buses) [3].

The higher frequency in state estimation execution requires the development of faster state estimation algorithms. The larger size of the supervised networks will increase the demand on the numerical stability of the algorithms. At same time, conventional centralised state estimation methods have reached a development stage in which important improvements in either speed or numerical robustness are not likely to occur. These facts, together with the technical developments in fast data communication network technology, opens up the possibility of parallel and distributed implementations of the state estimation algorithms [4]-[5].

The nature geographically distributed of power system applications can get a benefit from this form of decentralised computer architecture, in which several remote processors perform local state estimation in network areas. The results are send to a central computer that refines the calculation. The power system under consideration may be partitioned into k areas. Each area is governed by a local control center.

The measurement data in each area will be collected in each individual local control center that has at least one computer system for data acquisition, data processing, and computation [9]. The computer systems of adjacent areas are connected by fast data communication links, and these decentralised computer systems form a computer network.

2. WLS STATE ESTIMATION PROBLEM

Mathematically, the state estimation problem of a N-bus power system is described below as a WLS problem:

$$\min J(x) = \sum_{i=1}^{m} w_i (z_i - h_i(x))^2 = [z - h(x)]^T W [z - h(x)] \quad (1)$$

Subject to:

$$z = h(x) + e \quad (2)$$

where z is a (m*1) measurement vector, x is a (n*1) true state vector, $h(.)$ is a (m*1) vector of non-linear functions, e is a (m*1) measurements error vector, m is the number of measurements, and n is the number of state variables. The usual choice for state variables are the voltage and phase angles magnitudes, while the measurements are the active and reactive power flows, node injections and voltage magnitudes. The weights w_i represent the weight associated with measurement z_i. These weights are chosen as proportional to the accuracy of the measurements: the higher the accuracy of a measurement the bigger its weight. The solution of this optimisation problem gives the estimated state $\hat{x}$, which must satisfy the following optimality condition:

$$\frac{\partial J(x)}{\partial x} = 0 \Rightarrow H^T(\hat{x})W\left[z - h(\hat{x})\right] = 0 \tag{3}$$

where $$H(x) = \frac{\partial h(x)}{\partial x}$$

is the Jacobean matrix of the measurement function h(x). The solution of the non-linear equation (3) may be obtained by an iterative method in which a linear equation of following type is solved at each iteration to compute the correction, $x^{t+1} = x^t + \Delta x^t$,

$$\left[G(x^t)\right]\Delta x^t = H^T(x^t)W\left[z - h(x^t)\right] \tag{4}$$

where $G(x)$ is called the gain matrix and is usually chosen as

$$G(x) = H^T(x)WH(x)$$

Eq.(4) is called the normal equation of the WLS problem.

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

$$z_p = h_p(\theta, v) + e_p \tag{5}$$

$$z_q = h_q(\theta, v) + e_q \tag{6}$$

where θ (n_θ*1) and v (n_v*1) are the vectors of true voltage magnitudes and phase angles, p and q indicating partitions of vectors and matrices corresponding to active and reactive measurements, respectively, n_θ = N-1, n_v = N, and N is the number of network nodes. This naturally decoupled characteristic, make this method suitable for parallel processing implementation, with a great reducing of the required computation time.

3. PARALLEL AND DISTRIBUTED STATE ESTIMATOR PROBLEM

If we decompose the power network into "K" areas, connected through boundary buses which belongs simultaneously to both adjacent areas, the state estimation problem introduced in (5) and (6) can be presented as

$$z_p^k = h_p^k(\theta^k, v^k) + e_p^k, \qquad k = 1,....K \tag{7}$$

$$z_q^k = h_q^k(\theta^k, v^k) + e_q^k, \qquad k = 1,....K \tag{8}$$

where z_p^k and z_q^k are vectors of active and reactive measurements in area k; θ^k and v^k are vectors of voltage phase angles and magnitudes in area k, including the ones corresponding to the boundary buses. The number of boundary buses may be kept to a minimum and there are no injection measurements in the overlapping area buses. This is not a limitation, because actual injection measurement buses in overlapping areas, can be replaced by fictitious buses with no injection measurements connected to the actual buses, now placed outside the overlapping area, by zero impedance lines [10]. Then, the problem of parallel state estimation is to use the computer network associated with the measurement data collected in each local control center to solve the following weighted least square (WLS) problem in a distributed way:

$$\min \sum_{k=1}^{K}\left[z_p^k - h_p^k(.)\right]^T\left[R_p^k\right]^{-1}\left[z_p^k - h_p^k(.)\right] = 0 \tag{9}$$

$$\min \sum_{k=1}^{K}\left[z_q^k - h_q^k(.)\right]^T\left[R_q^k\right]^{-1}\left[z_q^k - h_q^k(.)\right] = 0$$

The iterative solution of above problem is:

$$\theta^k(i+1) = \theta^k(i) + \left[G_p^k\right]^{-1}\left[H_p^k\right]^T\left[R_p^k\right]^{-1}\left[z_p^k - h_p^k(\theta_k(i), v_k(i))\right]$$
k=1,....K (10)

$$v^k(i+1) = v^k(i) + \left[G_q^k\right]^{-1}\left[H_q^k\right]^T\left[R_q^k\right]^{-1}\left[z_q^k - h_q^k(\theta_k(i), v_k(i))\right]$$
k=1,...,K (11)

where

$$G_p^k = \left[H_p^k\right]^T\left[R_p^k\right]^{-1}H_p^k$$
$$G_q^k = \left[H_q^k\right]^T\left[R_q^k\right]^{-1}H_q^k$$

and

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

are the Jacobean matrix, calculated for the initial conditions and kept constant in the iterative process.

In the boundary buses, the elements $(\overset{b}{\theta},\overset{b}{v})$ obtained in (10) and (11) must be affected with a weight medium of the values calculated in the neighbouring areas k and j [8], and take the form

$$\overset{b}{\theta}{}^{k}(i+1)=\theta^{k}(i+1)+\Delta\theta^{k}(i+1) \qquad (12)$$

$$\overset{b}{v}{}^{k}(i+1)=v^{k}(i+1)+\Delta v^{k}(i+1) \qquad (13)$$

Where

$$\Delta\theta_r^k(1+1)=\frac{g_{rr}^k}{g_{rr}^k+g_{rr}^j}\left[\theta_r^k(i+1)-\theta_r^j(i+1)\right] \qquad (14)$$

$$\Delta v_r^k(1+1)=\frac{g_{rr}^k}{g_{rr}^k+g_{rr}^j}\left[v_r^k(i+1)-v_r^j(i+1)\right] \qquad (15)$$

g_{rr}^k and g_{rr}^j are diagonal elements corresponding to boundary bus r of the inverse gain matrices of the neighbouring area k and j, respectively.

4. ANALYSIS OF COMPUTATION EXPERIMENTS

The Parallel and Distributed State Estimation methodology analysed in this paper was tested and simulated with a PVM 3.1 (Parallel Virtual Machine) software, with program coded in Fortran 77 and running in a DEC Alpha machine with a Ultrix operating system. The distributed computer system, connected in a network, used in practice for parallel or distributed areas processing, was simulated with recurrence to PVM performances [6], that enables one to distribute tasks on various processors, to control message-passing between tasks, to synchronise tasks, etc. The convergence, accuracy and numerical efficiency of the proposed simulation study are presented in the following sections.

A. Parallel Processing in the Integral Version

The algorithm implemented for this integral study version is represented in Figure 1.

The nature decoupled of equation (10) and (11) make the algorithm suitable for parallel implementation. The algorithm presented in flowchart, calculates the θ and v, update at every iteration in a synchronous way. The IEEE 14, 57 and 118 bus standard networks were used to perform this test. Two levels of global redundancy were specified for each measurement system: normal and low level. Table 1 shows the data for each test case. In this table J is the sum of squared errors in the estimates of measured variables.

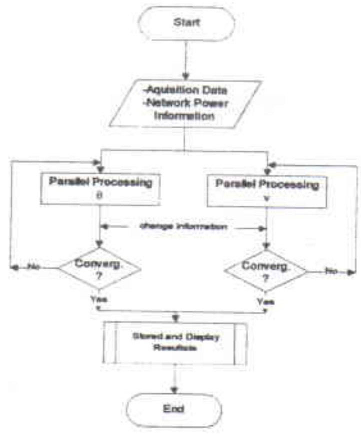

Fig. 1. Parallel Processing. Integral version.

TABLE I
TEST CASE DATA

Test Case	N° of Bus	Redun dancy	WLS		MDE		P Process.	
			t(s)	J	t(s)	J	t(s)	J
A1	14	1.9	0.13	24.6	0.10	28.6	0.10	28.6
A2	14	2.6	0.17	50.8	0.11	52.9	0.38	52.9
B1	57	1.7	5.40	91.5	1.60	100	1.60	100
B2	57	2.3	9.00	172	2.70	185	2.70	185
C1	118	2.3	115	419	30.0	425	34	425
C2	118	3.2	220	802	70.0	806	74	806

All test simulations converge in 2 iterations for standard WLS method and 8 iterations for standard decoupled estimator (MDE) and Parallel Processing. The convergence is obtained at 0.001 pu and 0.001 rad. for module and phase of voltage. From we can see that Figure 3, the Parallel Processing in integral version is not so accurate like the MDE method. In a synchronous process, due to idle times, the algorithm has to wait until the state vector is updated before it starts a new iteration. If we run the above algorithm in an asynchronous way, the precision of state estimation vector will be drastically deteriorated.

B. Parallel Processing in the Distributed Version

The geographical distribution of the power network and the increasing sophistication of the modern microprocessor-based remote terminal units (RTUs) are a motivation for the adoption of significant local processing of information by such RTUs before communication to the control centre. Synchronous computation become too expensive when the processors are geographically distributed [7]. So, asynchronous concurrent processing is an attractive alternative. We analysed this fact, dividing the test cases presented in Table 1 in some areas and processing the equations (10) and (11) for each area, like shown in flowchart of Figure 2. For the boundary buses, in the end of the asynchronous iterative process, we applied the restriction

(12) and (13). The characteristics of each area, for the same test cases, presented in Table 1, are shown in Table 2.

TABLE 2
PARALLEL PROCESSING OF DISTRIBUTED AREAS.

Test Case	N° of Bus / Area	Boundary Buses	Area Redundancy
A1	5 – 6 – 7	4	1.6 / 1.7 / 1.6
A2	5 – 6 – 7	4	2.5 / 1.8 / 2.1
B1	30 – 33	6	1.6 / 1.5
B2	30 - 33	6	2.2 / 2.0
C1	34 – 35 – 33 – 29	12	2.0 /2.0 /2.0 /2.0
C2	34 – 35 – 33 - 29	12	3.1 /3.0 /3.0 /3.0

For all the tests presented in Table 2, we simulated the parallel processing of distributed areas in the standard version WLS and in the decoupled method MDE, in the asynchronous way. The convergence obtained for 0.001 pu and 0.001 rad, the processing time, accuracy and numerical efficiency are shown in Table 3 for WLS version, and Table 4 for MDE version.

The results presented demonstrate that in parallel distributed state estimation, we can get an elevated reduction of processing time, for essentially the same number of iterations, compared with integral methods showed in Table 1. The accuracy of results, generally, is better for cases with more redundancy of measurements and for WLS state estimation version. The improvement in processing time for MDE method, compensate the small depreciate of results, compared with WLS version.

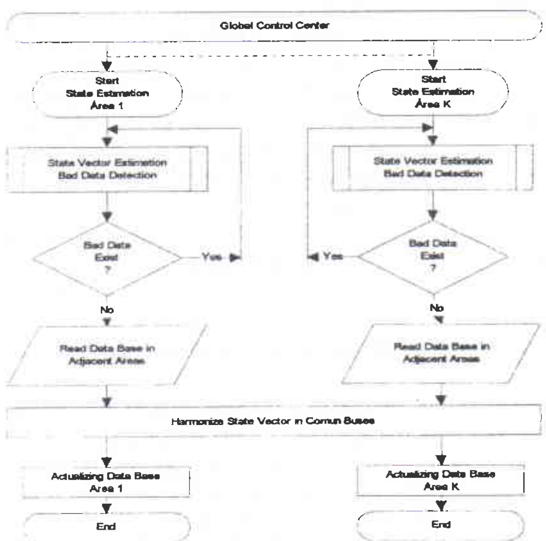

Fig. 2: Parallel Processing, Distributed Version

In Figure 3 we can see the performance of Parallel Processing in the Distributed Version (PPD), applicated to test case C2 (118 buses), comparing the processing time for standard WLS and MDE state estimation methods and the Parallel Processing in the Integral (PPI) and Distributed version.

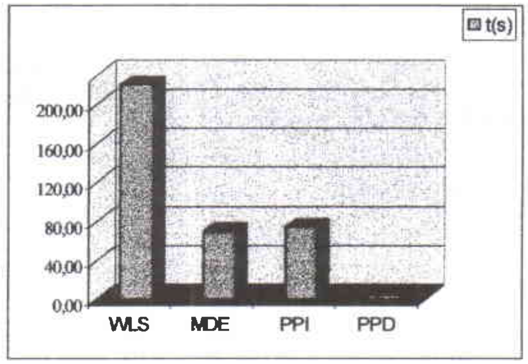

Fig.3. Parallel Processing Improvement

TABLE 3
PARALLEL PROCESSING OF DISTRIBUTED AREAS.
ESTIMATION ACCURACY FOR WLS VERSION

Test Case	N° of Iter	Time (s)	Average error in phase angles (rad*1000)	Average error in voltage magnitud (pu*1000)	J
A1	2 ; 4 ; 3	0.10	0.38 - 1.87 – 1.24	0.76 - 1.8 – 1.29	6.4 – 3.5 – 11.6
A2	2 ; 3 ; 5	0.12	0.28 - 0.87 – 0.59	0.98 - 1.03 – 0.76	12.7 – 12.2 – 22.3
B1	2 ; 3	1.15	0.92 – 1.90	1.30 – 2.00	35.5 – 50.1
B2	2 ; 3	1.70	0.88 – 1.70	0.95 – 1.06	71 – 86
C1	2; 2; 2; 4	8.00	0.59-1.69-0.35-1.02	0.81-0.95-0.81-1.29	51-69-111-174
C2	2; 2; 2; 3	15.00	0.36-0.84-0.18-0.3	1.02-0.95-0.96-0.3	152-143-226-290

TABLE 4
PARALLEL PROCESSING OF DISTRIBUTED AREAS.
ESTIMATION ACCURACY FOR MDE VERSION

Test Case	N° of Iter	Time (s)	Average error in phase angles (rad*1000)	Average error in voltage magnitud (pu*1000)	J
A1	4 ; 3 ; 6	0.098	0.48 - 1.88 – 1.09	0.86 - 1.9 – 1.5	6.4 – 7.3 – 11.6
A2	4 ; 3 ; 6	0.10	0.40 - 0.85 – 0.57	1.04 - 1.02 – 1.03	12.7 – 16 – 24
B1	6 ; 8	0.37	1.08 – 2.1	1.45 – 2.07	38.3 – 49.1
B2	5 ; 8	0.54	1.02 – 1.9	1.0 – 0.78	81 – 95
C1	5; 9; 11; 6	2.00	0.56-2.37-0.30-1.55	0.81-0.97-0.82-0.96	63-75-114-146
C2	5; 6; 11; 5	4.00	0.41-1.85-0.18-0.31	0.95-0.96-0.94-1.08	167-161-228-335

C. Bad Data Processing

A distributed bad data detection and identification schema is used for a distributed computing environment. The method we employ here is based on reduced model proposed in [11]. Therefore, if the measurement meters are properly collocated, each of the k areas is a maximal non-critical area, and each local computer system can process bad data of corresponding area independently. In this case, we will perform bad data detection based on statistical hypothesis by testing the value of $J(\hat{x})$ (has chi-square distribution), and the value of normalized residue, that is:

$$J_k(\hat{x}_k) < \alpha_k \quad ; \quad \left|r_{kj}^N\right| < \beta_{kj} \qquad (16)$$

where $r^N_{k,j}$ is the normalized residue of j^{th} bus in area k, α_k is the objective function detection threshold for area k, and $\beta_{k,j}$ is the normalized residue detection threshold for j^{th} bus in area k. Then, bad data are assumed to be present if any one of the above two tests fails. Once the existing bad data is detected in an area, we will employ an existing bad data identification method, such as the HTI (Hypothesis Test Identification) method, to identify either single or multiple bad data in that area.

5. CONCLUSIONS

In this paper was introduced and tested some methodologies for parallel state estimation, based in conventional algorithms like standard WLS version and standard decoupled MDE version. The results of computational experiments indicate that for integral processing of state estimation, the parallelism of algorithms does not bring any improvement, compared with the conventional decoupled MDE algorithm.

A distributed computing is the better way to adopt the parallel computing in power systems energy. This fact was simulated divided the IEEE standard test cases in some areas and with help of PVM software tool, that enable the simulation of distribute tasks on various processors, message passing between tasks, synchronise tasks, etc.

Because the idle times of processors, synchronous computations become too expensive when the processors are geographically distributed. So, we tested the asynchronous processing, and for boundary buses, we apply the restrictions indicated in (12) and (13). The computational results show that with this distributed methods we get a very high improvement in manner of time processing compared with integral standard version.

The only drawback is the discrepancy in values of boundary bus state variables estimated using different sets of measurements, but in cases with higher redundancy levels, the values of the discrepancies are acceptable and the effect on computational efficiency is minimal.

6. REFERENCES

[1] Do Couto Filho, M.B.; Leite da Silva, A.M. e Falcão, D.M., - *Bibliography On Power System State Estimation (1968-1989)*. IEEE Trans. on PWRS, Vol.5, n° 3, August 1990.

[2] A. Bose, K.E. Clements – *Real Time Modeling of Power Networks.* Proceedings of the IEEE, Vol. 75, n°12, December 1987.

[3] Masiello, R.D. e Schweppe, F.C. - *A Tracking Static State Estimator*, IEEE Trans. on PWRS, Vol. PAS-90, March / April 1971.

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

[5] Felix F. Wu and al, *Parallel Processing in Power Systems Computation.* IEEE Trans. On PWRS, Vol.7, August 1992.

[6] Jack Dongarra, Adam Berguelin. – *PVM: Parallel Virtual Machine.* Vaidy Sunderam, 1994.

[7] V.C. Ramesh, - *On Distributed Computing For On-Line Power System Applications.* Electric Power & Energy Systems, Vol. 18, 1996.

[8] Falcão, Djalma M. - *Parallel And Distributed Processing Applications in Power Systems Simulation and Control.* COPE, Universidade Federal do Rio de Janeiro, July 1995.

[9] Falcão, D.M.; Felix F. Wu e Liam Murphy, - *Parallel And Distributed State Estimation.* IEEE PES Summer Meeting, San Francisco, July 1994.

[10] Monticelli, A. e Garcia, A. - *Modeling Zero Impedance Branches in Power System State Estimation,* IEEE Trans. on PWRS, Vol. 6, November,1991.

[11] Korres, G.N. e Contaxis, G.C. – *Reduced Model for Bad Data Processing,* IEEE Trans. on PWRS, Vol. 6, May,1991.

7. BIOGRAPHIES

José António Beleza Carvalho was born in Oporto, Portugal, in 1959. He received the "B.S." degree in Electrical Engineering from the Institute of Electrical Engineering of Oporto in 1991 and M.Sc. degree from the Faculty of Electrical Engineering of Oporto University in 1994. He is presently Professor at Institute of Electrical Engineering of Oporto, in Power System Area. His research interests are in Power Systems State Estimation and in the Parallel and Distributed Computations.

Fernando Pires Maciel Barbosa was born in Oporto, Portugal, in 1948. He received the "Licenciatura" degree, after a five years course, in Electrical Engineering, from the Faculty of Electrical Engineering of Oporto University (FEUP) in 1971 and M.Sc. and Ph. D. degrees from UMIST in 1977 and 1979 respectively. He is presently Professor at FEUP in Power System Area. His research interests are in the area of power system analysis and control, with special emphasis on state estimation and transient stability.