Sensor Placement for Real-Time Power Flow Calculations in Transmission Networks

E. Feinberg, M. Li, R. Samulyak

Stony Brook University
Stony Brook, NY
Email: Eugene.Feinberg@stonybrook.edu
Email: Muqi.Li@stonybrook.edu
Email: Roman.Samulyak@stonybrook.edu

B. Fardanesh, G. Stefopoulos

New York Power Authority
White Plains, NY
Email: Bruce.Fardanesh@nypa.gov
Email: George.Stefopoulos@nypa.gov

***Abstract*—This paper describes a sensor placement algorithm for real-time parallel power flow computations for transmission networks. In particular, Phasor Measurement Units (PMUs) can be such sensors. Graph partitioning is used to decompose the system into several subsystems and to locate sensors in an efficient way. Power flow calculations are then run in parallel for each area. Test results on the IEEE 118- and 300-bus systems show that the proposed approach is suitable for real-time applications.**

Keywords–Parallel computing; power system; power flow calculation; PMU placement.

I. INTRODUCTION

Fast load flow analysis is essential for the successful implementations of advanced real-time control of transmission systems.The natural tool for this is parallel computing. To this end, various parallel methods have been proposed. Many traditional approaches [1][2][3] use factorization and the forward-backward solution of linear equations to achieve parallelism. As many serial computations are needed for such approaches, their parallel efficiency is not high.

Another approach to alleviating the computational burden is to decompose a large problem into a number of small problems and perform computations for each sub-problem in parallel. These smaller sub-problems are usually coordinated by a master process. Rafian et al. [4] presented a method for load-flow analysis based on tearing the network into 2-3 subsystems. In every iteration, the subsystems are solved in parallel and will communicate with a coordinating program. After the communication, the coordinating program is then conducted to determine the global solution for the original system. Chan [5] et al. proposed a parallel solution based on piecewise method. The Jacobian matrix is converted into a bordered block diagonal form. The block diagonal form leads to subproblems that can be solved independently, after which a problem corresponding to the border is solved to coordinate the subproblems and obtain a solution to the original problem. Amano et al. [6] employed a block-parallel method for load-flow analysis. In particular, the Jacobian matrix is constructed by applying the epsilon decomposition algorithm, which eliminates weak coupling elements from the matrix. One drawback of this approach is that the transformation of the matrix into a balanced diagonal matrix takes time; further, the speed of convergence is affected by the choice of partition method.

Traditional power flow calculations use only power injection measurements as the input. However, at the present time it is also possible to have synchronized voltage measurements at buses. For example, it is possible to use Phasor Measurement Units (PMUs) for this. These devices provide accurate real-time measurements at multiple points on the grid. A number of significant improvements in control and analytical capabilities have been made possible by this technology. However, due to their advanced features and the need for communications infrastructure, PMUs are relatively costly to implement and maintain. Hence it is typically not economical to install a PMU at each bus [7]. This has motivated a growing literature on the optimal placement of PMUs, and various PMU placement algorithms have been developed for different situations. Xu and Abur [8] have modeled the PMU placement problem as an integer linear programming problem with the constraint that the entire power system should be observable. A mixed integer linear programming formulation was introduced by Aminifar et al. [9] for PMU placement that accounts for line and PMU contingencies. Gou [10] presented a model that allows for redundant PMU placement and incomplete observability, which can be solved by integer linear programming. Chakrabarti and Kyriakides [11] proposed an exhaustive search algorithm for PMU placement that takes so-called zero injection buses, i.e., buses with neither generation nor load, into account. In particular, systems with such buses may need fewer PMUs.

This paper investigates the possibility of using additional sensor measurements, e.g., from PMUs, to decompose the power system into several parts and conduct power flow calculation in parallel. The idea of using PMU is motivated by decomposition method, e.g., [4][5], for solving power flow problem. Instead of using a master process to coordinate the subsystems in each iteration, we achieve the coordination directly with PMU measurements. Using this approach, no data needs to be transferred between subsystems, which results in a simple and efficient solution. The placement of PMUs is based on solving a graph partition problem. By placing a relatively small number of additional PMUs, a large power system can be divided into several non-overlapping smaller subsystems that can be solved in parallel. Numerical results on 118 and 300 bus power networks demonstrate the efficacy of the proposed approach in significantly reducing the computation time.

The rest of the paper is organized as follows. Section II

Copyright (c) IARIA, 2015. ISBN: 978-1-61208-406-0

briefly describes the power flow problem, and Section III introduces the proposed method. Sections IV describes the details of PMU placement. Section V illustrates the implementation of the algorithm via numerical examples on two power systems.

II. Power flow Analysis

Power flow analysis, commonly known as load flow analysis, is an important part of power system analysis. The goal of the power flow analysis is to obtain complete voltage angle and magnitude information for each bus in a power system under balanced three-phase steady state conditions. The power balance equations can be written as:

$$P_i = \sum_{j=1}^{n} |V_i||V_j|(G_{ij}cos\theta_{ij} + B_{ij}sin\theta_{ij}), \quad (1)$$

$$Q_i = \sum_{j=1}^{n} |V_i||V_j|(G_{ij}sin\theta_{ij} - B_{ij}cos\theta_{ij}). \quad (2)$$

Here, P_i and Q_i are the real and reactive power injections at bus i; $|V_i|$ is the bus voltage magnitude at bus i; θ_{ij} is the voltage angle difference between buses i and j; G_{ij} and B_{ij} are the real and imaginary parts of elements in the bus admittance matrix corresponding to buses i and j. Equations (1) and (2) constitute a set of nonlinear equations, and the number of equations is approximately twice the number of network buses. Expanding the above equations using Taylor series and ignoring the higher order terms results in the following equations:

$$\begin{bmatrix} \Delta P_2 \\ \vdots \\ \Delta P_n \\ \Delta Q_2 \\ \vdots \\ \Delta Q_n \end{bmatrix} = \begin{bmatrix} \frac{\partial P_2}{\partial \theta_2} & \cdots & \frac{\partial P_2}{\partial \theta_n} & \frac{\partial P_2}{\partial |V_2|} & \cdots & \frac{\partial P_2}{\partial |V_n|} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial P_n}{\partial \theta_2} & \cdots & \frac{\partial P_n}{\partial \theta_n} & \frac{\partial P_n}{\partial |V_2|} & \cdots & \frac{\partial P_n}{\partial |V_n|} \\ \frac{\partial Q_2}{\partial \theta_2} & \cdots & \frac{\partial Q_2}{\partial \theta_n} & \frac{\partial Q_2}{\partial |V_2|} & \cdots & \frac{\partial Q_2}{\partial |V_n|} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial Q_n}{\partial \theta_2} & \cdots & \frac{\partial Q_n}{\partial \theta_n} & \frac{\partial Q_n}{\partial |V_2|} & \cdots & \frac{\partial Q_n}{\partial |V_n|} \end{bmatrix} \begin{bmatrix} \Delta \theta_2 \\ \vdots \\ \Delta \theta_n \\ \Delta |V_2| \\ \vdots \\ \Delta |V_n| \end{bmatrix} \quad (3)$$

Equation (3) can be written in the form:

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = J \begin{bmatrix} \Delta \theta \\ \Delta |V| \end{bmatrix}, \quad (4)$$

where J is the Jacobian (or the matrix of partial derivatives) displayed in (3). The linearized system of equations is solved to determine the next approximation of voltage magnitude $|V|$ and angle θ, and the process continues until a stopping condition is met.

III. Proposed Real-Time Power Flow Calculation Method

In this paper, we propose an approach for power flow calculation based on power system decomposition and PMU placement. The proper placement of PMUs will decrease the dimensions of the sub-problems and improve the efficiency of computational procedures. Our method contains following steps:

A. Step 1: Partition Power System and PMU Placement

A power system is decomposed into k non-overlapping subsystems of approximately the same size using a graph partition algorithm. PMUs will be installed at selected boundary buses to make sure the voltage phasors at all boundary buses are known. In other words, PMUs are placed to make all boundary buses observable [12]. Given a particular bus, installing a PMU obviously makes that bus observable. In addition, all adjacent buses to that bus become observable since their voltage phasors can be calculated from the transmission line current measured by the PMU and transmission line parameters. The procedure of power system decomposition and PMU placement will be described in Section IV.

Let S denote the set of buses in a power system. Suppose this power system is decomposed into k subsystems. We denote by S_i the set of buses in subsystem $i = 1, \ldots, k$, where $S_i \cap S_j = \emptyset$ for $i \neq j$ and $\cup_{i=1}^{k} S_i = S$. Thus, each bus in the subsystem i belongs to one of the following types:

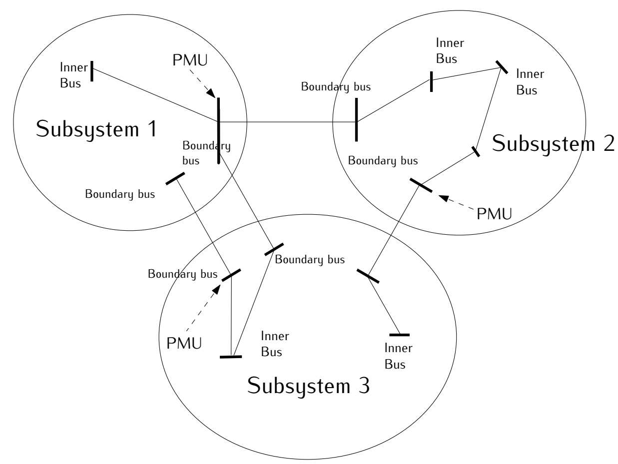

Figure 1. Power system decomposition and bus classification for k =3

- Inner Bus: All its neighboring buses also belong to subsystem i.
- Boundary Bus: At least one of its neighboring buses belongs to a different subsystem.

This is illustrated in Figure 1.

B. Step 2: Select Reference Buses

In order to perform power flow analysis for each subsystem, we need to choose a reference bus for each subsystem. Since the PMUs are placed in such a way that all boundary buses are observable, the voltage magnitude and phase angle for each boundary bus is known. This means that any boundary bus can serve as a reference bus for the subsystem it belongs to. We experimented with performing power flow calculation using different boundary buses as reference buses, and found that the computation time was not affected by the different selection of reference buses.

C. Step 3: Update Power at Boundary Buses

Before subsystem i can be solved independently of the others, the real and reactive power at each boundary bus needs to be updated to account for power flows from neighboring subsystems.

Copyright (c) IARIA, 2015. ISBN: 978-1-61208-406-0

In particular, consider a boundary bus b that belongs to subsystem i. Let P_b and Q_b be the real and reactive power, respectively, at the boundary bus b. For any bus c connected to b, let $P_{b,c}$ and $Q_{b,c}$ be the real and reactive line power flow, respectively, on the line from bus b to bus c. Also, let $A(b)$ denote the set of boundary buses connected to bus b and not belonging to subsystem i. For any boundary bus b_1 connected to b the line power flows P_{b,b_1} and Q_{b,b_1} can be calculated from PMU measurements. The power flowing between bus b and its neighbors in subsystem i can be adjusted as follows:

$$\begin{aligned} P_b^{new} &= P_b - \sum_{c \in A(b)} P_{b,c}, \\ Q_b^{new} &= Q_b - \sum_{c \in A(b)} Q_{b,c}. \end{aligned}$$

D. Step 4: Calculation for Subsystems and Aggregate the Results

Power flow calculations are performed for each subsystem with respect to its reference bus. The voltage phasors of each reference bus are determined by PMU measurement. After each subsystem is solved in parallel, the solution to the entire system is then obtained by aggregating the solutions for each subsystem.

It is possible that some of the subsystems consist of multiple connected components. In this case the number of connected component m is greater than k. In our computational experiments, this increase of components does not significantly increase the computational time.

IV. PMU PLACEMENT ALGORITHM

The proposed real-time power flow calculation method decomposes a large power system into several subsystems, which are then solved in parallel. The implementation cost associated with such a decomposition depends on the number of PMUs installed at boundary buses. Having roughly the same number of buses per subsystem is desirable from the standpoint of balancing computational load. Thus our objective is to minimize the number of PMUs installed at boundary buses subject to the condition that each subsystem has approximately the same number of buses.

The basic structure of the PMU placement algorithm contains two steps. The first step is to divide the power system into k parts using a graph partitioning algorithm. This algorithm attempts to minimize the number of lines whose incident buses belong to different subsystems, while keeping the number of buses per subsystem approximately equal. The second step is to place PMUs based on power system decomposition. Given the partition k, the PMU locations are obtained by solving an integer linear programming problem.

A. Step 1: Power System Decomposition

Viewing the system as a graph $G = (V, E)$, where the vertex set V is the set of buses and the edge set E is the set of transmission lines, we obtain a k-way partitioning problem. The k-way partitioning problem divides a graph into k subgraphs with roughly the same number of vertices such that the edge cut, i.e., the number of edges connecting different subgraphs, is minimized. This problem is NP-hard, and several heuristics for its solution have been developed; see [13]. In this paper, we implement two heuristics for graph partitioning. One is a spectral partitioning algorithm, another one is a multilevel k-way partitioning algorithm.

Spectral partitioning: The spectral partitioning algorithm uses the eigenvectors of the adjacency matrix of a graph to find partitions. In this paper, we are using a spectral factorization based algorithm [14]. This spectral partitioning algorithm consists of four steps.

1) Form the Adjacency Matrix: For a power system with n buses, let $A = \{a_{i,j}\}$ be the corresponding $n \times n$ graph adjacency matrix, where $a_{ij} = 1$, when bus i and bus j are connected by a transmission line, and $a_{ij} = 0$ otherwise.
2) Adjacency Matrix Normalization: Normalize the non-negative symmetric matrix A to obtain a doubly stochastic matrix A', i.e., $A' = \{a'_{ij}\}$ satisfies $\sum_i a'_{ij} = \sum_j a'_{ij} = 1$ for each j and i.
3) Compute Eigenvectors: Compute the k largest eigenvectors u_i, $i = 1, 2, ..., k$, of matrix A'. It is convenient to define the $n \times k$ matrix $U := [u_1, u_2, ..., u_k]$.
4) Clustering: Obtain a partition of the network into k subsystems by clustering the n rows of U into k clusters using the k-means algorithm [15].

Multilevel k-way partitioning: We used an implementation of the multilevel k-way partitioning algorithm from METIS [16]. The algorithm consists of three major steps:

1) Graph coarsening: Given the original graph $G_0 = (V_0, E_0)$, if weight information is not provided, it assumed that the vertex weights and edge weights are all equal to 1. Let $|V|$ be the number of vertices in V. A series of successively smaller graphs $G_i = (V_i, E_i)$, $i = 1...m$, is derived from the input graph such that $|V_{i-1}| > |V_i|$. Each successive graph G_i is constructed from the previous graph G_{i-1} by collapsing together a set of pairs of adjacent vertices; these sets can be obtained from finding a maximal matching. If two vertices are merged, the weights need to be updated in order to preserve the structure of the previous graph. The weight of the new vertex is set equal to the sum of the weights of its constituent vertices. If two merged vertices are adjacent to the same neighbor, then the two edges will be replaced by a new edge whose weight is the sum of weights of the edges it replaced. The graph coarsening step ends when the coarsest graph G_m is smaller than a given threshold.
2) Initial partitioning: A k-way partition of the coarsest graph G_m is computed such that each sub-graph contains roughly same vertex weight. This is done using a relatively simple approach such as the multilevel bisection algorithm.
3) Uncoarsening and refinement: The partition of the smallest graph G_m is projected back to G_0 through a successively larger graphs $G_{m-1}, G_{m-2}, ..., G_1$. This projection step reverses the process in step 1 to obtain the uncoarsen graph. After each projection step, the partition is refined using an algorithm based on the Kernighan-Lin method [17] that iteratively moves vertices between sub-graphs as long as such moves

improve the quality of the partition.

B. Step 2: PMU placement based on decomposition

Once a partition of the system is obtained, PMUs need to be installed to provide measurements for boundary buses. In most cases, the number of PMUs needed is not necessarily equal to the number of branches connecting different subsystems. Based on the approach in [8], the optimal PMU placement problem can be formulated as an Integer Linear Programming (ILP) problem.

Given a partition of the system with n boundary buses, we can define an n by n constraint matrix M for the boundary buses. The entries of M are defined as follows:

$$M_{i,j} = \begin{cases} 1, & \text{if } i = j \\ 1, & \text{if boundary buses } i \text{ and } j \\ & \text{are connected} \\ 0, & \text{otherwise} \end{cases}$$

The PMU placement problem can be formulated as follows:

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{n} c(i) x_i \\ \text{subject to} \quad & MX \geq \hat{1} \\ & x_i \in \{0, 1\}, \; i = 1, \ldots n, \end{aligned} \tag{5}$$

where n is the number of boundary buses, $c(i)$ is the cost of placing a PMU at boundary bus i, $\hat{1}$ is a vector of ones, and X is a binary decision vector whose entries are:

$$x_i = \begin{cases} 1, & \text{if a PMU should be installed at boundary bus } i \\ 0, & \text{otherwise.} \end{cases}$$

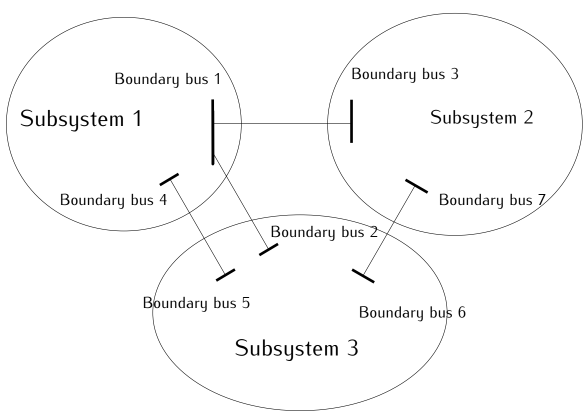

Figure 2. Example for optimal PMU placement

For example, consider the power system shown in Figure 2 that has been decomposed into three subsystems. There are 4 transmission lines between 7 boundary buses. The optimal PMU placement problem can be solved as follows: First, initialize the constraint matrix M for the boundary buses. Building the M matrix for Figure 2 yields:

$$M = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Then the inequalities in (5) takes the following form:

$$\begin{cases} x_1 + x_2 + x_3 & \geq 1 \\ x_1 + x_2 & \geq 1 \\ x_1 + x_3 & \geq 1 \\ x_4 + x_5 & \geq 1 \\ x_4 + x_5 & \geq 1 \\ x_6 + x_7 & \geq 1 \\ x_6 + x_7 & \geq 1 \end{cases}$$

The first constraint means that a PMU should be placed either on bus 1, bus 2 or bus 3 to make bus 1 observable. Similarly, the second constraint implies that a PMU should be installed at either bus 1 or 2 to make bus 2 observable. After solving the ILP problem for the boundary buses, the placement of PMUs for the entire system is obtained.

V. Numerical Results

We use the IEEE 118 and 300 bus systems from [18] to illustrate the performance of our parallel algorithms. The 118 bus system was decomposed into 2, 4, and 8 subsystems, while the 300 bus system was decomposed into 2, 4, 8, and 16 subsystems. The power system was partitioned by both the spectral algorithm [14] and the multilevel k-way method [16] described in Section IV. The PMU measurements at the boundary buses were emulated using solutions obtained by traditional serial power flow methods. After decomposition, each of the resulting sub-networks was solved using Newton's method [19]. The convergence criterion was set to 10^{-5}, and the maximum iteration number for Newton's method was set to 10. The system information is shown in Table I. The partition results and computation times for different numbers of sub-networks are summarized in Tables II and III.

TABLE I. Network Information for two power system

System	118 bus	300 bus
number of nodes	118	300
number of branches	186	411
number of generator buses	53	68

TABLE II. Test Results on IEEE 118 bus system

Method	Partition number	Max subsystem size	Edge-Cut size [a]	PMU number	Calculation time (sec.)
Spectral Method	1	118	0	0	0.183
	2	79	5	3	0.090
	4	38	15	10	0.029
	8	22	26	13	0.016
Multilevel k-way	1	118	0	0	0.183
	2	78	5	3	0.089
	4	33	14	10	0.026
	8	19	29	14	0.014

[a] Edge-cut size: the number of branches connecting different subsystems

The results obtained using our methods were compared with the corresponding results obtained using serial Newton's method. The maximum deviation of node voltage in our method compared to Newton's method was less than 10^{-4} p.u., which illustrates that our method is accurate and feasible.

TABLE III. TEST RESULTS ON IEEE 300 BUS SYSTEM

Method	Partition number	Max subsystem size	Edge-Cut size [a]	PMU number	Calculation time (sec.)
Spectral Method	1	300	0	0	1.440
	2	184	6	5	0.570
	4	98	11	7	0.163
	8	56	19	13	0.075
	16	30	39	26	0.029
Multilevel k-way	1	300	0	0	1.440
	2	173	6	6	0.530
	4	97	11	10	0.162
	8	51	24	20	0.058
	16	25	49	32	0.025

[a] Edge-cut size: the number of branches connecting different subsubsystems

A. Speedups compared to serial method

The computation times for the power flow calculation are presented in Tables II and III. In particular, the computation time T_s corresponding to partition number 1, the case where no partitioning is done, was obtained by the serial Newton's method. The rate of speedup can be obtained by the formula $S = T_s/T_p$, where T_p is the computation time of parallel method. Figures 3 and 4 show these speedups. Comparing the the details of these timing studies, we present the following conclusions:

1. High speedups and parallel efficiency are achieved in both the 118 and 300 bus systems by spectral and multilevel k-way method. Using spectral method as example, when the system is divided into two subsystems, a speedup by about a factor of 2 was obtained for both systems. Installing more PMUs usually leads to faster power flow calculations. In the 300-bus system, the traditional serial method takes 1.440 seconds, while with 5 PMUs the calculation time is reduced to 0.570 seconds; here the speedup rate is 2.53. The speedup rate increases to 49.66 by using 26 PMUs. These speedups suggest that the parallel algorithm proposed in this paper is efficient.

2. The speedup associated with the parallel method increased as the size of the power system increased. For the case of splitting system into two subsystems, by using spectral method, the calculation time for the 118 bus system was sped up by a factor of 2.03, while for the 300 bus system it was sped up by a factor of 2.53. For the case of splitting system into four subsystems, the calculation time for the 118 bus system was 6.31 times faster, and for the 300 bus system it was 8.83 times faster. This suggests that the algorithms presented in this paper may perform well on larger power systems.

B. Performance of Spectral Partitioning vs. Multilevel k-way Partitioning

Different partition methods may result in different PMU placements. In particular, the practicality of the partitioning algorithm becomes especially important when real systems are considered. For large power systems, the optimal PMU placement problem is hard to solve exactly. A common practice is to compare the performance of different methods. Here we compared the spectral partitioning algorithm with the k-way partitioning method. As Figures 3 and 4 show, the multilevel k-way method outperforms the spectral partitioning algorithm in terms of calculation time. However, the multilevel k-way method usually requires more PMUs.

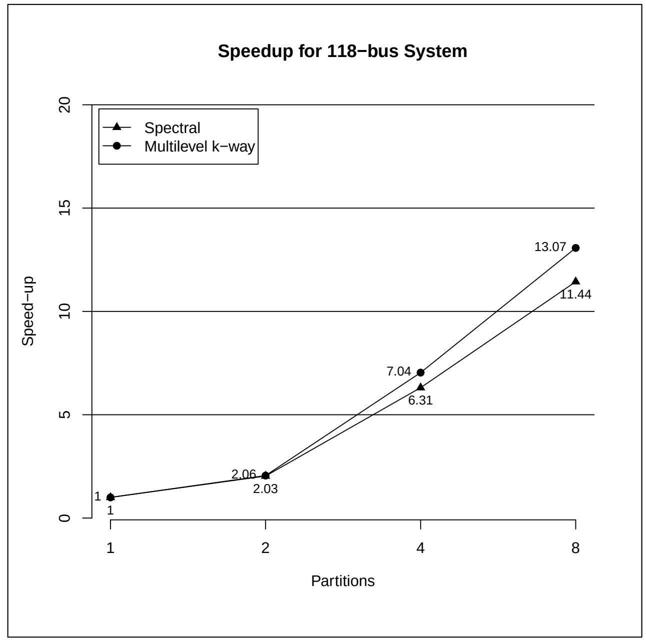

Figure 3. Speedup for 118 bus system

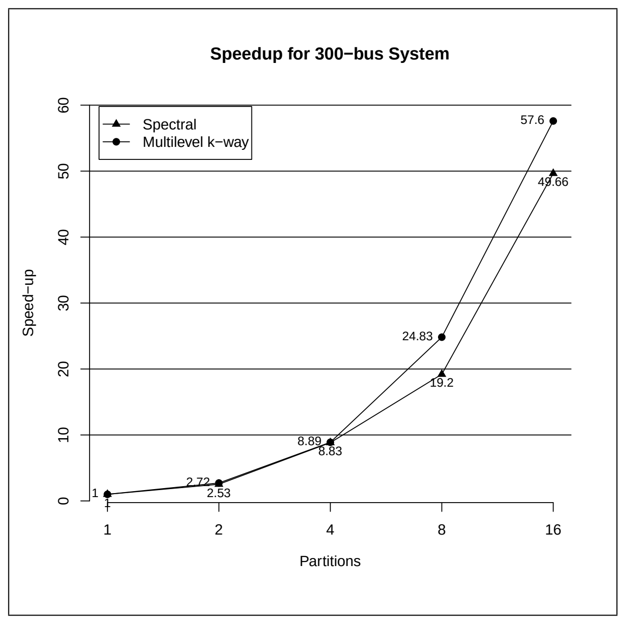

Figure 4. Speedup for 300 bus system

TABLE IV. COMPARISON OF THE SPECTRAL METHOD AND MULTILEVEL k-WAY METHOD FOR IEEE 300 BUS SYSTEM

Partition Number	Spectral method		Multilevel k-way method	
	Max Subsystem Size	PMU Number	Max Subsystem Size	PMU Number
2	184	5	173	6
4	98	7	97	10
8	56	13	51	20
16	30	26	25	32

Copyright (c) IARIA, 2015. ISBN: 978-1-61208-406-0

The partition results obtained via the spectral method and the multilevel k-way method for the 300 bus system are listed in Table IV. As shown in Table IV, with the same partition number, the max subsystem size obtained using the multilevel k-way method is smaller than the one obtained using the spectral algorithm. It indicates that the multilevel k-way method allocates the buses to subsystems in a more balanced way than the spectral method. More balanced subsystems may result in faster calculation time; see Figures 3 and 4. On the other hand, these allocations require more PMUs. There is a tradeoff between having balanced subsystems and using a small number of PMUs, considering the high cost of PMU implementation and maintenance in the long run, the spectral method may be more appropriate for our decomposition scheme.

VI. Conclusion

In this paper, we have described a new approach to power flow analysis. Our problem formulation explicitly takes into account the placement of PMUs. A graph partition approach was proposed to partition the power system and determine the placement of PMUs, after which the power flow problem can be solved in parallel. The effectiveness of the approach was illustrated on a 118 and 300 bus system using two different graph partitioning methods. In each of these cases, significant speedups compared to the serial Newton's method were obtained. The spectral algorithm requires fewer PMUs while keeping the approximately the same number of buses for each subsystem, thus it is more suitable for our decomposition approach.

Acknowledgment

This study was supported by NYSERDA (New York State Energy Research and Development Authority) contract 28814.

References

[1] S. Lin and J. Van Ness, "Parallel solution of sparse algerbraic equations," IEEE Transactions on Power Systems, vol. 9, no. 2, 1994, pp. 743–749.

[2] Y. Wang and H. Gooi, "New ordering methods for sparse matrix inversion via diagonalization," IEEE Transactions on Power Systems, vol. 12, no. 3, 1997, pp. 1298–1305.

[3] J. Wu and A. Bose, "Parallel solution of large sparse matrix equations and parallel power flow," IEEE Transactions on Power Systems, vol. 10, 1995, pp. 1343–1349.

[4] M. Rafian, M. Sterling, and M. Irving, "Decomposed load-flow algorithm suitable for parallel processor implementation," IEE Proceedings, vol. 132, no. 6, 1985, pp. 281–284.

[5] K. Chan, R. Dai, and C. Cheung, "A Coarse Grain Parallel Solution Method for Solving Large Set of Power Systems Network Equations ," Proc. Int. Conf. Power System Technology, vol. 4, 2002, pp. 2640–2644.

[6] M.Amano, A. Zecevic, and D. Siljak, "An improved block-parallel newton method via epsilon decompositions for load-flow calculations," IEEE Transactions on Power Systems, vol. 11, no. 03, 1996, pp. 1519–1527.

[7] M.Esmaili, K.Gharani, and H. Shayanfar, "Redundant observability PMU placement in the presence of flow measurements considering contingencies," IEEE Transactions on Power Systems, vol. 28, no. 4, 2013, pp. 3765–3773.

[8] B.Xu and A.Abur, "Observability Analysis and Measurement Placement for Systems with PMUs," in Power Systems Conference and Exposition, 2004. IEEE PES, vol. 2, 2004, pp. 943–946.

[9] F.Aminifar, A.Khodaei, M.Fotuhi-Firuzabad, and M.Shahidehpour, "Contingency-constrained PMU placement in power networks," IEEE Transactions on Power Systems, vol. 25, no. 1, 2010, pp. 516–523.

[10] B.Gou, "Generalized Integer Linear Programming Formulation for Optimal PMU Placement," IEEE Transactions on Power Systems, vol. 23, no. 03, 2008, pp. 1099–1104.

[11] S.Chakrabarti and E.Kyriakides, "Optimal placement of phasor measurement units for power system observability ," IEEE Transactions on Power Systems, vol. 23, no. 3, 2008, pp. 1433–1440.

[12] R. Kavasseri and S. K. Srinivasan, "Joint Placement of Phasor and Power Flow Measurements for Observability of Power Systems," IEEE Transactions on Power Systems, vol. 26, no. 4, 2011, pp. 1929–1936.

[13] P.-O. Fjllstrm, "Algorithms for Graph Partitioning: A Survey," Linkping Electronic Articles in Computer and Information Science, 1998.

[14] J. Hespanha, "An efficient MATLAB Algorithm for graph partitioning," Technical Report, University of California, Santa Barbara, CA, 2004, [retrieved: 03, 2015]. [Online]. Available: http://www.ece.ucsb.edu/~hespanha/techreps.html

[15] J. MacQueen, "Some methods for classification and analysis of multivariate observations," Proc. Fifth Berkeley Symp. on Math. Statist. and Prob., vol. 1, 1967, pp. 281–297.

[16] G. Karypis and V. Kumar, "A fast and highly quality multilevel scheme for partitioning irregular graphs," SIAM Journal on Scientific Computing, vol. 20, no. 1, 1999, pp. 359–392.

[17] B. W. Kernighan and S. Lin, "An Efficient Heuristic Procedure for Partitioning Graphs," Bell System Technical Journal, vol. 49, 1970, pp. 291–307.

[18] Power Systems Test Case Archive, [retrieved: 03, 2015]. [Online]. Available: https://www.ee.washington.edu/research/pstca

[19] H. Saadat, Power System Analysis. New York: McGraw-Hill, 1999.