Partition-Based Bus Renumbering Effect on Interior Point-Based OPF Solution

Mahdi Mehrtash, Amin Kargarian, Ali Mohammadi
Louisiana State University
Department of Electrical and Computer Engineering
Baton Rouge, United States
mmehrt3@lsu.edu

***Abstract*—The form of constraint coefficient matrix of an optimization problem significantly affects the solution procedure for finding the optimal results, especially when iterative algorithms are implemented. In power systems, the order of bus numbers affects the power system's graph adjacency matrix and accordingly affects the optimal power flow (OPF) problem's constraint coefficient matrix. Changing this constraint coefficient matrix might change the OPF solution time. In this paper, we show that the order of bus numbers affects the solution time of AC and DC OPF problems when an interior point method-based solver is used. We propose a partition-based bus renumbering algorithm to be implemented before solving the OPF problem. This algorithm constructs a well-patterned constraint coefficient matrix and speeds up the OPF solution procedure. Numerical results on the IEEE 118-bus system and the 13659-bus European transmission system show effectiveness of the proposed algorithm in reducing the OPF's solution time. Implementation of the proposed method leads to about 65% of timesaving when Matpower is used to solve OPF.**

***Index Terms*— Interior point method, optimal power flow, system partitioning, bus numbering.**

I. Introduction

A. Motivation

TO MATHEMATICALLY formulate and solve power system problems, each element of the system (e.g., buses, generating units, and lines) needs to be tagged with a specific name/number. Although a given power system has a specific topology, numerous options exist to name elements of the system. For instance, a bus in a specific location may be named either bus i or bus j. One can say that a power system is a graph in which bus numbers (i.e., order of vertex numbers) are not necessary assigned based on the graph topology. That is, if the graph is partitioned into several subgraphs, neighboring buses that belong to the same subgraph are not necessary numbered consecutively. For instance, in the IEEE 118-bus system, buses 17 and 113 are next to each other, while bus 112 is far from these two buses. The bus numbering does not change the solution of an optimization problem; however, we have found that it considerably changes the solution time. Random bus numbering leads to a scattered admittance matrix. Consequently, the system adjacency matrix and constraint coefficient matrix are not organized.

B. Literature Review

The optimal power flow (OPF) problem aims to find an optimal operating point of a power system in which generation cost and/or transmission loss is minimized subject to specific constraints on power and voltage variables [1]. The OPF problem is highly nonconvex and NP-hard in the worst case [2]. Formulation of the complete OPF model (AC-OPF), and the simplified version (DC-OPF) can be found in [3]. The OPF is an online operational problem. Meaning, it must solve online to keep the optimality of the operating point (OPF is usually solved every 5 minutes) [4]. Therefore, improving the solution time of the OPF problem is valuable for large/medium-scale power systems.

Various approaches have been presented in the literature to solve OPF. One of the most popular approaches, which has been widely used to solve OPF (and many other optimization problems), is the interior point method (IPM). Most of the open-source and commercial solvers, such as IBM ILOG CPLEX, MOSEK, LINDO, Xpress Optimizer, BPMPD, MIPS, MATLAB quadprog function, MATLAB fmincon function, etc., utilize the concept of IPM to solve optimization problems, such as OPF. Reference [5] discusses the importance of sparsity pattern and structure of a constraint matrix on exploiting the optimal solution of an optimization problem in the IPM-based solvers. It is proved that the pattern of the constraint matrix, which relates to the topology of the network graph, has a considerable impact on the direction of IPM-based solvers for exploiting the optimal solution in both linear and nonlinear programming problems [6, 7]. Therefore, initial graph partitioning and reordering the node numbers to construct a

This work is supported by the Louisiana Board of Regents under grant LEQSF (2016-19)-RD-A-10.

978-1-5386-1006-0/18/$31.00 ©2018 IEEE

well-patterned constraint matrix potentially speeds up the solution procedure of a solver, when applied to the OPF problem.

Traditionally, some efforts have been done for equations/variables reordering of the AC-OPF problem [8, 9]. Since the most time-consuming task in any iteration of IPM is the factorization of Hessian matrix that corresponds to the linearized KKT equations, some works have been done for the efficient solution of the linear system of equations to ensure the robustness of the factorization process and its speed-up [10, 11]. However, the idea of reordering bus numbers of a power system before constructing the mathematical model of the optimization problem to improving the IPM-based OPF solution time has not been considered in previous works.

C. Contribution

In this paper, we demonstrate that the order of the bus numbers changes the OPF problem's constraint coefficient matrix and considerably affects the solution time. We propose a partition-based bus renumbering (PBBR) algorithm to partition the power grid graph into several sub-graphs. We rename the bus numbers in a manner to have consecutive bus numbers in the same sub-graph. This algorithm is implemented before formulating and solving the OPF problem. Indeed the proposed algorithm is an offline procedure that helps us to construct a well-patterned constraint matrix. We do not change any physical constraints/objective of the OPF problem and any module of the interior point method. The proposed algorithm significantly reduces the solution time of DC-OPF and AC-OPF, when an IPM-based solver is used. Simulation results on the IEEE 118-bus system and the European 13659-bus system show effectiveness of the proposed bus renumbering algorithm. More than 60% of timesaving is achieved with the implementation of PBBR.

Note that changing the numbering order of buses only changes labels of the OPF inputs, not the results. However, the constraint coefficient matrix and the pattern of the feasibility region (i.e., feasible design space of the optimization problem) and hence exploiting path toward the optimal solution will change when the labels of the input parameters change (see reference [5-7]).

D. Paper Organization

The remainder of the paper is organized as follows. The proposed PBBR algorithm is presented in Section II. The numerical results are discussed in Section III. Concluding remarks are provided in Section IV.

II. Partition-Based Bus Renumbering Algorithm

Consider the 13659-bus PEGASE system [12]. Figure 1(a) shows the location of nonzero elements of the system's admittance matrix (Y-bus). Since the bus numbers are not organized, the admittance matrix has a random pattern. Such a random pattern will appear in Jacobian and the constraint coefficient matrices of the 13659-bus PEGASE system. Taking the inverse of the Jacobian matrix is time-consuming not only because of the size of the system but also because of the scattered pattern of this matrix. Intuitively, the same thing might happen for a solver when applying to solve OPF of the PEGASE system with such a scarred constraint coefficient matrix. We present an algorithm, which is based on a graph partitioning technique, to rearrange the bus numbers and, consequently, the admittance, Jacobian, and the constraint coefficient matrices.

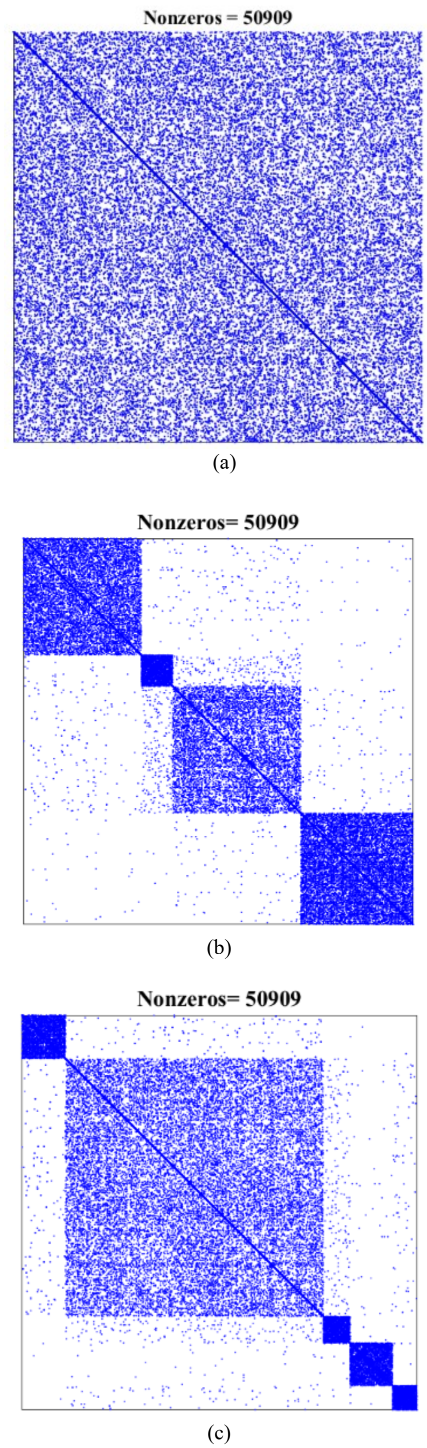


Fig. 1. Location of nonzero elements of 13659-bus PEGASE system's admittance matrix for (a) original bus numbering, (b) PBBR technique with 4 partitions, and (c) PBBR technique with 5 partitions.

In order to arrange a well-patterned admittance matrix, first, system partitions should be determined. Different partitioning methods have been proposed in the literature. K-means and spectral clustering are two popular partitioning method. We apply the spectral clustering method, which is proven to be advantageous for graph partitioning [13]. Reference [14] proposed a spectral clustering technique for large-scale graphs. A Matlab function (*grPartition*) is provided in [15] for efficient and fast partitioning of very large graphs using the spectral factorization method presented in reference [14]. The desired number of partitions and edge-weights matrix are inputs of *grPartition,* and as outputs, it returns the cost of partitioning and a vector with the cluster index for every node. By using this method, the graph might be decomposed from the nodes and/or edges. That is, the power grid can be decomposed from the perspective of buses and/or transmission lines. Partitioning the grid from the perspective of a bus leads to introducing two new buses to replace the original bus. Although it might be useful for few power system problems [16], in this paper, we only aim at reordering the bus numbers to reorganize the system Y-bus and constraint coefficient matrices. Thus, we need to avoid introducing new elements and rather focus on renumbering the existing elements. Hence, to prevent node slicing during the partitioning process, we set an arbitrary large cost (i.e., weight) to each bus.

To determine the partitioning cost from the transmission lines' perspective, dependency and closeness of the buses to each other need to be determined. Before partitioning, AC power flow is solved to obtain active and reactive power flow in each transmission line. We use the concept of apparent power to characterize dependency of the partitions. Indeed, if the apparent power exchanged between two neighboring partitions via the tie-lines is large, it indicates that the partitions are heavily dependent. It is not desirable to separate the buses which are strongly dependent and put them in different partitions. Such buses would be better to be in the same zone. Hence, we assign the apparent power of each line as the weight of the corresponding edge of the system graph. The partitioning cost from the perspective of transmission lines is the summation of apparent power flowing in the crossed branches (i.e., the branches that interconnect the partitions). Thus, we introduce index ϖ to determine dependency between the partitions. Where S_{lc} the apparent power of is crossed lines and S_l is the total apparent power of all transmission lines. This index indicates the quality of the partitioning process.

$$\varpi = \sum S_{lc} \times S_l^{-1} \quad (1)$$

$$lc \in \{\text{crossed lines}\}, \qquad l \in \{\text{all lines of the system}\}$$

We now form an edge-weights matrix $\boldsymbol{C}$ as follows, which is an input for *grPartition*.

$$\boldsymbol{C} = \begin{bmatrix} M_{b1} & S_{lb1-b2} & S_{lb1-b3} & \cdots & S_{lb1-bn} \\ S_{lb2-b1} & M_{b2} & S_{lb2-b3} & \cdots & S_{lb2-bn} \\ S_{lb3-b1} & S_{lb3-b2} & M_{b3} & \cdots & S_{lb3-bn} \\ & & \vdots & & \\ S_{lbn-b1} & S_{lbn-b2} & S_{lbn-b3} & \cdots & M_{bn} \end{bmatrix} \quad (2)$$

The total cost of partitioning is the summation of costs of bus slicing and branch cutting.

$$\Lambda = \sum_{ij} \varpi + \sum_{b} \mathrm{M}_b$$

where M_b is a big number that indicates the cost of slicing bus b. The above process decomposes the power grid into multiple partitions and determine that each bus b belongs to which partition. However, the system Y-bus and constraint coefficient matrices are not still well-parented. For instance, for the 13659-bus system, the Y-bus matrix is still similar to Fig. 1 (a). To create a well-pattern matrix, we propose to renumber the buses. The buses that belong to a partition are renumbered consecutively. For instance, assume that a system is separated into two partitions and $\{b_1, b_2, b_5\} \in \text{Partition}_1$ and $\{b_3, b_4, b_6\} \in \text{Partition}_2$. The renumber process reorganized the bus number as $\{b_1', b_2', b_3'\} \in \text{Partition}_1$ and $\{b_4', b_5', b_6'\} \in \text{Partition}_2$. The partitioning and renumbering process modifies the disorganized Y-matrix of the 13659-bus system shown in Fig. 1 (a) and creates the well-patterned matrices shown in Figs. 1 (b) and (c).

The steps of the proposed PBBR algorithm is summarized as follows:

Algorithm 1. Steps of the proposed PBBR technique	
Step 1.	**Initialization:** Read system data and set *k* to the desired number of partitions
Step 2.	**Run Power Flow:** Solve AC power flow equations
Step 3.	**Form *C* Matrix:** Create an edge-weights matrix ***C*** in which off-diagonal elements are apparent power of corresponding edges (lines), and diagonal elements are equal to a big enough value (to avoid slicing a bus in partitioning procedure)
Step 4.	**Run Spectral Clustering Partitioner:** Partition the graph using *grPartition* function in which *k* and ***C*** are inputs, and the cost of partitioning and the partitioned graph are outputs
Step 5.	**Form *NR*:** Create a vector, called ***NR***, based on partitioned graph, which includes new bus numberings (buses belong to the same partition have consecutive numbers in ***NR***)
Step 6.	**Update System Data:** Renumber buses according to ***NR***, and reorganize sending/receiving terminals of lines and location of generating units and loads based on the new numbering
Step 7.	**Run OPF**
Step 8.	**Reverting System Data:** Use ***NR*** and revert bus numbers to the original bus numbering

The proposed technique convert disorganized admittance and constraint coefficient matrices of a power system into well-patterned matrices. Thus, the IPM-based solvers solve the OPF problem much faster compared with the case that the matrices are not partitioned and renumbered. Note that the proposed PBBR technique is an offline process in which AC power flow is solved for a sample power demand. Hence, obtaining the ***NR*** vector (see Step 5 of Algorithm 1) is an offline procedure. However, OPF is an online problem, which is solved every 5 minutes as recommended by standards [4]. Therefore, performing the PBBR technique (an offline procedure) considerably reduces the solution time of the OPF problem (an online problem) in each interval. Moreover, it will be illustrated in the numerical results Section that decreasing or increasing the system demand has a negligible effect on the time-saving benefit that an operator gains by application of the proposed PBBR technique.

III. Numerical Results

The IEEE 118-bus system and the European 13659-bus transmission system are used to evaluate the effectiveness of the PBBR algorithm in reducing the computational burden of the DC- and AC-OPF problems. All simulations are carried out on a personal computer with an Intel(R) Xeon(R) CPU @2.6 GHz and 16 GB of RAM. To provide a fair comparison, MATPOWER 6.0 is used for all experiments in which Mathpower interior point solver (MIPS) is assigned for both DC-OPF and AC-OPF [17, 18].

A. 118-Bus System

The system has 118 buses, 54 generating units, 186 lines, and 91 load points. We study AC-OPF with ten scenarios. In the first scenario, AC-OPF is solved without graph partitioning and bus renumbering (i.e., the number of partitions $k = 1$). In scenarios two to ten, the system is partitioned respectively into two to ten subgraphs (i.e., $k = 2, \dots, 10$) and the bus renumbering algorithm is applied. The value of the objective function for all ten scenarios is \$129.66K. This shows that the proposed PBBR algorithm has no impact on the optimal solution of the optimization problem.

We analyze three different load levels to demonstrate the effectiveness of PBBR for various loading conditions. The base case uses the standard load values for the IEEE 118-bus system. We multiply the load on each bus to 0.8 and 1.2 to create two new load levels. The solver time with respect to k is shown in Fig. 2. The solver time significantly decreases after bus renumbering. For instance, the solver time is 0.672 (Sec.) for the base case loading condition without bus renumbering (i.e., $k = 1$), while the time goes down to 0.219 (Sec.) if $k = 2$. That is, 67.4% of time reduction. Increasing number of partitions from three to ten does not lead to a considerable change in the solver time. Not that we observed this behavior for many test cases; however, we do not generalize this for all systems.

B. 13659-Bus System

This is a large-scale realistic power system with 13659 buses, 4092 generating units and 20467 lines. We study the DC-OPF problem. We use shift factor values to formulate the power flow constraints. The PBBR algorithm is applied for ten scenarios as $k = 1, 2, \dots, 10$. The same values of cost function and decision variables are obtained for all ten scenarios. The solver time versus k is given in Fig. 3 for three loading conditions. The solver time reduces by 64.3% when the system is partitioned into two subgraphs and bus renumbering is applied compare with the scenario without graph partitioning. Note that the solver time for the scenario with $k = 5$ is more than that for $k = 4$. This is because of the differences in system's connectivity graph and sparsity matrix. Figures 1(b) and 1(c) respectively show the sparsity matrices for $k = 4$ and $k = 5$. Partitioning the system into four subgraphs leads to a more well-patterned sparsity matrix compared with that for the scenario with five subgraphs. It illustrates the influence of partitioning quality in the OPF solution time of the proposed PBBR method. That is, a higher quality in partitioning step leads to more saving in the OPF solution time step.

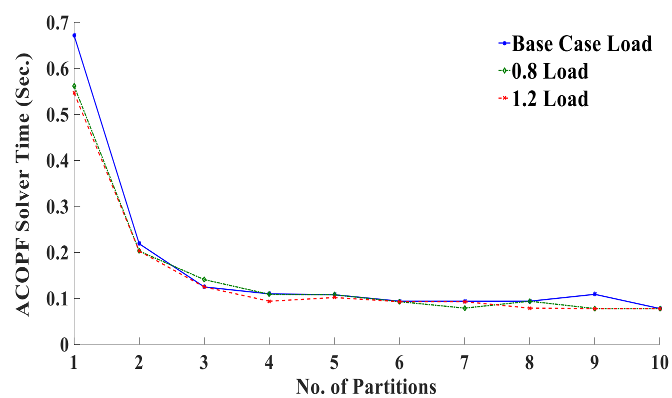


Fig. 2. Solution time of the AC-OPF solver for the 118-bus system with respect to the number of partitions (k).

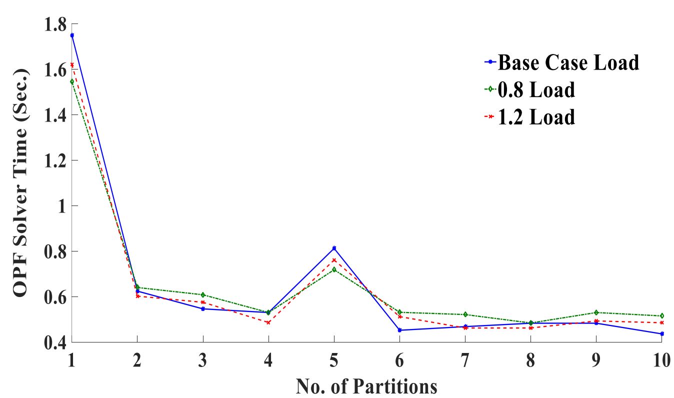
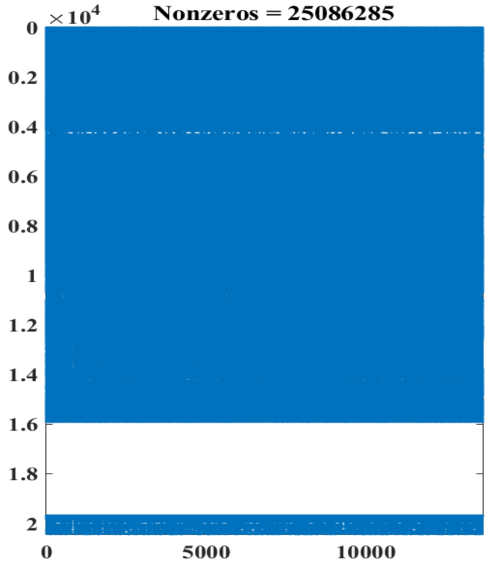
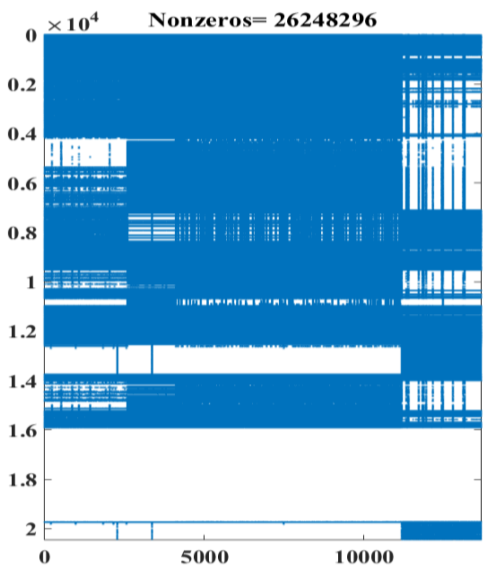


Fig. 3. Solution time of the DC-OPF solver for the 13659-bus system with respect to the number of partitions (k).

We analyze the shift factor (SF) matrix of the 13659-bus system. The SF matrix is used to model the network security constraints in the DC-OPF problem. Thus, the SF matrix constructs a large part of the constraint coefficient matrix of the optimization problem [19]. Locations and values of the SF matrix elements depend on the location of the reference bus and system topology. Therefore, changing bus numbering leads to changing in locations and values of the SF matrix elements. The effect of the PBBR algorithm on the SF matrix of the



(a)



(b)

Fig. 4. Locations of dominant elements (larger than $1e^{-3}$) of the shift factor matrix of the 13659-bus system for (a) the original bus numbering (i.e., $k = 1$) and (b) with application of PBBR with four partitions ($k = 4$).

13659-bus system is depicted in Fig. 4. Although the number of dominant elements slightly increases after bus renumbering, the neighboring non-zero elements of each subgraph are gathered together. The sparsity of the constraint matrix is enhanced and leads to constructing a more well-patterned constraint matrix. IPM-based solvers handle the constraint matrix of Figure 4(b) more efficient than the matrix of Figure 4(a). This significantly speeds up the solvers' solution procedure.

IV. Conclusion

The arrangement of bus numbers in power systems affects the OPF problem's constraint coefficient matrix. This consequently affects the optimization solution time when an interior point method-based solver is used. In this paper, a bus renumbering algorithm is proposed to rearrange the number of buses of a power system. The proposed algorithm, which is a fast offline procedure, is based on a graph partitioning technique (i.e., spectral clustering). The numerical results on the IEEE 118-bus systems and the 13659-bus PEGASE system show that implementation of the proposed algorithm prior to solving OPF leads to about 65% of reduction in the solution time compared with the classical OPF without bus renumbering.

Acknowledgment

The authors thank Professor Jacek Gondzio from the University of Edinburgh for his helpful insights.

References

[1] J. A. Momoh, *Electric Power System Applications of Optimization*. CRC press, 2008.

[2] J. Lavaei and S. H. Low, "Zero Duality Gap in Optimal Power Flow Problem," *IEEE Transactions on Power Systems,* vol. 27, no. 1, pp. 92-107, 2012.

[3] J. F. Marley, D. K. Molzahn, and I. A. Hiskens, "Solving Multiperiod OPF Problems Using An AC-QP Algorithm Initialized With an Socp Relaxation," *IEEE Transactions on Power Systems,* vol. 32, no. 5, pp. 3538-3548, 2017.

[4] M. B. Cain, R. P. O'neill, and A. Castillo, "History of Optimal Power Flow and Formulations," *Federal Energy Regulatory Commission,* pp. 1-36, 2012.

[5] J. Gondzio and A. Grothey, "Exploiting Structure in Parallel Implementation of Interior Point Methods for Optimization," *Computational Management Science,* vol. 6, no. 2, p. 135, 2009.

[6] J. Gondzio and A. Grothey, "Solving Non-Linear Portfolio Optimization Problems with the Primal-Dual Interior Point Method," *European Journal of Operational Research,* vol. 181, no. 3, pp. 1019-1029, 2007.

[7] E. D. Anderson, J. Gondzio, C. Mészáros, and X. Xu, "Implementation of Interior-Point Methods for Large Scale Linear Programs," in *Interior Point Methods of Mathematical Programming*: Springer, 1996, pp. 189-252.

[8] D. I. Sun, B. Ashley, B. Brewer, A. Hughes, and W. F. Tinney, "Optimal Power Flow by Newton Approach," *IEEE Transactions on Power Apparatus and systems,* no. 10, pp. 2864-2880, 1984.

[9] H. Wei, H. Sasaki, J. Kubokawa, and R. Yokoyama, "An Interior Point Nonlinear Programming for Optimal Power Flow Problems with a Novel Data Structure," *IEEE Transactions on Power Systems,* vol. 13, no. 3, pp. 870-877, 1998.

[10] V. De Sousa, E. C. Baptista, and G. da Costa, "Analysis of Numeric Conditioning of Hessian Matrix of Lagrangian Via Singular Values," in *Power Tech, 2007 IEEE Lausanne*, 2007, pp. 98-102: IEEE.

[11] F. Capitanescu, M. Glavic, D. Ernst, and L. Wehenkel, "Interior-Point Based Algorithms for The Solution of Optimal Power Flow Problems," *Electric Power systems research,* vol. 77, no. 5, pp. 508-517, 2007.

[12] C. Josz, S. Fliscounakis, J. Maeght, and P. Panciatici, "AC Power Flow Data in MATPOWER and QCQP Format: iTesla, RTE Snapshots, and PEGASE," *arXiv preprint arXiv:1603.01533,* 2016.

[13] A. K. Jain, "Data Clustering: 50 Years Beyond K-Means," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2008, pp. 3-4: Springer.

[14] J. P. Hespanha, "An Efficient Matlab Algorithm for Graph Partitioning," *University of California,* pp. 1-8, 2004.

[15] grpartition [Online]. Available: http://www.ece.ucsb.edu/~hespanha/software/grPartition.html

[16] J. Guo, G. Hug, and O. K. Tonguz, "Intelligent Partitioning in Distributed Optimization of Electric Power Systems," *IEEE Transactions on Smart Grid,* vol. 7, no. 3, pp. 1249-1258, 2016.

[17] R. D. Zimmerman and C. E. Murillo-Sánchez, "Matpower 6.0 User's Manual," ed: December, 2016.

[18] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, "Matpower: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education," *IEEE Transactions on power systems,* vol. 26, no. 1, pp. 12-19, 2011.

[19] A. J. Wood and B. F. Wollenberg, *Power Generation, Operation, and Control*. John Wiley & Sons, 2012.