A Robust Parallel Distributed State Estimation for Large Scale Distribution Systems

Ugur Can Yilmaz, *Student Member, IEEE*, and Ali Abur, *Life Fellow, IEEE*

***Abstract*—The growing need and interest in real-time monitoring of large distribution networks motivated by the rapid population of renewable sources, EVs and etc. demand a computationally efficient state estimation framework. This article presents an improved computational framework for implementing a robust state estimator using a multi-core processor. The main contribution of the article is the proposed computational framework along with two partitioning strategies which enable fast and robust state estimation for large scale radial and/or meshed distribution systems. Formulation of the proposed method and its implementation are described in detail. Performance of the estimator is tested by simulations first using a small 84-bus radial distribution system. Then the method's scalability is demonstrated by simulations on two very large scale distribution networks one configured radially and the other meshed each containing over 12 500 buses.**

***Index Terms*—Clustering, distributed estimation, network partitioning, parallel processing, state estimation.**

I. Introduction

RECENT increase in the penetration of renewable energy sources along with energy storage technologies changes requirements on the configuration, monitoring, control and secure operation of distribution networks. The existing radial and centralized infrastructure of distribution networks with unidirectional power flows towards passive loads is being transformed into the so-called "intelligent grids" taking advantage of innovative measurement devices, distribution system automation, advanced power electronics, and other advanced technologies [1]. One of the significant requirements of the transformation is the real-time monitoring of the system via a robust and computationally efficient distribution system state estimator (DSSE) [2]. There are DSSE techniques adapting transmission system state estimation to distribution network where [3] was the first proposing node voltage based DSSE, [4] yields the branch current based DSSE, [5] utilizes branch power based DSSE, and [6] proposes change of state variables to develop DSSE. DSSE based on evolutionary algorithms are also presented in the literature using artificial neural network [7] or fuzzy logic [8]. These DSSE algorithms should be scalable as the distribution system sizes grow and network models become large and complex.

As the system size and model complexity grow in the past decades, methods exploiting implicit parallelism in implementation of the state estimation solution have been investigated [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23]. The challenge common to all the proposed methods are twofold: 1) strategic partitioning of the system and 2) maintaining robustness against bad data. Partitioning of the system is critical since the largest zone will inadvertently constitute the bottleneck dictating the limit on the overall solution cpu time assuming parallel state estimation (SE) executions for each zone. Hence, partitioning is expected to yield zones as evenly sized as possible. On the other hand, when solving SE for a given zone, certain measurements incident to zone boundaries will have to be discarded weakening the local redundancy and thus bad data processing capability at the zone boundaries. A boundary bus is defined as a bus having at least one neighboring bus belonging to another zone. Buses which are not boundary buses are designated as internal buses.

In [10], a multi-level SE framework is introduced by decoupling the SE solution according to the voltage level of the network. A multi-area SE solution is described in [11], [12]. While the approach works without any PMU measurements, having at least one PMU per area facilitates the coordination level computations. A local SE is executed for each area, then a centralized coordinator estimator is executed to estimate resultant system states and also detect any bad data missed by individual area SE. This work does not provide any specific partitioning strategy but uses the existing utility zones for the system.

In [13], a central coordinator receives the local estimation results of non-overlapping areas or subsystems partitioned based on a geographical basis. Similarly, the authors of [14] use the estimation results of multiple geographically partitioned non-overlapping areas, and the concept of switching communication graphs to ensure all area's local estimates converge to the global least square estimate with probability 1. Based on the alternating direction method of multipliers, a systematic cooperation between local control centers is proposed in [15] where its convergence is guaranteed regardless of local observability or parameter tuning; however, network partitioning is assumed to be given. Some researchers focus on convergence issues in very large scale networks [16], [17]. There may be special cases

Manuscript received 29 January 2023; revised 24 April 2023; accepted 2 July 2023. Date of publication 5 July 2023; date of current version 21 February 2024. This work was supported by the U. S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) through Solar Energy Technologies Office under Award DE-EE0009356. Paper no. TPWRS-00123-2023. *(Corresponding author: Ali Abur.)*

The authors are with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02151 USA (e-mail: yilmaz.u@northeastern.edu; abur@ece.neu.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TPWRS.2023.3292552.

Digital Object Identifier 10.1109/TPWRS.2023.3292552

where the estimation problem may become ill-conditioned due to the highly different weights used for measurements. In [16] a gradient algorithm is implemented in a hierarchical manner when the distribution network is divided into multiple areas featuring sub-tree topology. The approach developed in [17] allows detection, identification, and isolation of the anomaly during the state estimation solution so that a large percentage of the system states can still be estimated, while isolating the subsystem containing the anomaly causing the divergence.

Further DSSE related partition methods to optimize the SE performance are presented in the literature. The authors of [18] exploit highly accurate μPMU devices and apply machine learning tools to train neural networks for a given set of already installed measurement units to partition the system using the vertex-cut method. In [19], an optimal partitioning approach is proposed by using community discovery algorithm to create overlapping areas with the objective of reducing the complexity of the information exchange between areas. [20] employs branch-line layer method which layers branches by their topological locations. The method considers PMU deployment by postorder-traversal algorithm which belongs to depth-first search method; thus, the method may not be computationally feasible for very large scaled networks. Affine Propagation Clustering Algorithm is used in [21] where optimal clustering center dynamically changes to optimize partition in an iterative manner.

The computational framework proposed in [22] is based on multiple copies of the network and uses different partitions in each copy. Assuming available PMU measurements in each zone a parallel and robust estimation solution capable of rejecting bad data is developed. However, partitioning is carried out using a simple strategy which makes the approach vulnerable to highly inefficient implementations based on the given system and measurement configuration. In [23], a two-stage multi-area SE for distribution system is proposed using an optimal partitioning algorithm. While a local SE is executed for each sub-area at the first stage, information exchange is achieved in the second stage; however, bad data detection is not addressed.

The main contributions of this article are given below.

- Design of effective partitioning algorithm to enable implementation of scalable state estimation solution for very large systems.
- Implementation and experimental validation of strategic partitioning algorithms which facilitate computationally efficient state estimation solution for large scale networks.
- Improvement of the method in [22], where the entire system is partitioned, and solved in each copy whereas in the proposed approach, only a single strategic zone is created and solved in each additional copy, eliminating unnecessary computations significantly improving overall computational performance.
- Facilitating monitoring of large scale networks in real-time by making the SE execution time nearly independent of system size where required number of processor cores are also minimized.

The article is organized as follows. Section II describes formulation of the SE problem, parallel SE framework, steps of

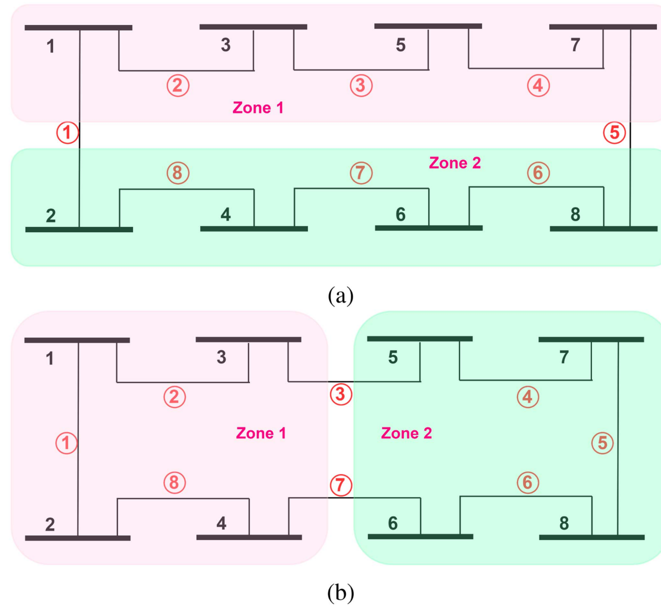


Fig. 1. Sample power network partitioned using two different configurations.

proposed partitioning algorithms. Section III presents the simulation results for strictly radial and meshed networks containing over ~12 thousand buses. Conclusions are drawn in Section IV.

II. The Proposed Method

The most commonly implemented and used SE algorithm in control centers is the Weighted Least Squares (WLS) based SE. It is also the least robust method against measurement errors, yielding biased estimates even in the presence of single bad data [24]. Although there are effective post estimation bad data processing tools [25], [26], they are computationally expensive and thus not widely popular. To avoid bad data vulnerability, this work will employ Least Absolute Value (LAV) based SE algorithm due to its automatic bad data rejection capability [27]. A brief overview of LAV based SE formulation will be given in Section II-A.

The primary goal of the proposed Massively Parallel Distributed (MPD) SE is to make the computation time of the estimation almost independent of the system size. To this end, the power network is partitioned into zones where each zone executes its own local SE. However, this approach inadvertently weakens measurement redundancy at the one boundaries due to the unusable measurements incident to more than one zone. The undesirable impact of this is the increased possibility of missing bad measurements incident to boundary buses. A boundary bus may have neighboring buses belonging to other zones, while an internal bus has all its neighbors in the same zone. In Fig. 1(a), a sample network is partitioned into 2 zones where buses 1,2,7,8 are the boundary buses. Measurements on branches 1 and 5 will not be used by the local estimators in either zone.

The impact of reduced redundancy caused by the system partitioning can be addressed as follows. Since, unlike the measurements incident to boundary buses, measurements incident to internal buses will not be affected by system partitioning, multiple copies of the system will be generated where it is ensured that each bus is an internal bus in at least one copy of the system. Using a different partitioning in Fig. 1(b), measurements on branches 1 and 5 will now be included in the respective local estimations.

As evident from the above simple case, successful implementation of this approach is closely linked to cleverly tailored partitioning algorithm. To minimize the computational burden, a strategical partitioning algorithm is developed. While Algorithm 1 is proposed for partitioning of radial structured networks, Algorithm 2 is developed for partitioning of meshed networks. Both of the algorithms aim to partition the system as evenly as possible while minimizing the number of boundary buses. Moreover, unlike the approach of [22], SE will be executed for only a single strategically selected zone in each additional copy. The aim of Algorithm 3 is to generate a minimum number of system copies while ensuring that each bus is an internal bus in at least one copy. Formulation and structure of MPD SE is presented in Section II-B. The details of Algorithms 1–3 are given in Section II-C, II-D and II-E.

A. Formulation of LAV Based SE

A brief overview of the LAV estimation formulation will be included here for completeness. Consider the measurement equations for a given power system:

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

where z is the $(m \times 1)$ measurement vector, x is the $(n_s \times 1)$ state vector, ϵ is the $(m \times 1)$ measurement error vector, $h(.)$ is the nonlinear function relating states to measurements, n_s is the number of system states, m is the number of measurements. LAV-based state estimation problem can be formulated as [27]:

$$\min \sum_{i=1}^{m} |r_i| \tag{2}$$

$$\text{subject to} \quad z_i = h_i(x) + r_i$$

where, r_i is the residual of i^{th} measurement, i.e., $r_i = z_i - h(\hat{x}_i)$. Using the first-order approximation of $h_i(x^0)$ around x^0, the problem can be rewritten as a sequence of linear programming (LP) problems as:

$$\begin{aligned} \min \quad & c^T \cdot Y \\ & A \cdot Y = b \\ & Y \geq 0 \end{aligned} \tag{3}$$

where;

$A = [H, -H, I, -I]$,
$c^T = [0_n, 0_n, 1_m, 1_m]$,
$b = \Delta z$,
0_n: $(1 \times n)$ vector of "0"s where n is the number of states,
1_m: $(1 \times m)$ vector of "1"s where m is the number of measurements
H : the Jacobian matrix.

B. Massively Parallel Distributed (MPD) SE Framework

The primary goal of the parallel distributed implementation of the robust LAV estimator is to make the computation time of the SE almost independent of the system size. This can be accomplished by first partitioning the network into multiple zones where the solution of each zone will be executed separately.

The flowchart of the overall MPD SE algorithm is given in Fig. 2. As shown in the flowchart, different partitioning algorithms are employed depending on the system topology. Once the original copy of the system is partitioned into n_0 zones using either Algorithms 1 or 2, additional system copies are generated using Algorithm 3. The reasoning behind the use of two different partitioning algorithms and their details are given in the following sections.

In the original (first) copy of the system, the system is partitioned into multiple zones; however, only one strategic zone is created in each of the remaining copies to minimize the computational burden. In other words, the buses which are located outside of the strategic zone in the additional copies are disregarded to avoid unnecessary computations. The details of Algorithms 1, 2, and 3 are given in the following sections. Total number of independent SE runs will be the total number of zones in all copies. The total number of zones, n_{zones}, is determined by the summation of the number of zones in the original copy and the number of additional copies of the system as given below:

$$n_{\text{zones}} = (n_0) + (n_{\text{copies}} - 1) \tag{4}$$

where n_{copies} is the total number of copies, and n_0 is the number of zones in the original (first) copy.

Following system partitioning, the formulation of the SE needs to be modified. Once the power network is partitioned into zones, each zone will execute its own SE with local measurements. The LAV-based SE problem of (3) is solved for all zones in all copies as n_{zones} independent sub-problems.

$$\begin{aligned} \min \quad & c_{c/z}^T \cdot Y_{c/z} \\ & A_{c/z} \cdot Y_{c/z} = b_{c/z} \\ & Y_{c/z} \geq 0 \end{aligned} \tag{5}$$

$A_{c/z} = [H_{c/z}, -H_{c/z}, I, -I]$,
$c_{c/z}^T = [0_{n_{c/z}}, 0_{n_{c/z}}, 1_{m_{c/z}}, 1_{m_{c/z}}]$,
$b_{c/z} = \Delta z_{c/z}$ where $z_{c/z}$ is the local measurement vector in copy c zone z,
$0_{n_{c/z}}$: $(1 \times n_{c/z})$ vector of "0"s where $n_{c/z}$ is the number of states in copy c zone z,
$1_{m_{c/z}}$: $(1 \times m_{c/z})$ vector of "1"s where $m_{c/z}$ is the number of measurements in copy c zone z,
$H_{c/z}$: the Jacobian matrix for copy c zone z.

Since there are multiple estimates of the states in different copies, the resultant estimate is obtained by taking the mean value of **only** the estimates of **internal buses** in each copy as shown in (6). This yields a robust estimate of the system state, free of gross errors.

$$\hat{x}_i = (1/n^*_{\text{zones}}) \cdot \sum_{c=1}^{n^*_{\text{zones}}} x^c_{i,\text{internal}} \tag{6}$$

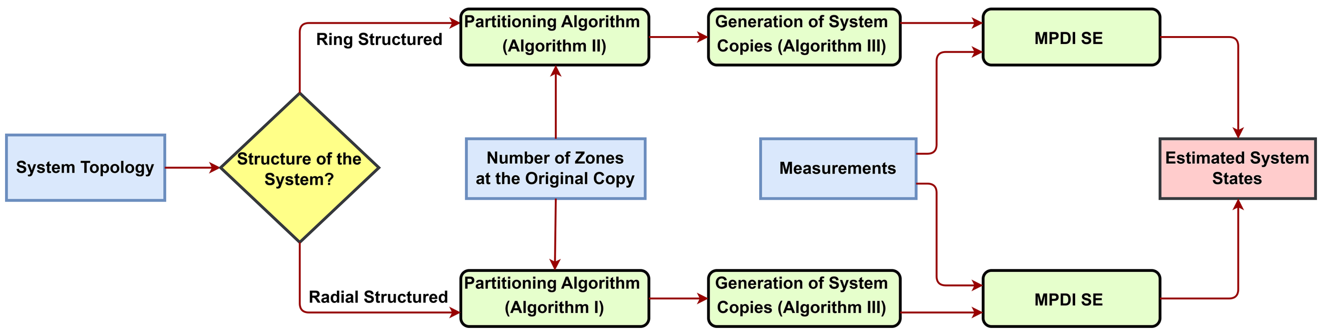


Fig. 2. Flow Chart of the MPD SE procedure.

where $x^c_{i,internal}$ is the estimation of the state of an internal bus in copy c, and n^*_{zones} is the number of zones that x_i is the state of an internal bus.

C. Partitioning Fully Radial Systems (Algorithm 1)

The aim of this partitioning algorithm is to obtain (n_0) zones in the original copy of the system in such a way that the sizes of the zones are approximately equal while minimizing the number of boundary buses. If the sizes of the zones differed significantly, the processors would be used inefficiently. In other words, as the largest zone determines the computation time of the SE, the objective is to minimize the size of the largest zone. Furthermore, number of boundary buses determines the required number of additional system copies.

Conventionally, a graph search based method is used to partition any graph into zones; however, in this work, a new method which exploits the radial structure of the system is developed. Removal of any branch in a radial network will split the system into two islands, considering the number of radial branches is always one less than the number of buses. Therefore, instead of using a computationally complex graph-based method that is not capable of partitioning the graph into similar-sized sub-networks, a new iterative partitioning algorithm is developed.

The pseudo-code of the developed partitioning algorithm is given below as Algorithm 1. The algorithm finds the branch whose removal minimizes the size difference of two islands. Once the original network is divided into two sub-networks, the process is applied to each sub-network resulting their partitioning into two smaller sub-networks. This process can be repeated until the network is divided into at least n_0 zones. Note that repeating the process n times will yield 2^n zones. Hence, the system will be partitioned into n_0' zones which is equal to n_0 rounded up to nearest power of two or simply n_0 if $log_2(n_0)$ is an integer.

Algorithm 1: Fully Radial Partitioning(N, n_0).

Require: N : network topology, n_0 : number of zones at the original copy of the system,
Ensure: N_k^1 and N_k^2 are the two islanded sub-networks of N_k where k is the number of sub-networks in N_k.
$n_z = 1$ ▷Set number of zones to 1
while $n_z \leq n_0$ **do**
  for $j \in k$ **do**
    $branch - m = \min_{m \notin N_j}(size(N_j^1) - size(N_j^2))$
      ▷Find branch-m whose removal minimizes the size difference of two sub-networks
    remove $branch - m$
  end for
  $n_z = n_z \times 2$
end while
return N_{n_z}

Algorithm 2: Ring Partitioning (N, n_0).

Require: N : network topology, n_0 : number of zones at the original copy of the system,
Ensure: N_k is the network with k-many sub-networks
Construct graph using N
Get coordinates of buses from graph
Execute graph-based-partitioning for n_0 zones
return N_{n_0}

D. Partitioning Meshed Systems (Algorithm 2)

Given that configuration of distribution systems are also changing from strictly radial to partially meshed structures, Partitioning of meshed systems may be considered as a minimum cut problem since removal of a minimum number of branches will also minimize the number of boundary buses. A fast k-way partitioning algorithm which minimizes the cost of cutting an edge is proposed in [28]. This algorithm is used as a graph-based partitioning algorithm. The drawback of the minimum cut k-way partition algorithm is that it yields unevenly sized zones. To use the graph-based partitioning algorithm, a graph is constructed using MATLAB's built-in functions and system topology. The coordinates of the buses are obtained using the constructed graph in MATLAB environment as shown below as Algorithm 2.

E. Generating Additional System Copies (Algorithm 3)

Once the original copy of the system is partitioned into zones, loss of certain measurements incident to boundary buses leads to reduced redundancy at the zone boundaries which may affect the robustness of the estimator. Generation of additional system copies aims to ensure that each bus is an internal bus in at least

Algorithm 3: Generation Algorithm of Zones in the Additional System Copies (N, N_b, max_{zone}).

Require: N : network topology, N_b : list of boundary buses, max_{zone} : the size of the largest zone in copy 1.
Ensure: N_c is the zone generated in copy c

```
c = 1
while N_b ≠ ∅ do
    select bus = N_b(1)
    s_z = 1                          ▷Set the size of the zone to 1
    N_c = neighbor(bus)              ▷Find all buses incident to the selected bus
    while s_z ≤ max_zone do
        N_c = N_c ∪ neighbor(N_c)
        calculate s_z = size(N_c)
    end while
    find internal bus list, N_internal,c, in the N_c,
    remove N_internal,c ∩ N_b from N_b
    c= c + 1
end while
Delete redundant copies if any
return N_{n_z}
```

one copy of the system. Instead of partitioning the additional copies into multiple zones, only one strategic zone is created in each additional system copy.

The steps of the algorithm generating the strategic zones are described below as Algorithm 3. System topology, boundary bus list, and size of the largest zone in the original copy are the inputs used for Algorithm 3. Bus list representing the desired strategic zone is initialized by selecting any bus from the list of boundary buses. The list is incrementally enlarged by adding neighbors of the buses already included in the strategic zone. The enlargement procedure is continued until the size of the strategic zone reaches the size of the largest zone in the original copy. Once the construction of the strategic zone is completed, any existing internal buses in the current copy of the system are removed from the boundary bus list. Then, new system copies are generated in the same manner until the boundary bus list is empty. Furthermore, there might be redundant copies where all of the buses are internal buses in at least one copy. Such redundant copies are detected and deleted in the end to eliminate unnecessary computations.

III. Simulations

Computational performance and robustness of the developed estimator and associated partitioning algorithms are evaluated by extensive simulations. Tests are carried out first using a 84-bus utility distribution system, followed by a very large-scale network (VLSN) containing 12589 buses. Two versions of the VLSN are created, the original meshed network and its radial version which is formed by removing all loops form the original meshed version. Impact of varying number of zones in network partitioning on the cpu performance of the MPD SE is shown by simulations. Computation times corresponding to the conventional centralized SE solution are also provided to contrast them with those observed for the proposed approach. All tests are conducted in MATLAB environment using Intel(R) Core i7 2.30 GHz computer with 16 cores.

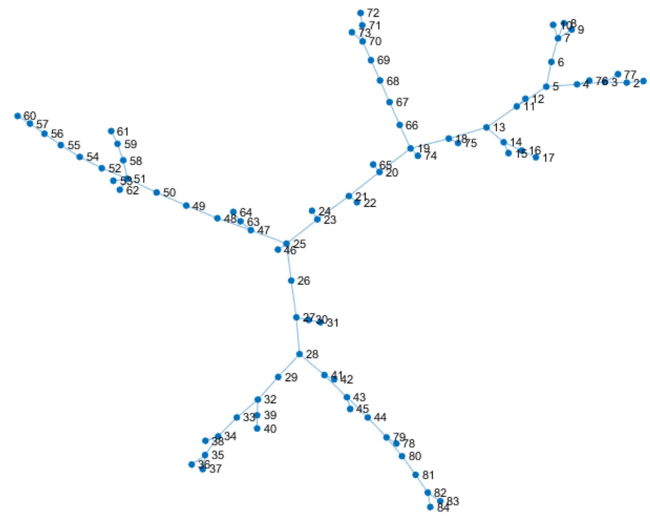
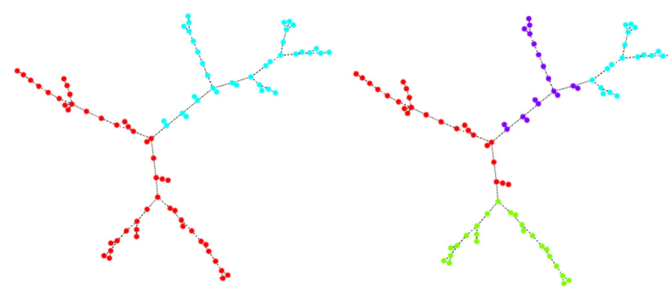


Fig. 3. Real world network with 84 buses visualized in MATLAB environment.



(a) (b)

Fig. 4. The network partitioned into (a) 2 zones, (b) 4 zones.

As indicated above, original meshed VLSN is converted into a strictly radial network by removing certain branches (to avoid loops) to test the scalability of the proposed method in large radial networks.

A. 84 Bus Radial Network

A 84-bus strictly radial utility distribution network whose network graph is constructed in the MATLAB environment to visualize the network is shown in Fig. 3. As a first step, the original copy of the network is partitioned into 4 zones using Algorithm 1. Fig. 4(a) illustrates how removal of branch 23 – 25 splits the network into two zones first, followed by the second partitioning which results in a total of 4 zones shown in Fig. 4(b).

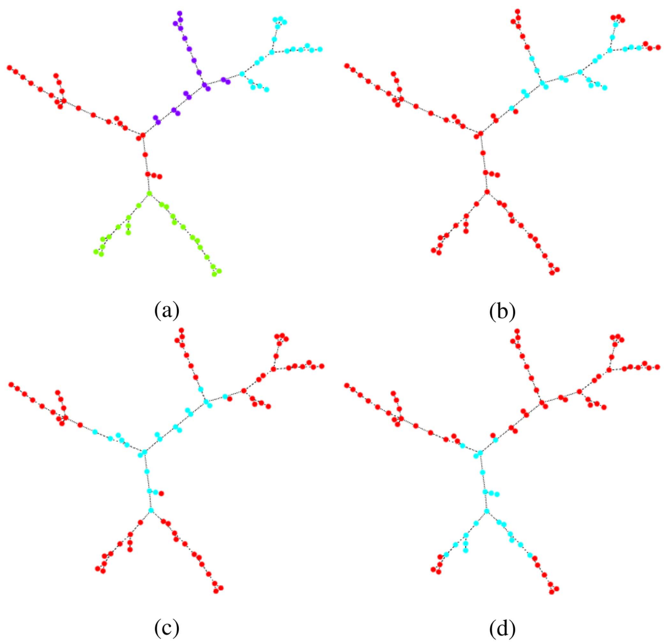
Once the original copy is partitioned into pre-defined n_0 zones, the boundary buses are marked so that Algorithm 3 can be executed to create additional system copies. In this case, three additional system copies are generated to ensure that each bus is an internal bus in at least one copy. The sizes of each zone in every copy are given in Table I where the zones created

TABLE I
SIZE OF EACH ZONE IN EVERY SYSTEM COPY

	Zone 1 of Copy 1	Zone 2 of Copy 1	Zone 3 of Copy 1	Zone 4 of Copy 1	Zone of Copy 2	Zone of Copy 3	Zone of Copy 4
Size of the Zone	24	23	19	18	22	21	21
Total Size of the System	84						

TABLE II
COMPUTATIONAL PERFORMANCE OF MPD AND CENTRALIZED SE SOLUTION

	Zone 1 of Copy 1	Zone 2 of Copy 1	Zone 3 of Copy 1	Zone 4 of Copy 1	Zone of Copy 2	Zone of Copy 3	Zone of Copy 4
Local SE Run Time (s)	0.0031	0.0028	0.0023	0.0021	0.0026	0.0026	0.0025
Centralized SE Run Time (s)	0.0111						



(a) (b) (c) (d)

Fig. 5. (a) Copy 1 (Original Copy), (b) Copy 2, (c) Copy 3, (d) Copy 4 of the network partitioned into 4 zones.

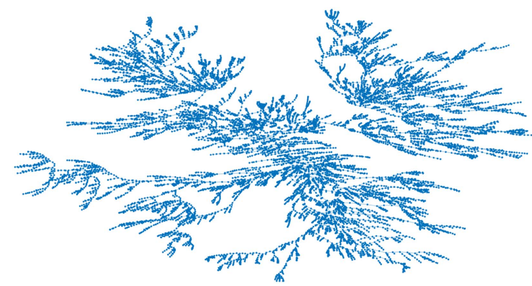
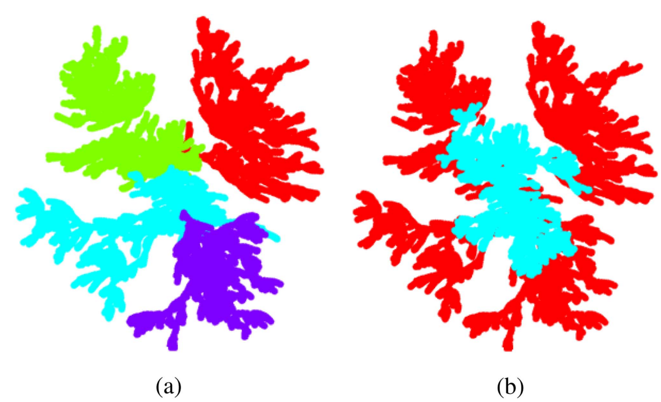


Fig. 6. Radial Structured VLSN Visualized in MATLAB environment.



(a) (b)

Fig. 7. (a) Copy 1 (Original Copy), (b) Copy 2 of Radial Structured VLSN partitioned into 4 zones.

in the additional system copies are strictly smaller than the largest zone in the original copy. The 4 zones of the original copy are indicated by different colors in Fig. 5(a) whereas the strategic zones of Copy 2,3,4 are shown in Fig. 5(b)–(d), respectively.

MPD SE is executed with the proposed zone partitioning algorithm, and computational performance is shown in Table II. The resultant computation time of the MPD SE with 4 zones in the original copy is determined by the largest zone (Zone 1 of Copy 1). Comparing 0.0031 sec. of the proposed method with 0.0111 sec. of the centralized solution, a clear but modest (one third) reduction in cpu time is observed by partitioning the system into 4 zones. In the following sections, much larger size networks will be tested to evaluate the scalability of the method in more realistic meshed or radial networks.

B. 12589 Bus Radial Network

A strictly radial network with 12589 buses is used to test the scalability of the proposed method for large radial networks. The radial VLSN is shown in Fig. 6. First, the system is partitioned into 4 zones using Algorithm 1 and subsequent use of Algorithm 3 yields only one additional copy. The original copy and Copy 2 of the VLSN when $n_0 = 4$ is illustrated in Fig. 7(a) and (b), respectively. When the original copy is partitioned into 8 zones, three additional copies of the system are generated using Algorithm 3, and all copies are shown in Fig. 8.

TABLE III
COMPUTATIONAL PERFORMANCE OF MPD AND CENTRALIZED SE SOLUTION IN RADIAL VLSN

	Centralized Solution	MPD SE (4 zones)	MPD SE (8 zones)	MPD SE (16 zones)
Computational Time	524.21 seconds	13.33 seconds	4.33 seconds	1.47 seconds
Required # of Cores	1	5	11	21

TABLE IV
COMPUTATIONAL PERFORMANCE OF MPD AND CENTRALIZED SE SOLUTION IN MESHED VLSN

	Centralized Solution	MPD Solution with 5 zones	MPD Solution with 20 zones
Computational Time	582.6 seconds	10.77 seconds	1.16 seconds
Required # of Cores	1	10	30

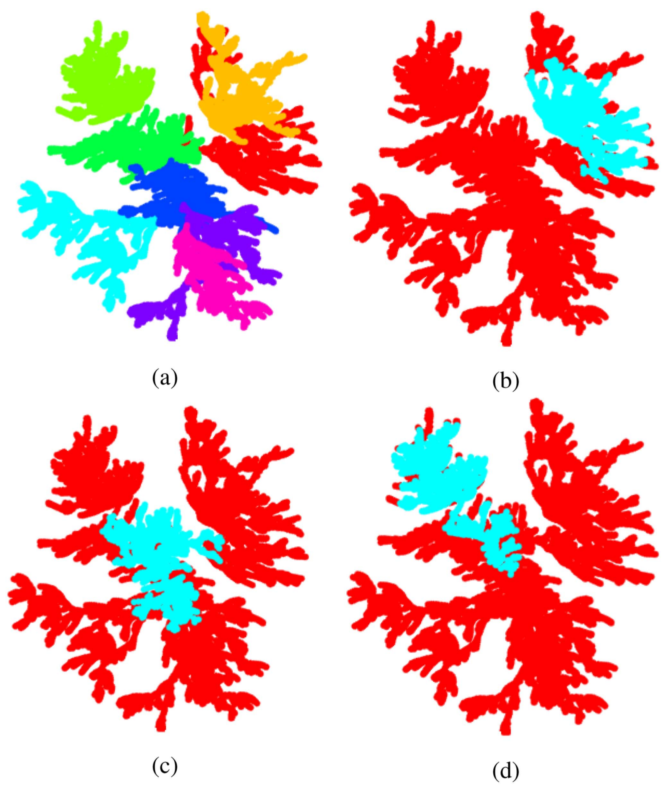


Fig. 8. (a) Copy 1 (Original Copy), (b) Copy 2, (c) Copy 3, (d) Copy 4 of Radial Structured VLSN partitioned into 8 zones.

As shown in Table III, while the centralized SE solution employs single core and solves SE in approximately 520 seconds, cpu time decreases and the number of required processor cores increases when the proposed partitioned method is used. When the original copy is partitioned into 8 zones, a total of 11 cores is required where the total computational time is almost $1/120$ of the centralized solution. When n_0 is increased to 16 zones in the original copy, overall cpu time decreases down to nearly a second. Note that by using Algorithm 1 instead of graph search-based Algorithm 2, similar-sized zones are obtained which reduces the computational time significantly. The number of cores required for parallel processing is calculated by the number of independent local SE runs in each case. When the number of zones in the original copy is 8, in addition to the 8 zonal SE solutions there will be 3 zonal SE solutions corresponding to the 3 strategic zones created in the 3 additional system copies. Thus, a total of 11 independent and parallel SE solutions will be obtained by 11 processors.

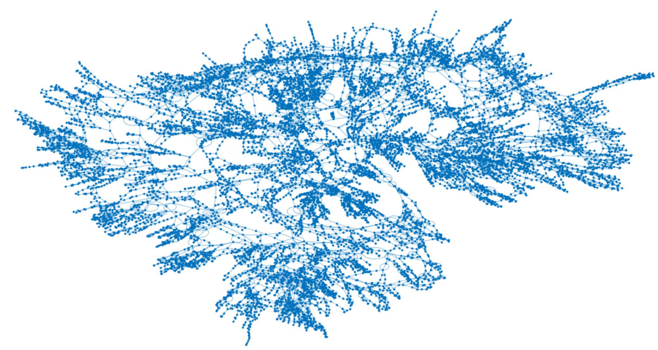


Fig. 9. Meshed Structured VLSN Visualized in MATLAB environment.

C. 12589 Bus Meshed Network

Finally, the proposed method is tested using the original meshed network with 12589 buses. The meshed configuration of the VLSN is illustrated in Fig. 9. In this case, the original copy of the network is partitioned into 5 zones using Algorithm 2 which employs graph search-based partitioning. This is followed up by applying Algorithm 3 which yields 5 additional copies as shown in Fig. 10.

As given in Table IV the centralized SE solution using a single core solves SE in approximately 580 seconds. The cpu time decreases and the number of required processor cores increases as the system is partitioned. While the original copy is partitioned into 5 zones, a total of 10 cores is required where the total computational time is almost $1/60$ of the centralized solution. If instead n_0 is set equal to 20 zones in the original copy, cpu time will decrease to nearly a second.

D. Assessment of Robustness Against Bad Data

Bad data rejection capability of the proposed method is experimentally validated first on the 84-bus radial network and then using the 12K-bus meshed VLSN. Moderately redundant measurement sets are used for both systems, and 100 Monte Carlo (MC) simulations are conducted where a single gross error

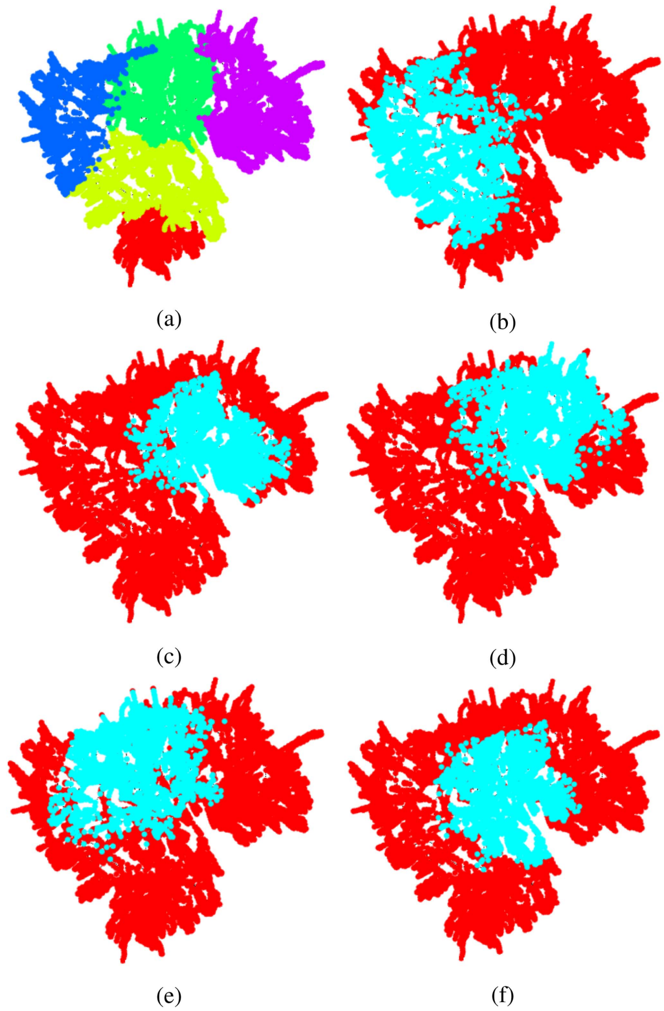


Fig. 10. (a) Copy 1 (Original Copy), (b) Copy 2, (c) Copy 3, (d) Copy 4, (e) Copy 5, (f) Copy 6 of Ring Structured VLSN partitioned into 5 zones.

TABLE V
MSE UNDER UNBIASED/BIASED MEASUREMENT SET

	84 bus network	VLSN
MSE under unbiased measurement set	7.98×10^{-14}	4.58×10^{-5}
$avg(MSE)$ of 100 simulations under biased measurement set	1.57×10^{-13}	7.71×10^{-5}

is introduced to a randomly selected measurement. Mean square error of the estimated states is recorded for each solution. Average of these MSE values for the 100 MC simulations with and without gross errors are given in Table V. Given results strongly support effectiveness of the proposed method in providing robust solutions under random gross errors.

IV. CONCLUSION

This article presents a computationally scalable and statistically robust distributed state estimation framework for very large scale networks. The developed approach is expected to facilitate real-time monitoring of large complex distribution networks and wide-area applications. The two partitioning algorithms that are presented in the article enable efficient parallelisation of computations both for radial and meshed networks. Simulation results experimentally validate the main contribution of the article, i.e. scalability and robustness against bad data of the developed computational framework and associated estimator.

REFERENCES

[1] R. Anderson, "The distributed storage-generation "smart" electric grid of the future," in *Proc. Renewable Energy Transformation*, 2004, pp. 1–7.

[2] G. T. Heydt, "The next generation of power distribution systems," *IEEE Trans. Smart Grid*, vol. 1, pp. 225–235, Dec. 2010.

[3] M. Baran and A. Kelley, "State estimation for real-time monitoring of distribution systems," *IEEE Trans. Power Syst.*, vol. 9, no. 3, pp. 1601–1609, Aug. 1994.

[4] M. E. Baran and A. W. Kelley, "A branch-current-based state estimation method for distribution systems," *IEEE Trans. Power Syst.*, vol. 10, no. 1, pp. 483–491, Feb. 1995.

[5] Y. Deng, Y. He, and B. Zhang, "A branch-estimation-based state estimation method for radial distribution systems," *IEEE Power Eng. Rev.*, vol. 22, no. 8, pp. 71–71, Oct. 2002.

[6] A. G. Expósito, A. de la Villa Jaén, and J. L. R. Izaga, "An alternative state estimation formulation for radial distribution networks," in *Proc. IEEE Lausanne Power Tech*, 2007, pp. 396–400.

[7] A. Bernieri, G. Betta, C. Liguori, and A. Losi, "Neural networks and pseudo-measurements for real-time monitoring of distribution systems," *IEEE Trans. Instrum. Meas.*, vol. 45, no. 2, pp. 645–650, Apr. 1996.

[8] V. Miranda, J. Pereira, and J. T. Saraiva, "Experiences in state estimation models for distribution systems including fuzzy measures," Dec. 1995. [Online]. Available: https://www.osti.gov/etdeweb/biblio/179131

[9] T. V. Cutsem and M. Ribbens-Pavella, "Critical survey of hierarchical methods for state estimation of electric power systems," *IEEE Trans. Power App. Syst.*, vol. PAS- 102, no. 10, pp. 3415–3424, Oct. 1983.

[10] A. Gomez-Exposito, A. Abur, A. D. L. V. Jaen, and C. Gomez-Quiles, "A multilevel state estimation paradigm for smart grids," *Proc. IEEE Proc. IRE*, vol. 99, pp. 952–976, Jun. 2011.

[11] T. Van Cutsem, J. L. Horward, and M. Ribbens-Pavella, "A two-level static state estimator for electric power systems," *IEEE Trans. Power App. Syst.*, vol. PAS-100, no. 8, pp. 3722–3732, Aug. 1981.

[12] L. Zhao and A. Abur, "Multiarea state estimation using synchronized phasor measurements," *IEEE Trans. Power Syst.*, vol. 20, no. 2, pp. 611–617, May 2005.

[13] G. N. Korres, "A distributed multiarea state estimation," *IEEE Trans. Power Syst.*, vol. 26, pp. 73–84, Feb. 2011.

[14] J. Wang and T. Li, "Distributed multi-area state estimation for power systems with switching communication graphs," *IEEE Trans. Smart Grid*, vol. 12, no. 1, pp. 787–797, Jan. 2021.

[15] V. Kekatos and G. B. Giannakis, "Distributed robust power system state estimation," *IEEE Trans. Power Syst.*, vol. 28, no. 2, pp. 1617–1626, May 2013.

[16] X. Zhou, Z. Liu, Y. Guo, C. Zhao, J. Huang, and L. Chen, "Gradient-based multi-area distribution system state estimation," *IEEE Trans. Smart Grid*, vol. 11, no. 6, pp. 5325–5338, Nov. 2020.

[17] P. Ren and A. Abur, "Avoiding divergence in multi-area state estimation," *IEEE Trans. Power Syst.*, vol. 34, no. 4, pp. 3178–3187, Jul. 2019.

[18] A. S. Zamzam and N. D. Sidiropoulos, "Physics-aware neural networks for distribution system state estimation," *IEEE Trans. Power Syst.*, vol. 35, no. 6, pp. 4347–4356, Nov. 2020.

[19] R. Fu, Z. Xu, B. Wong, H. Qian, L. Ju, and W. Jiang, "Multi-area state estimation of distribution networks based on optimal partition," in *Proc. IEEE Sustain. Power Energy Conf.*, 2021, pp. 1686–1691.

[20] P. Yuan, W. Zhang, and J. Chen, "A network partition approach for distributed three-phase state estimation of active distribution networks," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, 2018, pp. 1–5.

[21] Y. Tan, H. Wang, Y. Tian, G. Chen, P. Li, and J. Han, "Distributed state estimation of distribution network with adaptive network partition," in *Proc. IEEE 5th Int. Elect. Energy Conf.*, 2022, pp. 1437–1442.

[22] C. Xu and A. Abur, "A massively parallel framework for very large scale linear state estimation," *IEEE Trans. Power Syst.*, vol. 33, no. 4, pp. 4407–4413, Jul. 2018.

[23] R. Fu, Z. Xu, B. Wong, H. Qian, L. Ju, and W. Jiang, "Multi-area state estimation of distribution networks based on optimal partition," in *Proc. IEEE Sustain. Power Energy Conf.: Energy Transition Carbon Neutrality*, 2021, pp. 1686–1691.

[24] L. Mili and P. J. Rousseeuw, "Robust state estimation of electric power systems," *IEEE Trans. Circuits Syst. I: Fundam. Theory Appl.*, vol. 41, no. 5, pp. 349–358, May 1994.

[25] A. G. An Monticelli, "Reliable bad data processing for real-time state estimation," *IEEE Power Eng. Rev.*, vol. PER- 3, no. 5, pp. 31–32, May 1983.

[26] A. Monticelli and A. Garcia, "Reliable bad data processing for real-time state estimation," *IEEE Trans. Power App. Syst.*, vol. PAS- 102, no. 5, pp. 1126–1139, May 1983.

[27] A. Abur and A. G. Expósito, *Power System State Estimation: Theory and Implementation*. Boca Raton, FL, USA: CRC Press, 2004.

[28] J. P. Hespanha, "An efficient MATLAB algorithm for graph partitioning," Dept. Elect. Comput. Eng., Univ. California, Santa Barbara, CA, USA, Tech. Rep., Oct. 2004. [Online]. Available: https://web.ece.ucsb.edu/~hespanha/techrep.html

Ugur Can Yilmaz (Student Member, IEEE) received the B.S. and M.S. degrees from Orta Dogu Teknik Üniversitesi, Ankara, Turkey, in 2018 and 2021 respectively. He is currently working toward the Ph.D. degree with Northeastern University, Boston, MA, USA. His main research interests include power system monitoring, modeling and power system state estimation.

Ali Abur (Life Fellow, IEEE) received the B.S. degree in electrical engineering from Orta Dogu Teknik Üniversitesi, Ankara, Turkey, and the M.S. and Ph.D. degrees from Ohio State University, Columbus, OH, USA. He is currently a Professor with Electrical and Computer Engineering Department, Northeastern University, Boston, MA, USA.