Robust Hidden Topology Identification in Distribution Systems

Haoran Li, Student Member, IEEE, Yang Weng, Member, IEEE, Yizheng Liao, Student Member, IEEE, Brian Keel, Senior Member, IEEE, Kenneth E. Brown, Member, IEEE

Abstract—With more distributed energy resources (DERs) connected to distribution grids, better monitoring and control are needed, where identifying the topology accurately is the prequisite. However, due to frequent re-configurations, operators usually cannot know a complete structure in distribution grids. Luckily, the growing data from smart sensors, restricted by Ohm’s law, provides the possibility of topology inference. In this paper, we show how line parameters of Ohm’s equation can be estimated for topology identification even when there are hidden nodes. Specifically, the introduced learning method recursively conducts hidden-node detection and impedance calculation. However, the assumptions on uncorrelated data, availability of phasor measurements, and a balanced system, are not met in practices, causing large errors. To resolve these problems, we employ Cholesky whitening first with a proof for measurement decorrelations. For increasing robustness further, we show how to handle practical scenarios when only measurement magnitudes are available or when the grid is three-phase unbalanced. Numerical performance is verified on multi-size distribution grids with both simulation and real-world data.

I. INTRODUCTION

Distributed energy resources (DERs) are broadly defined as renewable energy sources, electricity storage, and intelligent loads. They can offer more controllability for system operators and more choices for end-users. Furthermore, proper deployment of DERs brings economic benefits, such as reduction of network investment and increase clean energy share [1]. Therefore, DER penetration has a consistent increase. The New York State Energy Research & Development Authority (NYSERDA) estimates a total 10,745 GWh of commercial PV by 2030 for the U.S. [1]. However, numerous challenges also come. For example, DERs like rooftop solar panels can generate inverse power [2]. High penetration of PVs affects instant system power balancing [3]. For the low-voltage network, DERs can cause the voltage rise and threaten the network reliability [4]. Thus, power engineers need new monitoring, control and operating methods to face these profound changes.

Distribution grid topology is a foundation for intelligent control and operation such as power flow study [5] and stability analysis. However, the network structures aren’t always available. Firstly, distribution grid reconfigures frequently. For example, in a city distribution network, routine reconfiguration helps to achieve a good radial topology from a meshed network [6], [7]. The frequency of a topology change ranges from once per eight hours with PVs [8] or once a month for medium-voltage grids [9] to once a season [10]. Secondly, some special changes, like the outages and manual maintenance, may not be reported immediately [2]. Further, considering the high cost, instruments like topology sensors aren’t widely installed in the system. Finally, complete information about new DER components may be unavailable. For example, some plug-and-play components belong to users and the utility doesn’t have the right to reach the breaker or switch information [2].

Fortunately, there are growing real-time measurements in distribution systems. For example, metering devices like micro-PMUs [11], frequency measurement devices [12], advanced sensors, and advanced metering infrastructure (AMI) are continuously being applied to distribution grids [13], [14]. Those meters are largely employed to enhance the observability of distribution systems [15]. Especially, topology detection is implemented via measurements, including voltage, current [16] and load data.

For this reason, structure learning with electric data in distribution networks has been visited recently. Initial researches are based on strict assumptions. For example, [17], [18] need the information of all switch locations and find right combinations; [19], [20] require the admittance matrix for a state-estimation-based learning method. These assumptions are unrealistic, since operators may not own complete information of circuit breakers and admittance matrix. Some more recent works overcome these assumptions but require data from all buses. For example, [21] gives a relaxation for the non-convex maximum likelihood estimation with grid’s voltage covariances; [22] uses conditional-independence tests of voltage among all nodes to select candidate lines; [2], [23] requires voltage magnitude of each node for calculating edge weights (e.g., mutual information) as the metric to structure learning; [24], [25] analyze relationships of second moments with each nodal voltage magnitude and power injections, still requiring data from all the buses.

Nonetheless, in systems like secondary distribution grids, sensors are often limited to end-user nodes (observed nodes). One reason is that they are mainly installed for services like price controllable demand and load monitoring [26]. There are also studies in recovering the network with hidden nodes. [26], [27] introduce the second order statistics of observed nodes as a metric to learn the topology, but they need complete information of line impedance. [28] estimates line impedance and employ a graphical-learning algorithm [29] for tree recovery. However, their assumption of uncorrelated-power injections is disobeyed in the real world due to common customer behavior. This leads to a weak performance of their algorithms with realistic data. In addition, their requirement of voltage angles may not be satisfied in many grids due to limited deployment of micro PMUs.

In this paper, we aim at learning the structure of radial
distribution grids only with end-user voltage and current data. For such purpose, we first convert the grid to a graphical model. Each leaf node has a unique path to the root so we can trace back from leaves to the root. In this process, edge weight is used as a metric to discover hidden nodes along the path. Then, recursive grouping (RG) algorithm [29] is used to iteratively detect hidden nodes with edge weights of current-node subset and calculate edge weights related to the hidden nodes.

A key step of RG algorithm is estimating line impedance (i.e., edge weight) of the end-node subset, which requires the nodal current deviation from the statistical mean to be pairwise uncorrelated. However, realistic data still presents low correlations, thus leading to accumulative errors. To eliminate correlations, a whitening technique is employed to make the whitened current deviations uncorrelated. With observed measurements, we prove Cholesky whitening preserves the values of whitened data due to its upper-diagonal structure. Namely, Cholesky whitening implemented on partial data gives exactly the corresponding-partial block of the whitened data from both the observed and the hidden. Therefore, an accurate estimation of line impedances from partial measurements is guaranteed.

Finally, this paper increases the robustness of the learning algorithm by handling the scenarios where only measurement magnitudes are available or the grid is unbalanced. For the first scenario, we employ clustering techniques to select measurements with similar angles that can be therefore discarded. Subsequently, we utilize induction method to prove modulus of impedance with path information is still a feasible metric for RG algorithm to obtain the whole tree. For the second scenario, we propose an approximation for the Ohm’s law so that our structure learning methods can work. The performance of the method is verified on multiple IEEE distribution systems. Real-world load data and the 115-bus grid from Pacific Gas and Electric Company (PG&E) are also included. The result of simulated data and real-time measurements shows high accuracy of our algorithms.

The rest of the paper is organized as follows: Section II illustrates the impedance estimation and RG algorithm. Section III introduces Cholesky whitening to get rid of real-world correlations. Section IV considers the magnitude-available case or unbalanced networks. Section VI exhibits experiments. Section VI makes the conclusions.

II. A GRAPHICAL STRUCTURE LEARNING
A. Graphical Model of Distribution Grids
A graphical model organizes measurements in the form of joint probability distribution without any preference. It is suitable for structure learning since the same topology constrains a period of data without bias. We model the radial network as \( G = (V, E) \), \( V \) denotes the node set that represents grid buses and \( E \) denotes the edge set for distribution lines. Among all the buses, we denote observed node set \( O \) to represent nodes with meters. The hidden-node set \( H \) includes nodes without power consumptions and lacks measurements. Such a representation is termed as a latent tree [29]. In a latent tree, the intermediate nodes can’t exhibit any identifiable

![Latent Tree Diagram](image)

**Figure 1.** An example of a latent tree. The root node is the slack bus and all the observed nodes consist of \( O \). Due to Assumption 1, intermediate node \( a \) has 2 degrees and only can be treated as an observed node. We introduce the following concepts for the convenience of later derivations: path \( P_{hg} \) is defined as the set of lines that connect node \( h \) and \( g \); if node \( a \) lies on the path from node \( g \) to the slack bus, we call node \( a \) the parent of node \( g \) and node \( g \) the child of node \( a \); we call the set of child nodes a sibling group; the sibling group of node \( f \) is designated as \( C(f) \). Here, \( C(f) = \{c, l\} \). information if they lie on an edge without extra branches. Thus, we focus on identifying hidden nodes with at least 3 branches:

**Assumption 1.** In our study scope, the hidden nodes have a degree larger than 2.

The latent tree model with Assumption 1 can be easily understood by Fig. 1, where the root node is the slack bus and all the observed nodes consist of \( O \). Due to Assumption 1, intermediate node \( a \) has 2 degrees and only can be treated as an observed node. We introduce the following concepts for the convenience of later derivations: path \( P_{hg} \) is defined as the set of lines that connect node \( h \) and \( g \); if node \( a \) lies on the path from node \( g \) to the slack bus, we call node \( a \) the parent of node \( g \) and node \( g \) the child of node \( a \); we call the set of child nodes a sibling group; the sibling group of node \( f \) is designated as \( C(f) \). Here, \( C(f) = \{c, l\} \).

In the graphical model for distribution grids, we treat the voltage and current injections of each bus as complex random variables represented via a collection of data. They are stored in \( N \times L \) matrices \( V = (v_1, v_2, \cdots, v_L) \) and \( I = (i_1, i_2, \cdots, i_L) \), where \( N \) is the total number of time slots, \( L \) is the number of buses and \( v_l \) and \( i_l \) represent collections of voltage and current from time 1 to \( N \) for bus \( l \). These measurements are constrained via Ohm’s law whose parameters, i.e., line impedances, are assumed to be edge weights in the graphical model.

B. Edge Weight Estimation for Structure Learning
For the latent tree, additive edge weights for the path among observed nodes are enough to recover the whole tree via mathematical manipulation like summation and subtraction, which is illustrated in the recursive grouping (RG) algorithm [29] in Appendix VII-A. Consequently, in this part, we illustrate how to estimate the required edge weights with voltage and current measurements. Let \( Y \) be the admittance matrix of the grid and the nodal network equation is \( (I)^T = Y(V)^T \) from time 1 to \( N \), where \( T \) is the transpose operator.
example, in Fig. 1, we have physically additive weight for the edge in the latent tree. For

C. Edge Weight-based Structure Learning

RG algorithm in Appendix VII-A. Joining (NJ) and Quartet-based Methods [29]. We introduce

the distance current data, (1) can find $Z$ Assumption 2, $2^0 (\text{where } H)$

Based on Assumption 2, we apply a inner product procedure to the above equation:

$$i_b^H v_a = \sum_{l=2}^{L} Z(a,l) i_b^H i_l = Z(a,b) i_b^H i_b, \quad \text{(1)}$$

where $H$ represents the conjugate transpose. We claim $i_b^H i_l = 0 \ (\forall l \neq b)$ because: 1) $\Delta i_b$ is uncorrelated of $i_l$ due to Assumption 2, 2) the mean $E[i_l] = 0, \forall 1 \leq l \leq L$. Then, $i_b^H i_l = N \cdot E[i_b^H i_l] = N \cdot E[i_b^H] \cdot E[i_l] = 0$. With voltage and current data, (1) can find $Z(a,b) \ (a, b \in O)$.

Considering the property of $Z(a,b)$ in Lemma 1, we capture the distance $d_{ab}$, representing the total impedance of path $P_{ab}$,

$$d_{ab} = Z(a,a) + Z(b,b) - 2Z(a,b), \quad \forall a, b \in O. \quad \text{(2)}$$

C. Edge Weight-based Structure Learning

The obtained distance in (2) consists of line impedance, a physically additive weight for the edge in the latent tree. For example, in Fig. 1, we have $d_{eg} = d_{ea} + d_{ag}$. With additive distances among observed nodes, many methods can conduct structure learning, e.g., Recursive Grouping (RG), Neighbor Joining (NJ) and Quartet-based Methods [29]. We introduce RG algorithm in Appendix VII-A.

To transform $\mathbf{Y}$ into the impedance matrix, we need to eliminate the column and row in $\mathbf{Y}$ with respect to the slack bus so that $\mathbf{Y}$ is invertible [30]. If we consider the deviation of voltage and current from the statistical mean, Ohm’s equation is still valid due to its linearity. In the following derivation, we only consider deviation data and keep the notation unchanged. Let node 1 be the slack bus and we know $v_1 = 0_{N \times 1}$, which brings no statistical information. Thus, we can eliminate the first column of $\mathbf{V}$ and $\mathbf{I}$ and the first row and column of $\mathbf{Y}$ matrix, but the equation still holds. Without loss of generality, we use the same notations for the eliminated matrices in the following derivations. After the elimination, we denote $\mathbf{Z} = \mathbf{Y}^{-1}$. Consequently, we have $(\mathbf{V})^T = \mathbf{Z}(1)^T$.

While the target is to estimate the path impedance, Lemma 1 [27], [30] links it to the entry of $\mathbf{Z}$.

**Lemma 1.** In a radial distribution grid, $Z(a,b)$ is the sum for line impedances of the common path between nodes $a$, $b$ to the slack bus, $a, b \in \{2, 3, \cdots, L\}$ (node 1 is the slack bus).

Mathematically, we have: $Z(a,b) = \sum_{(ik) \in P_{a1} \cap P_{b1}} z_{ik}$, where $P_{a1}$ and $P_{b1}$ are paths from nodes $a$ and $b$ to the slack bus. $z_{ik}$ is the impedance of line $(i,k)$. To specify each entry of $\mathbf{Z}$, we zoom in every column of $\mathbf{V}$: $v_a = \sum_{l=2}^{L} Z(a,l) i_l \ (\forall a \in \{2, 3, \cdots, L\}$. This equation can be left-multiplied by another current-deviation vector $i_b^H$ and if current deviations are uncorrelated, we acquire the entry $Z(a,b)$. Therefore, the following assumption is made to give an unrealistic impedance estimation. We show how to relax this assumption and obtain an accurate result in Section III.

**Assumption 2.** Current deviations of different buses are pairwise uncorrelated in distribution grids.

Based on Assumption 2, we apply a inner product procedure to the above equation:

$$i_b^H v_a = \sum_{l=2}^{L} Z(a,l) i_b^H i_l = Z(a,b) i_b^H i_b, \quad \text{(1)}$$

where $H$ represents the conjugate transpose. We claim $i_b^H i_l = 0 \ (\forall l \neq b)$ because: 1) $\Delta i_b$ is uncorrelated of $i_l$ due to Assumption 2, 2) the mean $E[i_l] = 0, \forall 1 \leq l \leq L$. Then, $i_b^H i_l = N \cdot E[i_b^H i_l] = N \cdot E[i_b^H] \cdot E[i_l] = 0$. With voltage and current data, (1) can find $Z(a,b) \ (a, b \in O)$.

Considering the property of $Z(a,b)$ in Lemma 1, we capture the distance $d_{ab}$, representing the total impedance of path $P_{ab}$,

$$d_{ab} = Z(a,a) + Z(b,b) - 2Z(a,b), \quad \forall a, b \in O. \quad \text{(2)}$$

C. Edge Weight-based Structure Learning

The obtained distance in (2) consists of line impedance, a physically additive weight for the edge in the latent tree. For example, in Fig. 1, we have $d_{eg} = d_{ea} + d_{ag}$. With additive distances among observed nodes, many methods can conduct structure learning, e.g., Recursive Grouping (RG), Neighbor Joining (NJ) and Quartet-based Methods [29]. We introduce RG algorithm in Appendix VII-A.
A. Whitening in Impedance Estimation

Under Assumption 3, this subsection adopts whitening transformation to eliminate current-deviation correlations for observed nodes. Normally, whitening is conducted via a whitening matrix \( W \), which can be directly calculated from current-deviation data. The whitening process is as follows: 
\[
(W)^T = \hat{W}(I)^T = \hat{W}Y(V)^T,
\]
where \( I \) is the whitened measurements with orthogonal column vectors.

The eliminated admittance matrix \( Y \) is invertible and whitening transformations like Cholesky whitening and zero-phase component (ZCA) whitening are invertible [31]. To acquire \( Z \) matrix, we reach to the whitened Ohm’s equation: 
\[
(V)^T = ZM(I)^T = K(I)^T,
\]
where we denote \( K = ZM \) and \( M \) is the inverse of \( W \) matrix.

However, we can only obtain an observed whitening matrix \( \tilde{W}_O \) with observed current-deviation measurements. We utilize the notation \( \tilde{\cdot} \) to represent matrix calculated from the observed measurements and the subscript \( \mathcal{O} \) to represent the sub-matrix corresponding to observed nodes in the matrix of all nodes. For example, usually \( \tilde{W}_O \neq W_O \) since \( \tilde{W}_O \) is only obtained from observed measurements:
\[
(\tilde{I}_O)^T = \tilde{W}_O(I_O)^T,
\]
where \( I_O \) and \( \tilde{I}_O \) denote the observed current deviations and whitened-observed current deviations. In Section III-B, Theorem 1, we will prove that under Cholesky whitening, \( \tilde{I}_O = I_O \). Therefore, we apply the inner product to obtain entry of \( K \): 
\[
K_O(a, b) = \tilde{I}_b^H a \hat{I}_b^H b, \quad \forall a, b \in \mathcal{O},
\]
where \( \hat{I}_b \) is the \( b^{th} \) column vector of \( I \) and \( K_O \) is the observed block of \( K \) matrix. Then, we give an estimation for the observed block of \( Z \) matrix, denoted as \( \hat{Z}_O \):
\[
\hat{Z}_O = K_O M_O^{-1} = K_O \tilde{W}_O.
\]

In the following subsection, we will prove Cholesky whitening can offer a relatively accurate estimation with (4).

Unlike \( Z \) matrix, the estimated impedance matrix \( \hat{Z}_O \) is asymmetric. The estimated distance between two observed nodes \( \forall a, b \in \mathcal{O} \) can be gained as follows:
\[
d_{ab} = \hat{Z}_O(a, a) + \hat{Z}_O(b, b) - \hat{Z}_O(a, b) - \hat{Z}_O(b, a).
\]

B. Choice of Whitening: Cholesky Whitening

We prove Cholesky whitening gives an accurate result via the above estimation process. Generally, a whitening matrix \( W \) aims at transforming the covariance matrix to be the identity matrix, then we have: 
\[
W^H W = \Sigma^{-1}, \quad \Sigma = \text{covariance matrix of } I.
\]
According to Cholesky decomposition, we know: 
\[
U^H U = \Sigma^{-1}, \quad U \text{ is the unique upper diagonal matrix}.
\]
Combining these two equations, we obtain: 
\[
M = W^{-1} = U^{-1}, \quad \text{and } M \text{ is also an upper diagonal matrix}.
\]
Then, we claim the following theorem.

**Theorem 1.** If we arrange bus numbers in the observed nodes set \( \mathcal{O} \) from \( k \) to \( L \) (\(|\mathcal{O}| = L - k + 1\)), under Cholesky whitening, 
\[
\hat{M}_O = M_O, \quad \hat{W}_O = W_O, \quad \text{and } \hat{I}_O = I_O.
\]

The proof can be seen in Appendix VII-B. Based on Theorem 1, we know the calculated \( K_O \) matrix in III-A is the same as the corresponding block in \( K \). To derive the error term, we rewrite \( K = ZM \) in the following form:
\[
\begin{bmatrix}
K_1 & K_2 \\
K_3 & K_O
\end{bmatrix}
= 
\begin{bmatrix}
Z_1 & Z_2 \\
Z_3 & Z_O
\end{bmatrix}
\cdot
\begin{bmatrix}
M_1 & M_2 \\
M_3 & M_O
\end{bmatrix},
\]
where \( M_O = \tilde{M}_O \) by Theorem 1. Thus, \( K_O \) can be written next:
\[
K_O = Z_3 M_2 + Z_O \tilde{M}_O.
\]

Consequently, the distance-estimation error comes from off-diagonal block \( M_2 \) of the \( M \) matrix. The following theorem claims that under Assumption 3, we have \( M_2 = 0_{(k-1) \times (L-k+1)} \).

**Theorem 2.** If current-deviation correlations only exist among observed nodes, the off-diagonal block in the inverse of Cholesky whitening matrix (i.e., \( M_2 \) in \( M \) matrix) is a zero matrix.

The proof can be seen in Appendix VII-C. Theorem 2 claims the feasibility of \( Z \) matrix estimation in (4). Finally, we come to our structure learning Algorithm 2.

**Algorithm 2** Cholesky Whitening-based Structure Learning

**Input:** The current and voltage measurements of observed nodes \( \mathcal{O} \).

**Output:** Structure of the latent tree and line parameter estimations.

1. Whiten the current deviations of the observed nodes: 
\[
\tilde{I}_O = W_O I_O.
\]
2. Calculate \( \tilde{I}_b^H a \hat{I}_b^H b, \quad \forall a, b \in \mathcal{O} \).
3. for \( \forall a, b \in \mathcal{O} \), do
4. \( K(a, b) = \tilde{I}_b^H a \hat{I}_b^H b \).
5. \( Z_O = K_O W_O \).
6. \( d_{ab} = Z_O(a, a) + Z_O(b, b) - Z_O(a, b) - Z_O(b, a), \quad \forall a, b \in \mathcal{O} \).
7. end for
8. Use the distance \( d_{ab} \), \( \forall a, b \in \mathcal{O} \) to form the estimated distance matrix \( D \) of observed nodes.
9. \( G = \{V, E\} \leftarrow RG(D, \mathcal{O}) \).

IV. STRUCTURE LEARNING WITHOUT ANGLE OR IN AN UNBALANCED NETWORK

In addition to the correlation issue, there are other realistic challenges for the above learning process: 1) only voltage and current magnitudes are available in many grids, and 2) many distribution grids are unbalanced. In this subsection, we propose methods to address these two challenges and further increase the robustness of the learning process.

A. Data Selection to Eliminate Angle Information

For challenge 1), the absence of angle information forces us to select proper measurements so that the impact of angle differences is reduced in the structure learning. We consider adding modulus in equation (1): 
\[
|Z(a, b)| = |v_a i_b^H / i_b i_b^H|, \quad \forall a, b \in \mathcal{O}, \text{where } | \cdot | \text{ represents the modular operation to a complex number or the element-wise modular}
operation to a complex vector or matrix. To eliminate angle information in the right-hand side, the following assumption is introduced:

Assumption 4. In a collection of data, the angles of voltage and current are nearly unchanged in distribution grids.

Assumption 4 can be achieved via carefully picking up of data. The voltage is usually stable when the topology is unchanged. The abrupt change of voltage magnitudes indicates the change of the topology. In addition, we prefer data with a smooth nodal-reactive-power change that brings an ignorable change for current angles. Rather than segmentation for a continuous time period, clustering techniques are employed to select data from the historical dataset. For time slot \( n \), we map the sampled data into a high dimensional space

\[
\mathbf{z}[n] = (\mathbf{v}_O[n], \lambda \mathbf{q}_O[n]),
\]

where \( \mathbf{v}_O[n] \) and \( \mathbf{q}_O[n] \) represent the collection of voltage magnitudes and nodal-reactive-power deviations from the mean for observed nodes in time \( n \), respectively, and \( \lambda \) is a weight term. The reactive power deviation is utilized for normalization, but the variance information is preserved. Consequently, we use methods like k-means or hierarchical clustering to form clusters in the mapped space with Euclidean distance as the metric. When the real-time point comes, the cluster to which the new point belongs offers an appropriate data collection for Assumption 4.

Based on Assumption 4, angles can be totally eliminated and we have \( |\mathbf{Z}(a,b)| = |\mathbf{v}_a|/|\mathbf{v}_b| \). If we consider the whitening process, we can obtain the approximated \( |\mathbf{K}_O| \) and \( |\mathbf{W}_O| \) matrices from voltage and current magnitudes. Similar to (4), we give an approximation for \( |\mathbf{Z}_O| = |\mathbf{K}_O| \cdot |\mathbf{M}_O^2| = |\mathbf{K}_O| \cdot |\mathbf{W}_O| \). We will show this is a good approximation in the numerical experiments Section V-C.

B. Structure Learning without Angles

Since the modulus \( |\mathbf{Z}(a,a)| \) obtained above represents the modulus of sum impedances along the path from node \( a \) to the slack bus, the availability of each path from leaf nodes to the root suggests the possibility to recover the tree.

With the modulus of components in \( \mathbf{Z} \) matrix, we give a distance estimation for observed nodes \( a \) and \( b \): \( d_{ab} = |\mathbf{Z}(a,a)| + |\mathbf{Z}(b,b)| - 2|\mathbf{Z}(a,b)| \). Though \( d_{ab} \) can’t represent the true impedance between nodes \( a \) and \( b \) now, we use induction method to prove that it helps the RG algorithm to recover the tree. The proof can be seen in Appendix VII-D.

C. Structure Learning in an Unbalanced Network

For unbalanced networks in challenge 2), we give a general form to approximate Ohm’s law via the impedance model of four-wire (with neutral conductor) in Fig. 3 [32]. We use the corresponding lowercase to represent phase \( A, B, C, N \) and the ground \( G, G' \):

\[
\begin{bmatrix}
 v_{ag}' \\
v_{bg}' \\
v_{cg}' \\
v_{ng}' \\
v_{gg}'
\end{bmatrix} =
\begin{bmatrix}
 z_{aa} & z_{ab} & z_{ac} & z_{an} & z_{ag}' \\
z_{ba} & z_{bb} & z_{bc} & z_{bn} & z_{bg}' \\
z_{ca} & z_{cb} & z_{cc} & z_{cn} & z_{cg}' \\
z_{na} & z_{nb} & z_{nc} & z_{nn} & z_{ng}' \\
z_{ga} & z_{gb} & z_{gc} & z_{gn} & z_{gg}'
\end{bmatrix}
\begin{bmatrix}
 i_a \\
i_b \\
i_c \\
i_n \\
i_g
\end{bmatrix},
\]

where \( v_{kg}' \), \( \forall k \in \{a,b,c,n,g\} \) represents the nodal phase voltage to the ground, \( i_k \) represent the phase current to the ground, \( z_{kk} \) represents the self impedance, and \( z_{ik} \) (\( \forall i \neq k \)) represents the mutual impedance. Carson assumes the sum of the wire current returns from the ground \( i_g = \sum_{k \in \{a,b,c,n,g\}} i_k \), analyzes their electromagnetic relations and give the unit self and mutual impedance for each wire and the ground [32]. We can assume we have the prior information for the impedance ratio. In phase \( A \), we write \( \lambda_{ak} = z_{ak}/z_{aa}, \forall k \in \{a,b,c,n,g\} \). If phase \( k \) is missing, \( \lambda_{ak} = 0 \). Thus, we have

\[
v_{ag}' = z_{aa} \sum_{k \in \{a,b,c,n,g\}} \lambda_{ak} i_k = z_{aa} i'_a,
\]

where \( i'_a \) is the weighted current in phase \( A \) and (9) is the equivalent Ohm’s equation. Similar process can be conducted for other phases. Due to the linearity of (9), the weighted current can be extended to the nodal current injection and the network Ohm’s equation still holds. Therefore, for each time slot and for each node, we utilize equation (9) to acquire the nodal phase voltage and the weighted current, which form the phase voltage matrix \( \mathbf{V}' \) and equivalent current matrix \( \mathbf{I}' \).

Finally, our structure learning method can be conducted with observed measurements \( \mathbf{V}_O \) and \( \mathbf{I}_O \).

V. Numerical Experiments

We test simulated data and real-world power data from Pacific Gas and Electric Company (PG&E) on IEEE distribution 8-, 19-, 33-bus and PG&E 115-bus systems. In addition, we test the three-phase unbalanced case (123-bus system from GridLABD) with realistic three-phase data from PG&E. For simulated data, we consider independent Gaussian distribution to generate current deviations: \( \Delta \mathbf{I} \sim \mathcal{N}(\mu, \Sigma) \), where \( \mu = [0, \cdots, 0]^T \) and \( \Sigma \) is a diagonal matrix with all diagonal elements to be 0.025. In addition, the current injections for the hidden nodes are zero. We consider \( N = 8760 \) samples to represent one-year data. PG&E load profile contains hourly real power consumption of 123, 000 residential loads in North California, USA [2]. As for reactive power \( q_i \), at bus \( i \), we consider a random power factor at time \( n \), \( p_f(n) \sim \text{Unif}(0.85, 0.95), \forall n \in \{1, 2, \cdots, 8760\} \). Then, we have: \( q_i(n) = p_r(n) \sqrt{1 - p_f(n)^2} / p_f(n) \). We assume the hidden nodes don’t have power consumptions and input the real-world data into the AC power flow solver in MATPOWER to obtain the voltage and current phasor.
For the magnitude data, we show how to select the right data. The input data can be phasor data or magnitude data. Cholesky whitening-based structure learning with real-world data is an effective method in recovering the topology of smart grids.

We define the average hidden-nodes-recovery rate ($ah(\%)$) and average correct-connection-recovery rate ($ac(\%)$) to weigh the performance of the algorithm. For a fixed number of hidden nodes ($hn$), $ah = 100 \times \sum_{k=1}^{K} (N_{hk}^{rk} / N_{hk}) / K$, $ac = 100 \times \sum_{k=1}^{K} (N_{hk}^{rk} / N_{hk}) / K$, where $N_{hk}^{rk}$, $N_{hk}^{rk}$, $N_{hk}^{rk}$ and $N_{hk}^{rk}$ are the number of recovered hidden nodes, true hidden nodes, recovered connections and true connections, respectively. $K$ is the total combinations of $hn$ hidden nodes. For example, Fig. 4 is the 19-bus system modified from the IEEE 18-bus distribution. If we want to calculate $ah$ and $ac$ versus $hn = 3$ in Fig. 4, we should consider different combinations of 3 hidden nodes: (3, 4, 12), (3, 12, 13) and (12, 13, 15). Accordingly, we input measurements of neighboring nodes (e.g., 2, 5, 11, 13 and 19 for the combination (3, 4, 12)) as the observed nodes.

### A. General Performance for Balanced and Unbalanced Grids

This subsection presents the general performance of Cholesky whitening-based structure learning with real-world data. The input data can be phasor data or magnitude data. For the magnitude data, we show how to select the right input collection in Part V-C. For 19-bus system in Fig. 4, we consider different hidden node combinations and obtain Table I. Further, we test different systems: 8-, 19-, 33-, 115- and 123-bus networks with $nh = 1, 2, 3$. The 123-bus network is unbalanced. The result is shown in Table II. The blank data means there is no complete combination for that case. The structure learning algorithm finds the correct connectivities with angle information in balanced networks, but obtains relatively lower accuracy with only magnitude input or in the unbalanced network. This is reasonable due to impedance-approximation error.

| Network size | 8   | 19  | 33  | 115 | 123 |
|--------------|-----|-----|-----|-----|-----|
| $nh = 1$     |     |     |     |     |     |
| Input phasor | $ah(\%)$ | 100 | 100 | 100 | 100 |
|              | $ac(\%)$ | 100 | 100 | 100 | 100 |
| $nh = 2$     |     |     |     |     |     |
| Input phasor | $ah(\%)$ | 100 | 100 | 100 | 100 |
|              | $ac(\%)$ | 100 | 100 | 100 | 100 |
| $nh = 3$     |     |     |     |     |     |
| Input phasor | $ah(\%)$ | 100 | 100 | 100 | 100 |
|              | $ac(\%)$ | 100 | 100 | 100 | 100 |
| $nh = 4$     |     |     |     |     |     |
| Input magnitude | $ah(\%)$ | 100 | 100 | 80.56 | 75 |
|              | $ac(\%)$ | 100 | 100 | 80.38 | 77.78 |

### B. Effectiveness of Whitening

In this section, we illustrate the effectiveness of whitening. Generally, the error comes from non-zero terms of inner product $\Delta s_i^H \Delta s_i$, $\forall b \neq l$. Fig 5 demonstrates the ratio $|\Delta s_i^H \Delta s_i|/|\Delta s_i^N \Delta s_i|$. The result shows that the simulated data and the Cholesky-whitened data are near uncorrelated while the realistic data before whitening is slightly correlated. Moreover, the sum of the ratio for non-whitened data (except node 8) is 194.69%.

### C. Performance Evaluation with Magnitude Data

In this subsection, we illustrate that clustering to pick up angle-consistent data can give a good approximation for $|Z_O|$ matrix in Section IV-A. Firstly, we show the clustering technique helps to eliminate angle information. With measurements of 19-bus system, k-means clustering is employed when $k = 3$ for the mapped
For the independent simulated data, the average errors are 1.83% (real distance without whitening), 1.93% (imaginary distance without whitening), 1.61% (real distance with whitening) and 1.81% (imaginary distance with whitening).

For the real-world data, the average errors are 11.64% (real distance without whitening), 9.45% (imaginary distance without whitening), 3.35% (real distance with whitening) and 3.31% (imaginary distance with whitening).

Figure 6. The distance-estimation errors of the observed nodes in 19-bus system.

Figure 7. 3D visualization for measurement points $z[n]$ in IV-A. We find 3 clusters (orange, green and blue clusters) for k-means when $k = 3$.

Figure 8. The box diagram of current angles. We consider 19 buses’ current angles with 2 collections of data: 1) one year’s data when $N = 8760$, 2) the orange clusters’ data when $N = 994$. The blue box represents the angle deviation calculated from one year’s data and the orange box is obtained from the orange cluster.

Figure 9. The heat map of the error matrix for $|Z|_O$. Specifically, we calculate the element-wise percentage error between the estimated matrix in Section IV-A and the true matrix. The maximum error is 23.48% and most of the errors (in the dotted-red frame) are below 20%.

Points $z[n]$ in IV-A. In this case, $z[n]$ is formed via the voltage magnitude and reactive power injections of observed buses in Fig. 4. Fig. 7 visualizes the mapped data in 3D dimension with 3 parts: orange, green and blue clusters.

To demonstrate the function of clustering, we draw the box diagram for current angles in the 19-bus system. In Fig. 8, we consider 2 collections of data: 1) one year’s data when $N = 8760$, 2) the orange clusters’ data when $N = 994$. While the one year’s data has large deviations in blue boxes, the orange cluster’s data only has small changes in orange boxes.

Subsequently, we illustrate the heat map of the error matrix for $|Z|_O$, and the observed nodes are the same as Fig. 4. Specifically, we calculate the element-wise percentage error between the estimated matrix in Section IV-A and the true matrix in Fig. 9. The maximum error is 23.48% and most of the errors (errors in the dotted-red frame) are below 20%.

Finally, we input voltage and current magnitudes of the orange cluster in Fig. 7 to the learning algorithm. The result is in Part V-A, Table I and II. The proposed algorithm still has a good performance for different hidden node combinations and various networks.

ACKNOWLEDGEMENT

The first two authors would like to acknowledge the support and work done by Salt River Project.

VI. CONCLUSIONS

The distributed energy resources (DERs) are increasing in distribution grids, bringing high requirements for monitoring and control. To achieve these targets, the topology is the foundation for the system operator.

Due to frequent topology reconfiguration, this paper introduces a learning algorithm to identify the radial topology in real-time. Starting from end nodes (observed nodes), the
Recursive Grouping (RG) algorithm to detect hidden nodes among a target subset of nodes and recursively update the current target subset. In RG algorithm, line impedance is the metric to identify node relationships. To obtain the input impedances for RG (i.e., impedances among end nodes), we propose an estimation process. In this estimation, the correlation of measurements generates errors so we introduce Cholesky whitening to eliminate the correlation. Finally, we handle the cases when only voltage and current magnitudes are available or the network is three-phase unbalanced. We test our algorithms on various distribution systems with simulated data and real-world data and observe high performance in our proposed methods.

REFERENCES

[1] N. Y. I. S. Operator and D. G. Energy, “A review of distributed energy resources,” Sep 2014.

[2] Y. Weng, Y. Liao, and R. Rajagopal, “Distributed energy resources topology identification via graphical modeling,” *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 2682–2694, Jul 2017.

[3] D. Lew, M. Asano, J. Boemer, C. Ching, U. Focken, R. Hydzik, M. Lange, and A. Motley, “The power of small: The effects of distributed energy resources on system reliability,” *IEEE Power and Energy Magazine*, vol. 15, no. 6, pp. 50–60, Nov 2017.

[4] P. D. F. Ferreira, P. M. S. Carvalho, L. A. F. M. Ferreira, and M. D. Lucci, “Distributed energy resources integration challenges in low-voltage networks: Voltage control limitations and risk of cascading,” *IEEE Transactions on Sustainable Energy*, vol. 4, no. 1, pp. 82–88, Jan 2013.

[5] S. Abhyankar, Q. Cui, and A. J. Flueck, “Fast power flow analysis using a hybrid current-power balance formulation in rectangular coordinates,” in *2014 IEEE PES T D Conference and Exhibition*, Apr 2014, pp. 1–5.

[6] C. Rudin, D. Waltz, R. N. Anderson, A. Boulanger, A. Salleb-Aouissi, M. Chow, H. Dutta, P. N. Gross, B. Huang, S. Jerome, D. F. Isaac, A. Kressner, R. J. Passonneau, A. Radeva, and L. Wu, “Machine learning for the new york city power grid,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 2, pp. 328–345, Feb 2012.

[7] C. Rudin, S. Ertekin, R. Passonneau, A. Radeva, A. Tomar, B. Xie, S. Lewis, M. Riddle, D. Pangravsinij, and T. McCormick, “Analytics for power grid distribution reliability in new york city,” *Interfaces*, vol. 44, no. 4, pp. 364–383, 2014.

[8] R. A. Jabr, “Minimum loss operation of distribution networks with photovoltaic generation,” *IET Renewable Power Generation*, vol. 8, no. 1, pp. 33–44, Jan 2014.

[9] O. F. Fajardo and A. Vargas, “Reconfiguration of mv distribution networks with multicast and multipoint alternative supply, part ii: Reconfiguration plan,” *IEEE Transactions on Power Systems*, vol. 23, no. 3, pp. 1401–1407, Aug 2008.

[10] E. A. Bueno, C. Lyra, and C. Cavellucci, “Distribution network reconfiguration for loss reduction with variable demands,” in *2004 IEEE/PES Transmission and Distribution Conference and Exposition: Latin America (IEEE Cat. No. 04EX956)*, Nov 2004, pp. 384–389.

[11] A. von Meier, D. Culler, A. McEachern, and R. Arghandeh, “Micro-synchrophasors for distribution systems,” in *ISGT* 2014, Feb 2014, pp. 1–5.

[12] Z. Zhong, C. Xu, B. J. Billian, L. Zhang, S. J. S. Tsai, R. W. Conners, V. A. Centeno, A. G. Phadke, and Y. Liu, “Power system frequency monitoring network (fnet) implementation,” *IEEE Transactions on Power Systems*, vol. 20, no. 4, pp. 1914–1921, Nov 2005.

[13] Y. Weng and R. Rajagopal, “Probabilistic baseline estimation via gaussian process,” in *2015 IEEE Power Energy Society General Meeting*, Jul 2015, pp. 1–5.
some \( q \), \( \Pi_q \) may consist of a single node, like node \( b \) in Fig. 10. After this partition, we update the target set for further grouping process, i.e., “Node Updating” in Fig. 10.

The “Node Updating” process is conducted via the following criteria: 1) if the node in \( \mathcal{Y} \) doesn’t connect to any other node like node \( b \) in Fig. 10, we include it in the new target set \( \mathcal{Y}_{\text{new}} \). 2) If the node in \( \mathcal{Y} \) is the parent node like node \( a \) in Fig. 10, we include it in \( \mathcal{Y}_{\text{new}} \) since its relationship with the hidden nodes isn’t figured out. 3) The hidden nodes detected in the last “Node Grouping” process are included in \( \mathcal{Y}_{\text{new}} \) like node \( f_1 \) and \( f_2 \). Therefore, we construct the target node set in the next iteration, i.e., node \( \{a, b, f_1, f_2\} \) in Fig. 10.

To enable the updated nodes set to be “observed”, we need to recompute the distance between the hidden node \( h \) in \( \mathcal{Y} \) and all other nodes \( p \in \mathcal{Y} \) in Step 7. We denote \( \mathcal{Y}_{\text{old}} \) to be the observed-node set in the previous iteration. Let \( i, j \in \mathcal{C}(h) \) be two children of \( h \), and let \( k \in \mathcal{Y}_{\text{old}} \setminus \{i, j\} \) be an arbitrary node. Knowing that \( d_{ih} - d_{jk} = d_{ik} - d_{jk} = \Phi_{ijk} \) and \( d_{ih} + d_{jk} = d_{ij} \), we calculate the distance between \( i \) and \( h \) as follows:

\[
d_{ih} = \frac{1}{2} (d_{ij} + \Phi_{ijk}).
\] (10)

For any other node \( p \in \mathcal{Y} \), we compute \( d_{hp} \) by discussing \( p \) is hidden or not:

\[
d_{hp} = \begin{cases} 
  d_{ip} - d_{ih}, & p \in \mathcal{Y}_{\text{old}} \\
  d_{ik} - d_{ih} - d_{pk}, & p \text{ is hidden, and } k \in \mathcal{C}(p).
\end{cases}
\] (11)

Subsequently, we introduce the Recursive Grouping (RG) algorithm (Algorithm 3), termed as \( \mathcal{RG} \). The iterative updating of \( \mathcal{Y} \) in \( \mathcal{RG} \) makes sure that relationships of all the nodes can be identified. For input distance matrix \( \mathbf{D} \), \( \mathbf{D}(a, b) \) and \( \mathbf{D}(b, a) \) correspond to the distance \( d_{ab} \). Fig. 11 illustrates the process of RG.

### B. Proof of Theorem 1

**Proof:** For the whitening process, we have:

\[
\mathbf{M} \mathbf{M}^H = \Sigma,
\] (12)

where \( \mathbf{M} \) is the true Cholesky whitening matrix. (12) is visualized as follows:

\[
\begin{bmatrix}
M_{11} & \ldots & \ldots & M_{1L} \\
M_{k1} & \ldots & \ldots & M_{k1L} \\
\vdots & \ddots & \ddots & \vdots \\
M_{1L} & \ldots & M_{kL} & M_{LL}
\end{bmatrix}
= \begin{bmatrix}
\Sigma_{11} & \ldots & \ldots & \Sigma_{1L} \\
\Sigma_{21} & \ldots & \ldots & \Sigma_{2L} \\
\vdots & \ddots & \ddots & \vdots \\
\Sigma_{1L} & \ldots & \Sigma_{kL} & \Sigma_{LL}
\end{bmatrix}.
\] (13)

We arrange the bus number of the observed nodes set \( \mathcal{O} \) from \( k \) to \( L \). The observed covariance matrix, termed as \( \Sigma_{\mathcal{O}} \), contains values \( \Sigma_{kk}, \Sigma_{k,k+1}, \ldots, \Sigma_{LL} \) that can be directly calculated with the observed measurements. Therefore, \( \Sigma_{\mathcal{O}} = \ldots \).
Algorithm 3 Recursive Grouping Algorithm for Structure Learning: \(\mathcal{RG}\)

**Input:** Distances matrix \(D\), Observed nodes set \(\mathcal{O}\).

**Output:** Structure of the latent tree and line parameter estimations.

1. Initialize \(\mathcal{Y} := \mathcal{O}\).
2. while \(|\mathcal{Y}| \geq 3\) do
3.    Compute \(d_{ijk} = d_{ik} - d_{jk}\) for all \(i, j, k \in \mathcal{Y}\).
4.    Using the “Node Grouping” procedure and partition \(\mathcal{Y}\) into \(\{\Pi_q\}_{q=1}^Q\), \(\mathcal{Y}_{\text{new}} = \bigcup_{k=1}^Q \Pi_k\).
5.    For each \(q = 1, \ldots, Q\) with \(|\Pi_q| > 2\), if \(\Pi_q\) contains a parent node \(u\), update \(\mathcal{Y}_{\text{new}} = \mathcal{Y}_{\text{new}} \bigcup \{u\}\) Otherwise, introduce a new hidden node \(h\), connect \(h\) to every node in \(\Pi_q\), and set \(\mathcal{Y}_{\text{new}} = \mathcal{Y}_{\text{new}} \bigcup \{h\}\).
6.    Update the observed nodes set: \(\mathcal{Y}_{\text{old}} = \mathcal{Y}\) and \(\mathcal{Y} = \mathcal{Y}_{\text{new}}\).
7.    For each new hidden node \(h \in \mathcal{Y}\), compute the distance \(d_{hp}\) for all \(p \in \mathcal{Y}\) using equation (10) and (11).
8. end while
9. if \(|\mathcal{Y}| = 2\) then
10.   Connect the two remaining nodes in \(\mathcal{Y}\) with an edge and then stop.
11. else if \(|\mathcal{Y}| = 1\) then
12.   Stop
13. end if

\[\Sigma_{\tilde{O}}, \tilde{\Sigma}_{\tilde{O}}\] is also conjugate-symmetric and uniquely Cholesky-decomposable. If we assume Cholesky decomposition of \(\Sigma_{\tilde{O}}\) is \(\Sigma_{\tilde{O}} = M_{\tilde{O}}M_{\tilde{O}}^T\) (\(M_{\tilde{O}}\) is a unique upper-diagonal matrix), we can conclude that:

\[\tilde{M}_{\tilde{O}} = \begin{bmatrix} M_{kk} & \cdots & M_{kL} \\ \vdots & \ddots & \vdots \\ M_{LL} & \cdots & M_{LL} \end{bmatrix}.\]

Then, \(\tilde{M}_{\tilde{O}} = M_{\tilde{O}}\). Given that \(W = M^{-1}\) is also an upper-diagonal matrix, we rewrite \(MW = I_d\) (\(I_d\) is the identity matrix) in blocks form:

\[\begin{bmatrix} M_1 & M_2 \\ M_{\tilde{O}} & \tilde{M}_{\tilde{O}} \end{bmatrix} \begin{bmatrix} W_1 & W_2 \end{bmatrix} = \begin{bmatrix} I_d & I_d \end{bmatrix}.\]

Therefore, we have \(W_{\tilde{O}} = \tilde{M}_{\tilde{O}}^{-1} = \tilde{W}_{\tilde{O}}\). This conclusion helps us to rewrite the whitening process:

\[\begin{bmatrix} (\tilde{I}_1)^T \\ (\tilde{I}_O)^T \end{bmatrix} = \begin{bmatrix} W_1 & W_2 \end{bmatrix} \begin{bmatrix} I_d \\ \tilde{W}_{\tilde{O}} \end{bmatrix} \begin{bmatrix} (\tilde{I}_1)^T \\ (\tilde{I}_O)^T \end{bmatrix},\]

where \((\tilde{I}_1)^T\) and \((\tilde{I}_O)^T\) correspond to measurements of unobserved nodes and \((I_O)^T\) and \((I_{\tilde{O}})^T\) correspond to measurements of observed nodes. Considering (3), we have: \((\tilde{I}_O)^T = \tilde{W}_{\tilde{O}}(I_O)^T = (I_O)^T\).

### C. Proof of Theorem 2

**Proof:** We can rewrite (13) in a block-matrix form:

\[\begin{bmatrix} M_1 & M_2 \\ M_{\tilde{O}} & \tilde{M}_{\tilde{O}} \end{bmatrix} \begin{bmatrix} M_1^H & M_2^H \\ \tilde{M}_{\tilde{O}}^H \end{bmatrix} = \begin{bmatrix} \Sigma_1 & \Sigma_2 \\ \Sigma_3 & \Sigma_4 \end{bmatrix},\]

(14)

where \(\Sigma_4\) is the covariance matrix for current deviations of observed nodes, and \(\Sigma_2\) is the covariance matrix of current deviations between observed nodes and hidden nodes. If current-deviation correlations only exist among observed nodes (Assumption 3), we can assume \(\Sigma_4\) is a dense matrix and \(\Sigma_2 = 0_{(k-1) \times (L-k+1)}, \Sigma_3 = 0_{(L-k+1) \times (k-1)}\).

According to (14), we have:

\[\begin{bmatrix} \tilde{M}_O & \tilde{M}_{\tilde{O}}^H \\ M_2 & \tilde{M}_{\tilde{O}} \end{bmatrix} = \begin{bmatrix} 0_{(k-1) \times (L-k+1)}, \tilde{M}_O \end{bmatrix} = \begin{bmatrix} 0_{(k-1) \times (L-k+1)} \\ 0_{(L-k+1) \times (k-1)} \end{bmatrix},\]

(15)

Then we have: \(M_2 \times \tilde{M}^H = \tilde{M}_O \times M_{\tilde{O}}^H = 0_{(k-1) \times (L-k+1)}\), which implies:

\[M_2 \times \Sigma_4^H \times M_{\tilde{O}}^H = 0_{(k-1) \times (k-1)},\]

\[M_2 \times V \times \Lambda \times V^H \times M_{\tilde{O}}^H = 0_{(k-1) \times (k-1)},\]

\[M_2 \times V \times \Lambda \times (M_2 \times V)^H = 0_{(k-1) \times (k-1)},\]

where \(V\) is a \((L-k+1) \times (L-k+1)\) matrix containing all the eigenvectors of \(\Sigma_4\) and \(\Lambda\) is a diagonal matrix with positive eigenvalues. Since \(M_2 \times V\) and \((M_2 \times V)^H\) are conjugate symmetric and \(\Lambda\) is a diagonal matrix with positive values, we know:

\[M_2 \times V = 0_{(k-1) \times (L-k+1)}\]

(16)

According to the orthogonal property of eigenvectors, we have: \(V \times V^H = \text{Id}_{(L-k+1) \times (L-k+1)}\) (\(\text{Id}\) is the identity matrix). We use \(V^H\) right multiply (16), and finally, we can obtain \(M_2 = 0_{(k-1) \times (L-k+1)}\).

### D. Proof of Feasibility for Structure Learning without Angle

**Proof:** Fig. 10 illustrates the learning process. Firstly, we consider observed nodes set \(\mathcal{O}\) as the current-active set \(\mathcal{Y} = \mathcal{O}\). We include nodes \(a, b, c_1, c_2, d_1, d_2, g \in \mathcal{O}\) to represent all the possibilities of observed nodes.
According to Lemma 2 in Appendix VII-A, we first verify the ability to identify the parent-child relationship of node \( a \) and \( g \), namely, the correctness of Lemma 2 (i). For every \( k \in \mathcal{O} \setminus \{a, g\} \), \( \Phi_{bkg} = d_{ab} - d_{bg} = |Z(a, g)| - |Z(g, g)| \), where the last equality holds by \( Z(a, k) = Z(g, k) \) from Lemma 1. Similarly, the distance between nodes \( a \) and \( g \) is \( d_{ag} = |Z(g, g)| - |Z(a, g)| \) by \( Z(a, a) = Z(a, g) \). Thus, for arbitrary \( k \in \mathcal{O} \setminus \{a, g\} \), we have \( \Phi_{bkg} = d_{bg} = -\Phi_{gak} \), which holds if \( i \) is the parent of \( g \). The above proof is applicable to every parent-child node pair when the child node is a leaf node.

For the other direction of Lemma 2 (i), we prove the contraposition. There exists non-parent-child node pair \( g \) and \( b \), and another observed node \( c_1 \) such that \( \Phi_{bc_1} = d_{bc} - d_{gb} = 2|Z(b, g)| - |Z(g, g)| < 0 \) by \( Z(b, c_1) = Z(g, c_1) \) and \( |Z(b, g)| < |Z(g, g)| \). In general, Lemma 2 (i) holds for our defined distance.

Then, we verify Lemma 2 (ii) for leaf nodes \( c_1 \) and \( l_1 \) in the sibling group. For every \( k \in \mathcal{O} \setminus \{c_1, l_1\} \), we have \( \Phi_{c1l1,k} = d_{c1k} - d_{l1k} = |Z(c_1, c_1)| - |Z(l_1, l_1)| \), where the last equality holds by \( Z(c_1, k) = Z(l_1, k) \). Thus, no matter which \( k \) we choose, \( \Phi_{c1l1,k} \) is a constant. Similarly, the distance between \( c_1 \) and \( l_1 \) is:

\[
\begin{align*}
d_{c1l1} &= |Z(c_1, c_1)| + |Z(l_1, l_1)| - 2|Z(c_1, l_1)|, \\
&\geq |(|Z(c_1, c_1)| - |Z(c_1, l_1)|) - (|Z(l_1, l_1)| - |Z(c_1, l_1)|)|, \\
&\geq |\Phi_{c1l1,k}|.
\end{align*}
\]

The above proof is applicable for every 2 leaf nodes when they are in the same sibling group.

Then, we prove the other direction of Lemma 2 (ii). Consider the contraposition, there exist non-sibling node pair \( b \) and \( c_1 \), and another observed node \( k \in \mathcal{O} \setminus \{b, c_1\} \) such that: \( \Phi_{bc1,k} = |Z(c_1, c_1)| - 2|Z(c_1, k)| < 0 \). Therefore, if \( k \) is varying, \( \Phi_{bc1,k} \) is not a constant. In general, Lemma 2 (ii) holds for our defined distance.

Secondly, using Lemma 2, we find the parent-child node pair \( a \) and \( g \), sibling group \( c_1 \) and \( l_1 \) with a hidden parent \( f_1 \), and a single node \( b \). Without loss of generality, they represent all the possible groups for \( \mathcal{O} \) by RG algorithm in Appendix VII-A.

Subsequently, nodes \( a, b \), and \( f_1 \) represent all possible types to form the new current-active set \( \mathcal{Y}_{new} \). To utilize the induction idea to prove the feasibility of our defined distance in RG algorithm, we only need to prove distances among the new set \( \mathcal{Y}_{new} \) still have the same form defined as \( d_{hh} = |Z(h, h)| + |Z(k, k)| - 2|Z(h, k)|, \forall h, k \in \mathcal{Y}_{new} \). However, there are 3 types of distance: 1) the distance between two nodes in \( \mathcal{Y} \), 2) the distance between a node in \( \mathcal{Y} \) and a detected hidden node, and 3) the distance between 2 detected hidden nodes. Thus, we include another sibling group \( c_2, l_2 \) and their parent \( f_2 \) to form the type 3) distance \( d_{f1,f2} \).

For distance type 1) like \( d_{ab} \), it is directly calculated by the distance definition. For distance type 2), without loss of generality, we consider \( d_{k_1, f_1}, \forall k_1 \in \mathcal{Y} \cap \mathcal{Y}_{new} \), defined as \( d_{k_1, f_1} = d_{k_1, c_1} - d_{c_1, f_1} \). Since \( \Phi_{c1l1,k} \) is a constant, we employ definition in (10), Appendix VII-A to calculate \( d_{c1l1,f1} \):

\[
d_{c1l1,f1} = \frac{1}{2}(\Phi_{c1l1,k} + d_{c1l1}) = |Z(c_1, c_1)| - |Z(c_1, f_1)|,
\]

where the last equality holds by \( Z(c_1, l_1) = Z(c_1, f_1) \). Therefore, we calculate \( d_{k_1, f_1}, \forall k_1 \in \mathcal{Y} \cap \mathcal{Y}_{new} \):