Learning Topology of Distribution Grids using only Terminal Node Measurements

Deepjyoti Deka†, Scott Backhaus†, and Michael Chertkov†
†Los Alamos National Laboratory, USA
Email: deepjyoti@lanl.gov, backhaus@lanl.gov, chertkov@lanl.gov

Abstract—Distribution grids include medium and low voltage lines that are involved in the delivery of electricity from substation to end-users/loads. A distribution grid is operated in a radial/tree-like structure, determined by switching on or off lines from an underlying loopy graph. Due to the presence of limited real-time measurements, the critical problem of fast estimation of the radial grid structure is not straightforward. This paper presents a new learning algorithm that uses measurements only at the terminal or leaf nodes in the distribution grid to estimate its radial structure. The algorithm is based on results involving voltages of node triplets that arise due to the radial structure. The polynomial computational complexity of the algorithm is presented along with a detailed analysis of its working. The most significant contribution of the approach is that it is able to learn the structure in certain cases where available measurements are confined to only half of the nodes. This represents learning under minimum permissible observability. Performance of the proposed approach in learning structure is demonstrated by experiments on test radial distribution grids.

Index Terms—Distribution Networks, Power Flows, Tree learning, Voltage measurements, Missing data, Complexity

I. INTRODUCTION

The power grid is operationally divided hierarchically into transmission and distribution grids. While the transmission grid connects the generators and includes high voltage lines, the distribution grid comprises of medium and low voltage lines that connect the distribution substation to the end users/loads. Aside from low voltages, distribution grids are structurally distinguished from the transmission side by their operational radial structure. A distribution grid is operated as a tree with a substation placed at the root node/bus that is connected via intermediate nodes and lines to the terminal end nodes/households. This radial operational structure is derived from an underlying loopy network by switching on and off some of the lines (network edges) [1]. The specific structure may be changed from one radial configuration to another, by reversing the switch statuses. Fig. 1 presents an illustrative example of a radial distribution grid derived from an underlying graph.

Historically, the distribution grid has had limited presence of real time devices on the lines, buses and feeders [2]. Due to this, real-time monitoring of the operating radial structure and the states of the resident buses on the distribution side is not straightforward. These estimation problems, previously neglected, have become of critical importance due to introduction of new devices like batteries and electric vehicles, and intermittent generation resources like rooftop solar panels. Optimal control in today’s distribution grid requires fast topology and state estimation, often from limited real-time measurements. In this context, it needs to be mentioned that smart meters, micro-PMUs [3], frequency measurements devices (FNETs) [4] and advanced sensors (internet-of-things) capable of reporting real-time measurements are being deployed on the distribution side. However, in the current scenario, such devices are often limited to households/terminal nodes as their primary purpose for installation is services like price controllable demand and load monitoring. A majority of the intermediate lines and buses that connect the distribution substation to the terminal nodes do not have real-time measurement devices and are thus unobserved in terms of their structure and state. This hinders real-time topology and state estimation.

This paper is aimed at developing a learning framework that is able to overcome the lack of measurements at the intermediate nodes. Specifically, the primary goal of our work is to propose an algorithm to learn the operating radial topology using only real-time voltage magnitude measurements from the terminal nodes. The reliance only on measurements from terminal nodes is crucial as it makes our work applicable to realistic deployment of smart devices in today’s distribution grids. Further, our learning algorithm is able to tolerate a much higher fraction of missing nodes (with unavailable data) compared to prior work in this area. Our approach is based on provable relations in voltage magnitudes of triplets and pairs of terminal nodes that are used to discover the operational edges iteratively. Computationally, the algorithm has polynomial complexity in the number of nodes in the system.

A. Prior Work

Topology learning in radial distribution grids has received attention in recent times. Learning techniques in this area vary, primarily based on the operating conditions and measurements available for estimation. The authors of [5] use a Markov random field model for nodal phases to identify faults in the grids. [6] uses conditional independence tests to identify the grid topology in radial grids. [7] uses signs of elements in the inverse covariance matrix of nodal voltages to learn the operational topology. Signature/envelope based identification of topology changes is proposed in [8]. Such comparison based schemes are used for parameter estimation
in [9], [10]. In contrast with nodal voltage measurements, line flow measurements are used in a maximum likelihood based scheme for topology estimation in [11]. In previous work [1], [12], authors have analyzed topology learning schemes that rely on trends in second moments of nodal voltage magnitudes. Further, a spanning tree based topology learning algorithm is proposed in [13]. This line of work [1], [12], [13] is close in spirit to machine learning schemes [14], [15] developed to learn the structure of general probabilistic graphical models [16]. A major limitation of the cited literature is that they assume measurement collection at most, if not all, nodes in the system. To the best of our knowledge, work that discuss learning in the presence of missing/unobserved nodes [12], [13] assume missing nodes to be separated by greater than two hops in the grid. As discussed earlier, distribution grids often have real-time meters only at terminal nodes and none at intermediate nodes that may be adjacent (one hop away).

B. Contribution of This Work

In this paper, we discuss topology learning in the radial distribution grid when measurements are limited to real-time voltage readings at the terminal nodes (end-users) alone. All intermediate nodes are unobserved and hence assumed to be missing nodes. We analyze voltages in the distribution grid using a linear lossless AC power flow model [1], [12] that is analogous to the popular [17], [18] LinDistFlow equations. For uncorrelated fluctuations of nodal power consumption, we construct functions of voltage magnitudes at a pair or triple of terminal nodes such that their values depend on the edges that connect the group of nodes. These functions provide the necessary tool to develop our learning scheme that iteratively identifies operational edges from the leaf onward to the substation root node. We discuss the computational complexity of the learning scheme and show that it is a third order polynomial in the number of nodes. In comparison to existing work, our approach is able to learn the topology and thereby estimate usage statistics in the presence of much greater fraction of missing/unobserved nodes. In fact, in limiting configurations, our learning algorithm is able to determine the true structure and statistics even when half of the nodes are unobserved/missing. We demonstrate the performance of our algorithm through experiments on test distribution grids.

The rest of the manuscript is organized as follows. Section II introduces notations used in the paper and describes the grid topology and power flow models used for analysis in later sections. Section III mentions assumptions made and describes important properties involving voltage measurements at terminal nodes. Our algorithm to learn the operating radial structure of the grid is discussed in Section IV with detailed examples. We detail the computational complexity of the algorithm in Section V. Simulation results of our learning algorithm on test radial networks are presented in Section VI. Finally, Section VII contains conclusions and discussion of future work.

II. DISTRIBUTION GRID: STRUCTURE AND POWER FLOWS

Radial Structure: We denote the underlying graph of the distribution grid by the graph $G = (V, E)$, where $V$ is the set of buses/nodes and $E$ is the set of all undirected lines/edges. We term nodes by alphabets (a, b, ...). An edge between nodes a and b is denoted by (ab). An illustrative example is given in Fig.1. The operational grid consisting of one tree $T$ with nodes $V^T$ and operational edge set $E^T \subset E$ as shown in Fig.1. Our results are immediately extended to the case $K > 1$. In tree $T$, $P_a$ denote the set of edges in the unique path from node $a$ to the root node (reference bus). A node c is called a descendant of node a if the path from node c to the root passes through a, i.e., $P_c \subset P_a$. We use $D_a$ to denote the set of descendants of a and include node a in $D_a$ by definition. If edge $(ac) \in E^T$ and c is a descendant of a, we term a as parent and c as its child node. Further, as discussed in the Introduction, terminal nodes/leaf nodes in the tree represent end-users or households that are assumed to be equipped with real-time nodal meters. The remaining intermediate nodes (not terminal nodes) are not observed and hence termed as missing nodes. These definitions are illustrated in Fig1. Next we describe the notation used in the power flow equations.

Power Flow Models: We use $z_{ab} = r_{ab} + i\theta_{ab}$ to denote the complex impedances of line (ab) $(i^2 = -1)$ where $r_{ab}$ and $\theta_{ab}$ represent the line resistance and reactance respectively. Let real valued scalars, $v_a$, $\theta_a$, $p_a$ and $q_a$ denote the voltage magnitude, voltage phase, active and reactive power injection respectively at node a. At each node a, Kirchhoff’s law of power flow relate them as follows:

$$P_a = p_a + iq_a = \sum_{b,(ab) \in E^T} v_a^2 - v_a v_b \exp(i\theta_a - i\theta_b)$$

Note that Eq. (1) is nonlinear and non-convex. Under realistic assumption that losses of both active and reactive power losses on each line of tree $T$ is small, we can neglect second order terms in Eq. (1) to achieve the following linearized form [1], [7], [12]:

![Fig. 1. Radial distribution grid tree with substation/root node colored in red. Dotted grey lines represent open switches. Terminal leaf nodes (d) represent end-users from which real-time measurements are collected. The intermediate missing nodes (b) are unobserved. Here, nodes a and c are descendants of node a. Dashed lines represent the paths from nodes a and d to the root node.](image-url)
\[ p_a = \sum_{b \in \delta(a)} \left( b_{ab}(\theta_a - \theta_b) + g_{ab}(v_a - v_b) \right), \quad (2) \]
\[ q_a = \sum_{b \in \delta(a)} \left( -g_{ab}(\theta_a - \theta_b) + b_{ab}(v_a - v_b) \right) \quad (3) \]
where \( g_{ab} \) is the resistance between nodes \( a \) and \( b \), \( b_{ab} \) is the reactance, and \( v_a \) and \( \theta_a \) are the voltage magnitude and phase of node \( a \) respectively at the non-substation nodes.

As shown in Eqs. (2), (3), which are equivalent to the LinDistFlow equations for power flow in distribution grid [17]–[19], when deviations in voltage magnitude are assumed to be small. Similar to LinDistFlow model, Eqs. (2), (3) are lossless with sum power equal to zero (\( \sum_{a \in V} P_a = 0 \)). Further, note that both active and reactive power injections are functions of difference in voltage magnitudes and phases of neighboring nodes. Thus, the analysis of the system can be reduced by one node termed as reference node with voltage magnitude and phase at all other nodes being measured relative to this node. In our case, we take the substation/root node as the reference node with voltage magnitude 1 and phase 0.

In our case, we take the substation/root node as the reference node with voltage magnitude 1 and phase 0 respectively. The reference node’s injection also balances the reference node with voltage magnitude 1 and phase 0 such that

\[ H^{-1}_{1/s}(a,b) = \sum_{(cd) \in \mathcal{P}_a \cap \mathcal{P}_b} r_{cd} \]

We term this as the Linear Coupled Power Flow (LC-PF) model where \( p, q \) are the vectors of real power, reactive power injections, voltage magnitudes and phase angles respectively at the non-substation nodes. \( H_{1/s} \) and \( H_{1/s}^{-1} \) are the reduced weighted Laplacian matrices for tree \( T \) where reciprocal of resistances and reactances are used respectively as edge weights. The reduction is achieved by removing the row and column corresponding to the reference bus in the original weighted Laplacian matrix. We denote the mean of a random vector \( X \) by \( \mu_X = \mathbb{E}[X] \). For two random vectors \( X \) and \( Y \), the covariance matrix is denoted by \( \Omega_{XY} = \mathbb{E}[(X - \mu_X)(Y - \mu_Y)^T] \). Using Eq. (5), we can relate the means and covariances of voltage magnitudes with those of active and reactive injection as follows:

\[ \mu_v = H^{-1}_{1/s} \mu_p + H^{-1}_{1/s} \mu_q \]
\[ \Omega_v = H^{-1}_{1/s} \Omega_p H^{-1}_{1/s} + H^{-1}_{1/s} \Omega_q H^{-1}_{1/s} + H^{-1}_{1/s} \Omega_p H^{-1}_{1/s} \Omega_{pq} H^{-1}_{1/s} + H^{-1}_{1/s} \Omega_{qp} H^{-1}_{1/s} \]
\[ (7) \]

Using these statistical quantities, we discuss useful identities that arise in radial distribution grids in the next section that form the basis of our learning algorithms.

III. PROPERTIES OF VOLTAGE MAGNITUDES IN RADIAL GRIDS

First, we make the following assumption regarding statistics of power injections at the non-substation grid nodes, under which our results hold.

**Assumption 1:** Fluctuations of active and reactive power at different nodes are uncorrelated. Thus, \( \forall a \neq b \) non-substation nodes, \( \Omega_p(a,b) = \Omega_q(a,b) = \Omega_{pq}(a,b) = 0 \).

As considered in prior literature [7], [12], this assumption is reasonable over short-time intervals where fluctuations in nodal power usage at households/terminal nodes are independent and hence uncorrelated. Intermediate nodes that do not represent end-users are uncorrelated if they have independent usage patterns. Specifically, for intermediate nodes involved in separation of power into downstream lines and without any major nodal usage, the net power injection is contributed by leakage or device losses and hence uncorrelated from the rest. Note that Assumption 1 does not specify the class of distributions that can model individual node’s power injection. It is applicable when nodal injections are negative (loads), positive (due to local generation) or a mixture of both. In future work, we will relax this assumption and discuss learning in the presence of positively correlated end-user injection profiles.

Next, we mention an analytical statement relating the inverse of the reduced Laplacian matrix for a radial graph that we use in our later results.

**Lemma 1.** [11], [20] The reduced weighted Laplacian matrix \( H^{-1}_{1/s} \) for tree \( \mathcal{T} \) satisfies

\[ H^{-1}_{1/s}(a,b) = \sum_{(cd) \in \mathcal{P}_a \cap \mathcal{P}_b} r_{cd} \]

In other words, the \((a,b)\)th entry of \( H^{-1}_{1/s}(a,b) \) is equal to the sum of line resistances of edges common to paths from node \( a \) and \( b \) to the root. For example, in Fig. 1, \( H^{-1}_{1/s}(a,d) = r_{be} + r_{ed} \). Using Eq. (8), it follows immediately that if node \( b \) is the parent of node \( a \), then \( \forall c \)

\[ H^{-1}_{1/s}(a,c) - H^{-1}_{1/s}(b,c) = \begin{cases} r_{ab} & \text{if node } c \in \mathcal{D}_a \\ 0 & \text{otherwise} \end{cases} \]

Next, consider the function \( \phi_{ab} \) defined over two nodes \( a \) and \( b \) as \( \phi_{ab} = \mathbb{E}[(v_a - \mu_a)^2 - (v_b - \mu_b)^2] \). \( \phi_{ab} \) represents the variance of the difference in voltage magnitudes at nodes \( a \) and \( b \). Using Eq. (7) we can write \( \phi_{ab} \) in terms of power injection statistics in tree \( \mathcal{T} \) as follows

\[ \phi_{ab} = \Omega_v(a,a) - 2\Omega_q(a,b) + \Omega_q(b,b) \]
\[ = \sum_{d \in \mathcal{T}} (H^{-1}_{1/s}(a,d) - H^{-1}_{1/s}(b,d))^2 \Omega_p(d,d) + (H^{-1}_{1/s}(a,d) - H^{-1}_{1/s}(b,d))^2 \Omega_q(d,d) + 2(H^{-1}_{1/s}(a,d) - H^{-1}_{1/s}(b,d))(H^{-1}_{1/s}(a,d) - H^{-1}_{1/s}(b,d)) \Omega_{pq}(d,d) \]
\[ (11) \]

Note that Lemma 1 and Eq. (9) can be inserted in Eq. (11) to simply it. In fact, doing so lets us derive properties of \( \phi \) for terminal nodes in tree \( \mathcal{T} \) as discussed next.

**Theorem 1.** Let node \( b \) be the parent of nodes \( a \) and \( c \) in \( \mathcal{T} \) such that \((ab)\) and \((bc)\) are operational edges (see Fig. 2(a)). Then

\[ \phi_{ab} = \sum_{d \in \mathcal{D}_a} r_{ad}^2 \Omega_p(d,d) + x_{ab}^2 \Omega_q(d,d) + 2r_{ad}x_{ab} \Omega_{pq}(d,d) + \sum_{d \in \mathcal{D}_c} r_{bc}^2 \Omega_p(d,d) + x_{bc}^2 \Omega_q(d,d) + 2r_{bc}x_{bc} \Omega_{pq}(d,d) \]
\[ + \phi_{bc} \]
\[ (12) \]
reactance respectively on lines on the path from node a to
operational lines whose impedances appear on the right side
voltages at three terminal nodes.

The next theorem gives a result that relates the
equality in Eq. (13) is true only when
above condition can be checked for each potential parent
node \( k \), for each node \( a \in \mathcal{D}_k \). Using this in the formula
for \( \phi_{ac} \) in Eq. (11) gives us the relation. The equality
\( \phi_{ac} = \phi_{ab} + \phi_{bc} \) is verified by plugging values in Eq. (11)
for \( \phi_{ab} \), \( \phi_{bc} \) and \( \phi_{ac} \).

A few points are in order. First, note that the only
operational lines whose impedances appear on the right side
of Eq. (12) are \( \{ab\} \) and \( \{bc\} \). For the special case where \( a \) and \( c \) are terminal nodes with parent \( b \), the relation reduces to the following:

\[
\phi_{ac} = r_{ab} \Omega_p(a,a) + x_{ab}^2 \Omega_q(a,a) + 2 r_{ab} x_{ab} \Omega_{pq}(a,a) + r_{bc}^2 \Omega_p(c,c) + x_{bc}^2 \Omega_q(c,c) + 2 r_{bc} x_{bc} \Omega_{pq}(c,c)
\]

If the covariance of injections at \( a \) and \( b \) are known, the
above condition can be checked for each potential parent
node \( b \) in linear time to identify the true parent. Second, the
equality in Eq. (15) is true only when \( \mathcal{P}_a \cap \mathcal{P}_b = \mathcal{P}_b \). It is
replaced by a strict inequality for other configurations that we
omit discussing as they are outside the scope of our learning
conditions. The next theorem gives a result that relates the
voltages at three terminal nodes.

**Theorem 2.** Let terminal nodes \( a \) and \( b \) have common parent
node \( k_1 \). Let \( c \) be another terminal node such that \( c, k_1 \in \mathcal{D}_k \), and \( \mathcal{P}_a \cap \mathcal{P}_c = \mathcal{P}_k \) for some intermediate node \( k_2 \) (see
Fig. 2(b)). Let \( r_{k_2}^1 \) and \( x_{k_2}^1 \) denote the sum of resistance and reactance respectively on lines on the path from node \( a \) to node \( k_2 \), i.e., \( r_{k_2}^1 = \sum r_{e,f} \), \( x_{k_2}^1 = \sum x_{e,f} \). Define \( r_{k_2}^2 \), \( x_{k_2}^2 \) etc. in
the same way. Then

\[
\phi_{ac} - \phi_{bc} = \Omega_p(a,a) \left( (x_{k_2}^1)^2 - (r_{k_2}^1)^2 \right) + \Omega_q(a,a) \left( (x_{k_2}^1)^2 - (r_{k_2}^1)^2 \right) + \Omega_{pq}(a,a) \left( (x_{k_2}^1)^2 - (r_{k_2}^1)^2 \right)
+ \Omega_p(b,a) \left( (x_{k_2}^1)^2 - (r_{k_2}^1)^2 \right) + \Omega_q(b,a) \left( (x_{k_2}^1)^2 - (r_{k_2}^1)^2 \right) + \Omega_{pq}(b,a) \left( (x_{k_2}^1)^2 - (r_{k_2}^1)^2 \right)
+ \Omega_q(b,c) \left( (x_{k_2}^1)^2 - (r_{k_2}^1)^2 \right) + \Omega_{pq}(b,c) \left( (x_{k_2}^1)^2 - (r_{k_2}^1)^2 \right).
\]

**Proof:** As \( a \) and \( b \) are terminal nodes with same parent
node \( k_1 \), for each node \( d \neq a \neq b \), the paths from \( d \), \( a \) and \( b \) to
the root follow \( \mathcal{P}_a \cap \mathcal{P}_d = \mathcal{P}_b \cap \mathcal{P}_d \). Using Lemma 1, we thus
have \( \forall d \neq a, b, H_{1/r}(a,d) = H_{1/r}(b,d) \) and

\[
H_{1/r}(a,d) - H_{1/r}(a,c,d) = H_{1/r}(b,d) - H_{1/r}(b,c,d)
\]

As \( k_1 \) and \( c \) are descendants of node \( k_2 \) and \( \mathcal{P}_k \cap \mathcal{P}_c = \mathcal{P}_{k_2} \),

\[
H_{1/r}(a,d) - H_{1/r}(a,c) = r_{k_2}^1, H_{1/r}(b,d) - H_{1/r}(b,c) = r_{k_2}^1
\]

Further, using Eqs. (17) (18), we get

\[
H_{1/r}(a,b) - H_{1/r}(a,c) = H_{1/r}(a,b) - H_{1/r}(a,b) = r_{k_2}^1
\]

where \( r_{k_2}^1 \), \( r_{k_2}^2 \) etc. are defined in the statement of the
theorem. Similar relations can be written for \( H_{1/r} \) terms as
well. We now expand \( \phi_{ac} \) and \( \phi_{bc} \) using Eq. (11). Using
Eq. (16) (18) (19) in the expression for \( \phi_{ac} - \phi_{bc} \) gives Eq. (15).

Observe that the lines whose impedances appear on the
right side of Eq. (15) are \( \{ak_1\}, \{bk_1\} \) and the ones on the path
from node \( k_1 \) to \( k_2 \). If this path is known till the penultimate
node \( k \) before \( k_2 \) (see Fig. 2(b)), then Eq. (15) can be used to
learn edge \( \{kk_2\} \) through a linear search among candidate
nodes for \( k_2 \). In the next section, we discuss the use of this
relation to learn the path from terminal pairs with common
parent (here \( a \) and \( b \)) to the root iteratively. Further, note
that the value of \( \phi_{ac} - \phi_{bc} \) is independent of injections at
intermediate nodes and at terminal node \( c \) as long as \( c \) is a
descendant of \( k_2 \). Thus, Eq. (15) only helps identify \( c \)’s
relative position in the graph with respect to \( k_2 \). As shown
in the next section, we are able to locate \( c \)’s parent using a
post-order node traversal [21] in the grid graph.

**IV. ALGORITHM TO LEARN TOPOLOGY USING TERMINAL
NODE MEASUREMENTS**

Using the relations described in the previous section,
we discuss our algorithm to learn the operational
topology of the radial distribution grid. As mentioned in
the Introduction, voltage measurements are available only at
the terminal/leaf nodes. The observer also has access to power
injection statistics (variances) at the terminal nodes. These
statistics are either known from historical data or computed
empirically from power injection measurements collected
at the terminal nodes. All intermediate nodes are missing
and their voltage measurements and injection statistics are not
observed/available. Further, we assume that impedances of
all lines (open or operational) in \( E \) for the underlying loopy
graph \( \mathcal{G} \) are available as input. The task of the learning
algorithm is to identify the set \( E' \) of operational edges in the
radial grid \( \mathcal{G} \). The substation root node in tree \( \mathcal{G} \) is assumed to
be connected to a single intermediate node. Otherwise each
sub-tree connected to the substation node can be considered
as a disjoint tree. First, we make the following restriction
regarding the degree of missing intermediate nodes.

**Assumption 2:** All missing intermediate nodes are
assumed to have a degree greater than 2.

This assumption is necessary as without it, the solution
to the topology learning problem will not be unique. As
an example, consider the test distribution grid given in
Fig. 3 where the path from leaf node \( a \) to node \( d \) passes
through two intermediate nodes \( b \) and \( c \) of degree 2 each.
Both configuration \( A \) (operations edges \( (ab), (bc), (cd) \)) and
configuration \( B \) (operations edges \( (ac), (cb), (bd) \)) are fea-
sible operational topologies if voltage and power injection
measurements are available only at the terminal node \( a \).
In other words, different values of injection covariances at
nodes \( b \) and \( c \) can be taken such that relations between voltage
and injections are satisfied in either configuration. Similar
assumptions for uniqueness in learning general graphical
models are mentioned in [22]. For radial configurations that
only parents with two or more leaf nodes as children
are used to keep track of the current stage of the learnt graph. There are three major stages in the edge learning process in Algorithm 1.

**Working:** Algorithm 1 learns the structure of operational
tree \( T \) iteratively from leaf node pairs up to the root. Set
\( L \) represents the current set of terminal nodes/leaves with
unknown parents, while set \( M \) represents the set of all
missing intermediate nodes. Note that for each node \( a \), \( par_a \)
denotes its identified parent while \( des_a \) denotes a set of two
leaf nodes that are its descendants. \( \Phi \) represents the empty
set. The two sets (\( par \) and \( des \) ) are used to keep track of
the current stage of the learnt graph. There are three major
stages in the edge learning process in Algorithm 1.

**First,** we build edges between leaf nodes pairs \( a,c \in L \)
to their common missing parent node \( b \) in Steps 3 to 6. Here
the relation for \( \phi_{ab} \) derived in Theorem 1 and Eq. (14)
is used to identify the true missing parent in set \( M \). Note
that only parents with two or more leaf nodes as children
can be identified at this stage. Parents that have at most one
leaf node as child (other children being missing intermediate
nodes) are not identified by this check.

**Second,** we remove nodes with discovered parents from
the set of leaf nodes \( L \) then iteratively discover the edges
between intermediate nodes in Steps 10 to 24. At each
stage of the iteration, \( M_2 \) denotes the set of intermediate
nodes with unknown parents whose descendants (\( des \) ) were
discovered in the previous iteration, while \( M_1 \) denotes the
set of all intermediate nodes with unknown parent (\( par \) ) and
known descendants. Clearly \( M_2 \subset M_1 \). The parent \( k_2 \) of each
node \( k \) in \( M_2 \) is identified by using relation Eq. (15) in
Theorem 2 for its descendants \( a \) and \( b \). There are two cases
considered for candidate parent \( k_2 \): (A) \( k_2 \) belongs to \( M_1 \)
and has known descendant \( c \) (Step 15), and (B) \( k_2 \) belongs
to \( M_1 \) and has no discovered descendant (Step 18). For
the second case, the algorithm tries to find some leaf node
\( c \) that is potentially an undiscovered descendant of \( k_2 \)
by checking for the relation \( \phi_{ac} - \phi_{bc} \) specified in Eq. (15). If
the relation holds, node \( k_2 \) is denoted as the parent of \( k_1 \). As
it is not clear if leaf \( c \) is an immediate descendant (child) of
\( k_2 \), no edge to \( c \) is added. The iterations culminate when no
additional edge between intermediate nodes is discovered.

**Third,** in Steps 29 to 36 the parents of the remaining leaf
nodes in \( L \) are identified. Note that Eq. (15) also holds for
non-immediate descendant \( c \) of \( k_2 \), hence it cannot be used
directly to identify \( c \)'s location. To overcome this, for each
\( c \) in \( L \), we first check at descendants of \( k_2 \) before \( k_2 \). This
is ensured by creating a post-order traversal list \( W \) in
Step 28. In post-order traversal \( W \), each potential parent
is reached only after all its descendants have been reached. The
node \( k_2 \) which satisfies Eq. (15) in Step 32 now gives the true
parent of leaf \( c \). The algorithm terminates after searching for
all leaves in \( L \).

**Algorithm 1 Topology Learning using Terminal Node Data**

**Input:** Injection covariances \( \Omega_{\mu \mu}, \Omega_{\mu v}, \Omega_{v v} \) at terminal nodes
\( L \), Missing node set \( M = V^3 - L \), \( m \) voltage magnitude
observations \( v \) for nodes in \( L \), set of all edges \( \mathcal{E} \) with line
impedances.

**Output:** Operational Edge set \( \mathcal{E}^T \).

1: \( \forall a \in M \), compute \( \phi_{ab} = \mathbb{E}[(v_a - \mu_a) - (v_b - \mu_b)]^2 \)
2: \( \forall a \in V^3 \), define \( par_a \leftarrow \Phi, \; des_a \leftarrow \Phi \)
3: for all \( a \in L \) do
4: if \( par_a = \Phi \; \&\; \exists c \in L, c \notin \mathcal{M} \) s.t. \( \phi_{ab}, c \) satisfy Eq. (14) then
5: \( \mathcal{E}^T \leftarrow \mathcal{E}^T \cup \{ \{ab\}, \{bc\} \} \)
6: \( par_a \leftarrow b, \; par_c \leftarrow b, \; des_b \leftarrow a, c \)
7: end if
8: end for
9: \( L \leftarrow \{ a : a \in L, \; par_a = \Phi \}, \; tp \leftarrow 1, \mathcal{M}_1 \leftarrow \Phi \)
10: while \( tp > 0 \) do
11: \( M_2 \leftarrow \{ k_1 : k_1 \in M_1 - \mathcal{M}_1, \; par_{k_1} = \Phi, \; des_{k_1} \neq \Phi \} \)
12: \( \mathcal{M}_1 \leftarrow \{ k_1 : k_1 \in M, \; par_{k_1} = \Phi, \; des_{k_1} \neq \Phi \} \)
13: for all \( k \in \mathcal{M}_1 \) with \( a,b \in des_{k_2} \) do
14: \( k_1 \leftarrow par_a \)
15: if \( \exists k_2 \in \mathcal{M}_1 \) with \( \mathcal{M}_2 \cap \{ k, k_2 \} \neq \Phi \) with \( c \in des_{k_2} \), s.t.
\( \phi_{ac} - \phi_{bc} \) satisfy Eq. (15) then
16: \( \mathcal{E}^T \leftarrow \mathcal{E}^T \cup \{ \{kk\}_2 \}, \; par_k \leftarrow k_2 \)
17: else
18: if \( k \in \mathcal{M}_1, \exists k_2 \in \mathcal{M} - \mathcal{M}_1, c \in L, \; \phi_{ac} - \phi_{bc} \) satisfy
Eq. (15) then
19: \( \mathcal{E}^T \leftarrow \mathcal{E}^T \cup \{ \{kk\}_2 \}, \; par_k \leftarrow k_2, \; des_{k_2} \leftarrow des_k \)
20: end if
21: end if
22: end for
23: \( tp \leftarrow |\{ k_1 : k_1 \in \mathcal{M}_1, \; par_{k_1} \neq \Phi \}| \)
24: end while
25: if \( |\mathcal{M}_1| = 1 \) then
26: Join \( k \in \mathcal{M}_1 \) to root
27: end if
28: Form a post-order traversal node set \( W \) using \( par_a \) for
\( a : \forall a \in M, \; des_a \neq \Phi \)
for all \( c \in L \) do
29: for \( j \leftarrow 1 \) to \(|W|\) do
30: \( k_2 \leftarrow W(j) \) with \( a,b \in des_{k_2}, k_1 \leftarrow par_a \)
31: if \( \phi_{ac} - \phi_{bc} \) satisfy Eq. (15) then
32: \( \mathcal{E}^T \leftarrow \mathcal{E}^T \cup \{ \{kk\}_2 \}, \; W \leftarrow W \setminus \{k_2\}, j \leftarrow |W| \)
33: end if
34: end for
35: end for
36: end for

Fig. 3. Distribution grid tree where terminal node \( a \) is connected to node \( d \) via two unknown intermediate nodes of degree 2. Either configuration \( d, c, b, a \) or \( d, b, c, a \) for the intermediate nodes is a feasible structure given
no measurements at nodes \( b \) and \( c \).
nodes \( N \) node pairs belong to substation connects to the node with no parent. Finally, positions of leaf nodes are discovered. In the next step, edges between intermediate nodes are discovered iteratively. Finally, positions of leaf nodes with no leaf sibling are discovered. The second graph has the structure of a line graph with one additional leaf node connected to each intermediate node on the line. In either graph, as the number of nodes increases, the fraction of missing nodes increases to 50%. In other words, configurations exist such that Algorithm 1 is able to learn the grid structure with approximately half the nodes missing/unobserved.

This is crucial because 50% fraction of missing nodes represents the threshold beyond which state estimation is not possible even in the presence of knowledge of the operational topology. This is because the LC-PF Eqs. (4) cannot be solved if less than half the nodes are observed.

V. COMPUTATIONAL COMPLEXITY AND PERFORMANCE

Consider the original loopy graph \( \mathcal{G} \) to be complete (all node pairs belong to \( \mathcal{E} \)). We compute the computational complexity of operational tree in terms of the number of nodes \( N = |V|^2 \).

Algorithm Complexity: To measure the computational complexity, we focus on the iteration steps. Let the number of leaves in tree \( \mathcal{T} \) be \( l_0 \). Detecting edges between leaf pairs and their common parent (Steps [3] to [6]) takes \( O(l_0^2(N-l_0)) \) comparisons, which takes the form \( O(N^3) \) in the worst case. Next we analyze the complexity of identifying edges between intermediate nodes. Let the number of ‘new’ intermediate nodes with unknown parents (set \( M_2 \)) in each iteration of (Steps [10] to [24]) be \( l_i \). Each ‘new’ node is compared with the set of all intermediate nodes with unknown parents (\( M_1 \supseteq \mathcal{M}_2 \)) first. As addition of a ‘new’ node leads to removal of its children (\( \geq 1 \)) from \( M_1 \), the size of \( M_1 \) never increases more than its initial value (\( \leq l_0/2 \)) giving this step a worst case complexity of \( O(l_i l_0) \). Comparison of nodes in \( M_1 \) with missing nodes in \( \mathcal{M} - M_1 \) and leaf nodes in \( \mathcal{L} \) (Step [18] has worst case complexity \( O(l_i l_0(N-l_0)) \). Using \( l_0 \) as \( O(N) \) and \( \sum_i l_i = O(N) \), the computational complexity of all iterations in Steps [10] to [24] is thus \( O(N^3) \). Finally composing the post-order tree traversal and checking at node location \( O(N) \) for each leaf \( O(N) \) has complexity less than or equal to \( O(N^3) \). The overall complexity of the algorithm is thus \( O(N^3) \).

Maximum Missing Node Fraction: Consider the two radial graphs given in Fig. 5 that satisfy the condition in Assumption 2. The first is a binary tree of depth \( d \). All nodes till depth \( d - 1 \) are missing. The second graph has the structure of a line graph with one additional leaf node connected to each intermediate node on the line. In either graph, as the number of nodes increases, the fraction of missing nodes increases to 50%. In other words, configurations exist such that Algorithm 1 is able to learn the grid structure with approximately half the nodes missing/unobserved.

This is crucial because 50% fraction of missing nodes represents the threshold beyond which state estimation is not possible even in the presence of knowledge of the operational topology. This is because the LC-PF Eqs. (4) cannot be solved if less than half the nodes are observed.

VI. EXPERIMENTS

Here we demonstrate performance of Algorithm 1. We consider a radial network \( \mathcal{G} \) (Fig. 6) with 20 nodes and one substation. Out of those 20 nodes, 12 are terminal/leaf nodes while 8 are intermediate nodes which are missing. In each of our simulation runs, we construct complex power injection samples at the non-substation nodes from a multivariate Gaussian distribution that is uncorrelated between different nodes as per Assumption 1. Next we use LC-PF Eqs. (5) to generate nodal voltage magnitude measurements. To understand the performance of our algorithm, we introduce 30 additional edges (at random) to construct the loopy edge set \( \mathcal{E} \). The additional edges are given impedances comparable to those of operational lines. The input to the observer consists of voltage magnitude measurements and injection statistics at the terminal nodes and impedances of all the lines within the loopy edge set \( \mathcal{E} \). The average fraction errors (number of errors/size of the operational edge set) in reconstructing the grid structure is shown in Fig. 7(a) for different sizes of terminal voltage magnitude measurements used. Different curves in the plot depict different values of tolerances \( \tau_1 \) and \( \tau_2 \) to check the correctness of Eq. (14) and Eq. (15) in Algorithm 1. Note that the average fractional errors decreases steadily with increase in the number of measurements as empirical second moment statistics used in Algorithm 1 get more accurate. The values of tolerance to achieve the most accurate results are selected manually. We plan to develop a theoretical selection of the optimal tolerance values in future work.
sample complexity of the learning algorithm and optimizing the selection of tolerance values based on the number of samples are directions of our future research in this area. Further, we plan to expand this learning regime to cases with correlated injections.

REFERENCES

[1] D. Deka, S. Backhaus, and M. Chertkov, “Structure learning and statistical estimation in distribution networks - part i,” arXiv preprint arXiv:1501.04131, 2015.
[2] R. Hoffman, “Practical state estimation for electric distribution networks,” in Power Systems Conference and Exposition, 2006. PSCE’06. 2006 IEEE PES. IEEE, 2006, pp. 510–517.
[3] A. von Meier, D. Culler, A. McEachern, and R. Arghandeh, “Micro-synchrophasors for distribution systems,” in Innovative Smart Grid Technologies Conference (ISGT). IEEE, 2014, pp. 1–5.
[4] 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,” Power Systems, IEEE Transactions on, vol. 20, no. 4, pp. 1914–1921, 2005.
[5] M. He and J. Zhang, “A dependency graph approach for fault detection and localization towards secure smart grid,” Smart Grid, IEEE Transactions on, vol. 2, no. 2, pp. 342–351, 2011.
[6] D. Deka, S. Backhaus, and M. Chertkov, “Estimating distribution grid topologies: A graphical learning based approach,” in Power Systems Computation Conference (accepted), 2016.
[7] S. Bolognani, N. Bof, D. Michelotti, R. Muraro, and L. Schenato, “Identification of power distribution network topology via voltage correlation analysis,” in Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on. IEEE, 2013, pp. 1659–1664.
[8] G. Cavraro, R. Arghandeh, A. von Meier, and K. Poolla, “Data-driven approach for distribution network topology detection,” arXiv preprint arXiv:1504.00724, 2015.
[9] J. Peppanen, J. Grimaldo, M. J. Reno, S. Grijalva, and R. G. Harley, “Increasing distribution system model accuracy with extensive deployment of smart meters,” in PES General Meeting—Conference & Exposition, 2014 IEEE. IEEE, 2014, pp. 1–5.
[10] J. Peppanen, M. J. Reno, M. Thakkar, S. Grijalva, and R. G. Harley, “Leveraging ami data for distribution system model calibration and situational awareness,” 2015.
[11] R. Sevlian and R. Rajagopal, “Feeder topology identification,” arXiv preprint arXiv:1503.07224, 2015.
[12] D. Deka, S. Backhaus, and M. Chertkov, “Structure learning and statistical estimation in distribution networks - part ii,” arXiv preprint arXiv:1502.07820, 2015.
[13] ——, “Learning topology of the power distribution grid with and without missing data,” in European Control Conference (accepted), 2015.
[14] P. Ravikumar, M. J. Wainwright, J. D. Lafferty et al., “High-dimensional ising model selection using 1-regularized logistic regression,” The Annals of Statistics, vol. 38, no. 3, pp. 1287–1319, 2010.
[15] A. Anandkumar, V. Tan, and A. S. Willsky, “High-dimensional graphical model selection: tractable graph families and necessary conditions,” in Advances in Neural Information Processing Systems, 2011, pp. 1863–1871.
[16] M. J. Wainwright and M. I. Jordan, “Graphical models, exponential families, and variational inference," Foundations and Trends® in Machine Learning, vol. 1, no. 1-2, pp. 1–305, 2008.
[17] M. Baran and F. Wu, “Optimal sizing of capacitors placed on a radial distribution system,” Power Delivery, IEEE Transactions on, vol. 4, no. 1, pp. 735–743, Jan 1989.
[18] ——, “Optimal capacitor placement on radial distribution systems,” Power Delivery, IEEE Transactions on, vol. 4, no. 1, pp. 725–734, Jan 1989.
[19] ——, “Network reconfiguration in distribution systems for loss reduction and load balancing,” Power Delivery, IEEE Transactions on, vol. 4, no. 2, pp. 1401–1407, Apr 1989.
[20] J. Resh, “The inverse of a nonsingular submatrix of an incident matrix,” IEEE Transactions on Circuit Theory, vol. 10, p. 131.
[21] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms. The MIT Press, 2001.
[22] J. Pearl, Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann, 2014.
[23] U. Eminoglu and M. H. Hocaoglu, “A new power flow method for radial distribution systems including voltage dependent load models,” Electric Power Systems Research, vol. 76, no. 13, pp. 106 – 114, 2005.

[24] “Reds: Repository of distribution systems,” accessed: 2016-05-20. [Online]. Available: [http://www.dejazzer.com/reds.html](http://www.dejazzer.com/reds.html)