Topology Identification of Smart Microgrids

Bof Nicoletta, Michelotti Davide, Muraro Riccardo

I. Introduction

A. Pratical relevance of the problem

The widely use of distributed energy generation (solar panels, wind turbines, etc), the presence of loads with significant energy consumption (electric cars) and the need for reliability of energy supply in critical areas has led the emergence of Smart Grid (SG), which can act in real time to manage the power grid in an efficient manner, concerning various aspects and features. Several studies in this area have been made and are still under progress (optimization of consumption, reduction of waste energy, detection of faults and malicius attacks, etc), nevertheless the knowledge of the physical grid structure is a basic ingredient in all these studies.

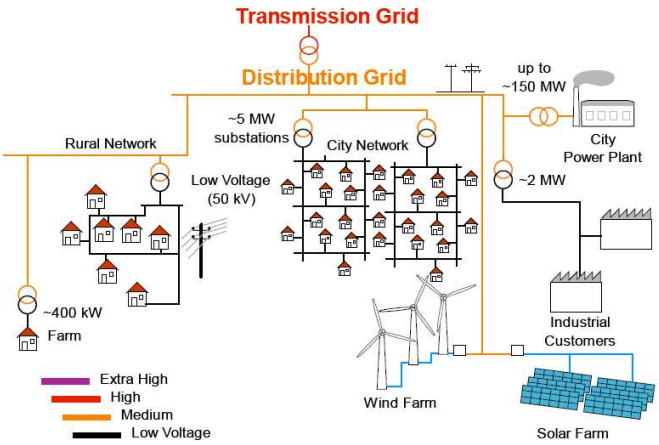

Figure 1. Hierarchy in energy dispatching: from transmission high-voltage power to distribution medium-low voltage power network.

As a matter of fact, knowing the arrangement of loads/generators on transmission lines is essential in order to make efficient electricity dispatchment, avoiding energy waste and voltage drops. In addition, through the knowledge of this graph is possible to implement a scheduling of connected devices to avoid overload of the lines. Depending on the context, the knowledge we have on grid topology is not always exhaustive. Surely the low voltage grid topology is in general not known, and nowadays there is no need of this knowledge since there is no practical interest due to the fact that it is not properly monitored. Since in future grid development PMUs devices will probably been widely used, the knowledge of grid topology will give important information for the optimization of electricity dispatchment. Moreover the network knowledge is even more relevant with the increasing complexity of networks, in which agents are not considered exclusively passive loads but can constitute a source of energy through common microgeneration devices.

B. Objective and translation of the problem

This study of predominantly theorical-simulation kind try to provide a static estimation algorithm for the identification of network topology of smart microgrids. The grid graph topology can be thought fixed during the computation time and in this sense it is a static estimation. We consider “microgrid” as the smallest portion available of the low-voltage power distribuition network that is managed autonomously from the rest of the network, having all the characteristics of interest. We assume that data and information of the grid are obtained from a network of Phasor Measurement Units (PMU’s).

Each node of the grid is supposed to be a PMU that under certain hypothesis provides angle measurements probabilistically distributed as a Gaussian random variable [1]. Therefore the electrical network can be regarded as a Gaussian probabilistic model. Due to this probabilistic model, we can give a graphical representation of the conditional dependence of measures, obtaining a graph which have a node for each PMU and represent the conditional dependences of nodes. This type of analysis is known as *graphical models,* and is based on the relationship existing between the concentration matrix (inverse of the sample covariance matrix) and the connections within the nodes in the graph. As a matter of fact the concentration matrix is sparse and non-zero elements imply the existence of a link between the corresponding nodes of the graphical representation.

According to the properties of the electric network, the concentration matrix contains not only elements referring to links between adjacent nodes, but also those referring to second-neighbours. The same information is given by the sparsity of the square Laplacian matrix of the electrical network. Due to this, and starting from the graph identified using the concentration matrix, a procedure to find only first-neighbours is needed. In theory (and under certain assumptions) some procedures which allow to determine the only root of a graph exist. However these do not suits our problem because we have only an estimator of the correlation matrix and the latter does not have exactly the same sparsity as L^2(if L is the Laplacian) . A different technique has then been elaborated to find the actual electrical graph, a technique mainly based on the fact that power grid has a tree structure.

In practice, known a series of measurements made on p nodes of the electric graph, at first we compute the sample covariance matrix and then, using techniques typical of graphical model we obtain an estimation of the topology of the graph, which will be further elaborated to obtain a tree structure for the graph itself. It is evident that the resulting graph regards interconnections between the PMU devices arranged on the network that provides an approximation of the physical structure of power grid that underlies.

C. State of art

At present there is no study that tries to solve the specific problem of power smart grid graph identification. Many aspects of the problem are based on typical approaches concerning graphical models which can be applied to electrical network too.

Ming Yuan and Yi Lin in their article [2] focus attention on selection and estimation of the the concentration matrix in Gaussian graphical model using penalized likelihood methods. The two provided methods they called *lasso-type estimator* and *nonnegative garrote-type estimator* lead to a sparse and shrinkage estimator of the concentration matrix and thus conduct model selection and estimation simultaneously. The implementation of this two methods can be done effectively by taking advantage of the efficient maxdet algorithm developed in convex semidefinite optimization. The competitive performance they proposed between two methods show that garrote-type estimator giving advantage for model-fitting when a good initial estimator is available as happens in our purpose of power grid.

Massive amount of measurements and their transmission across the grid by modern information technology make the grid prone to attacks. Hanie Sedghi and Edmond Jonckheere in their article [1] try to solve the problem of the most dreaded cyber-attacks on the electrical infrastructure rappresented by "false data injection" that compromises the PMU's data. Inadvertently they reveal some fundamental concepts useful for identification of power grid topology exploiting conditional mutual information in Gaussian Markov Random Field (GMRF). Therefore they explore the neighboring property of PMU angle measurements, then use so-called Conditional Covariance Test (CCT) on PMU angle measurements and show that, because of the walk-summability of grid graph, the output of CCT follows the grid topology. In fact, when the system is under false data injection attack, the output of CCT method misses some lines that are present in the grid graph. The approach used in this work has proven to be the closest to the problem of grid topology identification even though for other purposes and other point of view.

D. Our contribution

Starting from the power grid model suggested in [3], we reformulate the problem of graphical model selection for the purpose of grid topology identification. To do this we computed the theoretical sample covariance matrix of the power grid model. From its inverse, represented by the concentration matrix, we obtained some properties that this sparse matrix must have from theorical point of view. We have also found a relationship between the concentration matrix and the Laplacian matrix under some hypothesis and exploited this relation as much as we could to obtain information which could help in the reconstruction of the electric graph.

We introduced the penalized likelihood algorithm given by garrote-type estimator [2] on the concentration matrix obtained from power grid model. The problem of the tuning parameter, necessary for using garrote type estimator has not been completely solved yet, but applying our second step algorithm to reconstruct the likliest electrical tree, the choice of the tuning parameter does not seem critical.

E. Summary

Section II provides the mathematical preliminary and main notations used for the description of a graph relating a electric grid. The foundamental result refers to the approximate model of microgrid proposed in [3] that constitutes the basis for the following sections. Section III introduces the main ideas of graphical models and it provides some properties of the modelled power grid. The various aspects considered here can be summarized in: first and second neighbours dipendence between nodes, relation between Laplacian matrix and concentration matrix, explanation of garrote-type estimator. The goals collected in Section IV are twofold: first it provides the notion of bipartite graph and brings back the algorithm SBN (Squared of Bipartite graphs with a specified Neighbourhood), subsequently it explaines our algorithm for the determination of tree graph which we called MCT (Maximum Correlation Tree). Section V contains meaningful simulations obtained from real parameters of an existing power grid. They include two case: noiseless measurements and noisy measurements, obtained with a proper model of PMUs devices. In Section VI there are the conclusions of the work and some clues for future work. Finally in Appendix are collected insights and key demostrations of the results used during the treatment.

II. Model of a microgrid

A. Mathematical preliminaries and notation

Let $G = (V, E, \sigma, \tau)$ be a directed graph, where V is the set of nodes of cardinality $p = |V|$, E is the set of edges, and $\sigma, \tau : E \to V$ are two functions such that edge $e \in E$ goes from the source node $\sigma(e)$ to the terminal node $\tau(e)$. Two edges e and e' are consecutive if $\{\sigma(e), \tau(e)\} \cap \{\sigma(e'), \tau(e')\}$ is not empty. A path is a sequence of consecutive edges. We will often introduce complex-valued functions defined on the nodes and on the edges. These functions will also be intended as vectors in $\mathbb{C}^p$ (where $p = |V|$) and $\mathbb{C}^{|E|}$. Given a vector $\mathbf{u}$, we denote by $\bar{\mathbf{u}}$ its (element-wise) complex conjugate, and by $\mathbf{u}^T$ its transpose. Let moreover $A \in \{0, \pm 1\}^{|E| \times p}$ be the incidence matrix of the graph G, defined via its elements

$$[A]_{ev} = \begin{cases} -1 & \text{if } v = \sigma(e) \\ 1 & \text{if } v = \tau(e) \\ 0 & \text{otherwise.} \end{cases}$$

The second relevant matrix associated to a (un-weighed) graph G is the so-called Laplacian matrix $L \in \mathbb{Z}^{|E| \times p}$ defined via its elements

$$[L]_{ev} = \begin{cases} -1 & \text{if } v = \sigma(e) \text{ or } v = \tau(e) \\ -\sum_{i=1, i \neq v}^{p} [L]_{ei} & \text{if } v = e \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

so diagonal entries are the only positive values while the off-diagonal elements are negative equal to -1 or 0; L is

related with incidence matrix through $L = A^T A$. *If the graph G is connected* (i.e. for every pair of nodes there is a path connecting them), *then* $\mathbf{1}$ *is the only vector in both* $ker(A)$ *and* $ker(L)$ *[3], [4].* An undirected graph G is a graph in which for every edge $e \in E$, there exists an edge $e' \in E$ such that $\sigma(e') = \tau(e)$ and $\tau(e') = \sigma(e)$. *If the graph G is undirected then L is symmetric positive semidefinite* [4], thus results:

$$\begin{cases} L \geq 0 \\ L\mathbf{1} = 0 \\ L = L^T \end{cases} \tag{2}$$

If $\mathcal{W}$ is a subset of nodes, we define by $\mathbf{1}_{\mathcal{W}}$ the column vector whose elements are

$$[1_{\mathcal{W}}]_v \begin{cases} 1 & \text{if } v \in \mathcal{W} \\ 0 & \text{otherwise.} \end{cases}$$

Similarly, if w is a node, we denote by $\mathbf{1}_w$ the column vector whose value is 1 in position w, and 0 elsewhere, and we denote by $\mathbf{1}$ the column vector of all ones.

B. Model formulation of a microgrid

The microgrid introduced before may be modeled as an undirected graph G, in which edges represent the power lines, and nodes represent loads (with or without microgenerator) and the only point of connection of the microgrid to the transmission grid is called PCC (Point of Common Coupling). We limit our study to the steady state behavior of the system, when all voltages and currents are sinusoidal signals at the same frequency. Each signal can therefore be represented via a complex number $y = |y|e^{j\angle y}$ whose absolute value $|y|$ corresponds to the signal root-mean-square value, and whose phase $\angle y$ corresponds to the phase of the signal with respect to an arbitrary global reference. In this notation, the steady state of a microgrid is described by the following system variables:

- $\mathbf{u} \in \mathbb{C}^p$, where u_v is the grid voltage at node v;
- $\mathbf{i} \in \mathbb{C}^p$, where i_v is the current injected by node v;
- $\xi \in \mathbb{C}^{|E|}$, where ξ_e is the current flowing on the edge e.

The following constraints are satisfied by $\mathbf{u}, \mathbf{i}$ and ξ:

$$A^T \xi + \mathbf{i} = 0, \tag{3}$$

$$A\mathbf{u} + \mathbf{Z}\xi = 0, \tag{4}$$

where A is the incidence matrix of G, and $\mathbf{Z} = \text{diag}(z_e,\ e \in E)$ is the diagonal matrix of line impedances, z_e being the impedance of the microgrid power line corresponding to the edge e. Equation (3) corresponds to Kirchhoff's current law (KCL) at the nodes, while (4) describes the voltage drop on the edges of the graph. Each node v of the microgrid is then characterized by a law relating its injected current i_v with its voltage u_v. We model the PCC (which we assume to be the first node) as an ideal sinusoidal voltage generator at the microgrid nominal voltage U_N with arbitrary, but fixed, angle ϕ

$$u_0 = U_N e^{j\phi}. \tag{5}$$

We model loads and microgenerators (that is, every node v of the microgrid except the PCC) via the following law relating the voltage u_v and the current i_v

$$u_v \bar{i}_v = s_v |\frac{u_v}{U_N}|^{\eta_v}, \quad \forall v \in V \setminus \{0\}, \tag{6}$$

where s_v is the nominal complex power and η_v is a characteristic parameter of the node v. The model (6) is called exponential model and is widely adopted in the literature on power flow analysis. Notice that s_v is the complex power that the node would inject into the grid, if the voltage at its point of connection were the nominal voltage U_N. The parameter η_v depends on the particular device. For example, constant power, constant current, and constant impedance devices are described by $\eta_v = 0, 1, 2$, respectively.

C. Approximate model for microgrid

The task of solving the system of nonlinear equations given by (3), (4), (5), and (6) to obtain the grid voltages and currents, given the network parameters and the injected nominal powers $\{s_v, v \in V \setminus \{0\}\}$ at every node, has been extensively covered in the literature under the denomination of power flow analysis. A fundamental lemma for retrieving the solution is given in [3] :

Lemma 1. *Let L be the complex valued Laplacian $L := A^T \mathbf{Z}^{-1} A$. There exists a unique symmetric matrix $\mathbf{X} \in \mathbb{C}^{p \times p}$, $p = |V|$ such that*

$$\begin{cases} \mathbf{X}\mathbf{L} = I - \mathbf{1}\mathbf{1}_0^T \\ \mathbf{X}\mathbf{1}_0 = 0 \\ \mathbf{X} = \mathbf{X}^T \end{cases} \tag{7}$$

This matrix $\mathbf{X}$, called *Green like matrix,* depends only on the topology of the microgrid power lines and on their impedance. It has been observed that the Laplacian matrix weighted using $\mathbf{Z}^{-1}$ keeps the same properties (2) for unweighted graph [4]; as can be seen in Appendix F, the off- diagonal elements present negative real part and positive imaginary one, while diagonal elements have positive real part and negative imaginary one.

All the currents $\mathbf{i}$ and the voltages $\mathbf{u}$ of the microgrid are therefore determined by the equations

$$\begin{cases} \mathbf{u} = \mathbf{X}\mathbf{i} + U_N e^{j\phi}\mathbf{1} \\ \mathbf{1}^T \mathbf{i} = 0 \\ u_v \bar{i}_v = s_v |\frac{u_v}{U_N}|^{\eta_v}, \quad \forall v \in V \setminus \{0\} \end{cases} \tag{8}$$

where the first equation results from (3), (4), and (5) together with Lemma 1, while the second equation descends from (3), using the fact that $A\mathbf{1} = 0$ in a connected graph. We can see currents $\mathbf{i}$ and voltages $\mathbf{u}$ as functions $\mathbf{i}(U_N)$, $\mathbf{u}(U_N)$ of U_N. The following proposition provides the Taylor approximation of $\mathbf{i}(U_N)$ and $\mathbf{u}(U_N)$ for large U_N.

Proposition 2. *Let* $\mathbf{s}$ *be the vector of all nominal complex powers s_v, including*

$$s_0 := -\sum_{v \in V \setminus \{0\}} s_v. \tag{9}$$

Then for all $v \in V$ we have that

$$\begin{aligned} i_v(U_N) &= e^{j\phi}\left(\frac{\bar{s}_v}{U_N} + \frac{c_v(U_N)}{U_N^2}\right) \\ u_v(U_N) &= e^{j\phi}\left(U_N + \frac{[\mathbf{X}\bar{s}]_v}{U_N} + \frac{d_v(U_N)}{U_N^2}\right) \end{aligned} \tag{10}$$

for some complex valued functions $c_v(U_N)$ and $d_v(U_N)$ which are $O(1)$ as $U_N \to \infty$, i.e. they are bounded functions for large values of the nominal voltage U_N.

The affine approximation given in (10) which relates vectors of currents $\mathbf{i}$ and voltages $\mathbf{u}$ with the vector of all nominal complex powers $\mathbf{s}$, is verified in practice and corresponds to correct design and operation of power distribution networks, where indeed the nominal voltage U_N is chosen sufficiently large (subject to other functional constraints) in order to deliver electric power to the loads with relatively small power losses in lines. The proof of 1 and 2 is given in [3].

III. Graphical model selection

To estimate the electrical network graph, a Gaussian graphical model identification technique was used. Such technique allows to find out the graph representation $G = (V, E)$ from a p-dimensional Gaussian vector $\mathbf{Y} \sim \mathcal{N}(\mu, \Sigma)$, where V contains p vertices corresponding to the p coordinates of $\mathbf{Y}$ and the edges $E = (e_{ij})_{1 \le i < j \le p}$ describe the conditional independence relationship among $Y^{(1)}, \ldots, Y^{(p)}$. As a matter of fact the relationship $(i,j) \notin E \iff Y_i \perp Y_j \mid Y_k, k \neq i, j$ holds for every absent edge, therefore an edge in the graph is missing if and only if the element Y_i is completely uncorrelated from Y_j given all the others graph nodes. Of particular interest is the identification of zero entries in the so-called concentration matrix $C = \Sigma^{-1}$, since it has been proven that $C(i,j) = 0 \iff Y_i \perp Y_j \mid Y_k, k \neq i, j$, [5]. Therefore, graph estimation has been reduced to matching the non zero elements in the inverse of Σ with the graph edges. Given a finite realization $Y_1, \ldots, Y_N$ of a Gaussian vector $\mathbf{Y}$ with unknown mean μ and variance Σ, it's possible to apply the maximum likelihood estimator using data, obtaining $(\bar{\mu}, \hat{\Sigma})$. In this way the concentration matrix C can be naturally determined by $\hat{\Sigma}^{-1}$. However, this technique does not lead to sparse graph structure, since $\hat{\Sigma}$ is only an estimator of Σ. A possible solution to this problem will be given in the chapter III-C

A. Measure distribution and properties

Gaussian graphical model procedure can be applied to the identification of the electrical network because the voltage measures at each node are approximately Gaussian. As a matter of fact, the power needed at a given node is due to the request of many different loads which present also noises, and so it can be modelled as a Gaussian random variable [1]. Finally the relation which joins $\mathbf{s}$ (the vector containing the powers taken at every node) and $\mathbf{u}$ (the vector containing voltages) can be approximated by the affine relationship of equation (10), thus $\mathbf{u}$ is a vector of Gaussian random variables. In this way, the voltage measures taken at each node can be thought as realizations of correlated Gaussian random variables.

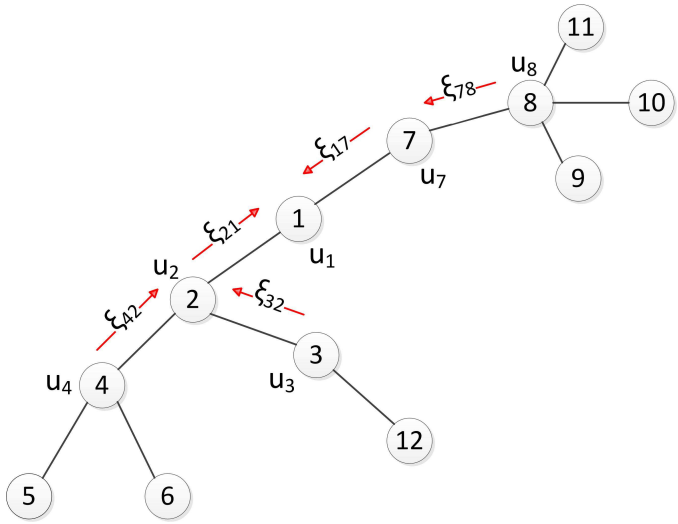

Figure 2. An electrical network used to show the conditional independence of measures.

If the real probability distribution of $\mathbf{u}$ is known its mean $\mu_{\mathbf{u}}$ and its covariance matrix $\Sigma_{\mathbf{u}}$ are given, the concentration matrix $C_{\mathbf{u}} = \Sigma_{\mathbf{u}}^{-1}$ presents non-zero entries among those elements of $\mathbf{u}$ which are conditionally dependent.

Considering the electrical graph, the voltage at a given node is a function of the voltages of its first and second neighbours. To understand why this property holds, a small example of electric network has been created, Figure 2. Supposing to know the voltage at each node, this can be expressed as a function of the current flowing from the node into the network, $u_j = f_j(i_j)$, if i_j and u_j are the current and the voltage at node j. As a consequence of Kirkchhoff's currents law, the sum of the currents entering a given node has to match the sum of the currents outgoing the same node. Moreover, the current flowing through an edge of the electric network is due to the voltage drop on the same edge, and can be evaluated by $\xi_{i,j} = \frac{u_i - u_j}{Z_{ij}}$, if Z_{ij} is the impedence of the line between node i and j .

Considering as an example node 1 :

$$\text{Node 1 eq.} \begin{cases} u_1 & = f_1(i_1) \\ i_1 + \xi_{21} + \xi_{71} & = 0 \end{cases}$$

$$\downarrow$$

$$u_1 = f_1(-\xi_{21} - \xi_{71})$$

$$\text{Left branch eq.} \begin{cases} \xi_{21} & = i_2 + \xi_{42} + \xi_{32} \\ \xi_{42} & = \frac{u_4 - u_2}{Z_{42}} \\ \xi_{32} & = \frac{u_3 - u_2}{Z_{32}} \\ i_2 = & f_2^{-1}(u_2) \end{cases}$$

$$\text{Right branch eq.} \begin{cases} \xi_{71} & = i_7 + \xi_{87} \\ \xi_{87} & = \frac{u_8 - u_7}{Z_{87}} \\ i_7 & = f_7^{-1}(u_7) \end{cases}$$

$$\begin{aligned} u_1 &= f_1(-\xi_{21} - \xi_{71}) \\ &= f_1(u_2, u_3, u_4, u_7, u_8) \end{aligned}$$

Now it's clear that the voltage at node 1, can be computed exactly if the voltages at nodes $2-3-4-7-8$ are known, and so node 1 is conditionally independent from any node of the electrical network except for the first and second neighbours, when the voltages at the latter ones are known. As a consequence, the evaluation of the electric network using a Gaussian model identification technique returns a graph wich contains not only the edges of the electrical networks, but also edges connecting a node with its second neighbours. In chapter IV-A and IV-B possible solutions to this problem are given.

All the previous reasonings hold if the PCC node is not considered in the calculation; as a matter of fact $\Sigma_{\mathbf{u}}$ is invertible. As soon as the PCC is introduced in the calculation, $\Sigma_{\mathbf{u}}$ and $\hat{\Sigma}_{\mathbf{u}}$ become singular, since PCC is an ideal voltage generator and its voltage is always constant, leading to the fact that its variance and its covariance with all the other elements in the vector $\mathbf{u}$ is null. It is therefore necessary to use a pseudoinverse of $\Sigma_{\mathbf{u}}$ (which will be denoted as $\Sigma_{\mathbf{u}}^{\dagger}$) to obtain a useful result, when the PCC is involved in network identification. The introduction of PCC is an important aspect both for finding the connection to the highest smart grid layer (the distribution grid) and for some considerations that will come out from the calculation involving also the PCC. To simplify further calculation the introduction of a hypothesys about micro-grid structure is now done: the PCC node is connected to the remaining electric network only through one edge. This is not a too restrictive hypothesis since it only means that there must be only one node connected to the PCC, and all the other nodes can be connected without any kind of constraint.

B. The Laplacian matrix relation

In the following part theoretical $\Sigma_{\mathbf{u}}$ and a relationship between one of its pseudoinverse and $\bar{L}L$, is found. The latter will be a corroboration that a non-zero entry in $\Sigma_{\mathbf{u}}^{\dagger}$ indicates that the 2 nodes involved are neighbours or second neighbours, as alredy observed. One hypothesis is assumed in this section: all the loads must be described by scorrelated Gaussian random variables with the same variance σ^2. This hypothesis is made to simplify the mathematical treatment of this matter, but the results of this work are still valid without them, as will be highlighted in the simulation chapter. Another non restrictive supposition is made: the PCC is the first one node, so the column vector $\mathbf{1_0}$ has 1 in first position and 0 in all the other $p-1$ elements. For space sake, only the final results are retrieved here, while all calculus can be found in Appendix. The theoretical covariance matrix (see Appendix A) results to be:

$$\Sigma_{\mathbf{u}} = \mathbf{E}\left[(\mathbf{u} - \mu_{\mathbf{u}})\overline{(\mathbf{u} - \mu_{\mathbf{u}})}^T\right] = \frac{\sigma^2}{U_N^2}\mathbf{X}\overline{\mathbf{X}}$$

Since $\frac{\sigma^2}{U_N^2}$ is only a multiplicative factor, it is ignored in the following passages. Applying the same process used to calculate $\mathbf{X}$ which is a pseudoinverse of L, a pseudoinverse A of $\mathbf{X}\overline{\mathbf{X}}$ is found. A has the following property:

$$\begin{cases} \mathbf{X}\overline{\mathbf{X}}A = I - \mathbf{1}\mathbf{1_0^T} \\ A = \bar{A}^T \\ A\mathbf{1} = 0 \end{cases}$$

As A can be computed deleting in $\mathbf{X}\overline{\mathbf{X}}$ the row and column related with the PCC, the usefulness of the hypothesis about the PCC linkage becomes now clear. If the PCC was connected directly with two nodes i and j, it would be impossible to have in $\Sigma_{\mathbf{u}}^{\dagger}$ a non zero entry between i and j as it should be, because the information about neighbours involving PCC has been delated during the calculation. When reconstructing the whole concentration matrix with the method presented in Appendix B, the elements belonging to $\mathbf{S}^{-1}$ are not changed. With the hypothesis that a single edge starts from the PCC the possibility of this event disappear.

The following formula gives a relationship between A and $\bar{L}L$:

$$A = \bar{L}L\left(I - \mathbf{X}\mathbf{1}\mathbf{1_0^T}L\right)^{-1}. \tag{11}$$

As shown in the Appendix C the existence of the inverse is assured almost always; moreover it can be shown that A has the same elements of $\bar{L}L$ except for those element in position $(PCC, PCC), (i, i), (i, PCC)$, if i is the first neighbour of the PCC node. This relation assures that A and $\bar{L}L$ have the same sparsity, so $\Sigma_{\mathbf{u}}^{\dagger} = \frac{U_N^2}{\sigma^2}A$ has the same sparsity of $\bar{L}L$ too. This is an alternative way to demonstrate the presence in $\Sigma_{\mathbf{u}}^{\dagger}$ of non-zero entries only between those elements which in the electric graph are first or second neighbours, because $\bar{L}L$ has the same sparsity of L^2 and then its non-zero entries indicate first and second neighbours. $\bar{L}L$ is thought to have the same sparsity as L^2 since every simulation done as confirmed this, and we also expect this to be a general consideration, with only some exceptions due to very particular matrixes. Moreover formula (11) implies a precise relationship between A and L, so the estimation of $\Sigma_{\mathbf{u}}^{\dagger}$, which differs from A for a multiplicative factor, can give information also about L. This is an important consideration because knowing the Laplacian matrix means also having information about line impedences, (1) (however finding an estimator of $\Sigma_{\mathbf{u}}^{\dagger}$ so good as to have the right values is very difficult).

In conclusion, under the hypothesis concerning the phase of the variance of loads, having a good estimator of $\Sigma_{\mathbf{u}}$ not only allows to reconstruct the electrical graph, but also gives important information about impedences of the line.

C. Concentration matrix estimation

In this section will be illustrated the garrote-type estimator theory used to determine an estimator of the concentration

matrix usable for Gaussian graphical modelling. This method has been used because of the sparsity property of the concentration matrix, given the fact that the inverse of the sampled covariance matrix $\hat{\Sigma}_{\mathbf{u}}^{-1}$ doesn't lead to a sparse matrix. This approach belong to the penalized likelihood method that leads to a sparse and shrinkage estimator of the concentration matrix, which has to be positive definite, and thus conducts model selection and estimation simultaneously. The implementation of this methods is nontrivial because of the positive definite constraint on the concentration matrix, so we were forced to use the maxdet algorithm developed in convex optimization. To achieve sparse graph structure a nonnegative garrote-type estimator has been used. Such method is appliable only on invertible matrix and in our case$\hat{\Sigma}_{\mathbf{u}}$ is singular. However, as proven in Appendix B all the information brought by $\hat{\Sigma}_{\mathbf{u}}$ can be obtained by its submatrix found deleting the PCC row and column. Calling this matrix $\tilde{\Sigma}_{\mathbf{u}}$, this is non singular and the garrote-type estimator can be applied to its inverse, $\tilde{C}$. It's known that $\tilde{C}$ is a reliable estimator of C, the theoretical concentration matrix (not considering the PCC node).

The shrinkage estimator of $\hat{C}$ can be defined through $\hat{c}_{ij} = d_{ij}\widetilde{c}_{ij}$, where the symmetric matrix D is the minimizer of

$$\begin{array}{rl} \min\limits_{D} & \{-log|\hat{C}| + tr(\hat{C}\tilde{\Sigma}_{\mathbf{u}})\} \\ \text{subject to} & \sum_{i\neq j} d_{ij} \leq t \qquad d_{ij} \geq 0 \\ & \hat{C} > 0 \end{array} \tag{12}$$

Where t is a tuning parameter, [2]. Equivalently, using the Lagrangian form, this can be written as

$$\begin{array}{rl} \min\limits_{D} & \{-log|\hat{C}| + tr(\hat{C}\tilde{\Sigma}_{\mathbf{u}}) + \lambda \sum_{i\neq j} \frac{\hat{c}_{ij}}{\widetilde{c}_{ij}}\} \\ \text{subject to} & \frac{\hat{c}_{ij}}{\widetilde{c}_{ij}} \geq 0 \\ & \hat{C} > 0 \end{array}$$

where λ is another tuning parameter related to t. A further step of this method is to estimate the PCC neighbourhood using the procedure explained in Appendix B and determine the estimated concentration matrix for the whole network starting from the output of the garrote type estimator.

The main property of this estimator is that for a relatively large sample it leads up to the consistency as claimed in theorem 3.

Theorem 3. *If we denote with $\hat{C}$ the minimizer of (12) with initial estimator $\widetilde{C}$, $n\lambda \to \infty$ and $\sqrt{n}\lambda \to 0$ as $n \to \infty$, then $\Pr(\hat{c}_{ij} = 0) \to 1$ if $c_{ij} = 0$, and other elements of $\hat{C}$ have the same limiting distribution as the maximum likelihood estimator on the true graph structure.*

Theorem 3 indicates that the garrote-type estimator enjoys the so-colled oracole property: it selects the right graph with probability tending to one and at the same time gives a root-n consistent estimator of the concentration matrix. Due to the nonlinearity of the objective function and the positive-definitness constraint, the problem is non trivial. For its solution we can lead back our problem to the maxdet problem, which has the following general form:

$$\begin{array}{rl} \min\limits_{x\in\mathbb{R}^m} & b^T x - log|G(x)| \\ \text{subject to} & G(x) > 0 \\ & F(x) \geq 0 \end{array} \tag{13}$$

where $b \in \mathbb{R}^m$. Moreover, $G : \mathbb{R}^m \to \mathbb{R}^{l\times l}$ and $F : \mathbb{R}^m \to \mathbb{R}^{l\times l}$ are affine:

$$\begin{aligned} G(x) &= G_0 + x_1 G_1 + \cdots + x_m G_m \\ F(x) &= F_0 + x_1 F_1 + \cdots + x_m F_m \end{aligned}$$

where F_i and G_i are symmetric matrices. It is not hard to see that the garrote-type estimator solves the problem respect to the minimizer D, as showed in Appendix D. The problem solution can be determine using the Matlab toolbox YALMIP that can handle optimization and control oriented SDP problems. Moreover, this software can work with complex-valued data and constraints, necessary in our project. So far we focused on the calculation of the minimizer for any fixed tuning parameter t. Usually, the optimum choice of this value depends directly from the problem so it will be handle in simulation chapter

IV. Netwrok Graph Estimation

A. Roots of Bipartite Graphs

Now, given the concentration matrix $\hat{C}$, determined from the garrote algorithm, and taking advantage of the sparsity relation between this one and L^2, we want to apply algebraic graph theory at the problem in order to determine a matrix with the same sparsity as the laplacian matrix L concerning the network graph model.

Definition 4. H is a root of G = (V,E) if there exists a positive integer k such that x and y are adjacent in G if and only if their distance in H is at most k. If H is a $k-th$ root of G, then we write $G = H^k$ and call G the $k-th$ power of H.

Ordinarily, it is a difficult task to determine whether a given graph G has a $k-th$ root or not. Also, the number of $k-th$ roots could be exponential in the size of the input graph. However, we focus the analysis on bipartite graph that allow, under certain hypothesis, to proof the uniqueness of their root.

Definition 5. A bipartite graph (or bigraph) is a graph whose vertices can be divided into two disjoint sets U and V such that every edge connects a vertex in U to one in V. Therefore, U and V are each independent sets.

Proposition 6. *Let B be a bipartite graph such that $B^2 = G$. If $uv \in E(G)$ and u,v are on different sides of B, then $uv \in E(B)$.*

Moreover, let $B = (X, Y, E)$ be a bipartite graph with X and Y as the partite sets. Suppose we fix the partite sets of the bipartite roots of G. Then, from Proposition 6, the edge set of the bipartite root is forced. In fact, the unique bipartite root candidate is $B = (X, Y, E)$ with $E(B) = \{uv | uv \in E(G), u \in X, v \in Y\}$ as seen in Figure 3.

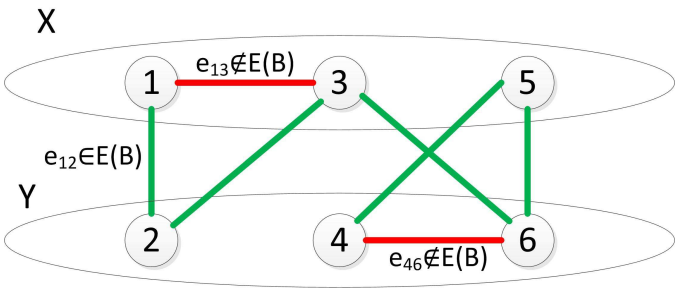

Figure 3. Example of the SBN functioning where the green edge represent the selected one while the red one doesn't belong to the root.

Furthermore, in the grid modellization chapter we stated that an electrical graph is always a tree so the proposition below holds.

Proposition 7. *Trees are bipartite*

So, if the neighbourhood of a generic node in L is known, it's possible to find out the unique root of the graph seeing that the two disjoint partition are univocally determine. Now, we propose the algorithm applied in this project taken from [6] called SBN (Squares of Bipartite graphs with a specified Neighbourhood). The hypothesis made until now require to have knowledge about the PCC neighbourhood and this information can be gather easily when the installation of the node is made. The main issue in this root finding procedure is that the matrix obtaines using the garrote-type estimator has only asymptotically the same sparsity of L^2; to apply the algorithm it is necessary that the given matrix represents the square of a bipartite graph. The estimated concentration matrix obtained by the garrote-type estimator presents some false positive and some false negative, and with null probability it still represents the square af a tree. Thus, different solutions must be evaluate if a non ideal characterization of the concentration matrix is available.

Algorithm 1 Root of bipartite graph with specified neighbourhood

$C_1 \leftarrow v$
$C_2 \leftarrow U$
$V_2 \leftarrow C_1 \cup C_2$
k←2
while (V_k is a proper subset of V(G)) do
    $C_{k+1} \leftarrow N_G(C_{k-1}) - V_k$
    $V_{k+1} \leftarrow V_k \cup C_{k+1}$
    $k \leftarrow k + 1$
end
$X \leftarrow \bigcup_i C_{2i+1}$
$Y \leftarrow \bigcup_i C_{2i}$
$E \leftarrow \{xy | x \in X, y \in Y \, and \, xy \in E(G)\}$

B. Maximum Correlation Tree

Since in real applications it's not possible to have enough measures to estimate the covariance matrix in such a good way as to have the estimated concentration matrix pointing out only the true zero entries, the previous method doesn't seems applicable. A different kind of algorithm has been created in order to reconstruct the electrical network having a concentration matrix which can present false non zero entries. The electrical network has a tree structure, so the algorithm tries to find out the most likely tree starting from the concentration matrix.The algorithm exploit the difference between absolute values in the concentration matrix; indeed, as it is pointed out in [1], the entry in the concentration matrix concerning a second-neighbour is smaller than the entry for a first neighbour. Taking advantage of this fact, the algorithm gives back a matrix whose non zero entries determine the tree with the strongest edges. Another fundamental fact used by the algorithm is that a graph containing p nodes is a tree if and only if two out of these 3 conditions are met: it is connected, it has $p-1$ edges and it has no cycle, [7]. A description of the algorithm now follows. The algorithm is given the matrix $\hat{C}$ and the number of the PCC node. Matrix $\hat{C}$ contains information about the value of the correlation between nodes. The algorithm starts guessing that the graph has $p-1$ edges corresponding to the $p-1$ highest absolute value elements in $\hat{C}$ (excluded the diagonal entries). Then it explores the graph starting from the PCC node; if during the exploration the algorithm finds a cycle in the graph, it keeps in memory the nodes which form the same. After the graph exploration, nodes can be divided in 2 sets, V_e which contains the nodes reached by the exploration, and $V_u$which contains the unreached nodes. If V_u isn't empty, the graph results disconnected, so it cannot be a tree; the algorithm tries then to find a connection between the 2 partitions. Among all the edges between V_u and V_e the algorithm adds to E the one with the highest absolute value in $\hat{C}$. If all the entries in $\hat{C}$ are such that $\hat{C}(i,j) = 0$ with $i \in V_e \wedge j \in V_u$ there is no possible tree constructable with the given matrix so the algorithm stops giving an error. If during the exploration the algorithm has found a cycle, it compares the weight of all edges which compose the cycle and delete from E the edge with the smallest absolute value. The algorithm repeats the exploration of the graph and the subsequent operations until it finds a tree or until it realizes that it's impossible to bui ld one. Figures 4-5-6 give an illustration of the functioning of the algorithm.

Since the algorithm starts with a graph containing the "heaviest" possible edges, at every iteration it adds the edge with the highest absolute value among those which can be inserted, and it deletes the smallest edge between those which form cycles, thus the algorithm gives the best (in terms of weigth of the edges) tree which can be found with the given matrix $\hat{C}$. From now on, we will report on this algorithm with the name of MCT (Maximum Correlation Tree). This algorithm can obviously commit some mistakes, due to the fact that the given matrix is affected by noise and it's obtained b y an estimation procedure, but in the simulation chapter it will be clear that as long as the number of measures increases, the results get better (since the matrix $\hat{C}$ is more precise). This method however can give a quite reliable estimation of the electrical network with a restrained number of measures.

Taking advantage of the fact that second-neighbours have smaller entries in $\hat{C}$ than first-neighbours, one can think to

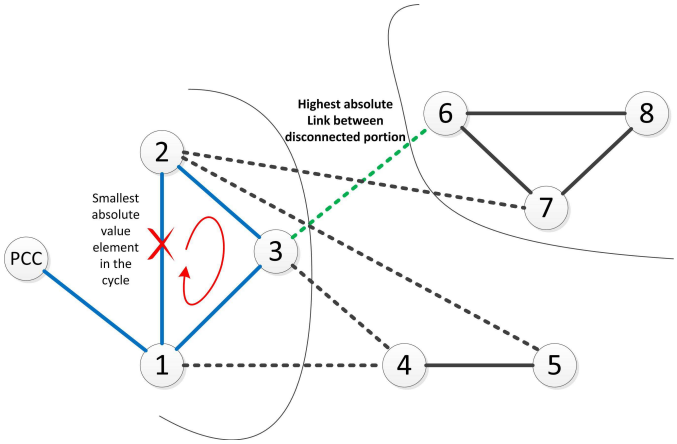

Figure 4. First iteration of the algorithm. The light blue links between nodes represent the edges visited from root PCC during the exploration, black links represent the edges of the graph and dotted lines represents those edges wich can be add to the graph to make the graph connected.

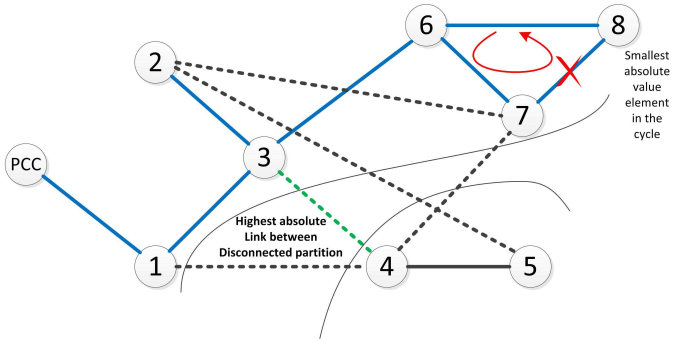

Figure 5. Second iteration of the algorithm

just create a threshold to select the $p-1$ edges with the highest absolute value; as will be showed by the simulation, this very simple way of reasoning gives good results only asymptotically, as should be expected. The algorithm which has been created starts with a threshold but guarantee that the output matrix represents a tree.

A quite important advantage concerning this algorithm is that it doesn't need a precise choice for the threshold of the garrote-type estimator. As a matter of fact the algorithm works well even if matrix $\hat{C}$ is not really sparse. The algorithm needs $\hat{C}$ to have a detectable difference beetween absolute value of first and second neighbours, but this property is mostly due to the number of measures and the noise of the same; the garrot-type estimator affects this difference but for a very wide range of the threshold t the algorithm works well.

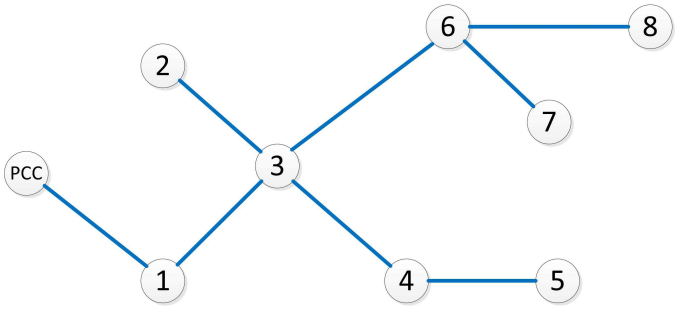

Figure 6. Final result

To improve the algorithm, a relation between signs of the real parts of non-zero elements in $\hat{C}$ has been seeked. As $\hat{C}$ is an estimation of $\Sigma_{\mathbf{u}}^{\dagger}$, and the latter equals $\bar{L}L$ except for a multiplicative positive factor, the signs of $\bar{L}L$ has been studied. Considering a non-weighted Laplacian matrix L_u, $\bar{L_u}L_u = L_u^2$ has always positive elements concerning the diagonal and second neighbours, while it has negative entries for first neighbour (the demonstration can be found in Appendix E.1). The introduction of weights in the Laplacian matrix makes this property difficult to demonstrate (the property looks at the signs of the real part, since weights in electrical networks are complex numbers), but in the special case of a chain-structured power network the proof is easy to find.

A chain-structered power network is a very simple type network, where nodes are arranged in a line and each one is connected to the previous node in the line and to the following one. Conforming to this topology, the weighted laplacian matrix has non zero elements only on the diagonal, the subdiagonal and the upperdiagonal positions. With such a laplacian matrix the proof that $\bar{L}L$ has the elements related to first neighbours with negative real part, and those related to second neighbours with positive real part is easy to find (see Appendix E.2).

Considering more complicated network structures, the mathematical treatment of the problem is quite difficult. The positivity of the real part of diagonal elements, and those concerning second neighbours is still easy (see Appendix E.3), but the negativity of the real part of first neighbour is difficult to prove and maybe it's not correct. However the sum in a column of $\bar{L}L$ of the real parts of elements concerning first neighbours has to be negative ($L\mathbf{1} = \mathbf{0}$), so there is at least one first neighbour with negative real part. Many simulations has been done to verify this property: using a quite generic topology for the electrical network (which will be introduced later), the values of the impedences of the line have been taken casually (as realization of gaussian variables) and the property about the sign of the real part of first neighbours in $\bar{L}L$ has always been respected. Even if simulations cannot give certainties for all the possible electric networks, the property seems to hold.

The property found could give some improvement to the algorithm because the matrix obtained by the garrote-type estimator is an estimation of $\Sigma_{\mathbf{u}}^{\dagger}$ and so, as stated before, elements concerning first neighbours should have negative real part; in this way the algorithm could exploit also the sign of the elements and not only the absolute value. However, the simulations done have not highlighted any real improvement in the result. Moreover, when the measures taken from PMU are noisy, in esteemed $\Sigma_{\mathbf{u}}^{\dagger}$ some second neighbours' elements appear to have negative real part too as a consequence of the noisy misures. Considering this two facts, and the absence of a proper mathematical demonstration, this expedient is not used in the solution of the problem.

V. Simulation

A. IEEE 37 Node Test Feeder

For our simulations we mainly considered the same grid topology implemented in [3], that is inspired from the standard testbed IEEE 37 node test feeder [8], which is an actual portion of power distribution network located in California. We assumed that load are balanced, and therefore all currents and voltages can be described in a single-phase phasorial notation. The topology network IEEE 37 considered is represented in the figure 7, obtained by renaming the labels of nodes with the convenient notation order used in the simulation program.

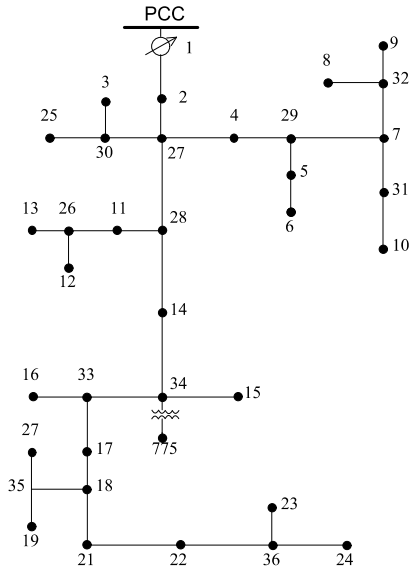

Figure 7. Schematic representation of the IEEE37 testbed.

B. General considerations

In the implementation of the overall estimation procedure it has been overlooked the effect of the variation in the garrote estimator tuning parameter. The main problem noticed in the project was determine an automatic parameter detection procedure. So far, it seems that an easy relation between raw data and t can't be found. Moreover, application of BIC method such in [2] didn't provide a useful estimation of the parameter. A manual choice of t has been done in order to obtain a good output in the garrote estimation although a pratical verification proven that there is a low sensitivity of the Maximum Correlation Tree algorithm respect to the tuning parameter t even in case of noisy data.

C. Noise free simulation

The plots given in simulations are graphical representation of matrixes that represent the sparsity of the same. If the (i, j) element of the matrix is zero, in the corrisponding position of the plot there is a blank circle and is not highlighted, while if the element is non-zero, the plot presents a coloured spot. The colours changes with the absolute value of the element, going from red (the smallest absolute value element in the matrix) to black (the highest absolute value). Moreover a black circumference is around those elements which are first-neighbours, while a grey one is around those which are second-neighbours in the actual network.

The first simulations done concerned ideal misures, with no noises. Loads at each node are modelled as scorrelated Gaussian random variables with same variance σ^2.

1) Simulation with 100 measures: With 100 misures the estimated concentration matrix is quite rough, and it's not possible to differentiate which entries are truly non zero. As can be expected, the results obtained with the garrote-type estimator are not satisfactory. The choice of the threshold has been done looking to the relationship between the threshold t and the number of non-zero entries in the matrix returned by the garrote-type estimator. As can be seen in Figure 8 for values of t between 60 and 80 the numbers of non-zero elements identified by the estimators seems to remain stable.

Figure 8. Number of non-zero entries respect to threshold t on garrote-type estimator for 100 measurements.

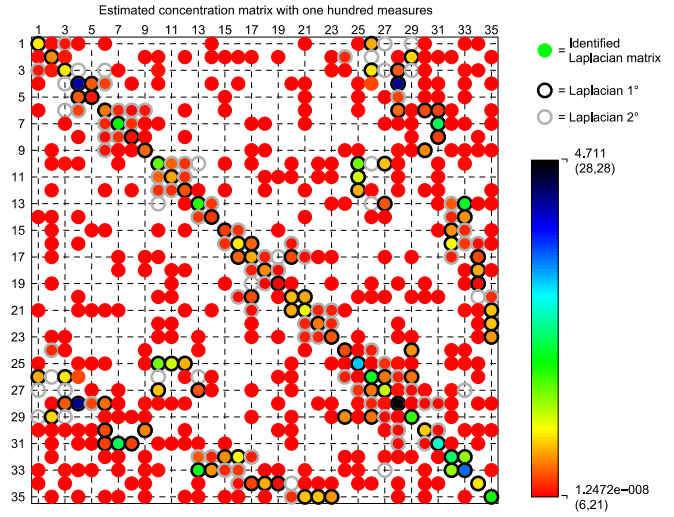

Figure 9. Representation of the concentration matrix output of garrote-type estimator for 100 measurements where the non-zeros entries are represented by full colored spots corresponding to the relative weight of the absolute value.

Using as threshold 70, the matrix obtained from the garrote-type estimator is not at all precise (Figure 9), there are lots of false positive and some second neighbours are not identified.

The algorithm SBN is unusable since there are too many false positives.

However using MCT, which can be always applied, the matrix returned has non-zero elements only in corrispondence of first-neighbours, so it finds the right graph (Figure 10)

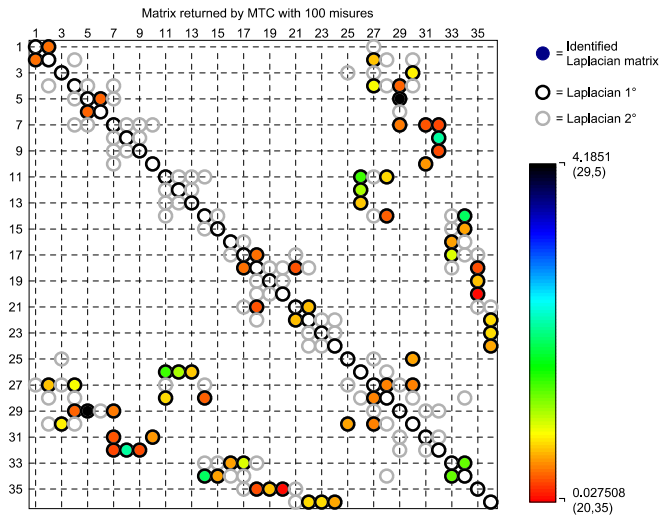

Figure 10. Tree graph reconstructed by MCT algorithm applied to the concentration matrix resulting from the garrote-type estimator of Figure 9.

2) Simulation with 50000 measures: With such a high numbers of measures the estimated concentration matrix is reliable, so we expect the garrote-type to work well; analyzing the same relationship used before to choose the threshold, 50 seems to be a nice choice for t (Figure 11).

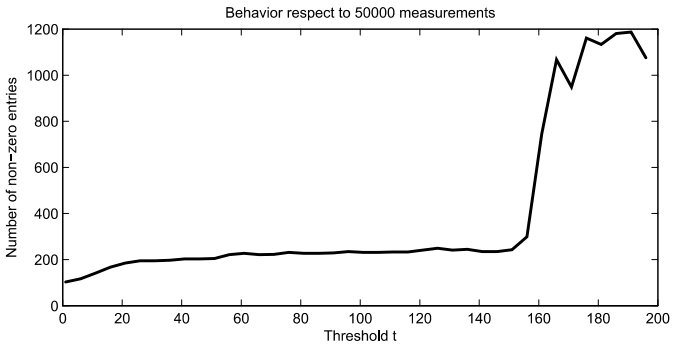

Figure 11. Number of non-zero entries respect to threshold t on garrote-type estimator for 100 measurements.

The matrix returned by the garrote-type estimator is almost right, but there are still some false positves and some false negatives (involving only second-neighbours) (Figure 12).

Even if the matrix returned by the estimator is almost right SBN doesn't work correctly, because the matrix given to the algorithm should be the square of a bipartite graph, and this is not the case. The MCT algortihm is obviously able to return the matrix representing the right graph.

D. Noisy Simulation

In real applications PMUs are affected by noises. There are three types of error in a measure taken by PMU: an error on

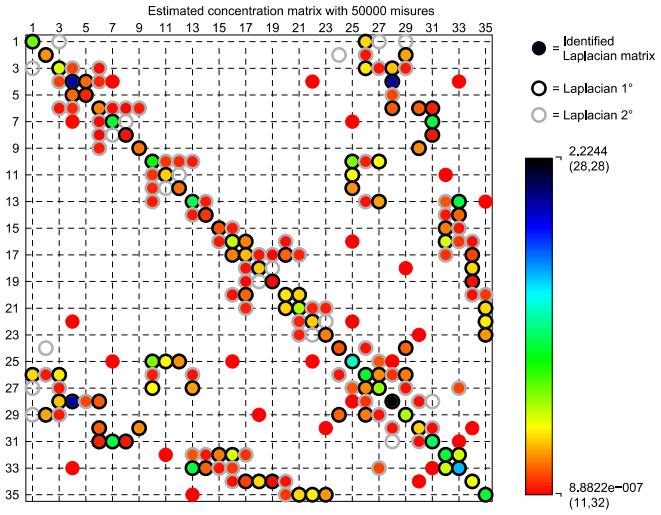

Figure 12. Representation of the concentration matrix output of garrote-type with a threshold of 50 obtained for 50000 measurements.

the syncrhonization between different PMUs, an error on the measure of the phase difference between voltage and current and an error on the amplitude of the signal. The first one can supposed to be time invariant and can be modelled as a Gaussian random variable with standard deviation 10^{-3}. The second one, which is mainly due to quantization and the way in which the phase difference is computed, is time variant, uncorrelated between any 2 misures and it has been supposed that 97% of the measures stand between $\pm 0.5°$from the actual value. The latter error, due to quantization and noise, is time variant, uncorrelated between any 2 misures and it has been supposed that 97% of the measures stand between 0.5% of the actual value. This values have been found on Factomart Catalogue).

Ultimately, in the simulation voltage measure at each node has been obtained as

$$u_n = e^{j\theta_{syin}} e^{j\theta}(1+\Delta)u$$

with u the measure computed without error, θ_{sync} the syncrhonization error (generated once for every simulation), θ the error on the phase difference (generated for every measure) and Δ the errore on the amplitude (generated for every measure).

1) Simulation with 10000 measures: The introduction of errors is critical. Even using a very high number of measures, in the estimated concentration matrix the difference between absolute value of the elements is quite small. The output of the garrote type doesn't present appreciable difference in abs olute value between first-neighbours' element and all the other no n-zero one (Figure 13).

MCT doesn't work well because the difference in absolute value of the elements in the estimated concentration matrix is too small. It returns a matrix which contains lots of false positive (Figure 14).

To diminish the consequence of noisy measures an elaboration of the data has been done. PMUs can work at a

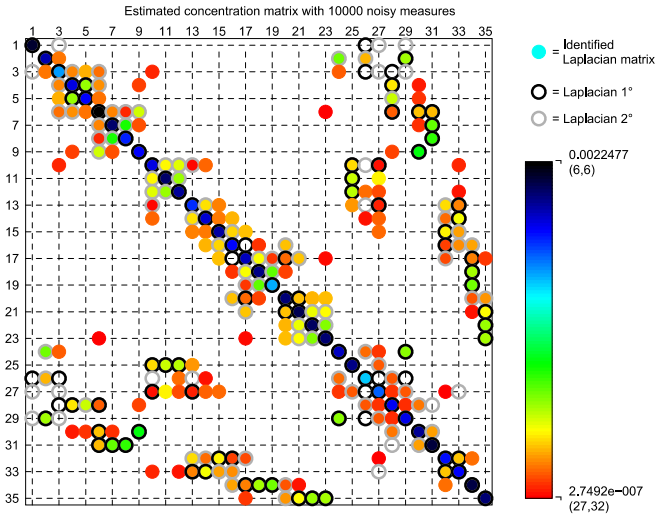

Figure 13. Simulation output of garrote-type estimator for 10000 measurements affected by noise (the threshold chosen is 150).

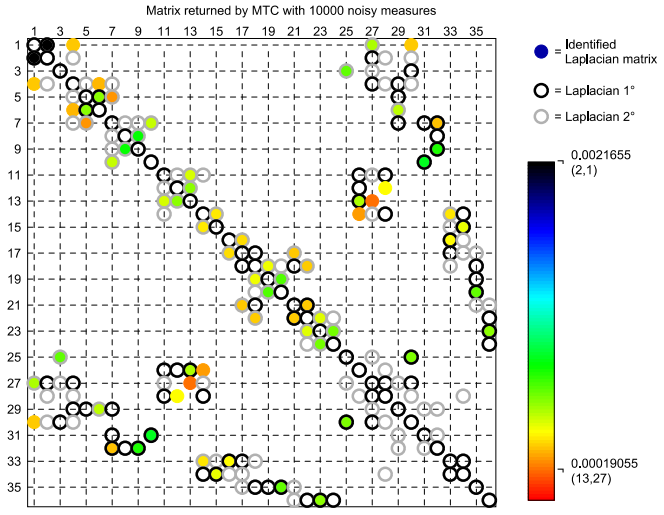

Figure 14. Tree graph reconstructed by MCT algorithm applied to noisy simulation give in Figure 13.

sampling frequency of 60 Hz; in a bit more than a second and a half, it can obtain 100 consecutive measures of the voltage at a node. With high probability loads' distributio n doesn't change in such a short period of time, so we can assume that these measures should all be equals because the electrical grid could be considered steady state. Starting from this consideration we average these measures obtaining a less noisy datum. The mean of the data is not a correct estimator of the actual value of the measure because noise modify also the phase of the measure itself; however, given that the error on the phase is small this procedure seems to work. As a matter of fact the error introduced by the multiplication for $e^{j\theta_{syin}}e^{j\theta}$ is quite small; the real part of $e^{j\theta_{syin}}e^{j\theta}$ is almost one, while the imaginary part is quite small (on the order of 10^{-3}). Considered this, doing the average on 100 measures and writing $e^{j\theta^i_{syin}}e^{j\theta^i}$ as $\delta_i + j\epsilon_i$ we get:

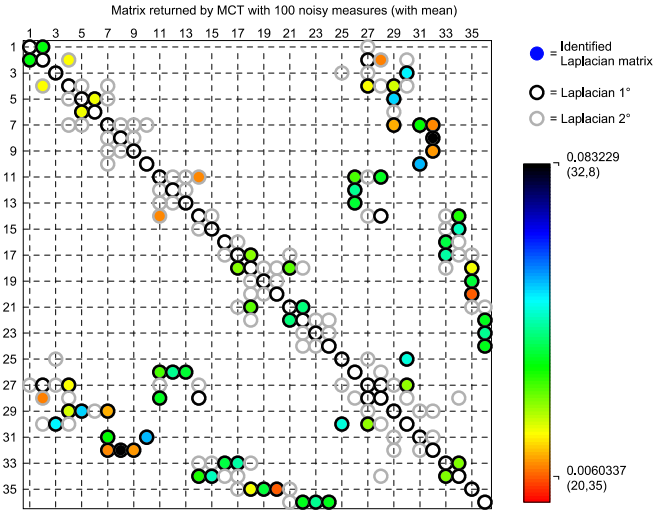

Figure 15. Tree graph reconstructed by MCT algorithm for 100 averanged measures (the threshold chosen for garrote estimator was 150).

$$\frac{(\delta_1 + j\epsilon_1)(1+\Delta_1)u + \ldots + (\delta_{100} + j\epsilon_{100})(1+\Delta_{100})u}{100} =$$
$$\frac{\delta_1 + \delta_2 + \ldots + \delta_{100}}{100}u + \frac{\delta_1\Delta_1 + \delta_2\Delta_2 + \ldots + \delta_{100}\Delta_{100}}{100}u +$$
$$+j\frac{\epsilon_1 + \epsilon_2 + \ldots + \epsilon_{100}}{100}u + j\frac{\epsilon_1\Delta_1 + \epsilon_2\Delta_2 + \ldots + \epsilon_{100}\Delta_{100}}{100}u$$

Given that $\delta_i \backsimeq 1$ the first term is almost equal to u, while the second one can be approximated as the sum of 100 realization of a Gaussian random variable with zero mean so it goes to zero. The third and fourth terms goes almost to zero because ϵ_i is very small. In the following 2 simulations each "measure" has actually been created doing an average on 100 consecutive measures with the same loads' condition as explained previously.

2) Simulation with 100 measures: In this simulation 100 "measures" have been used. The average techniquegive a very good improvement to the identification procedure. As a matter of fact the concentration matrix estimated by the garrot-type estimator is far more better than the previous simulation (even if the numbers of measures is drastically diminished). The MCT algorithm is able to estimate almost perfectly the topology of the electrical graph (Figure 15), obtaining only 3 false positive (which are moreover second neighbours) so the algorithm appears to be still valid if we manipulate data. In Figure 16, the identified graph is shown to see which edges are wrong.

3) Simulation with 1000 measures: Doing a simulation with 1000 "measures" the results of the whole algorithm is perfect. MCT returns a matrix containing only the actual links of the electrical graph. We can thus affirm that the manipulation done on the data has given good results, and the whole procedure is still valid even with noisy measures.

4) Other simulations: Different type of simulations has been done. We have changed the values of the loads, modelling them as scorrelated gaussian random variables with different mean and the results were still good. Also the change in

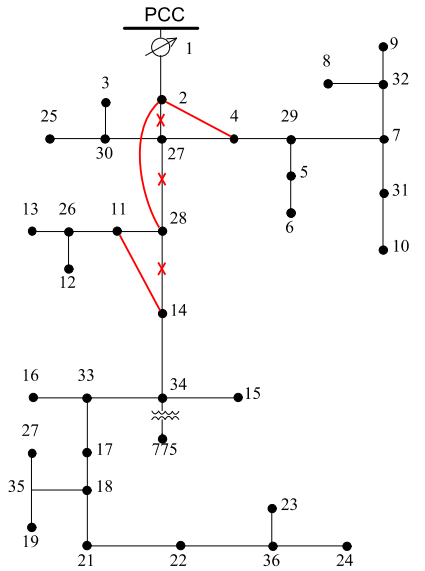

Figure 16. Topology identification corresponding tree graph obtain by MCT algorithm of Figure 15.

value of the impedences of the line has not really affected the identification. Concerning noises, it has been observed that the average expedient is not good if the phase error introduced by noise is quite higher than the one considered in the previous simulations, as can be expected. So the identification procedure needs the misures to be not too noisy, and this implies that PMUs must introduce small errors (at least on the phase).

VI. Conclusions and future developments

This work shows the possibility to identify the topology of an electric graph starting from measures taken by PMUs devices arranged in different places of the power grid. This result has been reached using a two-step-algorithm: first the concentration matrix has been estemated using a garrote-type estimator, and then the same matrix has been elaborated to get the most likely tree wich describes the electrical network (using MCT). The obtained results are satisfactory in the ideal case of noiseless measures, while the introduction of errors in measures make the procedure unreliable. However with some reasonable data-processing the results are still quite satisfactory. Neverthless data-processing is possible only if the phase error is adequately small. In practical case the procedure found works only if PMUs give accurate measures.

Future developments can involve the distribution of the algorithms in order to obtain a distributed algorithm that runs on each node. Another important feature to analyze is the possibility of creating an automatic procedure to calculate the threshold for the garrote-type estimator. In the end a more precise study on the values of the concentration matrix could give important information about the network line impedences.

References

[1] H. Sedghi and E. Jonckheere, "On the conditional mutual information in gaussian-markov structured grids," *Dept. of Electrical Engineering,University of Southern California, Los Angeles*.

[2] Y. L. Ming Yuan, "Model selection and estimation in the gaussian graphical model," 2007.

[3] S. Bolognani and S. Zampieri, "A distributed control strategy for reactive power compensation in smart microgrids," 2012.

[4] S. M. Francesco Bullo, Jorge Cortés, *Distributed Control of Robotic Networks*. Princeton University Press, 2009.

[5] S. L. Lauritzen, *Graphical Models*. Aalborg University: Clarendon Press - Oxford, 1996.

[6] L. C. Lau, "Bipartite roots of graphs," 2004.

[7] M. Fischietti, *Lezioni di ricerca operativa*. Edizioni libreria progetto Padova, second ed.

[8] W. H. Kersting, "Radial distribution test feeders," *IEEE Power Engineering Society Winter Meeting*.

[9] A. Ben-Israel and T. N. Greville, *Generalized Inverses: Theory and Applications*. Rutgers Center for Operations Research, Rutgers University, 640 Bartholomew Rd, Piscataway, 2001.

Appendix

A Theoretical covariance matrix $\Sigma_{\mathbf{u}}$

From the definition of the covariance matrix and using formula (10) concerning the approximated network we have:

$$\Sigma_{\mathbf{u}} = \mathbf{E}\left[(\mathbf{u}-\mu_{\mathbf{u}})\overline{(\mathbf{u}-\mu_{\mathbf{u}})}^T\right]$$

$$\mu_{\mathbf{u}} = \mathbf{E}[\mathbf{u}] = e^{j\varphi}U_N + \mathbf{X}\frac{\mathbf{E}[\overline{\mathbf{s}}]}{U_N}e^{j\varphi}$$

Obtaing in this way the theoretical coviariance matrix.

$$\Sigma_{\mathbf{u}} = \frac{1}{U_N^2}\mathbf{X}\mathbf{E}[(\overline{\mathbf{s}}-\overline{\mu_{\mathbf{s}}})(\mathbf{s}-\mu_{\mathbf{s}})^T]\overline{\mathbf{X}} = \frac{1}{U_N^2}\mathbf{X}Var(\mathbf{s})\overline{\mathbf{X}}$$

Remembering that the power needed to the PCC is the opposite of all powers needed by other nodes, $s_1 = -\sum_{i\neq 1} s_i$, and s_i are scorrelated and with the same variance σ, the variance of $\mathbf{s}$ has the following structure:

$$Var(\mathbf{s}) \;=\; \sigma^2\begin{bmatrix} n-1 & -1 & \cdots & -1 \\ -1 & 1 & 0 & 0 \\ \vdots & 0 & \ddots & \vdots \\ -1 & 0 & \cdots & 1 \end{bmatrix} = \sigma^2 D$$

$$D = \left[I + (n-2)\mathbf{1_0}\mathbf{1_0^T} + \mathbf{1_0}[0\ -1\cdots-1] + [0\ -1\cdots-1]^T\mathbf{1_0^T}\right]$$

Since $\mathbf{X1_0} = 0$, it's easy to prove the following equality:

$$\Sigma_{\mathbf{u}} = \frac{\sigma^2}{U_N^2}\mathbf{X}D\overline{\mathbf{X}} = \frac{\sigma^2}{U_N^2}\mathbf{X}\overline{\mathbf{X}}$$

It can be easily seen that the first row and the first column of $\mathbf{X}\overline{\mathbf{X}}$ is composed only by zero (considering that $\mathbf{X1_0} = 0$ and that $\mathbf{X} = \mathbf{X^T}$). In this way it can be written as $\mathbf{X}\overline{\mathbf{X}} = \begin{bmatrix} 0 & 0^T \\ 0 & \mathbf{S} \end{bmatrix}$, with S a $(p-1)\times(p-1)$ full-rank matrix.

B On the calculus of pseudoinverse A

The pseudoinverse A of the matrix $\mathbf{X}\overline{\mathbf{X}}$ is determine from the submatrix of $\begin{bmatrix} |\mathbf{X}|^2 & \mathbf{1} \\ \mathbf{1^T} & 0 \end{bmatrix}^{-1}$. Indeed, it can be seen that the equation below hold and it's possible to write down the system made by the constraint of the inverse operator.

$$\begin{bmatrix} \mathbf{X}\bar{\mathbf{X}} & \mathbf{1} \\ \mathbf{1^T} & 0 \end{bmatrix}\begin{bmatrix} A & v \\ v^T & c \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & 1 \end{bmatrix}$$

$$\begin{cases} \mathbf{X}\overline{\mathbf{X}}A = I - \mathbf{1}v^T \\ \mathbf{X}\overline{\mathbf{X}}v + \mathbf{1}c = 0 \\ \mathbf{1}^T A = 0 \\ \mathbf{1}^T v = 1 \end{cases} \quad\Leftrightarrow\quad \begin{cases} v = \mathbf{1_0} \\ c = 0 \\ \mathbf{X}\overline{\mathbf{X}}A = I - \mathbf{1}\mathbf{1_0^T} \\ A = \bar{A}^T \end{cases}$$

An interesting result can be found exploiting the structure of $\mathbf{X}\overline{\mathbf{X}}$:

$$\begin{bmatrix} \mathbf{X}\overline{\mathbf{X}} & \mathbf{1} \\ \mathbf{1^T} & 0 \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{0^T} & 1 \\ \mathbf{0} & \mathbf{S} & \mathbf{1} \\ 1 & \mathbf{1^T} & 0 \end{bmatrix}$$

Changing the last and second rows and columns, it becomes $\begin{bmatrix} 0 & 1 & \mathbf{0^T} \\ 1 & 0 & \mathbf{1^T} \\ \mathbf{0} & \mathbf{1} & \mathbf{S} \end{bmatrix}$

The inverse of this matrix leads to

$$\begin{bmatrix} \Delta & 1 & \alpha_1 & \cdots & \alpha_{p-1} \\ 1 & 0 & 0 & \cdots & 0 \\ \alpha_1 & 0 & & & \\ \vdots & \vdots & & \mathbf{S^{-1}} & \\ \alpha_{p-1} & 0 & & & \end{bmatrix}, \text{ with } \alpha_j = -\sum_{i=1}^{p-1} \mathbf{S^{-1}}(i,j) \text{ and } \Delta = \sum_{i,j=1}^{p-1} \mathbf{S^{-1}}(i,j).$$

Finally, changing back the second and last rows and columns, the inverse of the given matrix is

$$\begin{bmatrix} \Delta & \alpha_1 & \cdots & \alpha_{p-1} & 1 \\ \alpha_1 & & & & 0 \\ \vdots & & \mathbf{S^{-1}} & & \vdots \\ \alpha_{p-1} & & & & \vdots \\ 1 & 0 & \cdots & \cdots & 0 \end{bmatrix} \Rightarrow \quad A = \begin{bmatrix} \Delta & \alpha_1 & \cdots & \alpha_{p-1} \\ \alpha_1 & & & \\ \vdots & & \mathbf{S^{-1}} & \\ \alpha_{p-1} & & & \end{bmatrix}$$

The pseudoinverse matrix of $\mathbf{X}\overline{\mathbf{X}}$can therefore be found deleting the row and column related with the PCC, inverting the resulting matrix and reconstructing the elements of the PCC with easy calculations.

C Relationship between A and $\bar{L}L$

Starting from equation (7), it's possible to write down:

$$\bar{\mathbf{X}}\bar{L} = I - \mathbf{1}\mathbf{1_0^T} \Rightarrow \mathbf{X}\bar{\mathbf{X}}\bar{L}L = \mathbf{X}L - \mathbf{X}\mathbf{1}\mathbf{1_0^T}L \Rightarrow \mathbf{X}\bar{\mathbf{X}}\bar{L}L + \mathbf{X}\mathbf{1}\mathbf{1_0^T}L = I - \mathbf{1}\mathbf{1_0^T} \tag{1}$$

$$\mathbf{X}\bar{\mathbf{X}}A = \mathbf{X}\bar{\mathbf{X}}\bar{L}L + \mathbf{X}\mathbf{1}\mathbf{1_0^T}L \tag{2}$$

The given relationship is linear but matrix $\mathbf{X}\bar{\mathbf{X}}$ is singular, so pseudoinverses has to be used to solve the problem. The solution of a linear system like $CXB = D$, where the unknown matrix is X and C and B are singular matrix, exists if and only if there are two matrices $C^{(1)}$ and $B^{(1)}$ such that $CC^{(1)}C = C$, $BB^{(1)}B = B$ and $CC^{(1)}DB^{(1)}B = D$. $C^{(1)}$ and $B^{(1)}$ are a particular type of pseudoinverses (not unique). If the solution exists, the matrix which solves the system is

$$X = A^{(1)}DB^{(1)} + Y - A^{(1)}AYBB^{(1)}$$

where Y is an arbitrary matrix of appropriate dimension. For further details see [9]. From the properties of A follow that $\mathbf{X}\bar{\mathbf{X}}A\mathbf{X}\bar{\mathbf{X}} = \mathbf{X}\bar{\mathbf{X}}$, so A is a possible pseudoinverse of $\mathbf{X}\bar{\mathbf{X}}$ that can be used in the solution of (2). Starting from (2), the existence of a solution with A as pseudoinverse can be proved in the following way:

$$\mathbf{X}\bar{\mathbf{X}}A(\mathbf{X}\bar{\mathbf{X}}\bar{L}L + \mathbf{X}\mathbf{1}\mathbf{1_0^T}L) = \mathbf{X}\bar{\mathbf{X}}A\mathbf{X}\bar{\mathbf{X}}\bar{L}L + \mathbf{X}\bar{\mathbf{X}}A\mathbf{X}\mathbf{1}\mathbf{1_0^T}L$$

From the property of $\mathbf{X}\bar{\mathbf{X}}$ and A it's easy to see that the first term in the sum equals $\mathbf{X}\bar{\mathbf{X}}\bar{L}L$. Using (1) and the property of A to analyze the second therm:

$$\mathbf{X}\bar{\mathbf{X}}A - \mathbf{X}\bar{\mathbf{X}}A\mathbf{X}\bar{\mathbf{X}}\bar{L}L - \mathbf{X}\bar{\mathbf{X}}A\mathbf{1}\mathbf{1_0^T} = I - \mathbf{1}\mathbf{1_0^T} - \mathbf{X}\bar{\mathbf{X}}\bar{L}L = \mathbf{X}\mathbf{1}\mathbf{1_0^T}L$$

so

$$\mathbf{X}\bar{\mathbf{X}}A\mathbf{X}\bar{\mathbf{X}}\bar{L}L + \mathbf{X}\bar{\mathbf{X}}A\mathbf{X}\mathbf{1}\mathbf{1_0^T}L = \mathbf{X}\bar{\mathbf{X}}\bar{L}L + \mathbf{X}\mathbf{1}\mathbf{1_0^T}L.$$

The existence of a solution has thus been proved. The general solution is

$$A = A\mathbf{X}\bar{\mathbf{X}}\bar{L}L + A\mathbf{X}\mathbf{1}\mathbf{1}_\mathbf{0}^\mathbf{T}L + Y - A\mathbf{X}\bar{\mathbf{X}}Y$$

with Y an arbitrary matrix of appropriate dimensions. Among all the possible choices for Y the null matrix is considered. Analyzing the first term,

$$A\mathbf{X}\bar{\mathbf{X}}\bar{L}L = (I - \mathbf{1}\mathbf{1}_\mathbf{0}^\mathbf{T})\bar{L}L = \bar{L}L$$

the solution can be written as:

$$A(I - \mathbf{X}\mathbf{1}\mathbf{1}_\mathbf{0}^\mathbf{T}L) = \bar{L}L \tag{3}$$

Without any loss of generality, suppose that the node which is linked with the PCC is labelled as 2. In this way the first row of L has the following form: $[\ \alpha \quad -\alpha \quad 0 \quad \cdots \quad 0\ ]$, $\alpha \neq 0$, since the sum of the elements of any row in L has to be 0. Moreover, remembering that the first row of $\mathbf{X}$ is made by zero entries, the sum of all the elements of the first row is 0. Calling $a_i = \sum_{j=1}^{p} \mathbf{X}(i,j)$, it's easy to prove that

$$(I - \mathbf{X}\mathbf{1}\mathbf{1}_\mathbf{0}^\mathbf{T}L) = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ \alpha a_2 & 1+\alpha a_2 & 0 & \cdots & 0 \\ \alpha a_3 & -\alpha a_3 & & & \\ \vdots & \vdots & & I_{p-2} & \\ \alpha a_p & -\alpha a_p & & & \end{bmatrix}$$

This matrix is invertible if and only if $1 + \alpha a_2 \neq 0$, that is $\alpha \neq -1/a_2$. This condition appears to be almost always met (at least in the simulation done for this work has always been met), so (3) can be solved using the inverse:

$$A = \bar{L}L(I - \mathbf{X}\mathbf{1}\mathbf{1}_\mathbf{0}^\mathbf{T}L)^{-1}$$

The inverse of the latter matrix has the following form:

$$(I - \mathbf{X}\mathbf{1}\mathbf{1}_\mathbf{0}^\mathbf{T}L)^{-1} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -\frac{\alpha a_2}{1+\alpha a_2} & \frac{1}{1+\alpha a_2} & 0 & \cdots & 0 \\ \frac{\alpha a_3 + 2\alpha^2 a_3 a_2}{1+\alpha a_2} & \frac{\alpha a_3}{1+\alpha a_2} & & & \\ \vdots & \vdots & & I_{p-2} & \\ \frac{\alpha a_p + 2\alpha^2 a_p a_2}{1+\alpha a_2} & \frac{\alpha a_p}{1+\alpha a_p} & & & \end{bmatrix}$$

Due to the form of the inverse, A turns out to have the same element of $\bar{L}L$ except for the elements in position $(1,1)$, $(1,2)$, $(2,1)$, $(2,2)$.

D MaxDet and Nonnegative Garrote Estimator

We want to show the relation between the maxdet problem and the nonnegative Garrote estimator in order to apply YALMIP functions in the optimization of our semi-definite problem. Now, taking into account equation (12) and (13), we can see that

$$\begin{aligned} G(x) &= \hat{C} \\ &= \begin{bmatrix} d_{11}\tilde{c}_{11} & \cdots & d_{1n}\tilde{c}_{1n} \\ \vdots & \ddots & \vdots \\ d_{n1}\tilde{c}_{n1} & \cdots & d_{nn}\tilde{c}_{nn} \end{bmatrix} \\ &= d_{11}\begin{bmatrix} \tilde{c}_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} + d_{kh}\begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \tilde{c}_{kh} & \vdots \\ 0 & \cdots & 0 \end{bmatrix} + \cdots + d_{nn}\begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \tilde{c}_{nn} \end{bmatrix} \end{aligned}$$

and $b^T x = tr(\hat{C}\bar{A}) = \begin{bmatrix} \tilde{c}_{11}\bar{a}_{11} & \tilde{c}_{12}\bar{a}_{12} & \dots & \tilde{c}_{hk}\bar{a}_{hk} & \dots & \tilde{c}_{nn}\bar{a}_{nn} \end{bmatrix} \cdot \begin{bmatrix} d_{11} & d_{12} & \dots & d_{hk} & \dots & d_{nn} \end{bmatrix}^T$. So, it has been proven that $\hat{C}$ can be written as an affine function of d_{ij}, while $tr(\hat{C}\tilde{\Sigma}_\mathbf{u})$ can be written as a linear function

of d_{ij}, as needed by MaxDet problem formulation. Concerning the constraint $\sum_{i\neq j} d_{ij} \leq t \;\; d_{ij} \geq 0$ we can write a similar relation as made for G(x):

$$
\begin{aligned}
F(x) &= F_0 + d_{11}F_{11} + d_{12}F_{12} + \cdots + d_{1n}F_{1n} + \cdots + d_{nn}F_{nn} \\
&= \begin{bmatrix} +t & 0 & \dots & 0 \\ 0 & \ddots & 0 & \vdots \\ \vdots & 0 & \ddots & \vdots \\ 0 & \dots & \dots & 0 \end{bmatrix} + d_{11} \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \vdots \\ \vdots & 0 & \ddots & \vdots \\ 0 & \dots & \dots & 0 \end{bmatrix} + d_{12} \begin{bmatrix} -1 & 0 & \dots & 0 \\ 0 & \ddots & 0 & \vdots \\ \vdots & 0 & 1 & \vdots \\ 0 & \dots & \dots & 0 \end{bmatrix} + \cdots \\
&= + \; d_{1n} \begin{bmatrix} -1 & 0 & \dots & 0 \\ 0 & \ddots & 0 & \vdots \\ \vdots & 0 & 1 & \vdots \\ 0 & \dots & \dots & 0 \end{bmatrix} + \cdots + d_{nn} \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & \ddots & 0 & \vdots \\ \vdots & 0 & \ddots & \vdots \\ 0 & \dots & \dots & 1 \end{bmatrix} \\
&= \begin{bmatrix} -\sum_{i\neq j} d_{ij} + t & & & & & & \\ & +d_{11} & & & & & \\ & & +d_{12} & & & & \\ & & & +d_{13} & & & \\ & & & & \ddots & & \\ & & & & & +d_{nn} \end{bmatrix}
\end{aligned}
$$

where F_{ij} with $i \neq j$ is the $(n^2+1)\times(n^2+1)$ matrix with -1 in position (1,1) and 1 in position $(n(i-1)+j+1, n(i-1)+j+1)$, whereas if $i = j$ the matrix F_{ij} is made by the element 1 in position $(n(i-1)+j+1, n(i-1)+j+1)$. Moreover, it's easy to show that the constraint $F(x) \geq 0$ gives:

$$F(x) \;\; \geq \;\; 0 \Longleftrightarrow z^T F(x) z \geq 0 \; \forall z$$

$$z^T F(x) z = z_1^2(-d_{12} - \cdots - d_{1n} - \cdots - d_{n,n-1} + t) + z_2^2 d_{11} + \cdots + z_N^2 d_{nn}$$

if $z_i = 0$ for all $i \neq 1$ and $z_1 = 1$, then

$$-\sum_{i\neq j} d_{ij} + t \geq 0$$

If $z_i = 0$ for all $i \neq 2$ and $z_2 = 1$, then

$$d_{11} \geq 0$$

and so forth. Thus, the semi definite constraint $F(x) \geq 0$ is equivalent to $\sum_{i\neq j} d_{ij} \leq t \;\; d_{ij} \geq 0$

E Numerical property of Laplacian element

E.1 Unweighted Laplacian

Theorem 1. *First neighbours in an unweighted Laplacian L_u are negative, while second neighbours and the elements on the diagonal are positive.*

Proof. From the property of the Laplacian matrix the following statements are easy to prove:

$$L_u(i,i) = -\sum_{k\neq i} L_u(i,k)$$

$$L_u(i,j) < 0 \Longleftrightarrow (i,j) \in G$$

Considering now the element (i,j), $i \neq j$ L_u^2 and supposing that $L_u(i,j) < 0$, that is nodes i, j are firt neighbour:

$$
\begin{aligned}
L_u^2(i,j) &= \sum_{k=1}^{n} L_u(i,k)L_u(k,j) = \sum_{k\neq i,j} L_u(i,k)L_u(k,j) + L_u(i,i)L_u(i,j) + L_u(i,j)L_u(j,j) \\
&= \sum_{k\neq i,j} L_u(i,k)L_u(k,j) - L_u(i,j)\sum_{k\neq i} L_u(i,k) - L_u(i,j)\sum_{k\neq j} L_u(k,j)
\end{aligned}
$$

Writing the summations explicitly:

$$
\begin{aligned}
L_u^2(i,j) &= L_u(i,1)L_u(1,j) + L_u(i,2)L_u(2,j) + \ldots + L_u(i,n)L_u(n,j) + \\
&+ L_u(i,j)\left[-L_u(i,1) - L_u(i,2) - \ldots - L_u(i,j) - \ldots - L_u(i,n)\right] + \\
&+ L_u(i,j)\left[-L_u(j,1) - L_u(j,2) - \ldots - L_u(i,j) - \ldots - L_u(j,n)\right]
\end{aligned}
$$

Writing first terms as $L_u(i,k)L_u(k,j) = \frac{1}{2}L_u(i,k)L_u(k,j) + \frac{1}{2}L_u(k,1)L_u(k,j)$:

$$
\begin{aligned}
L_u^2(i,j) &= L_u(i,1)\left[\frac{1}{2}L_u(1,j) - L_u(i,j)\right] + L_u(1,j)\left[\frac{1}{2}L_u(i,1) - L_u(i,j)\right] \\
&+ L_u(i,2)\left[\frac{1}{2}L_u(2,j) - L_u(i,j)\right] + L_u(2,j)\left[\frac{1}{2}L_u(2,j) - L_u(i,j)\right] + \ldots + \\
&+ L_u(i,n)\left[\frac{1}{2}L_u(n,j) - L_u(i,j)\right] + L_u(n,j)\left[\frac{1}{2}L_u(n,j) - L_u(i,j)\right] - 2L_u(i,j)^2
\end{aligned}
$$

Since in L_u all the elements outside the diagonal are either 0 or -1, and $L(i,j) = -1$, all the terms between brackets are positive ($\frac{1}{2}L(k,h) - L(i,j) > 0$). Finally the coefficient of each term in brackets is negative, so $L_u^2(i,j)$ is negative. Concerning terms representing second neighbours in L_u^2, they are positive because they can be written as:

$$
L_u^2(i,j) = \sum_{k=1}^{n} L_u(i,k)L_u(k,j) = \sum_{k\neq i,j} L_u(i,k)L_u(k,j) + L_u(i,i)L_u(i,j) + L_u(i,j)L_u(j,j)
$$

and $L_u(i,j) = 0$ since i and j are second negihbours; so

$$
L_u^2(i,j) = \sum_{k=1}^{n} L_u(i,k)L_u(k,j) = \sum_{k\neq i,j} L_u(i,k)L_u(k,j)
$$

which is a summation of positive (or null) terms. The diagonal elements in L_u^2 are positive because they can be written as norms. □

E.2 Weighted Laplacian in Chain network structure

In a chain-structured electric network sign property is easy to verify even in the weighted Laplacian

Figure 1: Infinite chain-structured graph

The weighted Laplacian matrix associated to an electric network as the one in Figure 1 is like

$$
L = \begin{bmatrix}
\ddots & -\epsilon & 0 & \vdots & \vdots & \iddots \\
-\epsilon & \alpha+\epsilon & -\alpha & 0 & 0 & \cdots \\
0 & -\alpha & \alpha+\beta & -\beta & 0 & \cdots \\
\cdots & 0 & -\beta & \beta+\gamma & -\gamma & 0 \\
\cdots & 0 & 0 & -\gamma & \gamma+\delta & -\delta \\
\iddots & \vdots & \vdots & 0 & -\delta & \ddots
\end{bmatrix}
$$

with $\alpha = a_\alpha + jb_\alpha,\ a_\alpha > 0,\ b_\alpha < 0$ (these considerations are valid for all the non-zero elements in the matrix). Calling $F = \bar{L}L$, the demonstration looks at just one element wich represent a first neighbour in F, for example the element in position $(4,3)$:

$$F(4,3) = -\bar{\beta}(\alpha+\beta) - (\bar{\beta}+\bar{\gamma})\beta = -(a_\beta - jb_\beta)(a_\alpha + jb_\alpha + a_\beta + jb_\beta) - (a_\beta - jb_\beta + a_\gamma - jb_\gamma)(a_\beta + jb_\beta)$$

Considering only the real part of $F(4,3)$:

$$-a_\beta a_\alpha - 2a_\beta^2 - b_\beta b_\alpha - 2b_\beta^2 - a_\gamma a_\beta - b_\gamma b_\beta < 0$$

The real part is then negative (the same proof holds for all the first neighbours in F). The elements corresponding to second neighbours in F are in the second subidiagonal and in the second upperdiagonal (this is a consequence of the form of the Laplacian). To study the sign of the real part of second neighbours the element in position (5,3) is analyzed:

$$F(5,3) = (-\bar{\gamma})(-\beta) = (a_\gamma - jb_\gamma)(a_\beta + jb_\beta)$$

The real part of $F(5,3)$ is then:

$$a_\gamma a_\beta + b_\gamma b_\beta > 0$$

as stated.

E.3 Laplacian of generic Networks

In a generic electrical network, $\bar{L}L$ has positive real part elements on the diagonal and for the entries concerning second neighbours. Calling $F = \bar{L}L$ and cosidering that $L = L^T$ the diagonal elements are real and positive:

$$F(i,i) = \sum_{k=1}^{p} \bar{L}(i,k)L(k,i) = \sum_{k=1}^{p} \bar{L}(k,i)L(k,i) > 0.$$

Considering the entry in F for a second neighbour, for example $F(i,j)$ with $L(i,j) = 0$:

$$F(i,j) = \sum_{k=1}^{p} \bar{L}(i,k)L(k,j) = \sum_{k\neq i,j} \bar{L}(i,k)L(k,j) + \bar{L}(i,i)L(i,j) + \bar{L}(i,j)L(j,j) = \sum_{k\neq i,j} \bar{L}(i,k)L(k,j)$$

Considering the real part of a generic term of the summation, with $L(i,k) = a + jb\,,\ a \leq 0,\ b \geq 0,\ L(k,j) = c + jd\,,\ c \leq 0,\ d \geq 0$

$$\bar{L}(i,k)L(k,j) = ac + bd \geq 0$$

The real part of the summation is therefore positive.

F Signs of the weighted Laplacian

In this Appendix the dimonstration concernig sign in weighted Laplacian matrix is proven. The property is as follow: considering L the real part of its non zero elements out of the diagonal is positive, and the imaginary part of the same is negative. Considering the impedence of any edge of the electric network, this has real and imaginary part which are positive (the line is a passive element). Due to this fact its inverse has positive real part and negative imaginary part. As a matter of fact

$$z_e = a_e + jb_e,\ a_e, b_e > 0 \ \Rightarrow \quad \frac{1}{z_e} = \frac{1}{a_e + jb_e} = \frac{a_e - jb_e}{a_e^2 + b_e^2}.$$

A generic element of the weighted Laplacian can be computed as

$$L(i,j) = \sum_{k=1}^{p-1} A(k,i)\frac{1}{z_k}A(k,j)$$

The incidence matrix A is such that only two elements per row are different from zero, a 1 and a -1. Exploiting this fact, each term of the summation can be zero or $-\frac{1}{z_k}$ and this proves the statement. This implies also that the elements on the diagonal has positive real part and negative imaginary one (since the sum of the elements in a row is 0).