

# Finding all voltage solutions to tree distribution networks

## Abstract

We present an algorithm that finds all voltage solutions to the power flow equations on trees. We prove that this algorithm will find all real solutions. Moreover, the algorithm finds a univariate polynomial whose roots govern the voltages of the entire tree. We give versions of this algorithm for both the lossless case and the loss case. We also bound the complexity of the algorithm in terms of the max degree of the network. We use this to show that in physically motivated radial networks, our algorithm runs much faster than in the worst-case, scaling more like the number of real solutions.

## 1 Introduction

## 2 Preliminaries

We consider an  $n$  bus electric power system. We label the buses as  $1, \dots, n$ . The topology of the network is given by an graph  $G$  and an admittance matrix  $Y = G + jB$  where  $j = \sqrt{-1}$ . The system is *lossless* if  $G = 0$ . Each bus  $i$  has an associated power injection given by  $P_i + jQ_i$ ,  $P_i, Q_i \in \mathbb{R}$ . The bus also has voltage phasor  $x_i + jy_i$  with  $x_i, y_i \in \mathbb{R}$ . We make a choice of *slack bus* in the system which provides a reference voltage angle. In our setup, we let bus 1 be the slack bus, with  $x_1 = 1, y_1 = 0$ . The  $PQ$  equations for power flow are then given by:

$$\sum_{k=1}^n G_{i,k}(x_i^2 + y_i^2 - x_i x_k - y_i y_k) + B_{ik}(x_k y_i - x_i y_k) = P_i \quad (2.1)$$

$$\sum_{k=1}^n -B_{ik}(x_i^2 + y_i^2 - x_i x_k - y_i y_k) + G_{ik}(x_k y_i - x_i y_k) = Q_i \quad (2.2)$$

We will consider  $G$  a directed rooted tree with the slack bus the root and edges pointing away from the root. For a vertex  $v$ , let  $p(v)$  denote the parent of a bus that isn't the slack bus, and let  $c(v)$  denote all children of a vertex  $v$ .

### 2.1 Lossless

Note that in the case that we are lossless, the equations become:

$$f_i = \sum_{k=1}^n B_{ik}(x_k y_i - x_i y_k) - P_i = 0 \quad (2.3)$$

$$g_i = \sum_{k=1}^n -B_{ik}(x_i^2 + y_i^2 - x_i x_k - y_i y_k) - Q_i = 0 \quad (2.4)$$

Note that if we let  $\alpha_{ik} = x_i x_k + y_i y_k$ ,  $\beta_{ik} = x_k y_i - x_i y_k$ ,  $R_i = x_i^2 + y_i^2$ , this becomes:

$$f_i = \sum_{k=1}^n B_{ik} \beta_{ik} - P_i = 0 \quad (2.5)$$

$$g_i = \sum_{k=1}^n -B_{ik}(R_i - \alpha_{ik}) - Q_i = 0 \quad (2.6)$$

To solve the  $PQ$  equations, we will use resultants from elimination theory. Let  $f = a_d y^d + a_{d-1} y^{d-1} + \dots + a_0$ ,  $g = b_e y^e + b_{e-1} y^{e-1} + \dots + b_0 \in \mathbb{R}[x_1, \dots, x_m, y]$ , so that the  $a_i, b_j$  are in  $\mathbb{R}[x_1, \dots, x_m]$ . Then the *Sylvester matrix* of  $f$  and  $g$  with respect to  $y$  is a  $(d+e) \times (d+e)$  matrix in  $\mathbb{R}[x_1, \dots, x_m]$  whose entries are formed as follows:

Fill the matrix by starting in the upper-left corner and filling in the coefficients of  $y$  in  $f$  (in decreasing order of power of  $y$ ). Form the second row by shifting the entries down one and across by one. Repeat until the coefficient  $a_0$  is in the right-most entry of a row. Do the same with the coefficients of  $g$ .

For example, if  $f = a_4 x^4 + a_3 x^3 + a_2 x^2 + a_1 x + a_0$ ,  $g = b_2 x^2 + b_1 x + b_0$ , the Sylvester matrix with respect to  $x$  is:

$$\begin{pmatrix} a_4 & a_3 & a_2 & a_1 & a_0 & 0 \\ 0 & a_4 & a_3 & a_2 & a_1 & a_0 \\ b_2 & b_1 & b_0 & 0 & 0 & 0 \\ 0 & b_2 & b_1 & b_0 & 0 & 0 \\ 0 & 0 & b_2 & b_1 & b_0 & 0 \\ 0 & 0 & 0 & b_2 & b_1 & b_0 \end{pmatrix}$$

The resultant of  $f, g$  with respect to  $y$ , denoted  $\text{res}_y(f, g)$  is then the determinant of the associated Sylvester matrix. Note that if  $f, g \in \mathbb{R}[x_1, \dots, x_m, y]$ , then  $\text{res}_y(f, g) \in \mathbb{R}[x_1, \dots, x_m]$ . Moreover, resultants satisfy the following useful fact. For  $z_1, \dots, z_m \in \mathbb{R}$ ,  $z_1, \dots, z_m$  is a root of  $\text{res}_y(f, g)$  iff there is some  $z \in \mathbb{R}$  such that  $(z_1, \dots, z_m, z)$  is a common zero of  $f$  and  $g$ . Informally,  $\text{res}_y(f, g)$  eliminates the variable  $y$  in the two equations.

It has been known for decades that one can in theory use iterated resultants to reduce multivariate equations to univariate equations, and then solve. Unfortunately, for generic systems of equations, this algorithm has a high time complexity since a given variable  $y$  must be eliminated from all equations. The number of resultants needed increases, as does the resulting degrees of the next stage of polynomials (since performing a resultant generally increases the degree).

Our main contribution comes from showing that in the case of power flow equations for a tree network, both the structure of the equations and topology of the graph allow us to significantly reduce the cost of this kind of approach. Moreover, we can guarantee that this method will require even fewer resultant computations if the tree has small average degree.

### 3 Algorithm

We first motivate and present the algorithm in the lossless setting. Recall that in the case that we are lossless, the equations become:

$$\begin{aligned} \sum_{k=1}^n B_{ik}(x_k y_i - x_i y_k) &= P_i \\ \sum_{k=1}^n B_{ik}(x_i^2 + y_i^2 - x_i x_k - y_i y_k) &= -Q_i \end{aligned}$$

We let  $\alpha_{ik} = x_i x_k + y_i y_k$ ,  $\beta_{ik} = x_k y_i - x_i y_k$ ,  $R_i = x_i^2 + y_i^2$ . Note that  $\alpha_{ik} = \alpha_{ki}$  and  $\beta_{ik} = -\beta_{ki}$ . Taking the sum over edges of  $G$ , this becomes:

$$\begin{aligned} B_{ip(i)}\beta_{ip(i)} + \sum_{k \in c(i)}^n B_{ik}\beta_{ik} &= P_i \\ B_{ip(i)}(R_i - \alpha_{ip(i)}) + \sum_{k=1}^n B_{ik}(R_i - \alpha_{ik}) &= -Q_i \end{aligned}$$

We also include the easily-verified identities:

$$\alpha_{ik}^2 + \beta_{ik}^2 = R_i R_k$$

Note that if we solve for  $\alpha_{ik}, \beta_{ik}, R_i$  for all  $(i, k) \in E$ , we will have solved for all  $x_i, y_i$ . To see this, first consider any bus  $i$  connected to the slack bus 1. Since  $x_1 = 1, y_1 = 0$ , we find that  $\alpha_{i1} = x_i, \beta_{i1} = y_i$ . More generally, if we have values for  $\beta_{ik}, x_k, y_k$  and  $(i, k) \in E$ , then we can use linear algebra to solve for  $x_i, y_i$  by using equations:

$$\begin{aligned}\alpha_{ik} &= x_i x_k + y_i y_k \\ \beta_{ik} &= x_k y_i - x_i y_k\end{aligned}$$

In the case that  $G$  is a tree, note that we can then start at the slack bus and propagate down the tree to solve for  $x_i, y_i$  given values of  $\alpha_{ik}, \beta_{ik}$ .

This change of variables has advantages over the  $x, y$  coordinates. Under this formulation, we have the following lemma that we can use to solve for the  $\beta_{ik}$ .

**Lemma 3.1.** *For buses  $i, k$ , we write  $i \rightarrow k$  if there is a path from  $i$  to  $k$  in the directed rooted tree  $G$ . Note that this is equivalent to saying that  $k$  is a descendant of  $i$ . The  $\beta_{ik}$  satisfy the following recursive identity:*

$$B_{ip(i)}\beta_{ip(i)} = \sum_{i \rightarrow k} P_k \quad (3.1)$$

*Proof.* If  $i$  is a leaf, this equation is a restatement of the  $P$  equation above. Otherwise, we rearrange the  $P$  equation to get:

$$\begin{aligned}B_{ip(i)}\beta_{ip(i)} &= P_i - \sum_{k \in c(i)} B_i \beta_{ik} \\ &= P_i - \sum_{k \in c(i)} B_i \beta_{ki} \\ &= P_i + \sum_{k \in c(i)} B_i \beta_{kp(k)} \\ &= P_i + \sum_{k \in c(i)} \sum_{k \rightarrow \ell} P_\ell \\ &= \sum_{i \rightarrow k} P_k\end{aligned}$$

□

Therefore, we can solve for the  $\beta_{ik}$  in linear time using a recursive algorithm. We now wish to solve for the  $\alpha_{ik}, R_i$ . We can treat the  $\beta_{ik}$  as constants by the above. We will solve by using elimination of variables to reduce this question to solving a univariate polynomial.

We first start with an example. Say that we have the following graph:

Consider the equations at a leaf node  $i$ . Then the  $Q$  equation and the associated identities above give us the equations:

$$\begin{aligned}B_{n,n-1}(R_n - \alpha_{n,n-1}) &= P_n \\ \alpha_{n,n-1}^2 + \beta_{n,n-1}^2 &= R_n R_{n-1}\end{aligned}$$

Note that these are the only equations in which  $R_i$  appears due to the topology of the graph. To eliminate  $R_i$ , we need only use the equations above to substitute, which leads to the new equation:

$$B_{n,n-1}(\alpha_{n,n-1}^2 + \beta_{n,n-1}^2 - \alpha_{n,n-1} R_{n-1}) = P_n R_{n-1} \quad (3.2)$$

Looking at the  $Q$  equation of bus  $j$ , we get:

$$B_{n-1,n-2}(R_{n-1} - \alpha_{n-1,n-2}) + B_{n-1,n}(R_{n-1} - \alpha_{n,n-1}) = Q_{n-1}$$

Note that this is linear in  $\alpha_{ij}$ . Therefore we can solve for  $\alpha_{ij}$  and substitute in to 3.2. The result is an equation with variables  $\alpha_{jk}, R_j$ . Hence, we've eliminated  $\alpha_{ij}, R_i$ . By iterating this procedure, we can eventually eliminate all variables except  $\alpha_{12} = x_2$ . Solving this polynomial, we get all possible values of  $x_2$ . Since we can use

## 4 Complexity

Let  $G$  be a graph representing our network. We wish to determine how many resultants we need to compute

## 5 Empirical Results

## 6 Conclusion