A Submodular Approach for Electricity Distribution Network Reconfiguration

Ali Khodabakhsh

University of Texas at Austin

ali.kh@utexas.edu

Co-authors: Ger Yang, Soumya Basu, Evdokia Nikolova,
Michael Caramanis, Thanasis Lianeas, Emmanouil Pountourakis

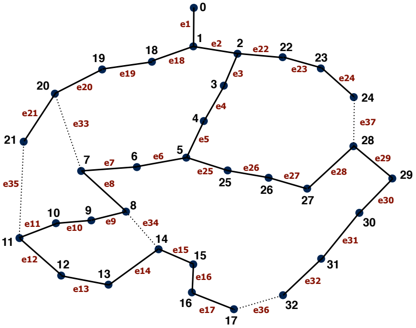
HICSS-51, January 4, 2018

Introduction

- Distribution networks are usually
  - built as interconnected mesh networks.
  - configured and operated as radial networks.
- Switches
  - allow dynamic reconfiguration of the distribution network.
  - opening or closing of a switch corresponds to the removal or addition of an edge.
- Distribution Network Reconfiguration (DNR): How to find a spanning tree (or a spanning forest) that optimizes an appropriate objective function, such as total losses or load balancing.

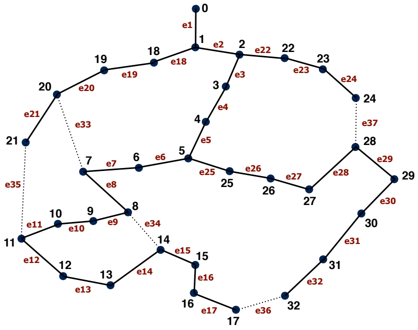
Problem Statement

- Given a graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$
  - $\mathcal{N}$: set of nodes/users/buses
  - $\mathcal{E}$: set of edges/lines/branches
- Given power demands (p_i, q_i) for all $i \in \mathcal{N} \backslash \{0\}$
- Given line resistances R_e



Problem Statement

- Given a graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$
  - $\mathcal{N}$: set of nodes/users/buses
  - $\mathcal{E}$: set of edges/lines/branches
- Given power demands (p_i, q_i) for all $i \in \mathcal{N} \setminus \{0\}$
- Given line resistances R_e



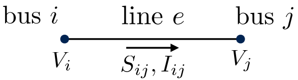
$\Rightarrow$ Find a spanning tree such that the total loss is minimized.

Contributions

1. We prove that the DNR problem is NP-hard.
2. We formulate the DNR problem as a supermodular minimization problem subject to a single matroid basis constraint.
3. We propose a polynomial time algorithm based on local search methods, and we give a performance bound on the result.
4. We show that our algorithm is equivalent to the branch exchange algorithm.

   $\Rightarrow$ This will be the first theoretic result on why the branch exchange algorithm performs well in practice.

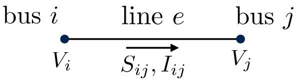
Power Flow Equations



- Loss of line $e = \{i, j\}$: $L_e = R_e \times |I_{ij}|^2$
- Relaxed model in radial networks

$$|V_i|^2 |I_{ij}|^2 = |S_{ij}|^2 = P_{ij}^2 + Q_{ij}^2$$

Power Flow Equations



- Loss of line $e = \{i, j\}$: $L_e = R_e \times |I_{ij}|^2$
- Relaxed model in radial networks

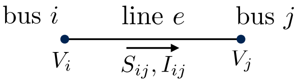
$$|V_i|^2 |I_{ij}|^2 = |S_{ij}|^2 = P_{ij}^2 + Q_{ij}^2$$

Assumption 1

Voltage variation is negligible: $|V_i| = 1 \; p.u., \forall i$

$$\Rightarrow L_e = R_e \times [P_{ij}^2 + Q_{ij}^2]$$

Power Flow Equations



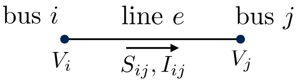
- Loss of line $e = \{i, j\}$: $L_e = R_e \times [P_{ij}^2 + Q_{ij}^2]$
- Let c_e be the successors of branch e.

Assumption 2

The impact of losses on line flows is negligible relative to the impact of power demands.

- $P_{ij} = \sum_{i \in c_e} p_i$

Power Flow Equations



- Loss of line $e = \{i, j\}$: $L_e = R_e \times [P_{ij}^2 + Q_{ij}^2]$
- Let c_e be the successors of branch e.

Assumption 2

The impact of losses on line flows is negligible relative to the impact of power demands.

- $P_{ij} = \sum_{i \in c_e} p_i$

$$\Rightarrow L_e = R_e \left[\left(\sum_{i \in c_e} p_i \right)^2 + \left(\sum_{i \in c_e} q_i \right)^2 \right]$$

Optimization Problem

- Finding a spanning tree with minimum total loss:

$$\min_{ST} \quad \sum_{e \in ST} R_e \left[\left(\sum_{i \in c_e} p_i \right)^2 + \left(\sum_{i \in c_e} q_i \right)^2 \right] \tag{P1}$$

Optimization Problem

- Finding a spanning tree with minimum total loss:

$$\min_{ST} \quad \sum_{e \in ST} R_e \left[\left(\sum_{i \in c_e} p_i \right)^2 + \left(\sum_{i \in c_e} q_i \right)^2 \right] \qquad \text{(P1)}$$

Theorem

Distribution Network Reconfiguration problem (P1) *is strongly NP-hard.*

Optimization Problem

- Finding a spanning tree with minimum total loss:

$$\min_{ST} \quad \sum_{e \in ST} R_e \left[\left(\sum_{i \in c_e} p_i \right)^2 + \left(\sum_{i \in c_e} q_i \right)^2 \right] \qquad \text{(P1)}$$

Theorem

Distribution Network Reconfiguration problem (P1) *is strongly NP-hard.*

$\Rightarrow$ We try to find an approximation algorithm.

Polynomial Description of Spanning Trees

- $x_e = \begin{cases} 1 & \text{edge } e \text{ is in the tree} \\ 0 & \text{otherwise} \end{cases}$
- $y_{ij}^k = \begin{cases} 1 & \text{there is a path from } i \text{ to } k \text{ starting with edge } (i,j) \\ 0 & \text{otherwise} \end{cases}$

$$\text{ST} \begin{cases} \sum_{e \in \mathcal{E}} x_e = n - 1 \\ y_{ij}^k + y_{ji}^k = x_e & \forall e = \{i,j\} \in \mathcal{E}, \forall k \in \mathcal{N} \\ x_e + \sum_{k \neq i,j} y_{ik}^j = 1 & \forall i,j \in \mathcal{N} : e = \{i,j\} \in \mathcal{E} \\ x_e, y_{ij}^k, y_{ji}^k \in \{0,1\} & \forall e = \{i,j\} \in \mathcal{E}, \forall k \in \mathcal{N} \end{cases}$$

Integer Program Formulation

$$
\begin{aligned}
\min \quad & \sum_{\{i,j\}\in\mathcal{E}} R_{ij} y_{ji}^0 \left[\left(\sum_{k\in\mathcal{N}} y_{ij}^k p_k \right)^2 + \left(\sum_{k\in\mathcal{N}} y_{ij}^k q_k \right)^2 \right] \\
s.t. \quad & \sum_{e\in\mathcal{E}} x_e = n-1 \\
& y_{ij}^k + y_{ji}^k = x_e \qquad \forall e=\{i,j\}\in\mathcal{E}, \forall k\in\mathcal{N} \\
& x_e + \sum_{k\neq i,j} y_{ik}^j = 1 \qquad \forall i,j\in\mathcal{N} : e=\{i,j\}\in\mathcal{E} \\
& x_e, y_{ij}^k, y_{ji}^k \in \{0,1\} \qquad \forall e=\{i,j\}\in\mathcal{E}, \forall k\in\mathcal{N}
\end{aligned}
\tag{P2}
$$

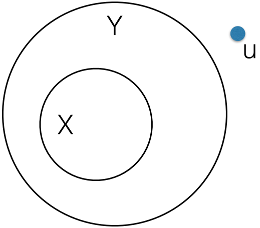
Set Function Definitions

Definition

A set function $f : 2^V \mapsto \mathbb{R}$ with a ground set V is supermodular if:

$$f(X \cup \{u\}) - f(X) \leq f(Y \cup \{u\}) - f(Y),$$

for every $X \subseteq Y \subseteq V$, $u \in V \setminus Y$.



Set Function Definitions

Definition

A set function $f : 2^V \mapsto \mathbb{R}$ with a ground set V is supermodular if:

$$f(X \cup \{u\}) - f(X) \leq f(Y \cup \{u\}) - f(Y),$$

for every $X \subseteq Y \subseteq V$, $u \in V \backslash Y$.

Definition

Let V be a finite set, and let $\mathcal{I}$ be a collection of subsets of V. The pair $\mathcal{M} = (V, \mathcal{I})$ is a matroid if the following conditions hold:

1. If $B \in \mathcal{I}$, then $A \in \mathcal{I}$ for all $A \subseteq B$,
2. If $A, B \in \mathcal{I}$ and $|A| < |B|$, then there exists $v \in B \backslash A$ such that $A \cup \{v\} \in \mathcal{I}$.

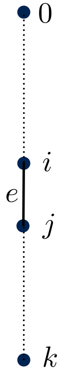
Supermodular Formulation

For any $A \subseteq \mathcal{E}$, define:

$$f(A) = \sum_{\{i,j\} \in \mathcal{E}} R_{ij} z_{ji}^{0} \left[\left(\sum_{k \in \mathcal{N}} z_{ij}^{k} p_k \right)^2 + \left(\sum_{k \in \mathcal{N}} z_{ij}^{k} q_k \right)^2 \right]$$

where z_{ij}^{k} is the number of paths from i to k starting with edge (i, j).

$$\begin{aligned} \min \quad & f(A) \\ s.t. \quad & A \text{ is a spanning tree} \end{aligned} \qquad \text{(P3)}$$



Theorems

$$\begin{aligned} \min \quad & f(A) \\ s.t. \quad & A \text{ is a spanning tree} \end{aligned} \tag{P3}$$

Theorems

$$\begin{aligned} \min \quad & f(A) \\ s.t. \quad & A \text{ is a spanning tree} \end{aligned} \tag{P3}$$

Theorem

$f(A)$ is a supermodular function over $\mathcal{E}$, if $p_i, q_i \geq 0$ for all i.

Theorems

$$\begin{aligned} \min \quad & f(A) \\ s.t. \quad & A \text{ is a spanning tree} \end{aligned} \tag{P3}$$

Theorem

$f(A)$ is a supermodular function over $\mathcal{E}$, if $p_i, q_i \geq 0$ for all i.

Theorem

For any connected graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$ there is a matroid $\mathcal{M}(\mathcal{E}, \mathcal{I})$ such that the bases of $\mathcal{M}$ are the spanning trees of $\mathcal{G}$.

Theorems

$$\begin{aligned} \min \quad & f(A) \\ s.t. \quad & A \text{ is a spanning tree} \end{aligned} \qquad \text{(P3)}$$

Theorem

$f(A)$ is a supermodular function over $\mathcal{E}$, if $p_i, q_i \geq 0$ for all i.

Theorem

For any connected graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$ there is a matroid $\mathcal{M}(\mathcal{E}, \mathcal{I})$ such that the bases of $\mathcal{M}$ are the spanning trees of $\mathcal{G}$.

Theorem (Lee, Mirrokni, Nagarajan, Sviridenko 2009)

There is a $(\frac{1}{6} - \epsilon)$-approximation algorithm for maximizing any non-negative submodular function over bases of matroid $\mathcal{M}$, when $\mathcal{M}$ contains at least two disjoint bases.

Algorithm

Input: Configuration $\mathcal{G}(\mathcal{N},\mathcal{E})$, bus demands (p_k, q_k), line resistances (R_{ij}), ϵ.
Initialize T with an arbitrary spanning tree.
while there exist $e \in \mathcal{E} \setminus T$ and $e' \in T$ such that $(T\setminus\{e'\}) \cup \{e\}$ is a spanning tree and $f((T\setminus\{e'\}) \cup \{e\}) < (1-\epsilon) f(T)$ **do**
    $T \leftarrow (T\setminus\{e'\}) \cup \{e\}$
end while
return T

Algorithm

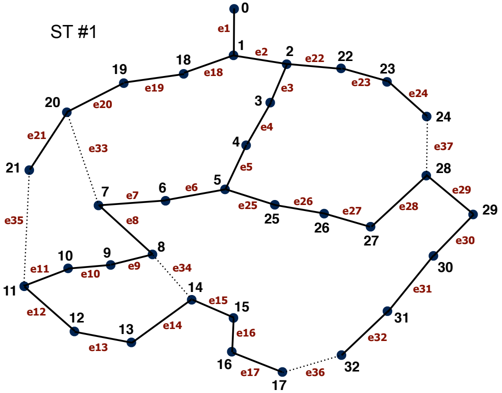
Input: Configuration $\mathcal{G}(\mathcal{N}, \mathcal{E})$, bus demands (p_k, q_k), line resistances (R_{ij}), ϵ.
Initialize T with an arbitrary spanning tree.
while there exist $e \in \mathcal{E} \backslash T$ and $e' \in T$ such that $(T \backslash \{e'\}) \cup \{e\}$ is a spanning tree and $f((T \backslash \{e'\}) \cup \{e\}) < (1-\epsilon) f(T)$ **do**
    $T \leftarrow (T \backslash \{e'\}) \cup \{e\}$
end while
return T

Theorem

Let $M = f(\mathcal{E})$ be an upper bound on f, and T^* be the optimal tree, then

$$M - f(T) \geq \left(\frac{1}{6} - \epsilon\right)\left(M - f(T^*)\right).$$

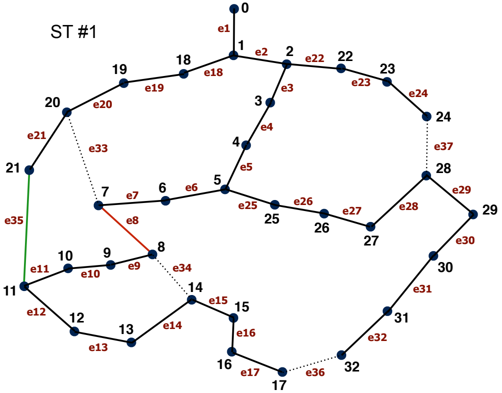
Example



Total Loss

1. 202 kW, exchange:(8,35)

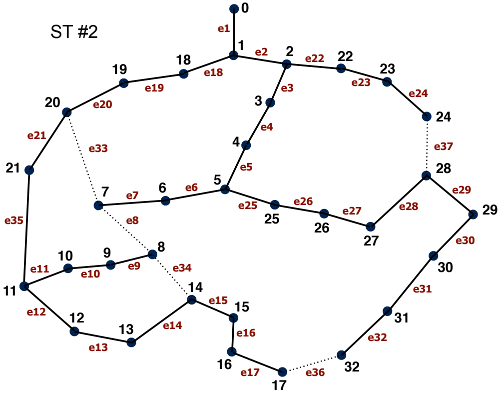
Example



Total Loss

1. 202 kW, exchange:(8,35)

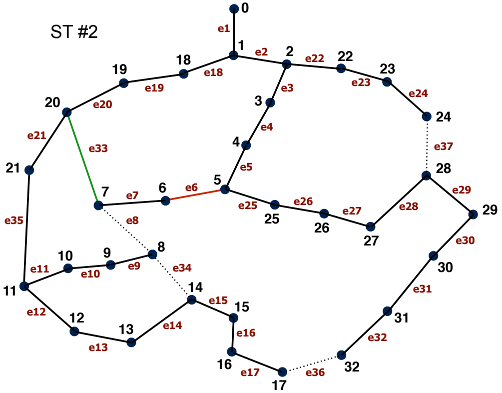
Example



Total Loss

1. 202 kW, exchange:(8,35)
2. 153 kW, exchange:(6,33)

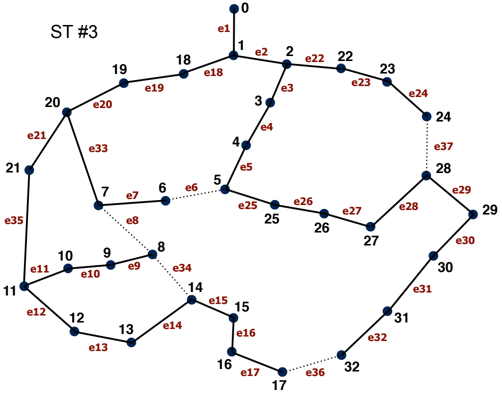
Example



Total Loss

1. 202 kW, exchange:(8,35)
2. 153 kW, exchange:(6,33)

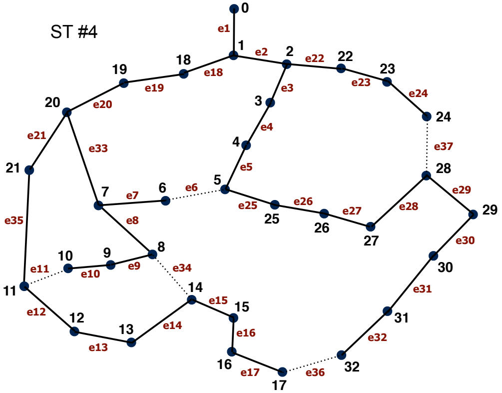
Example



Total Loss

1. 202 kW, exchange:(8,35)
2. 153 kW, exchange:(6,33)
3. 147 kW, exchange:(11,8)

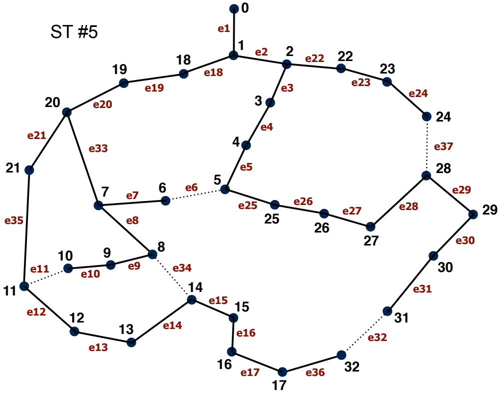
Example



Total Loss

1. 202 kW, exchange:(8,35)
2. 153 kW, exchange:(6,33)
3. 147 kW, exchange:(11,8)
4. 145 kW, exchange:(32,36)

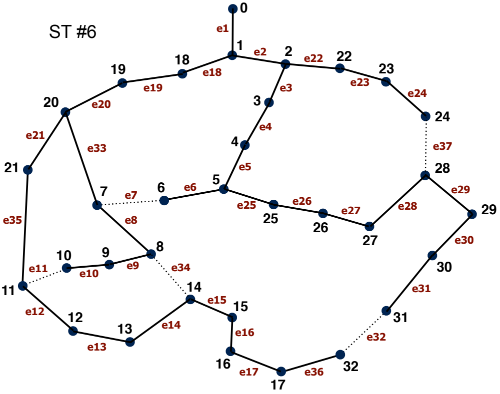
Example



Total Loss

1. 202 kW, exchange:(8,35)
2. 153 kW, exchange:(6,33)
3. 147 kW, exchange:(11,8)
4. 145 kW, exchange:(32,36)
5. 144 kW, exchange:(7,6)

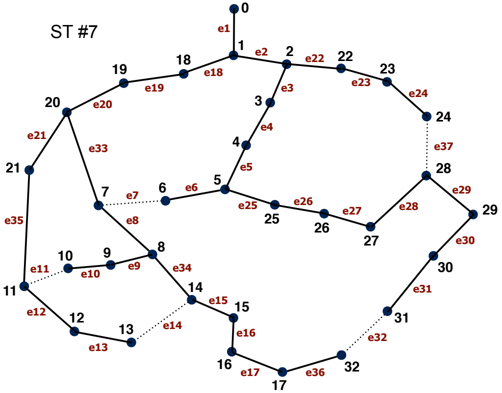
Example



Total Loss

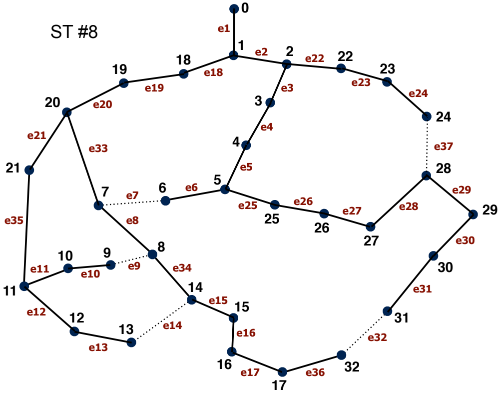
1. 202 kW, exchange:(8,35)
2. 153 kW, exchange:(6,33)
3. 147 kW, exchange:(11,8)
4. 145 kW, exchange:(32,36)
5. 144 kW, exchange:(7,6)
6. 142 kW, exchange:(14,34)

Example



Total Loss

1. 202 kW, exchange:(8,35)
2. 153 kW, exchange:(6,33)
3. 147 kW, exchange:(11,8)
4. 145 kW, exchange:(32,36)
5. 144 kW, exchange:(7,6)
6. 142 kW, exchange:(14,34)
7. 141 kW, exchange:(9,11)

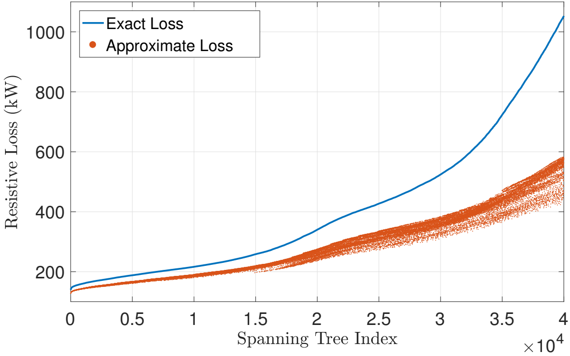


Total Loss

1. 202 kW, exchange:(8,35)
2. 153 kW, exchange:(6,33)
3. 147 kW, exchange:(11,8)
4. 145 kW, exchange:(32,36)
5. 144 kW, exchange:(7,6)
6. 142 kW, exchange:(14,34)
7. 141 kW, exchange:(9,11)
8. 139 kW

Validation of our assumptions that led to

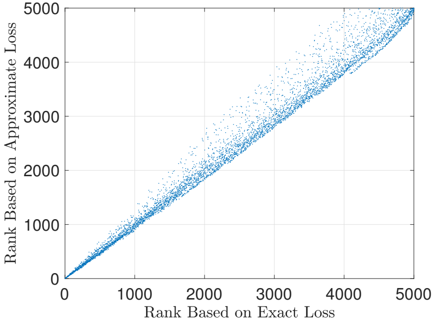
$$\text{Total Loss} = \sum_{e \in ST} R_e \left[\left(\sum_{i \in c_e} p_i \right)^2 + \left(\sum_{i \in c_e} q_i \right)^2 \right]$$



Simulation Results

Validation of our assumptions that led to

$$\text{Total Loss} = \sum_{e \in ST} R_e \left[\left(\sum_{i \in c_e} p_i \right)^2 + \left(\sum_{i \in c_e} q_i \right)^2 \right]$$



Closing Remarks

- We modeled the DNR problem as a combinatorial optimization problem (a supermodular minimization problem subject to a single matroid basis constraint).
- This enabled us to propose a polynomial time algorithm based on local search methods with a performance bound.
- This submodular framework can be extended to other problems to obtain new algorithms or to achieve performance bounds on existing heuristics.