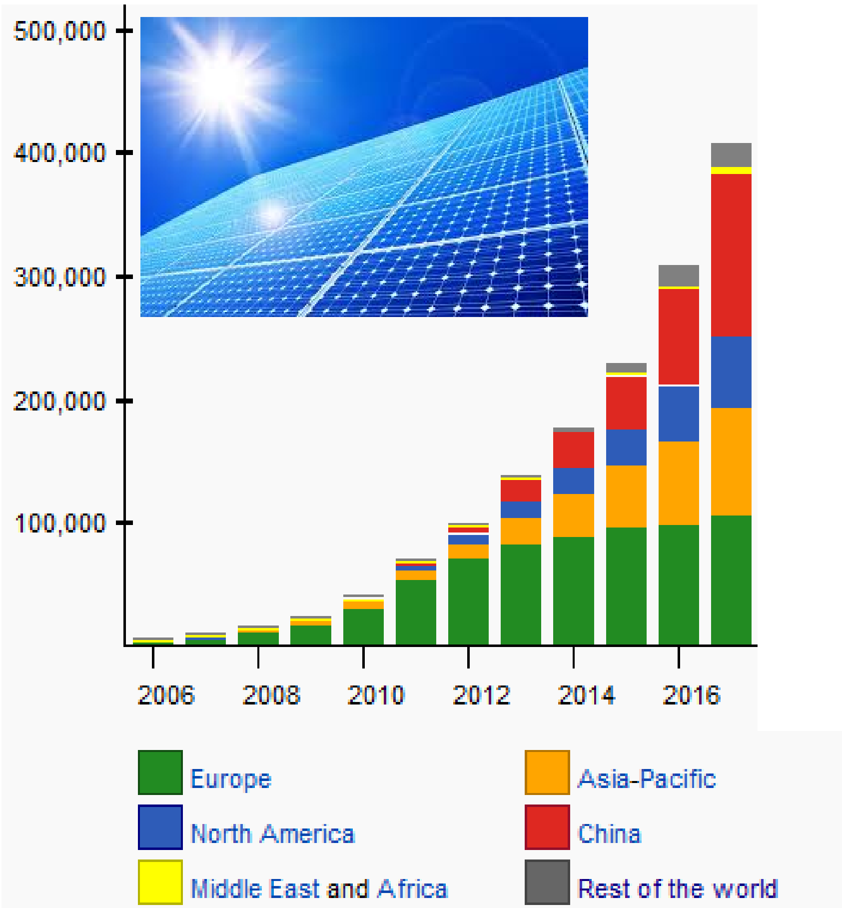
Computation-Efficient Optimization Algorithms for Autonomous Energy Systems

Xinyang Zhou, Chin-Yao Chang
Power System Engineering Center, NREL

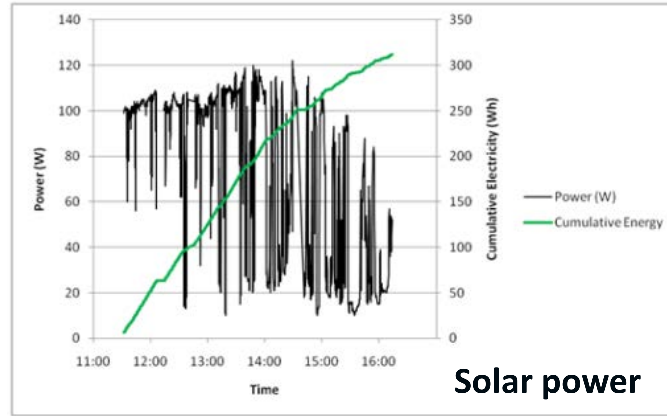
Aug. 19, 2020

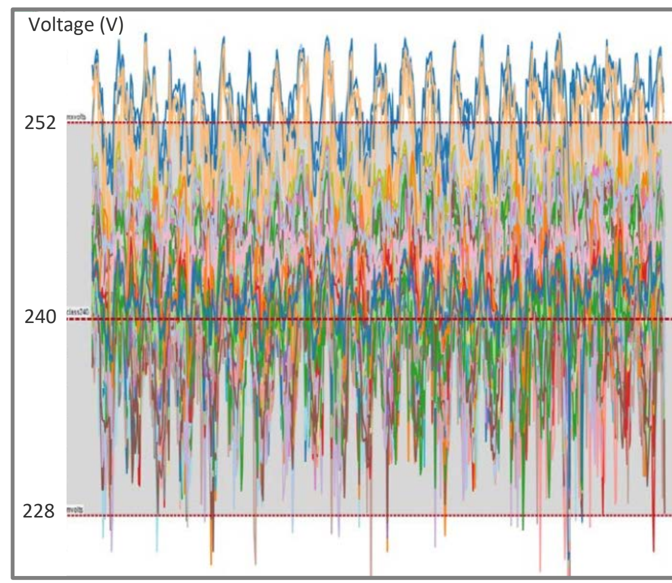
Background & Motivation

Background & Motivation



Cumulative photovoltaic capacity [MW]
Source: International Energy Agency, 2016

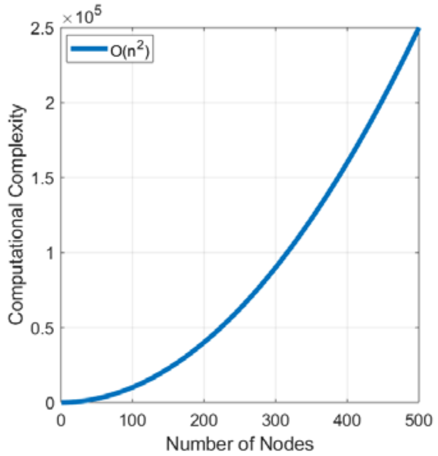


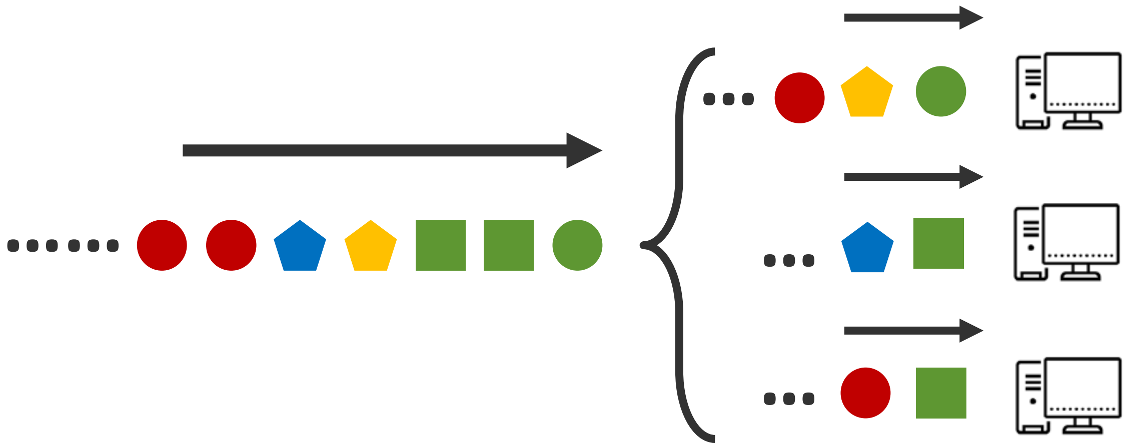


1-month voltage profile
Source: University of Hawaii

Background & Motivation

- Fast and optimal distribution systems voltage regulation
- Larger Systems: Increasing computational complexity





  – Distributed and parallel computation
  – Autonomous grid structure
    - No performance loss
    - Regional control and computation
    - Collaborating while preserving detailed information

Overall Goals

- Large distribution systems with deep renewable energy penetration
- Fast OPF solving
- Optimal solution without compromising performance (compared with centralized algorithms)

System Model & Design Intuition

Distribution System Modeling

- Radial Distribution Network
- Dist-Flow Model *[Baran 1989]*

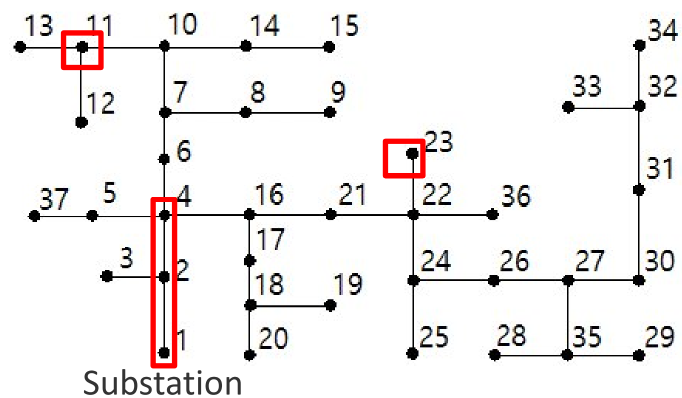
$$P_{ij} = -p_j + \sum_{k:(j,k)\in\mathcal{E}} P_{jk} + \cancel{r_{ij}\ell_{ij}},$$

$$Q_{ij} = -q_j + \sum_{k:(j,k)\in\mathcal{E}} Q_{jk} + \cancel{x_{ij}\ell_{ij}},$$

$$v_j = v_i - 2\left(r_{ij}P_{ij} + x_{ij}Q_{ij}\right) + \cancel{\left(r_{ij}^2 + x_{ij}^2\right)\ell_{ij}},$$

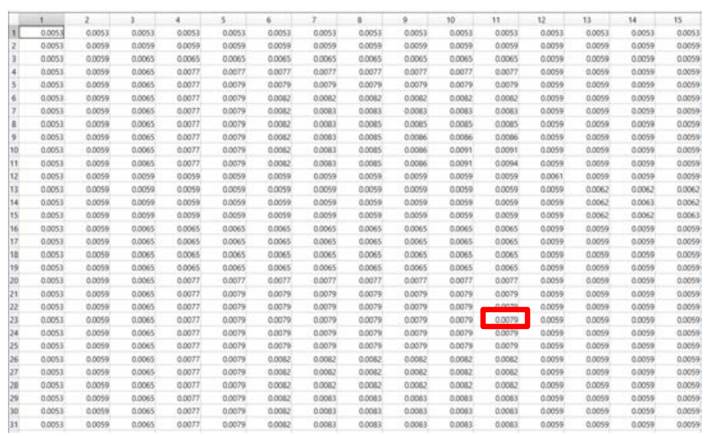
$$\cancel{\ell_{ij}v_i = P_{ij}^2 + Q_{ij}^2}.$$

~1% erros



- Lin-Dist-Flow Model *[Baran 1989, Farivar 2013]*

$$\boldsymbol{v} = R\boldsymbol{p} + X\boldsymbol{q} + \tilde{\boldsymbol{v}}$$

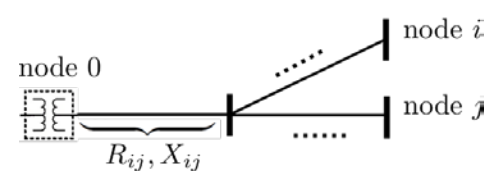


	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0.0053	0.0053	0.0053	0.0053	0.0053	0.0053	0.0053	0.0053	0.0053	0.0053	0.0053	0.0053	0.0053	0.0053	0.0053
2	0.0053	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059
3	0.0053	0.0059	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0059	0.0059	0.0059	0.0059
4	0.0053	0.0059	0.0065	0.0077	0.0077	0.0077	0.0077	0.0077	0.0077	0.0077	0.0077	0.0059	0.0059	0.0059	0.0059
5	0.0053	0.0059	0.0065	0.0077	0.0079	0.0079	0.0079	0.0079	0.0079	0.0079	0.0079	0.0059	0.0059	0.0059	0.0059
6	0.0053	0.0059	0.0065	0.0077	0.0079	0.0082	0.0082	0.0082	0.0082	0.0082	0.0082	0.0059	0.0059	0.0059	0.0059
7	0.0053	0.0059	0.0065	0.0077	0.0079	0.0082	0.0083	0.0083	0.0083	0.0083	0.0083	0.0059	0.0059	0.0059	0.0059
8	0.0053	0.0059	0.0065	0.0077	0.0079	0.0082	0.0083	0.0085	0.0085	0.0085	0.0085	0.0059	0.0059	0.0059	0.0059
9	0.0053	0.0059	0.0065	0.0077	0.0079	0.0082	0.0083	0.0085	0.0086	0.0086	0.0086	0.0059	0.0059	0.0059	0.0059
10	0.0053	0.0059	0.0065	0.0077	0.0079	0.0082	0.0083	0.0085	0.0086	0.0091	0.0091	0.0059	0.0059	0.0059	0.0059
11	0.0053	0.0059	0.0065	0.0077	0.0079	0.0082	0.0083	0.0085	0.0086	0.0091	0.0094	0.0059	0.0059	0.0059	0.0059
12	0.0053	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0061	0.0059	0.0059	0.0059
13	0.0053	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0062	0.0062	0.0062
14	0.0053	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0062	0.0063	0.0062
15	0.0053	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0059	0.0062	0.0062	0.0063
16	0.0053	0.0059	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0059	0.0059	0.0059	0.0059
17	0.0053	0.0059	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0059	0.0059	0.0059	0.0059
18	0.0053	0.0059	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0059	0.0059	0.0059	0.0059
19	0.0053	0.0059	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0065	0.0059	0.0059	0.0059	0.0059
20	0.0053	0.0059	0.0065	0.0077	0.0077	0.0077	0.0077	0.0077	0.0077	0.0077	0.0077	0.0059	0.0059	0.0059	0.0059
21	0.0053	0.0059	0.0065	0.0077	0.0079	0.0079	0.0079	0.0079	0.0079	0.0079	0.0079	0.0059	0.0059	0.0059	0.0059
22	0.0053	0.0059	0.0065	0.0077	0.0079	0.0079	0.0079	0.0079	0.0079	0.0079	0.0079	0.0059	0.0059	0.0059	0.0059
23	0.0053	0.0059	0.0065	0.0077	0.0079	0.0079	0.0079	0.0079	0.0079	0.0079	0.0079	0.0059	0.0059	0.0059	0.0059
24	0.0053	0.0059	0.0065	0.0077	0.0079	0.0079	0.0079	0.0079	0.0079	0.0079	0.0079	0.0059	0.0059	0.0059	0.0059
25	0.0053	0.0059	0.0065	0.0077	0.0079	0.0079	0.0079	0.0079	0.0079	0.0079	0.0079	0.0059	0.0059	0.0059	0.0059
26	0.0053	0.0059	0.0065	0.0077	0.0079	0.0082	0.0082	0.0082	0.0082	0.0082	0.0082	0.0059	0.0059	0.0059	0.0059
27	0.0053	0.0059	0.0065	0.0077	0.0079	0.0082	0.0082	0.0082	0.0082	0.0082	0.0082	0.0059	0.0059	0.0059	0.0059
28	0.0053	0.0059	0.0065	0.0077	0.0079	0.0082	0.0082	0.0082	0.0082	0.0082	0.0082	0.0059	0.0059	0.0059	0.0059
29	0.0053	0.0059	0.0065	0.0077	0.0079	0.0082	0.0083	0.0083	0.0083	0.0083	0.0083	0.0059	0.0059	0.0059	0.0059
30	0.0053	0.0059	0.0065	0.0077	0.0079	0.0082	0.0083	0.0083	0.0083	0.0083	0.0083	0.0059	0.0059	0.0059	0.0059
31	0.0053	0.0059	0.0065	0.0077	0.0079	0.0082	0.0083	0.0083	0.0083	0.0083	0.0083	0.0059	0.0059	0.0059	0.0059

Resistance/Reactance of common path

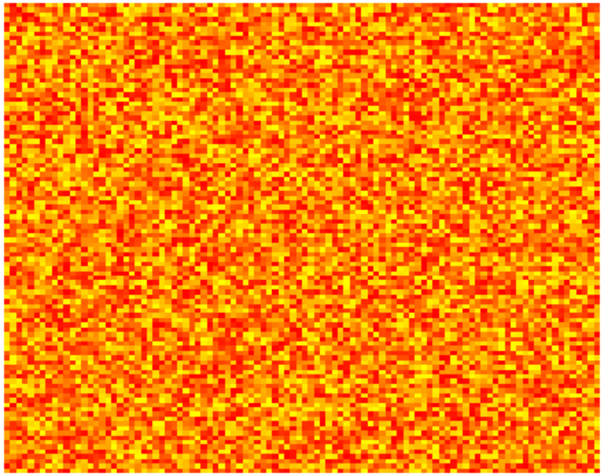
$$R_{ij} := \sum_{(\zeta,\xi)\in\mathcal{E}_i\cap\mathcal{E}_j} 2\cdot r_{\zeta\xi}$$

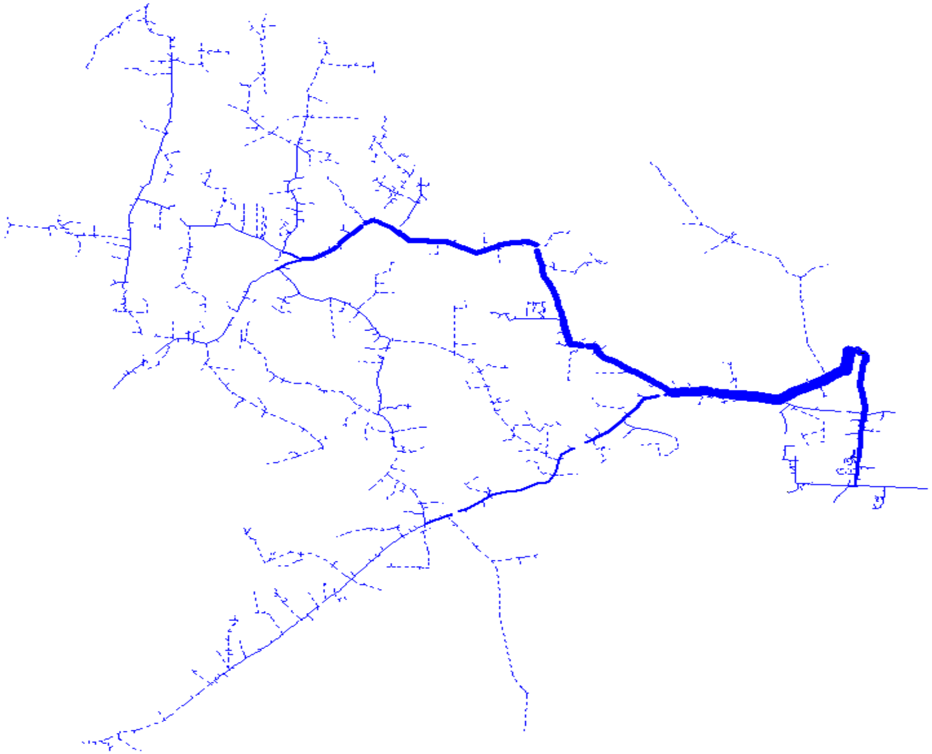
$$X_{ij} := \sum_{(\zeta,\xi)\in\mathcal{E}_i\cap\mathcal{E}_j} 2\cdot x_{\zeta\xi}$$



Design Intuition

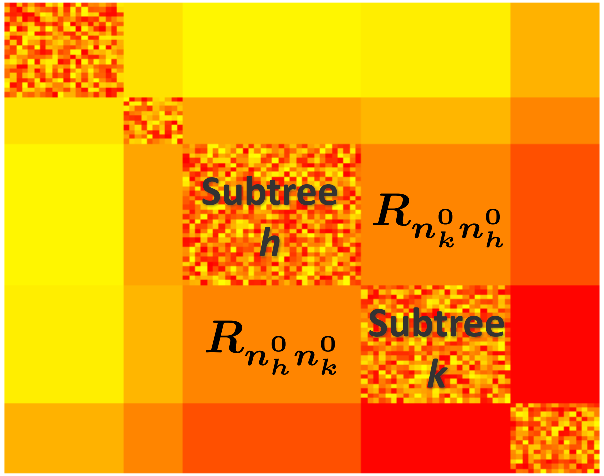
- General distribution network and sensitivity matrix
- Properties behind subtree-based network structure
- Go deeper

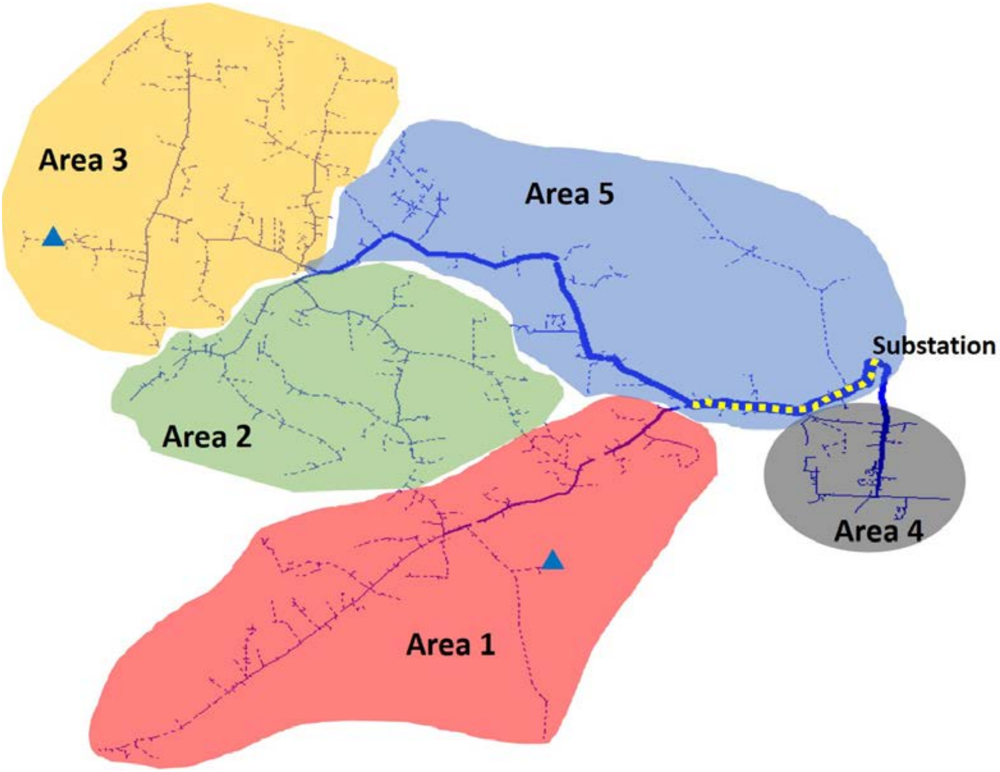




Design Intuition

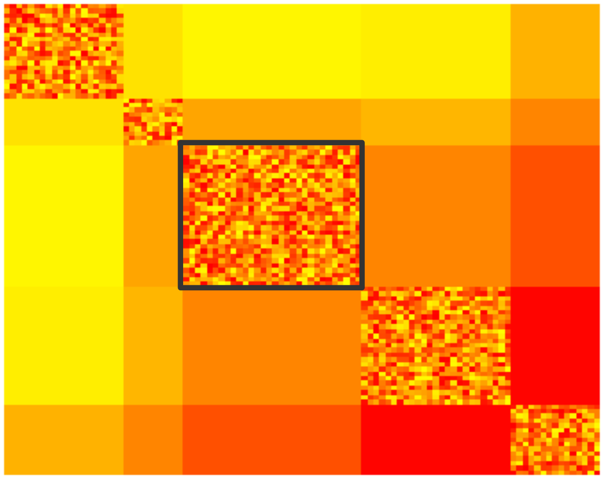
- General distribution network and sensitivity matrix
- Properties behind subtree-based network structure
- Go deeper

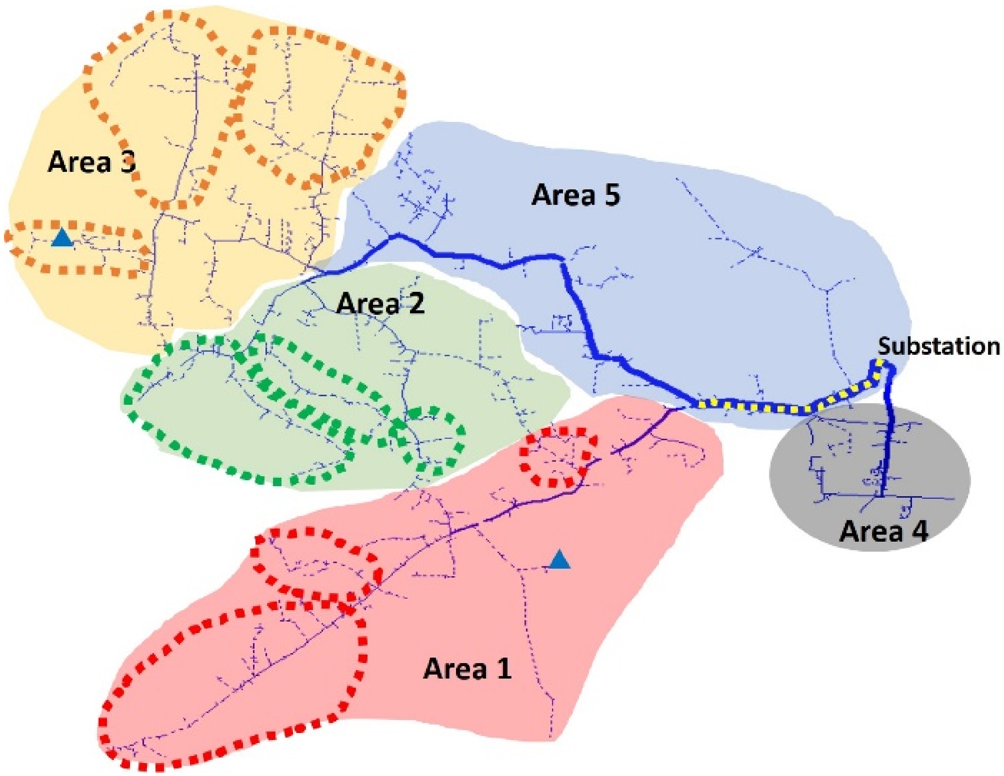




Design Intuition

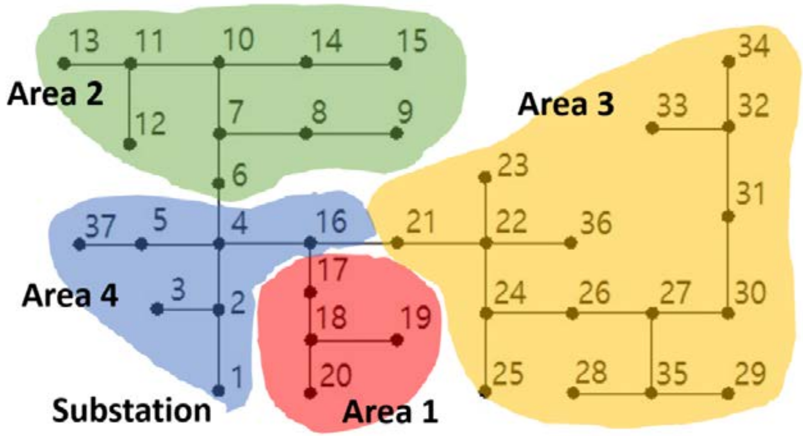
- General distribution network and sensitivity matrix
- Properties behind subtree-based network structure
- Go deeper

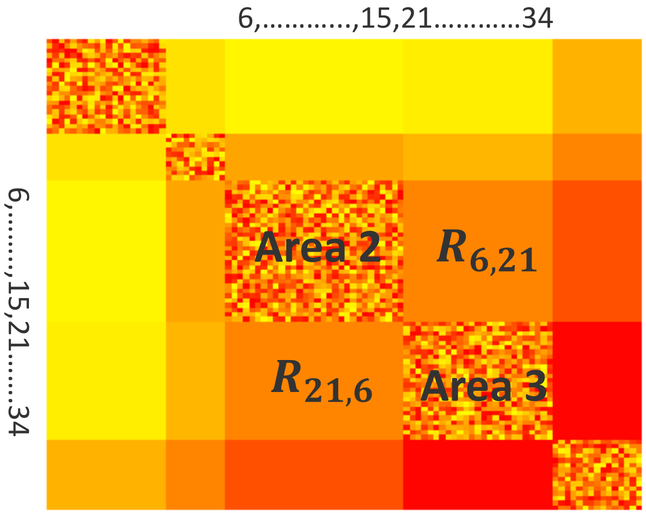




Example

- IEEE 37-Node Test Feeder





Example: Any node *i* in subtree Area 2 and any node *j* in subtree Area 3 share the identical common path leading back to the substation, i.e., lines 1-2-4. Therefore,
$R_{ij} = R_{6,21} = r_{12} + r_{24}, X_{ij} = x_{12} + x_{24}$ for any i within Area 2 and any j within Area 3

Solving Large OPF

Problem Formulation

- OPF Problem

$$\min_{\boldsymbol{p},\boldsymbol{q}} \quad \sum_{i\in\mathcal{N}} C_i(p_i,q_i) + C_0(P_0(\boldsymbol{p})),$$
$$\text{s.t.} \quad \underline{\boldsymbol{v}} \leqslant \boldsymbol{v}(\boldsymbol{p},\boldsymbol{q}) \leqslant \overline{\boldsymbol{v}},$$
$$(p_i,q_i) \in \mathcal{Y}_i, \forall i \in \mathcal{N}.$$

$$C_i(p_i,q_i) = (p_i - \hat{p}_i)^2 + (q_i - \hat{q}_i)^2$$
$$C_0(P_0(\boldsymbol{p})) = \alpha(P_0(\boldsymbol{p}) - \hat{P}_0)^2$$
$$P_0(\boldsymbol{p}) = -P_I - \sum_{i\in\mathcal{N}} p_i$$
$$\boldsymbol{v}(\boldsymbol{p},\boldsymbol{q}) = R\boldsymbol{p} + X\boldsymbol{q} + \tilde{\boldsymbol{v}}$$

- (Regularized) Lagrangian

$$\mathcal{L}_\eta(\boldsymbol{p},\boldsymbol{q};\overline{\boldsymbol{\mu}},\underline{\boldsymbol{\mu}}) = \sum_{i\in\mathcal{N}} C_i(p_i,q_i) + C_0(P_0(\boldsymbol{p}))$$
$$+\underline{\boldsymbol{\mu}}^\top(\underline{\boldsymbol{v}} - \boldsymbol{v}(\boldsymbol{p},\boldsymbol{q})) + \overline{\boldsymbol{\mu}}^\top(\boldsymbol{v}(\boldsymbol{p},\boldsymbol{q}) - \overline{\boldsymbol{v}}) - \frac{\eta}{2}\|\boldsymbol{\mu}\|_2^2$$

Gradient Algorithm

- Primal-Dual Gradient Algorithm

$$\boldsymbol{p}(t+1) = \Big[\boldsymbol{p}(t) - \epsilon\Big(\nabla_{\boldsymbol{p}} C(\boldsymbol{p}(t), \boldsymbol{q}(t)) - C_0'(P_0(\boldsymbol{p}(t))) \cdot \mathbf{1}_N + R^\top(\overline{\boldsymbol{\mu}}(t) - \underline{\boldsymbol{\mu}}(t))\Big)\Big]_{\boldsymbol{\mathcal{Y}}},$$

$$\boldsymbol{q}(t+1) = \Big[\boldsymbol{q}(t) - \epsilon\Big(\nabla_{\boldsymbol{q}} C(\boldsymbol{p}(t), \boldsymbol{q}(t)) + X^\top(\overline{\boldsymbol{\mu}}(t) - \underline{\boldsymbol{\mu}}(t))\Big)\Big]_{\boldsymbol{\mathcal{Y}}},$$

$$\underline{\boldsymbol{\mu}}(t+1) = \big[\underline{\boldsymbol{\mu}}(t) + \epsilon(\underline{\boldsymbol{v}} - \boldsymbol{v}(t) - \eta\underline{\boldsymbol{\mu}}(t))\big]_+,$$

$$\overline{\boldsymbol{\mu}}(t+1) = \big[\overline{\boldsymbol{\mu}}(t) + \epsilon(\boldsymbol{v}(t) - \overline{\boldsymbol{v}} - \eta\overline{\boldsymbol{\mu}}(t))\big]_+,$$

$$\boldsymbol{v}(t+1) = R\boldsymbol{p}(t+1) + X\boldsymbol{q}(t+1) + \tilde{\boldsymbol{v}}.$$

- Design Motivation
  - Centrally coordinated algorithm: increasing computation in the coupling term ($O(N^2)$)
  - Hierarchical structure

Gradient Algorithm

- Primal-Dual Gradient Algorithm

$$
\begin{aligned}
\boldsymbol{p}(t+1) &= \Big[\boldsymbol{p}(t) - \epsilon\Big(\nabla_{\boldsymbol{p}} C(\boldsymbol{p}(t),\boldsymbol{q}(t)) - C_0'(P_0(\boldsymbol{p}(t)))\cdot \mathbf{1}_N \\
&\qquad + R^\top(\overline{\boldsymbol{\mu}}(t) - \underline{\boldsymbol{\mu}}(t))\Big)\Big]_{\boldsymbol{\mathcal{Y}}}, \\
\boldsymbol{q}(t+1) &= \Big[\boldsymbol{q}(t) - \epsilon\Big(\nabla_{\boldsymbol{q}} C(\boldsymbol{p}(t),\boldsymbol{q}(t)) \\
&\qquad + X^\top(\overline{\boldsymbol{\mu}}(t) - \underline{\boldsymbol{\mu}}(t))\Big)\Big]_{\boldsymbol{\mathcal{Y}}}, \\
\underline{\boldsymbol{\mu}}(t+1) &= [\underline{\boldsymbol{\mu}}(t) + \epsilon(\underline{\boldsymbol{v}} - \boldsymbol{v}(t) - \eta\underline{\boldsymbol{\mu}}(t))]_+, \\
\overline{\boldsymbol{\mu}}(t+1) &= [\overline{\boldsymbol{\mu}}(t) + \epsilon(\boldsymbol{v}(t) - \overline{\boldsymbol{v}} - \eta\overline{\boldsymbol{\mu}}(t))]_+, \\
\boldsymbol{v}(t+1) &= R\boldsymbol{p}(t+1) + X\boldsymbol{q}(t+1) + \tilde{\boldsymbol{v}}.
\end{aligned}
$$

$$
\begin{aligned}
p_i(t+1) = \Big[p_i(t) - \epsilon\Big(\partial_{p_i} C_i(p_i(t), q_i(t)) - C_0'(P_0(\boldsymbol{p}(t))) \\
+ \sum_{j\in\mathcal{N}} R_{ij}(\overline{\mu}_j(t) - \underline{\mu}_j(t))\Big)\Big]_{\mathcal{Y}_i}
\end{aligned}
$$

- Design Motivation
  - Centrally coordinated algorithm: increasing computation in the coupling term ($O(N^2)$)
  - Hierarchical structure

Hierarchical Implementation

- Subtrees & useful properties
- Individual node update

$$p_i(t+1) = \Big[p_i(t) - \epsilon\Big(\partial_{p_i} C_i(p_i(t), q_i(t)) - C_0'(P_0(\boldsymbol{p}(t))) + \sum_{j \in \mathcal{N}} R_{ij}\left(\overline{\mu}_j(t) - \underline{\mu}_j(t)\right)\Big)\Big]_{\mathcal{Y}_i}$$

Hierarchical Implementation

- Subtrees & useful properties
- Individual node update

$$p_i(t+1) = \Big[p_i(t) - \epsilon\Big(\partial_{p_i} C_i(p_i(t), q_i(t)) - C_0'(P_0(\boldsymbol{p}(t))) + \sum_{j\in\mathcal{N}} R_{ij}\left(\overline{\mu}_j(t) - \underline{\mu}_j(t)\right)\Big)\Big]_{\mathcal{Y}_i}$$

For clustered node $i \in \mathcal{N}_k$

$$\sum_{j\in\mathcal{N}} R_{ij}(\overline{\mu}_j - \underline{\mu}_j)$$

$$= \sum_{j\in\mathcal{N}_k} R_{ij}(\overline{\mu}_j - \underline{\mu}_j) + \sum_{j\in\mathcal{N}\setminus\mathcal{N}_k} R_{ij}(\overline{\mu}_j - \underline{\mu}_j)$$

Reduce/Recycle computation!

within other areas

$$= \sum_{j\in\mathcal{N}_k} R_{ij}(\overline{\mu}_j - \underline{\mu}_j) + \sum_{h\in\mathcal{K}, h\neq k} R_{n_h^0 n_k^0} \sum_{j\in\mathcal{N}_h} (\overline{\mu}_j - \underline{\mu}_j)$$

within area k

$$+ \sum_{j\in\mathcal{N}_0} R_{n_k^0 j}(\overline{\mu}_j - \underline{\mu}_j)$$

unclustered nodes

$$= \sum_{k\in\mathcal{K}} R_{in_k^0} \sum_{j\in\mathcal{N}_k} (\overline{\mu}_j - \underline{\mu}_j) + \sum_{j\in\mathcal{N}_0} R_{ij}(\overline{\mu}_j - \underline{\mu}_j)$$

Hierarchical Implementation

- Subtrees & useful properties
- Individual node update

$$p_i(t+1) = \Big[p_i(t) - \epsilon\Big(\partial_{p_i} C_i(p_i(t), q_i(t)) - C_0'(P_0(\boldsymbol{p}(t))) + \sum_{j\in\mathcal{N}} R_{ij}\big(\overline{\mu}_j(t) - \underline{\mu}_j(t)\big)\Big)\Big]_{\mathcal{Y}_i}$$

For unclustered node $i \in \mathcal{N}_0$

$$\sum_{j\in\mathcal{N}} R_{ij}(\overline{\mu}_j - \underline{\mu}_j)$$

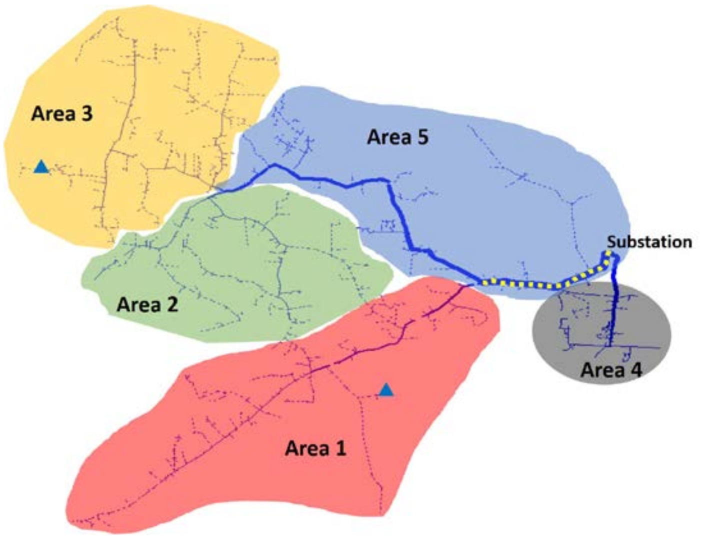
$$= \sum_{j\in\mathcal{N}_k} R_{ij}(\overline{\mu}_j - \underline{\mu}_j) + \sum_{j\in\mathcal{N}\backslash\mathcal{N}_k} R_{ij}(\overline{\mu}_j - \underline{\mu}_j)$$

$$= \sum_{j\in\mathcal{N}_k} R_{ij}(\overline{\mu}_j - \underline{\mu}_j) + \sum_{h\in\mathcal{K}, h\neq k} R_{n_h^0 n_k^0} \sum_{j\in\mathcal{N}_h} (\overline{\mu}_j - \underline{\mu}_j) + \sum_{j\in\mathcal{N}_0} R_{n_k^0 j}(\overline{\mu}_j - \underline{\mu}_j)$$

within all areas

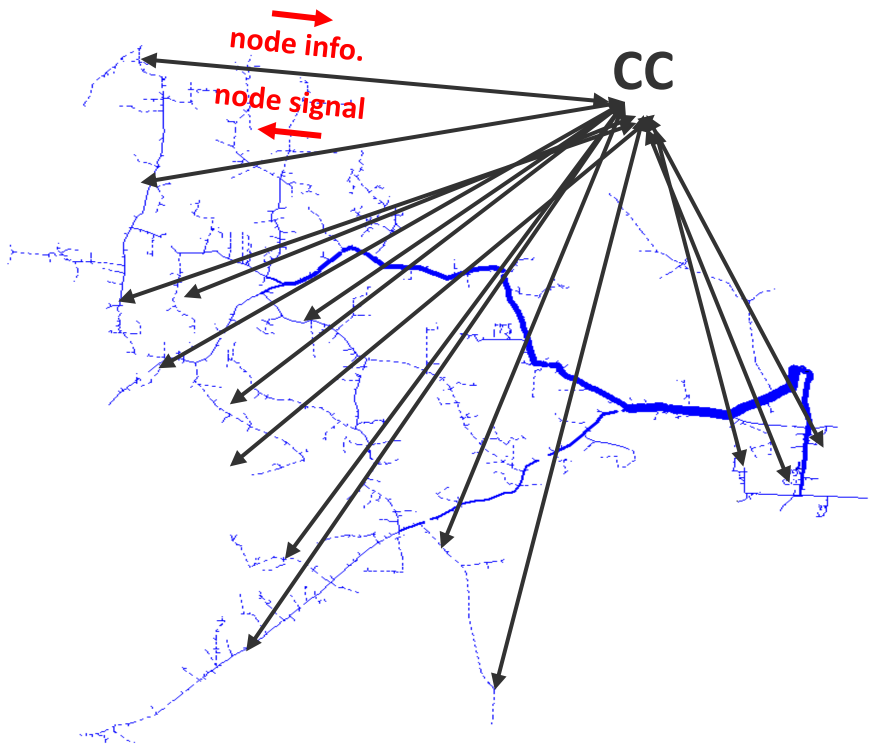
unclustered nodes

$$= \sum_{k\in\mathcal{K}} R_{in_k^0} \sum_{j\in\mathcal{N}_k} (\overline{\mu}_j - \underline{\mu}_j) + \sum_{j\in\mathcal{N}_0} R_{ij}(\overline{\mu}_j - \underline{\mu}_j)$$



Hierarchical Implementation

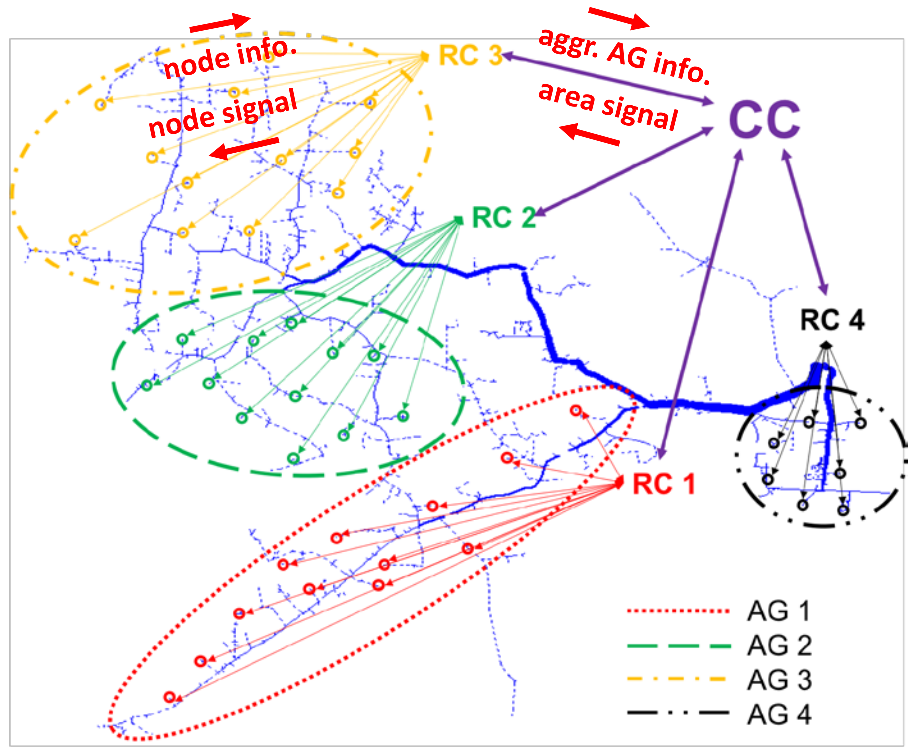
- Mathematically equivalent to the classic (distributed) gradient algorithm
- New implementation adapted for networked AGs structure
  - Design by exploring network and linearization structure
  - Parallel computation of coupling terms (computation bottleneck!)
    - Reducing and recycling computational load
    - Node-wise information preserved within AGs



Classic centrally coordinated distributed implementation

Hierarchical Implementation

- Mathematically equivalent to the classic (distributed) gradient algorithm
- New implementation adapted for networked AGs structure
  - Design by exploring network and linearization structure
  - Parallel computation of coupling terms (computation bottleneck!)
    - Reducing and recycling computational load
    - Node-wise information preserved within AGs



AG-based hierarchical distributed implementation

Hierarchical Implementation

- Equivalent to centrally coordinated implementation
- Computational complexity reduction
  - Ideal (probably unrealistic) situation: $O(N^2) \rightarrow O\left(N^{4/3}\right)$
  - Optimal/More clustering?
- Privacy preservation
  - Node-wise information preserved
  - Topology information within areas preserved

Multi-Phase System Extension

- Linearized multi-phase dist-flow *[Gan 2016]*

$$\boldsymbol{v}_\Xi = R_\Xi \boldsymbol{p}_\Xi + X_\Xi \boldsymbol{q}_\Xi + \tilde{\boldsymbol{v}}_\Xi$$

$$\begin{aligned} \boldsymbol{v}_\Xi &= [[v_1^\phi]_{\phi\in\Phi_1}^\top, \ldots, [v_N^\phi]_{\phi\in\Phi_N}^\top]^\top \in \mathbb{R}^{N_\Xi} \\ \boldsymbol{p}_\Xi &= [[p_1^\phi]_{\phi\in\Phi_1}^\top, \ldots, [p_N^\phi]_{\phi\in\Phi_N}^\top]^\top \in \mathbb{R}^{N_\Xi} \\ \boldsymbol{q}_\Xi &= [[q_1^\phi]_{\phi\in\Phi_1}^\top, \ldots, [q_N^\phi]_{\phi\in\Phi_N}^\top]^\top \in \mathbb{R}^{N_\Xi} \end{aligned}$$

$$\begin{aligned} \partial_{p_i^\phi} v_j^\varphi &= 2\mathfrak{Re}\{\overline{Z}_{ji}^{\varphi\phi} \cdot e^{-\jmath 2\pi(\varphi-\phi)/3}\} \\ \partial_{q_i^\phi} v_j^\varphi &= -2\mathfrak{Im}\{\overline{Z}_{ji}^{\varphi\phi} \cdot e^{-\jmath 2\pi(\varphi-\phi)/3}\} \end{aligned}$$

$$Z_{ji}^{\varphi\phi} = \sum_{(\zeta,\xi)\in\mathcal{E}_j\cap\mathcal{E}_i} z_{\zeta\xi}^{\varphi\phi}$$: impedance of common path!

- Multi-Phase OPF Problem

$$\begin{aligned} \min_{\boldsymbol{p}_\Xi, \boldsymbol{q}_\Xi} \quad & \sum_{i\in\mathcal{N}} \sum_{\phi\in\Phi_i} C_i^\phi(p_i^\phi, q_i^\phi) + C_0(P_0(\boldsymbol{p}_\Xi)), \\ \text{s.t.} \quad & \underline{v}_i^\phi \leqslant v_i^\phi(\boldsymbol{p}_\Xi, \boldsymbol{q}_\Xi) \leqslant \overline{v}_i^\phi, \phi\in\Phi_i, \forall i\in\mathcal{N}, \\ & (p_i^\phi, q_i^\phi) \in \mathcal{Y}_i^\phi, \phi\in\Phi_i, \forall i\in\mathcal{N}. \end{aligned}$$

$$P_0 = -\sum_{\phi\in\Phi_0} P_I^\phi - \sum_{i\in\mathcal{N}} \sum_{\phi\in\Phi_i} p_i^\phi$$

Multi-Phase System Extension

- (Regularized) Lagrangian

$$\mathcal{L}_\eta^\Phi(\boldsymbol{p}_\Xi, \boldsymbol{q}_\Xi; \overline{\boldsymbol{\mu}}_\Xi, \underline{\boldsymbol{\mu}}_\Xi) = \sum_{i \in \mathcal{N}} \sum_{\phi \in \Phi_i} C_i^\phi(p_i^\phi, q_i^\phi) + C_0(P_0(\boldsymbol{p}_\Xi))$$

$$+ \underline{\boldsymbol{\mu}}_\Xi^\top(\underline{\boldsymbol{v}}_\Xi - \boldsymbol{v}_\Xi(\boldsymbol{p}_\Xi, \boldsymbol{q}_\Xi)) + \overline{\boldsymbol{\mu}}_\Xi^\top(\boldsymbol{v}_\Xi(\boldsymbol{p}_\Xi, \boldsymbol{q}_\Xi) - \overline{\boldsymbol{v}}_\Xi) - \frac{\eta}{2}\|\boldsymbol{\mu}_\Xi\|_2^2.$$

- Primal-dual gradient algorithm

$$\begin{aligned}
p_i^\phi(t+1) &= \Big[p_i^\phi(t) - \epsilon\Big(\partial_{p_i^\phi} C_i^\phi(p_i^\phi(t), q_i^\phi(t)) - \\
&\quad C_0'(P_0(\boldsymbol{p}_\Xi(t))) + \sum_{j \in \mathcal{N}} \sum_{\varphi \in \Phi_j} \partial_{p_i^\phi} v_j^\varphi \big(\overline{\mu}_j^\varphi(t) - \underline{\mu}_j^\varphi(t)\big)\Big)\Big]_{\mathcal{Y}_i^\phi}, \\
q_i^\phi(t+1) &= \Big[q_i^\phi(t) - \epsilon\Big(\partial_{q_i^\phi} C_i^\phi(p_i^\phi(t), q_i^\phi(t)) + \\
&\quad \sum_{j \in \mathcal{N}} \sum_{\varphi \in \Phi_j} \partial_{q_i^\phi} v_j^\varphi \big(\overline{\mu}_j^\varphi(t) - \underline{\mu}_j^\varphi(t)\big)\Big)\Big]_{\mathcal{Y}_i^\phi}, \\
\underline{\mu}_i^\phi(t+1) &= [\underline{\mu}_i^\phi(t) + \epsilon(\underline{v}_i^\phi - v_i^\phi(t) - \eta\underline{\mu}_i^\phi(t))]_+, \\
\overline{\mu}_i^\phi(t+1) &= [\overline{\mu}_i^\phi(t) + \epsilon(v_i^\phi(t) - \overline{v}_i^\phi - \eta\overline{\mu}_i^\phi(t))]_+, \\
\boldsymbol{v}_\Xi(t+1) &= R_\Xi \boldsymbol{p}_\Xi(t+1) + X_\Xi \boldsymbol{q}_\Xi(t+1) + \tilde{\boldsymbol{v}}_\Xi.
\end{aligned}$$

Multi-Phase System Extension

- Decompose of the coupling terms

For clustered node $i \in \mathcal{N}_k$

$$2\sum_{j\in\mathcal{N}}\sum_{\varphi\in\Phi_j}\Re\{\overline{Z}_{ji}^{\varphi\phi}e^{-\jmath 2\pi(\varphi-\phi)/3}\}\cdot\left(\overline{\mu}_j^{\varphi}(t)-\underline{\mu}_j^{\varphi}(t)\right)$$

$$= 2\Re\Big\{\sum_{\varphi\in\Phi_0}e^{-\jmath 2\pi(\varphi-\phi)/3}\Big(\sum_{j\in\mathcal{N}^{\varphi}\cap\mathcal{N}_k}\overline{Z}_{ji}^{\varphi\phi}\left(\overline{\mu}_j^{\varphi}(t)-\underline{\mu}_j^{\varphi}(t)\right)$$

within area k

$$+\sum_{\substack{n_h^0\in N^{\varphi}\\ h\in\mathcal{K},h\neq k}}\overline{Z}_{n_h^0 n_k^0}^{\varphi\phi}\sum_{j\in\mathcal{N}^{\varphi}\cap\mathcal{N}_h}\left(\overline{\mu}_j^{\varphi}(t)-\underline{\mu}_j^{\varphi}(t)\right)$$

within other areas

$$+\sum_{j\in\mathcal{N}^{\varphi}\cap\mathcal{N}_0}\overline{Z}_{jn_k^0}^{\varphi\phi}\left(\overline{\mu}_j^{\varphi}(t)-\underline{\mu}_j^{\varphi}(t)\right)\Big)\Big\}$$

unclustered nodes

$$= 2\Re\Big\{\sum_{\varphi\in\Phi_0}e^{-\jmath 2\pi(\varphi-\phi)/3}\Big(\sum_{\substack{n_k^0\in N^{\varphi}\\ k\in\mathcal{K}}}\overline{Z}_{n_k^0 i}^{\varphi\phi}\sum_{j\in\mathcal{N}^{\varphi}\cap\mathcal{N}_k}\left(\overline{\mu}_j^{\varphi}(t)-\underline{\mu}_j^{\varphi}(t)\right)$$

$$+\sum_{j\in\mathcal{N}^{\varphi}\cap\mathcal{N}_0}\overline{Z}_{ji}^{\varphi\phi}\left(\overline{\mu}_j^{\varphi}(t)-\underline{\mu}_j^{\varphi}(t)\right)\Big)\Big\}$$

Multi-Phase System Extension

- Decompose of the coupling terms

For unclustered node $i \in \mathcal{N}_0$

$$2 \sum_{j \in \mathcal{N}} \sum_{\varphi \in \Phi_j} \mathfrak{Re}\{\overline{Z}_{ji}^{\varphi\phi} e^{-\jmath 2\pi(\varphi-\phi)/3}\} \cdot (\overline{\mu}_j^{\varphi}(t) - \underline{\mu}_j^{\varphi}(t))$$

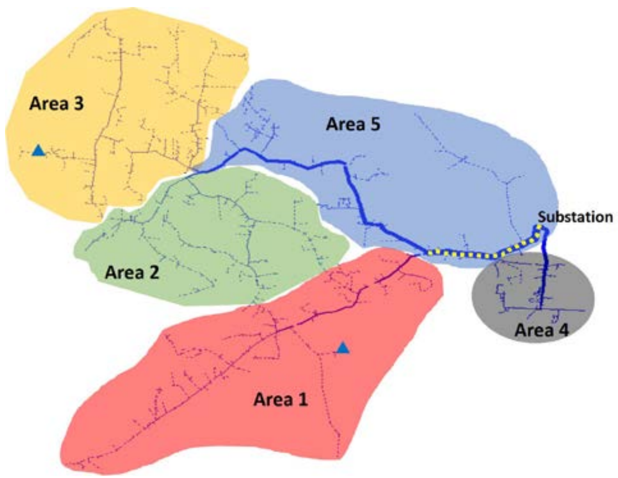
$$= 2\mathfrak{Re}\Big\{ \sum_{\varphi \in \Phi_0} e^{-\jmath 2\pi(\varphi-\phi)/3} \Big(\sum_{j \in \mathcal{N}^{\varphi} \cap \mathcal{N}_k} \overline{Z}_{ji}^{\varphi\phi} (\overline{\mu}_j^{\varphi}(t) - \underline{\mu}_j^{\varphi}(t)) + \sum_{\substack{n_h^0 \in N^{\varphi} \\ h \in \mathcal{K}, h \neq k}} \overline{Z}_{n_h^0 n_k^0}^{\varphi\phi} \sum_{j \in \mathcal{N}^{\varphi} \cap \mathcal{N}_h} (\overline{\mu}_j^{\varphi}(t) - \underline{\mu}_j^{\varphi}(t)) + \sum_{j \in \mathcal{N}^{\varphi} \cap \mathcal{N}_0} \overline{Z}_{jn_k^0}^{\varphi\phi} (\overline{\mu}_j^{\varphi}(t) - \underline{\mu}_j^{\varphi}(t)) \Big) \Big\}$$

$$= 2\mathfrak{Re}\Big\{ \sum_{\varphi \in \Phi_0} e^{-\jmath 2\pi(\varphi-\phi)/3} \Big(\sum_{\substack{n_k^0 \in N^{\varphi} \\ k \in \mathcal{K}}} \overline{Z}_{n_k^0 i}^{\varphi\phi} \sum_{j \in \mathcal{N}^{\varphi} \cap \mathcal{N}_k} (\overline{\mu}_j^{\varphi}(t) - \underline{\mu}_j^{\varphi}(t)) + \sum_{j \in \mathcal{N}^{\varphi} \cap \mathcal{N}_0} \overline{Z}_{ji}^{\varphi\phi} (\overline{\mu}_j^{\varphi}(t) - \underline{\mu}_j^{\varphi}(t)) \Big) \Big\}$$

within all areas

unclustered nodes

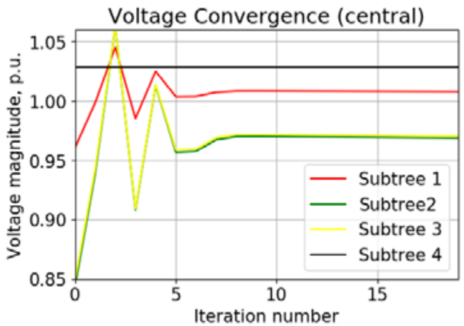
Numerical Setup

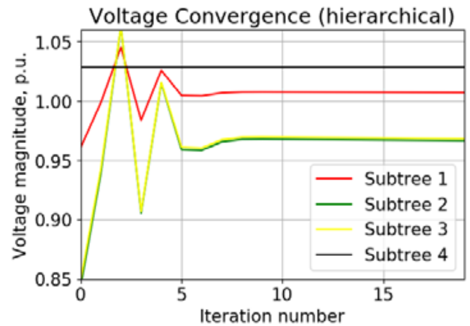


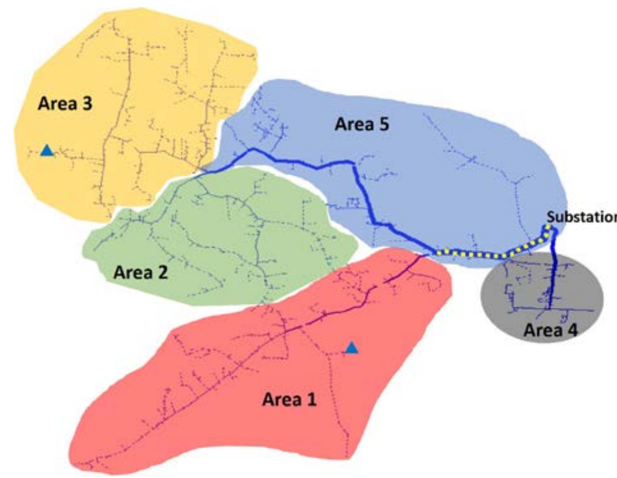
- Synthetic 11,000-node test feeder
  - IEEE 8,500
  - EPRI CKT7
- Primary side voltage control
  - 4,521-node network
  - 1,043 controllable loads
- Clustered into 4 areas consisting of 154—357 controllable loads
- Three-phase hierarchical distributed algorithm

OPF Numerical Results

- 2-level: 4-fold speed improvement



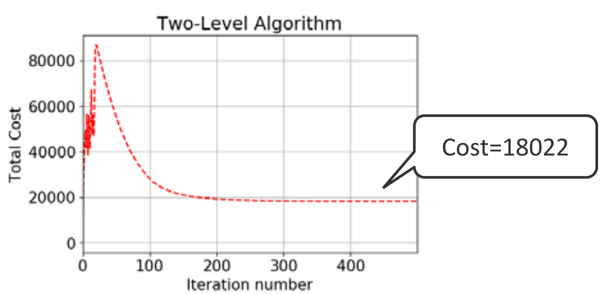


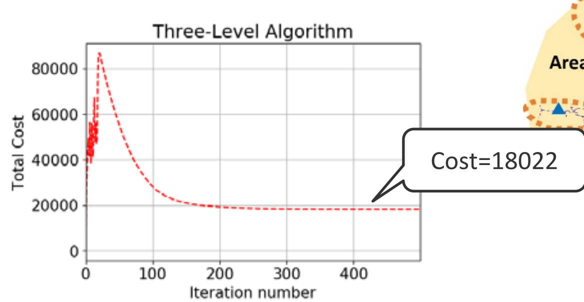


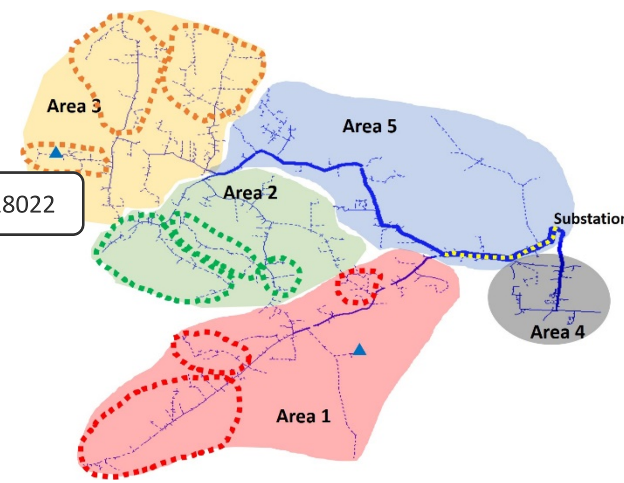
  - “Free” speed improvement
  - Parallel implementation: 2.5 times more

31.3% speed improvement

- 3-level: 31.3% further improvement

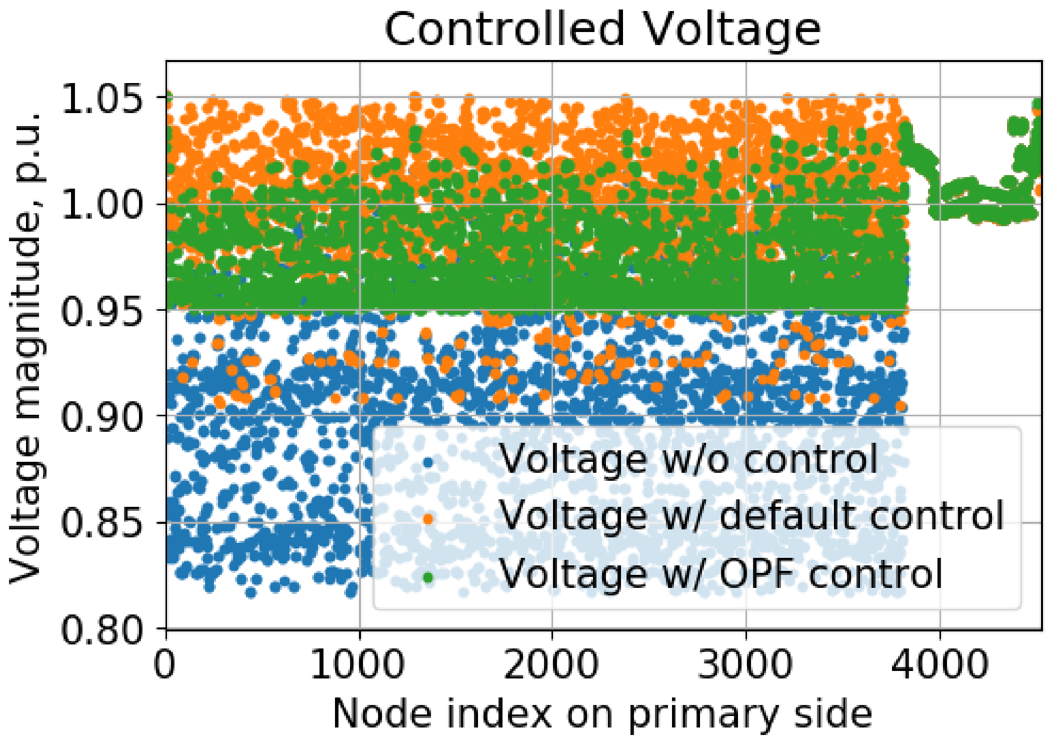






OPF Numerical Results

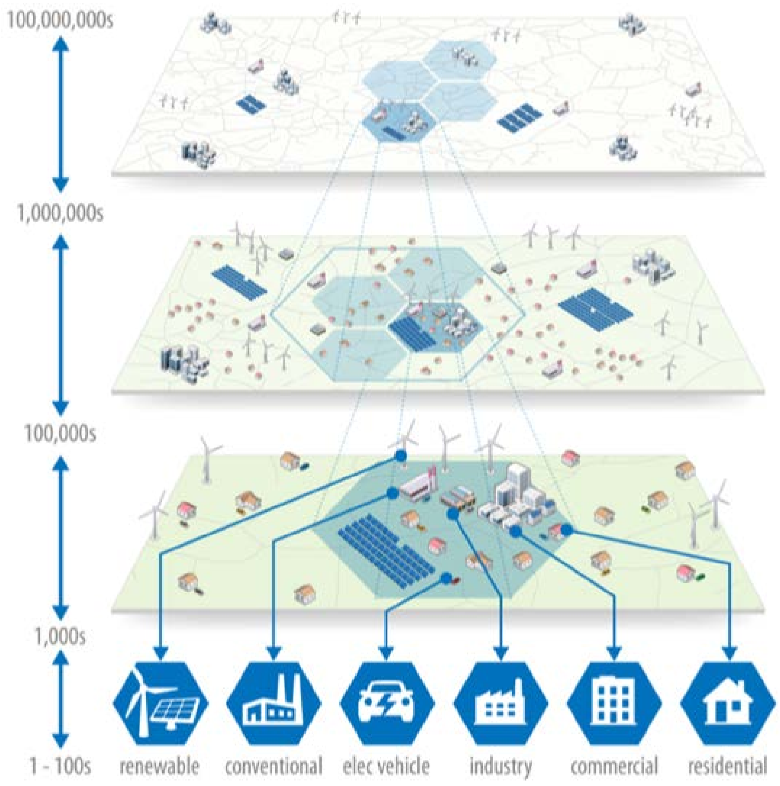
- Voltage regulation



Distributed Version of the Algorithm

Objective

- AES cellular control structure
  - Multi-layer hierarchy
  - Distributed coordination between cells in the same layer
    - ✓ ***Reduced reliance on central controller*** (flexibility, robustness, privacy)
- By merging
  - Complexity reduction method
  - Distributed feedback-based algorithm (review later)



Low complexity feedback-based algorithm *that is flexible to various communication scenarios*

Review of Distributed Feedback-Based Algorithm

- Recall the OPF Problem that we attempt to solve

$$
\begin{aligned}
&\min_{p,q} \sum_{i\in\mathcal{N}} C_i(p_i, q_i), \\
\text{s.t. } & v = Rp + Xq + \tilde{v}, \\
& \underline{v} \leqslant v \leqslant \bar{v}, \\
& (p_i, q_i) \in \mathcal{Y}_i, \quad \forall i \in \mathcal{N}
\end{aligned}
$$

- Primal-dual method $x_i = [p_i, q_i]^T$

$$
\begin{aligned}
\dot{x}_i &= \Pi_{\mathcal{T}_{\mathcal{Y}_i}^{x_i}} \left(- \begin{bmatrix} \nabla_{p_i} C_i + R_i^\top (\bar{\mu} - \underline{\mu}) \\ \nabla_{q_i} C_i + X_i^\top (\bar{\mu} - \underline{\mu}) \end{bmatrix} \right), \forall i \in \mathcal{N} \\
\dot{\bar{\mu}} &= \Pi_{\mathcal{T}_{\Re_+}^{\bar{\mu}}} \left(v(x) - \bar{v} \right) \\
\dot{\underline{\mu}} &= \Pi_{\mathcal{T}_{\Re_+}^{\underline{\mu}}} \left(\underline{v} - v(x) \right)
\end{aligned}
$$

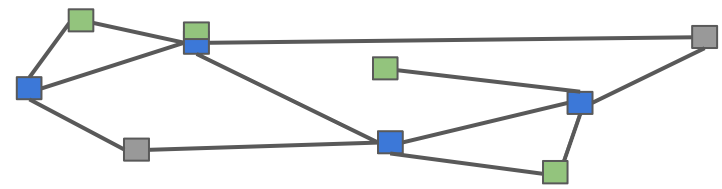
Need all-to-all communication because R and X are non-sparse

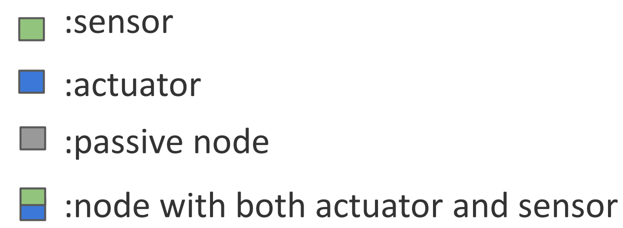
A distributed algorithmic framework applicable to systems with underlying non-sparse network graph

Abstraction of Power Networks

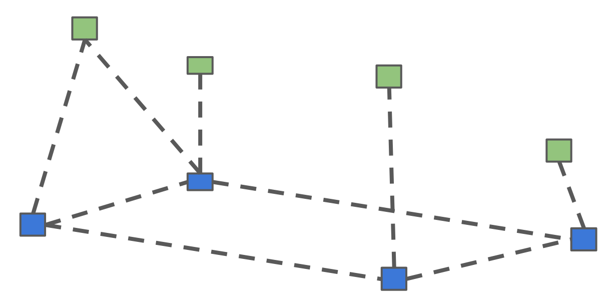
- For online OPF
  - **Sensor:** voltage magnitude
  - **Actuator:** active/reactive power
- Each sensor communicates with at least one actuator
- Actuators' network is connected (no need to match physical network)

Interconnected system network





Cyber layer network



Review of Distributed Feedback-Based Algorithm

- Main idea
  - Each actuator has an estimate of all the dual variables ($\bar{\mu}$ and $\underline{\mu}$)
    - Focus on $\bar{\mu}$ as similar logics apply to $\underline{\mu}$

Copy of $\bar{\mu}$ at node 1 (or $\bar{\lambda}^{(1)}$)

$$\bar{\mu} = [\bar{\mu}_1 \quad \cdots \quad \bar{\mu}_M] \Longrightarrow \begin{bmatrix} \bar{\lambda}_1^{(1)} & \bar{\lambda}_2^{(1)} & \cdots & \bar{\lambda}_M^{(1)} \\ \bar{\lambda}_1^{(2)} & \bar{\lambda}_2^{(2)} & \cdots & \bar{\lambda}_M^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{\lambda}_1^{(N)} & \cdots & \cdots & \bar{\lambda}_M^{(N)} \end{bmatrix} \in \mathbb{R}^{N \times M}$$

All the copies of $\bar{\mu}_2$ (or $\bar{\lambda}_2$)

  - The nodes use distributed communication reaches consensus for $\bar{\lambda}_i$ for all i (at an optimum of $\bar{\mu}_i, \mathbb{1}\bar{\mu}_i^*$)

Review of Distributed Feedback-Based Algorithm

- Proposed updating rule:

$$\dot{x}_i = \boxed{\mathcal{M}_i(x_i, \bar{\lambda}^{(i)})},$$
$$\dot{\bar{\lambda}}_k = \boxed{b_k \Pi_{\mathcal{T}_{\mathbb{R}^N_+}^{\bar{\lambda}_k}} \left(v_k - \bar{v}_k \right)} - \boxed{L \bar{\lambda}_k}, \quad \forall k = 1, \cdots, M.$$

Primal update only needs local variables

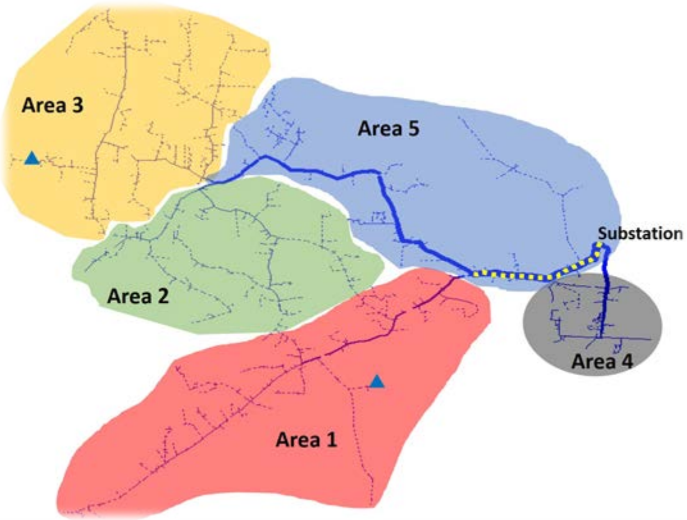
Distributed communication for consensus

Using voltage measurement to find a dual optimum $\bar{\mu}_i^*$

- Distributed communication is only used for the consensus
- Provably convergence
- Event-triggered version available
- Plug-and-play capability
- **Drawback:** requires NM number of variables (originally M)
- Recall the complexity reduction trick

Complexity Reduction on the Distributed Algorithm

- All the actuators in each area reach consensus of
  - Dual variables in the same area
  - Sum of the dual variables in each of other areas
- New partition of $\bar{\mu}$:



$$\bar{\mu} = [\underbrace{\bar{\mu}_{1,1} \quad \cdots \quad \bar{\mu}_{1,M_1}}_{\text{Area 1}} \quad \underbrace{\bar{\mu}_{2,1} \quad \cdots \quad \bar{\mu}_{2,M_2}}_{\text{Area 2}} \quad \cdots \quad \cdots \quad \bar{\mu}_{K,M_K}]$$

Area 1   Area 2   ......   Area K

$$\left[\begin{array}{ccc|ccc} \bar{\lambda}_{a,1}^{(1)} & \cdots & \bar{\lambda}_{a,M_a}^{(1)} & \bar{\lambda}_{a,M_a+1}^{(1)} & \cdots & \bar{\lambda}_{a,M_a+K-1}^{(1)} \\ \bar{\lambda}_{a,1}^{(2)} & \cdots & \bar{\lambda}_{a,M_a}^{(2)} & \bar{\lambda}_{a,M_a+1}^{(2)} & \cdots & \bar{\lambda}_{a,M_a+K-1}^{(2)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \bar{\lambda}_{a,1}^{(N_a)} & \cdots & \bar{\lambda}_{a,M_a}^{(N_a)} & \bar{\lambda}_{a,M_a+1}^{(N_a)} & \cdots & \bar{\lambda}_{a,M_a+K-1}^{(N_a)} \end{array}\right] \in \mathbb{R}^{N_a \times (M_a+K-1)}$$

Each of them surrogates the sum of the dual variables in another area

Complexity Reduction on the Distributed Algorithm

- Similar updating rule:

$$\dot{x}_{a,i} = \mathcal{M}_i(x_{a,i}, \bar{\lambda}_a^{(i)}),$$

$$\dot{\bar{\lambda}}_{a,k} = b_{a,k}\Pi_{\mathcal{T}_{\mathbb{R}_+^{N_a}}^{\bar{\lambda}_{a,k}}}\left(v_{a,k} - \bar{v}_{a,k}\right) - L_a\bar{\lambda}_{a,k}, \quad \forall k = 1, \cdots, M_a$$

$$\dot{\bar{\lambda}}_{a,k} = \sum_{j \in \mathcal{N}_h} b_{h,j}^{(k)}\Pi_{\mathcal{T}_{\Re_+^{N_a}}^{\bar{\lambda}_{a,k}}}(v_{h,j} - \bar{v}_{h,j}) - L_a\bar{\lambda}_{a,k},$$
$$\forall k = M_a + 1, \cdots, M_a + K - 1.$$

Consensus of the dual variables for every other node in the same area

Consensus of the sum of the dual variables in each of other areas

- Soft constraints on the distributed communication
  - Each sensor communicates with at least one actuator in every other area
  - Every row (and column) of the off-diagonal block L_{ik} has at least one non-zero entry
  - Off-diagonal blocks L_{ik} defines $b_{h,j}^{(k)}$

$$\begin{bmatrix} L_1 & L_{12} & \cdots & L_{1K} \\ L_{21} & L_2 & \cdots & L_{2K} \\ \vdots & \ddots & \ddots & \vdots \\ L_{K1} & L_{K2} & \cdots & L_K \end{bmatrix}$$

Complexity Reduction on the Distributed Algorithm

- Equivalent to the original algorithm
- Number of copies is greatly reduced:
  - From NM to $\sum_{a\in\mathcal{K}} N_a(M_a + K - 1)$

- ***The reduced amount of the multiplications are translated to less amount of variables in the distributed approach***

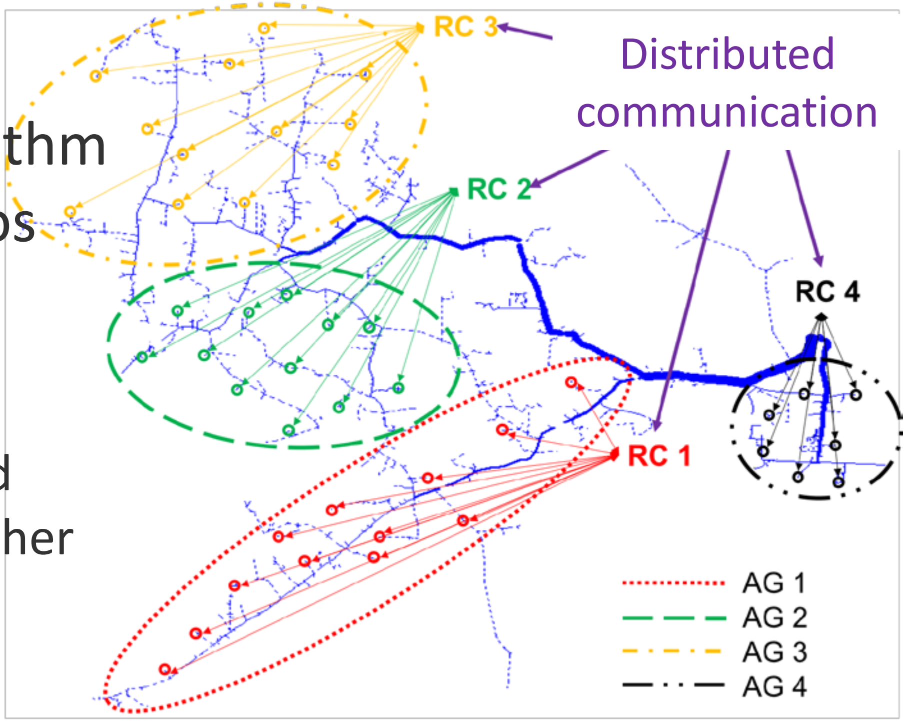
- Proof of convergence
  - Mild assumptions
    - The objective function is continuously differentiable and strongly convex
    - The constraint set is nonempty, compact, and convex
    - The problem is feasible and the Slater condition is satisfied
  - LaSalle's invariance principle to show convergence to invariant set where the copies of the dual variables are at the optimal point

Hierarchical Distributed Control Algorithm

- Assume each area has a regional coordinator (RC)
  - Distributed communication between RC
  - Number of variables further reduced from $\sum_{a\in\mathcal{K}} N_a(M_a + K - 1)$ to $K^2 + M$
  - Similar proof of convergence

- Flexibility to adjust the algorithm to various operation scenarios

- Future works:
  - Check how the feedback-based algorithm complement with other controls (e.g. black start, LTC)



Q&A

Related Publication

1. X. Zhou, Z. Liu, C. Zhao, and L. Chen, “Accelerated Voltage Regulation in Multi-Phase Distribution Networks Based on Hierarchical Distributed Algorithm”, *IEEE Trans. on Power System,* 2019. **(OPF)**
2. X. Zhou, Z. Liu, W. Wang, C. Zhao, F. Ding, L. Chen, “Hierarchical Distributed Voltage Regulation in Networked Autonomous Grids”, *American Control Conference*, 2019. **(OPF)**
3. X. Zhou, Z. Liu, C. Zhao, Y. Guo, and L. Chen, “Gradient-Based Multi Area Distributed Distribution System State Estimation”, Under review, *IEEE Trans. on Smart Grid*. **(DSSE)**
4. Y. Guo, X. Zhou, C. Zhao, Y. Chen, T. H. Summers and L. Chen, “On Optimal Power Flow Problems with State Estimation Feedback for Distribution Networks”, *American Control Conference,* 2020. **(Joint DSSE-OPF)**
5. C.-Y. Chang, M. Colombino, J. Cortes, and E. Dall’Anese, “Saddle-Flow Dynamics for Distributed Feedback-Based Optimization”, IEEE Control Systems Letters, vol. 3, no. 4, pp 948-953, 2019 **(Dist. Algo.)**
6. C.-Y. Chang, X. Zhou, A. Bernstein, “Computation-efficient algorithm for distributed feedback optimization of distribution grids,” submitted to 11th IEEE SmartGridComm. **(Low Complexity Dist. Algo.)**
7. X. Zhou, Y. Chen, Z. Liu, C. Zhao, and L. Chen, “A Multi-Level Algorithm for Large Distribution System Optimal Power Flow”, submitted to *IEEE Power Engineering Letter*. **(Multi-Level OPF)**