Measurement-Based Methods for Model Reduction, Identification, and Distributed Optimization of Power Systems

Seyed Behzad Nabavi

Department of Electrical and Computer Engineering
North Carolina State University

24/4/2015
Part I– Identification of Dynamic Reduced-Order Models of Power Systems
Introduction

- Mathematical modeling of dynamic equivalents of large-scale electric power systems has seen some 40 years of long and rich research history.

- Chow and Kokotovic established the relationship between the slow coherency and weak connections using singular perturbation theory.

- Slow coherency arises from the slower inter-area modes. These interarea modes, if not properly damped, lead to system separation and extensive loss of load.

**Figure:** R. Podemore: Coherency in Power Systems
Model-based Dynamic Equivalencing

- Recent evidences of blackouts have shown the discrepancy between the offline models and the response of the system.
Model-based Dynamic Equivalencing

- Dynamic equivalencing has seen 40 years of active research:
  - linear modal decomposition [Undrill, 71]
  - circuit-theoretic approaches [de Mello, 75]
  - machine aggregation [Germond, 78]
  - enumerative clustering algorithms [Zaborsky, 82]
  - software programs such as DYNEQ and DYNRED [Price, 95]

- Model based methods:
  - need the exact knowledge of the entire power system model,
  - are computationally challenging,
  - are based on idealistic assumption about system structure and clustering.
Measurement-based Dynamic Equivalencing

- PMUs provide high-resolution GPS-synchronized three-phase measurements of voltage, current, phasor, and frequency.

- System operators are, therefore, inclining more towards online models constructed from PMU (Phasor Measurement Unit) measurements.

- We next propose two algorithms to identify these dynamic equivalent models using PMU measurements:
  * Identification of the equivalent linear models
  * Identification of the equivalent nonlinear DAE models
Power System Swing Equation

Nonlinear Electromechanical Model:

\[
\dot{\delta}_i(t) = \omega_s(\omega_i(t) - 1),
\]
\[
M_i \dot{\omega}_i(t) = P_{m_i} - P_{e_i}(t) - D_i(\omega_i(t) - 1),
\]
\[
P_{e_k}(t) + \sum_{l \in \mathcal{N}_k} P_{kl}(t) - P_{L_k}(t) = 0, 
Q_{e_k}(t) + \sum_{l \in \mathcal{N}_k} Q_{kl}(t) - Q_{L_k}(t) = 0
\]

Linearized Kron-Reduced Model (around \((\delta_{i0}, 1)\)):

\[
\begin{bmatrix}
\Delta \dot{\delta}(t) \\ \Delta \dot{\omega}(t)
\end{bmatrix} =
\begin{bmatrix}
\begin{array}{c|c}
0_{n \times n} & \omega_s I_{n \times n} \\
\hline
M^{-1} L & -M^{-1} D 
\end{array}
\end{bmatrix}
\begin{bmatrix}
\Delta \delta(t) \\ \Delta \omega(t)
\end{bmatrix} + B d(t),
\]

where \(\Delta \delta \triangleq [\Delta \delta_1 \ldots \Delta \delta_n]^T\), \(\Delta \omega \triangleq [\Delta \omega_1 \ldots \Delta \omega_n]^T\), \(\Delta \delta, \Delta \omega \in \mathbb{R}^n\),

\[
M = \text{diag}(M_i) \in \mathbb{R}^{n \times n}, \quad D = \text{diag}(D_i) \in \mathbb{R}^{n \times n}, \quad d(t) : \text{unknown disturbance}
\]

\[
[L]_{i,j} = E_i E_j (G_{ij} \cos(\delta_{i0} - \delta_{j0}) - B_{ij} \sin(\delta_{i0} - \delta_{j0})) \quad i \neq j, \quad [L]_{i,i} = -\sum_{k=1}^{n} [L]_{i,k},
\]
Linear Dynamic Equivalent Models

\[
\begin{bmatrix}
\Delta \delta_{1,1} \\
\vdots \\
\Delta \delta_{m,1,1} \\
\Delta \delta_{1,2} \\
\vdots \\
\Delta \delta_{m,2,2} \\
\Delta \delta_{1,r} \\
\vdots \\
\Delta \delta_{m,r,r}
\end{bmatrix} = \mathbf{M}^{-1} \begin{bmatrix}
L_{11} & L_{12} & \cdots & L_{1r} \\
L_{21} & L_{22} & \cdots & L_{2r} \\
\vdots & \vdots & \ddots & \vdots \\
L_{r1} & L_{r2} & \cdots & L_{rr}
\end{bmatrix} \begin{bmatrix}
\Delta \delta_{1,1} \\
\vdots \\
\Delta \delta_{m,1,1} \\
\Delta \delta_{1,2} \\
\vdots \\
\Delta \delta_{m,2,2} \\
\Delta \delta_{1,r} \\
\vdots \\
\Delta \delta_{m,r,r}
\end{bmatrix} = (\mathbf{M}^*)^{-1} \begin{bmatrix}
[L^*]_{1,1} & [L^*]_{1,2} & \cdots & [L^*]_{1,r} \\
[L^*]_{2,1} & [L^*]_{2,2} & \cdots & [L^*]_{2,r} \\
\vdots & \vdots & \ddots & \vdots \\
[L^*]_{r,1} & [L^*]_{r,2} & \cdots & [L^*]_{r,r}
\end{bmatrix} \begin{bmatrix}
\Delta \delta_{1} \\
\vdots \\
\Delta \delta_{r}
\end{bmatrix}
\]

Aggregated Transmission Network Graph
Identification of Linear Equivalent Models

The reduced-order model:
\[
\begin{bmatrix}
\Delta \delta^S(t) \\
\Delta \omega^S(t)
\end{bmatrix} = \begin{bmatrix}
0_{r \times r} & \omega_s I_{r \times r} \\
(M^s)^{-1} L^s & -(M^s)^{-1} D^s
\end{bmatrix} \begin{bmatrix}
\Delta \delta^S(t) \\
\Delta \omega^S(t)
\end{bmatrix} + B^s d(t).
\]

Assumptions:
- The area partitioning for our system is known apriori.
- There is at least one PMU at a generator bus in each area \(S\).
- \(A^s\) and \(B^s\) are a controllable pair.

Objective:
- Finding the equivalent linear model of a power system from \(y_i(t), i \in S\):
  \[
y_i(t) = \{\tilde{V}_i(t), \tilde{l}_{i,j}(t)\}, \ i \in S, \ j \in N_i.
\]

Proposed Identification Steps:
- Extract \(\Delta \delta_k^S(t)\) for each area \(k\) from \(y_i(t), i \in S\).
- Identification of \(A^s\).
Extraction of $\Delta \delta_k^s(t)$ for Each Area $k$

- **Step 1:** Extract $\Delta \delta_i(t)$ from $y_i(t)$:

  \[
  E_i(t) \angle \delta_i(t) = jx_d' \angle \phi_i(t) + V_i(t) \angle \theta_i(t)
  \]

  $\Rightarrow \hat{\delta}_i(t) = \angle (jx_d' \angle \phi_i(t) + V_i(t) \angle \theta_i(t))$

  $\Delta \hat{\delta}_i(t) = \hat{\delta}_i(t) - \hat{\delta}_i(t_0)$.

- **Step 2:** Extract $\Delta \hat{\delta}_k^s(t)$ from $\Delta \hat{\delta}_{i,k}(t)$, (generator $i$ belonging to area $k$):

  \[
  \Delta \delta_{i,k}(t) = \Delta \delta_{i,k}^0(t) + \sum_{l=1}^{r-1} \rho_{il} e^{(-\sigma_l+j\Omega_l)t} + \rho_{il}^* e^{(-\sigma_l-j\Omega_l)t} + \sum_{l=r}^{n-1} \rho_{il} e^{(-\sigma_l+j\Omega_l)t} + \rho_{il}^* e^{(-\sigma_l-j\Omega_l)t},
  \]

  - $\Delta \delta_{i,k}^s(t)$, inter-area or slow modes
  - $\Delta \delta_{i,k}^f(t)$, intra-area or fast modes

  Use a modal decomposition technique such as Prony to decompose $\Delta \delta_{i,k}(t)$

  Form $\Delta \delta_{i,k}^s(t)$ by retaining only the modes in [0.1, 1] Hz.
Extraction of $\Delta \delta_k^s(t)$ for Each Area $k$

- We truncate $\Delta \delta_{i,k}(t)$ to extract $\Delta \delta_{i,k}^s(t)$.
- From the coherency assumption
  \[
  \Delta \delta_{1,k}^s(t) \approx \Delta \delta_{2,k}^s(t) \approx \cdots \approx \Delta \delta_{m_k,k}^s(t)
  \]
- We set $\Delta \delta_k^s(t) = \Delta \delta_{i,k}^s(t)$. 
Identification of $A^s$

- The reduced-order model:

$$\begin{bmatrix} \Delta \dot{\delta}^s(t) \\ \Delta \dot{\omega}^s(t) \end{bmatrix} = \begin{bmatrix} 0_{r \times r} & \omega_s I_{r \times r} \\ (M^s)^{-1} L^s & -(M^s)^{-1} D^s \end{bmatrix} \begin{bmatrix} \Delta \delta^s(t) \\ \Delta \omega^s(t) \end{bmatrix} + A^s B^s d(t).$$

- Solve the following NLS problem (assuming $d(t)$ is a momentary perturbation at $t = t_0$):

$$\min_{A^s} \int_{t_1}^{t_m} \| \begin{bmatrix} \Delta \delta^s(t, A^s) \\ \Delta \omega^s(t, A^s) \end{bmatrix} - \begin{bmatrix} \Delta \hat{\delta}^s(t) \\ \Delta \hat{\omega}^s(t) \end{bmatrix} \|_2^2 \, dt$$

where,

$$\begin{bmatrix} \Delta \delta^s(t, A^s) \\ \Delta \omega^s(t, A^s) \end{bmatrix} = \exp(A^s(t - t_1)) \begin{bmatrix} \Delta \hat{\delta}^s(t_1) \\ \Delta \hat{\omega}^s(t_1) \end{bmatrix},$$

- $\Delta \hat{\omega}^s(t)$ is calculated from the numerical differentiation of $\Delta \hat{\delta}^s(t)$ normalized by $\omega_s$. 
Identifiability Analysis of $A^s$

- Lemma: [Bellman and Astrom-70] Consider the system

\[ \dot{x} = Ax + Bu, \quad y = Cx \]

If the matrix $C$ is full column-rank and the system is controllable, then $A$ and $B$ can be determined uniquely from input output data.

- In our identification problem, we assume $(A^s, B^s)$ to be a controllable pair, and $C = I_{2n}$ (full column-rank), thus $A^s$ is identifiable.

- More results on identifiability analysis will be provided in Part III (joint work with Dr. P. P. Khargonekar).
A Case Study– NPCC 48 Machine Model
**A Case Study- NPCC 48 Machine Model**

- Defining the error:

  \[ J_a(k) = \frac{1}{t_m - t_1} \int_{t_1}^{t_m} |\Delta \delta_{k,\text{reduced}}(t) - \Delta \delta_{k,\text{actual}}(t)| \, dt. \]

- \( \sum_k J_a(k) = 10.2232(\text{deg}) \) for the model-based method, and \( \sum_k J_a(k) = 4.6017(\text{deg}) \) for our measurement-based method.
Identification of the Equivalent DAE Models

- The linear equivalent models are in the Kron’s form.
- This model is not a very suitable choice for:
  1. Identification of the individual equivalent parameters such as inertia \( (M_i) \)
  2. Shunt controller design purposes
  3. Describing the system behavior for large disturbances (transient stability)

\[
\dot{\delta}_i(t) = \omega_s(\omega_i(t) - 1), \\
M_i\ddot{\delta}_i(t) = P_{mi} - P_{ei} - D_i(\omega_i(t) - 1), \\
\dot{\omega}_i^s(t) = \omega_i^s(t) - \omega_s, \\
M_i^s\ddot{\omega}_i^s(t) = P_{mi}^s - P_{ei}^s - D_i^s(\omega_i^s(t) - 1),
\]
Identification of the Equivalent DAE Models

Assumptions:
- The area partitioning for our system is known apriori.
- The boundary buses of all areas are equipped with PMUs (denoted by $S$).

Objective:
- Finding the equivalent DAE model of a power system from $y_i(t)\ i \in S$:

$$y_i(t) = \{\tilde{V}_i(t), \tilde{I}_{i,j}(t)\}, \ i \in S, \ j \in \mathcal{N}_i.$$

Proposed Identification Steps:
- Finding the equivalent pilot bus voltages and currents.
- Estimating the equivalent area impedances.
- Estimating the equivalent generator parameters.
- Estimating the inter-area impedances.
Equivalent Pilot Bus Voltage and Current

Step 1: Use $y_k$ to calculate $\tilde{V}_{pk}(t)$ and $\tilde{I}_{pk}(t)$

$$\tilde{I}_{pk}(t) \triangleq I_{pk}(t)\angle \phi_{pk}(t) = \sum_{i \in B_k} \tilde{i}_i(t), \quad \tilde{V}_{pk}(t) \triangleq V_{pk}(t)\angle \theta_{pk}(t) = \frac{\sum_{i \in B_k} \tilde{V}_i(t)\tilde{I}_i^*(t)}{\tilde{I}_{pk}^*(t)}$$
Equivalent Pilot Bus Voltage and Current

- **Step 2:** Construction of $\tilde{V}_{p_k}^s(t)$ and $\tilde{I}_{p_k}^s(t)$
  - The modal decomposition of $\delta_i^s(t)$:
    \[
    \delta_i^s(t) \approx \sum_{l=1}^{2r} \rho_{jl} e^{\lambda_l t} + \sum_{k=1}^{2r} \sum_{l=1}^{2r} \rho'_{jkl} e^{(\lambda_l + \lambda_k) t} \Rightarrow V_{p_k}^s(t) = \sum_{l=1}^{2r} \alpha_{lk} e^{\lambda_l t} + \sum_{i=1}^{2r} \sum_{j=1}^{2r} \alpha'_{ijk} e^{(\lambda_i + \lambda_j) t},
    \]
  - Use Prony to decompose $V_{p_k}^s(t)$:
    \[
    V_{p_k}^s(t) = \sum_{l=1}^{N} \beta_{lk} e^{\gamma_l t}
    \]
  - Retain only those modal components within the [0.1,1] Hz. The sum of these selected modal components are classified as $V_{p_k}^s(t)$.
  - Apply the same procedure to extract $\theta_{p_k}^s(t)$, $I_{p_k}^s(t)$, and $\phi_{p_k}^s(t)$.
Equivalent Area Impedance

- KVL in the equivalent circuit:

\[ E_k^s(t) \angle \delta_k^s(t) = (r_k^s + jx_{d_k}^s)\tilde{I}_{p_k}^s(t) + \tilde{V}_{p_k}^s(t). \]

- For any time instance:

\[ \Phi_0 \triangleq |(r_k^s + jx_{d_k}^s)\hat{I}_{p_k}^s(t_0) \angle \hat{\phi}_{p_k}^s(t_0)) + \hat{V}_{p_k}^s(t_0) \angle \hat{\theta}_{p_k}^s(t_0)|, \]

\[ \vdots \]

\[ \Phi_m \triangleq |(r_k^s + jx_{d_k}^s)\hat{I}_{p_k}^s(t_m) \angle \hat{\phi}_{p_k}^s(t_m)) + \hat{V}_{p_k}^s(t_m) \angle \hat{\theta}_{p_k}^s(t_m)|. \]

- The estimation of \( r_k^s \) and \( x_{d_k}^s \) can be posed as the following NLS problem:

\[ \min_{x_{d_k}^s, r_k^s} \text{var}(\Phi_0, \ldots, \Phi_m), \]
Estimating the equivalent generator parameters

- Solve the following NLS problem

\[
\min_{M_k^s, D_k^s, P_{m_k}^s} \int_{t_0}^{t_m} |\delta_k^s(t) - \delta_k^s(t, M_k^s, D_k^s, P_{m_k}^s)|^2 dt,
\]

where

\[
\dot{\delta}_k^s(t) = \omega_k^s(t) - \omega_s,
\]

\[
M_k^s \dot{\omega}_k^s(t) = P_{m_k}^s - P_{e_k}^s - D_i^s(\omega_k^s(t) - 1),
\]

\[
\delta_k^s(t) = \delta_k^s(t_0), \quad \omega_k^s(t) = \omega_k^s(t_0), \quad P_{e_k}^s(t) = \text{Re}(\hat{E}_k^s(t) \angle \delta_k^s(t) \tilde{I}_{p_k}^s(t))
\]
Estimating the inter-area impedances

- KCL on equivalent pilot buses:
\[
Y^s \left[ \tilde{V}^s(t_0) | \cdots | \tilde{V}^s(t_m) \right] = \left[ \tilde{I}^s(t_0) | \cdots | \tilde{I}^s(t_m) \right],
\]

- Estimate \( Y^s \) by solving:
\[
\min_{Y^s} \| Y^s \tilde{V}^s - \tilde{I}^s \|^2_F,
\]
\[
s.t. \quad Y^s = (Y^s)^T
\]
A Case Study– IEEE 39 Bus Model

Area 1

Area 2

Area 3

Area 4

$H_1' = 510.6557$
$D_1' = 0.3227M_1'$

$H_2' = 82.3485$
$D_2' = 0.4417M_2'$

$H_3' = 267.2335$
$D_3' = 0$

$H_4' = 106.6757$
$D_4' = 1.6541M_4'$

Model Reduction, Identification, and Distributed Optimization of Power Systems
© 2015 by S. Nabavi
Future Work

- Investigating the utility of the reduced order models for shunt controller design purposes (such as Static Var Compensator (SVC)).

\[
G_k^s \quad \rightarrow \quad \tilde{V}_{pk}^s(t) \quad \rightarrow \quad \text{SVC} \\
\tilde{I}_{pk}^s(t) \quad \rightarrow \quad \text{SVC} \quad \rightarrow \quad \text{Coherent Area } k
\]
Part II– Distributed Optimization Algorithms for Wide-Area Oscillation Monitoring in Power Systems
Introduction

- In Part I, we describe methods to identify the equivalent models from PMU. In Part II, we use PMUs to identify the (inter-area) oscillation modes from PMUs in a distributed way.

- Majority of modal estimation algorithms are centralized such as: Eigenvalue Realization Algorithm (ERA) [Sanchez-Gasca-99], Prony analysis [Hauer-90], Robust Least Squares [Zhuo-08], and Hilbert-Huang transform [Messina-06].

- As the number of PMUs scales up into the thousands, the current state-of-the-art centralized architectures will no longer be sustainable.
Wide-Area Oscillation Monitoring

Using PMU measurements to estimate the frequency, damping factor and residue of the different electro-mechanical oscillation modes
Wide-Area Oscillation Monitoring

Using PMU measurements to estimate the frequency, damping factor and residue of the different electro-mechanical oscillation modes
Oscillation Monitoring

\[
f_0 + f_1 z^{-1} + f_2 z^{-2} + \cdots + f_n z^{-n} \over 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_n z^{-n}}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n}}
\]
Centralized Prony Method

Step 1. Find $a_1$ through $a_{2n}$

\[
\begin{bmatrix}
\Delta \theta_i(2n) \\
\Delta \theta_i(2n + 1) \\
\vdots \\
\Delta \theta_i(2n + \ell)
\end{bmatrix}
\begin{bmatrix}
c_i
\end{bmatrix}
= 
\begin{bmatrix}
\Delta \theta_i(2n - 1) & \cdots & \Delta \theta_i(0) \\
\Delta \theta_i(2n) & \cdots & \Delta \theta_i(1) \\
\vdots & \ddots & \vdots \\
\Delta \theta_i(2n + \ell - 1) & \cdots & \Delta \theta_i(\ell)
\end{bmatrix}
\begin{bmatrix}
-a_1 \\
a_2 \\
\vdots \\
-a_{2n}
\end{bmatrix}
\]

Finding the global $a$ using all available measurements by solving:

\[
\begin{bmatrix}
c_1 \\
\vdots \\
c_p
\end{bmatrix}
= 
\begin{bmatrix}
H_1 \\
\vdots \\
H_p
\end{bmatrix}
\begin{bmatrix}
a
\end{bmatrix}
\]

Solve this using Batch Least Squares - Centralized Prony Method

Step 2. Find the eigenvalues of $A$ (i.e., $-\sigma_i \pm j\Omega_i$) by
- Finding the roots of discrete-time transfer function ($z_1$ through $z_{2n}$)
- Converting them from discrete-time to continuous-time
Centralized Prony Method

\[ \theta_i \rightarrow (H_i, c_i), \quad i = 1, \ldots, p \]

\[ \Rightarrow a = \arg \min_{a} \frac{1}{2} \| \begin{bmatrix} H_1 \\ \vdots \\ H_p \end{bmatrix} a - \begin{bmatrix} c_1 \\ \vdots \\ c_p \end{bmatrix} \|_2^2 \]
Distributing the Prony Method

N Computational Areas:

\( \Delta \theta_{j,i} \): PMU \( i \) in area \( j \)

\( \Delta \theta_{j,i} \rightarrow H_{j,i}, \ c_{j,i} \)

\[\hat{H}_j \triangleq \begin{bmatrix} H_{j,1}^T & H_{j,2}^T & \cdots & H_{j,N_j}^T \end{bmatrix}^T,\]

\[\hat{c}_j \triangleq \begin{bmatrix} c_{j,1}^T & c_{j,2}^T & \cdots & c_{j,N_j}^T \end{bmatrix}^T\]

\( N_j \): is the total number of PMUs in Area \( j \),

Global Consensus Problem:

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{N} \frac{1}{2} \| \hat{H}_i a_i - \hat{c}_i \|_2^2 \\
\text{subject to} & \quad a_i - z = 0, \text{ for } i = 1, \ldots, N
\end{align*}
\]

Use Alternating Direction Method of Multipliers (ADMM) to solve it
Distributing the Prony Method

Three Distributed Cyber-Physical Architectures (Using ADMM):

- Standard ADMM
  - Asynchronous ADMM
- Hierarchical ADMM
- Distributed ADMM
Distributed Prony using Standard ADMM (S-ADMM)

\[
\begin{align*}
\text{minimize} & \quad \sum_{j=1}^{N} \frac{1}{2} \| \hat{H}_j a_j - \hat{c}_j \|^2 \\
\text{subject to} & \quad a_j - z = 0, \text{ for } j = 1, \ldots, N
\end{align*}
\]

Augmented Lagrangian:

\[
L_\rho = \sum_{j=1}^{N} \left( \frac{1}{2} \| \hat{H}_j a_j - \hat{c}_j \|^2 + w_j^T (a_j - z) + \frac{\rho}{2} \| a_j - z \|^2 \right),
\]

\(a_j, z\): the primal variable
\(w_j\): the dual variable
\(\rho\): penalty factor
Distributed Prony using S-ADMM

Iteration $k$

- Each PDC updates $a_j$ locally
  
  
  
  $a_j^{k+1} = ((H_j^k)^T H_j^k + \rho I)^{-1} ((H_j^k)^T c_j^k - w_j^k + \rho z^k)$
Distributed Prony using S-ADMM

Iteration $k$

- Each PDC updates $a_j$ locally
  \[ a_j^{k+1} = ((H_j^k)^TH_j^k + \rho I)^{-1}((H_j^k)^Tc_j^k - w_j^k + \rho z^k) \]
- PDC $j$ sends $a_j^{k+1}$ to the central PDC.
Distributed Prony using S-ADMM

Iteration $k$

- Each PDC updates $a_j$ locally
  
  $$a_j^{k+1} = ((H_j^k)^T H_j^k + \rho l)^{-1} ((H_j^k)^T c_j^k - w_j^k + \rho z^k)$$

- PDC $j$ sends $a_j^{k+1}$ to the central PDC.

- The central PDC receives $a_j^{k+1}$ from all PDCs.
Distributed Prony using S-ADMM

Iteration $k$

- Each PDC updates $a_j$ locally
  \[ a_{j}^{k+1} = ((H_j^k)^T H_j^k + \rho I)^{-1}((H_j^k)^T c_j^k - w_j^k + \rho z_j^k) \]
- PDC $j$ sends $a_{j}^{k+1}$ to the central PDC.
- The central PDC receives $a_{j}^{k+1}$ from all PDCs.
- The central PDC calculates $z_{j}^{k+1} = \frac{1}{N} \sum_{j=1}^{N} a_{j}^{k+1}$. 

Supervisory ISO

PDC 1

PDC 2

PDC 3

PDC 4

Estimated Prameters

PMU Measurements
Distributed Prony using S-ADMM

Iteration $k$

- Each PDC updates $a_j$ locally
  
  $$a_j^{k+1} = ((H_j^k)^T H_j^k + \rho I)^{-1} ((H_j^k)^T c_j^k - w_j^k + \rho z^k)$$

- PDC $j$ sends $a_j^{k+1}$ to the central PDC.

- The central PDC receives $a_j^{k+1}$ from all PDCs.

- The central PDC calculates $z^{k+1} = \frac{1}{N} \sum_{j=1}^{N} a_j^{k+1}$.

- The central PDC sends $z^{k+1}$ to local PDCs.
Distributed Prony using S-ADMM

Iteration $k$

- Each PDC updates $a_j$ locally
  \[ a_j^{k+1} = ((H_j^k)^T H_j^k + \rho I)^{-1}((H_j^k)^T c_j^k - w_j^k + \rho z^k) \]

- PDC $j$ sends $a_j^{k+1}$ to the central PDC.

- The central PDC receives $a_j^{k+1}$ from all PDCs.

- The central PDC calculates $z_j^{k+1} = \frac{1}{N} \sum_{j=1}^N a_j^{k+1}$.

- The central PDC sends $z_j^{k+1}$ to local PDCs.

- PDC $j$ calculates $w_j^{k+1}$ as
  \[ w_j^{k+1} = w_j^k + \rho (a_j^{k+1} - z_j^{k+1}) \]
Distributed Prony using S-ADMM

Iteration $k$

- Each PDC updates $a_j$ locally
  \[ a_j^{k+1} = \left( (H_j^k)^T H_j^k + \rho I \right)^{-1} \left( (H_j^k)^T c_j^k - w_j^k + \rho z_j^k \right) \]
- PDC $j$ sends $a_j^{k+1}$ to the central PDC.
- The central PDC receives $a_j^{k+1}$ from all PDCs.
- The central PDC calculates $z_j^{k+1} = \frac{1}{N} \sum_{j=1}^N a_j^{k+1}$.
- The central PDC sends $z_j^{k+1}$ to local PDCs.
- PDC $j$ calculates $w_j^{k+1}$ as
  \[ w_j^{k+1} = w_j^k + \rho (a_j^{k+1} - z_j^{k+1}) \]

The central PDC and local PDCs find the eigenvalues $-\sigma_i \pm j\Omega_i$ using $z_j^k$. 
Distributed Prony using A-ADMM

Iteration $k$

- Each PDC updates $a_j$ locally
  
  $$a_j^{k+1} = (H_j^k)^T H_j^k + \rho I)^{-1} ((H_j^k)^T c_j^k - w_j^k + \rho z^k)$$

- PDC $j$ sends $a_j^{k+1}$ and $w_j^k$ to the central PDC.

- The central PDC receives $a_j^{k+1}$ and $w_j^k$ from $S^k$, a subset of PDCs.

- The central PDC calculates
  
  $$z^{k+1} = \frac{1}{N} \sum_{j=1}^{N} a_j^{k+1} + \frac{1}{\rho} w_j^k$$

  $$j \notin S : a_j^{k+1} = a_j^k, w_j^k = w_j^{k-1}$$

- The central PDC sends $z^{k+1}$ to local PDCs.

- PDC $j$ calculates $w_j^{k+1}$ as
  
  $$w_j^{k+1} = w_j^k + \rho(a_j^{k+1} - z^{k+1}), \quad j \in S^k, \quad w_j^{k+1} = w_j^k, \quad j \notin S^k$$
Distributed Prony using H-ADMM

- Less communication and computation overhead for the central PDC for large number of PDCs.
- The same convergence properties as the S-ADMM.
Distributed Prony using D-ADMM

- Define a communication graph $\mathcal{G}(V, E)$.
- A different version of the original problem defined over $\mathcal{G}$:
  \[\minimize_{a_1, \ldots, a_N, z} \sum_{j=1}^{N} \frac{1}{2} \left\| \hat{H}_j a_j - \hat{c}_j \right\|^2 \]
  subject to $a_j - a_k = 0$, for $jk \in E(\mathcal{G})$
- Modified Augmented Lagrangian:
  \[
  L^{k}_\rho = \frac{1}{2} \sum_{j=1}^{N} \left( \left\| \hat{H}^k_j a_j - \hat{c}_j^k \right\|^2 + \rho \left( \sum_{v \in P_j} \left\| a^k_{v+1} - a_j - \frac{1}{\rho} w^k_{vj} \right\|^2 + \sum_{v \in S_j} \left\| a_j - a^k_v - \frac{1}{\rho} w^k_{vj} \right\|^2 \right) \right)
  \]
Resilient Distributed Prony using D-ADMM

- One of the challenges of using any distributed computational architecture is ensuring their resiliency to node attacks in the form of data manipulation.
- It is difficult for the ISO to detect a manipulated set of measurement broadcasting from a malicious local PDC.
- The D-ADMM architecture has the advantage that the primal and dual updates are done by local PDCs.
- Let us define the following residual errors:

\[ E_j^k \triangleq \| \hat{H}_j a_j^k - \hat{c}_j \|, \]
\[ E_{ji}^k \triangleq \| \hat{H}_j a_i^k - \hat{c}_j \|, \forall i \in \mathcal{N}_j \]

- let us consider \( \mathcal{G} \) to be a cycle.
Resilient Distributed Prony using D-ADMM

Each PDC $j$ receives the update of $a_i^{k+1}$ for all $l \in P_j$. 
Resilient Distributed Prony using D-ADMM

1. Each PDC $j$ receives the update of $a_l^{k+1}$ for all $l \in P_j$.
2. PDC $j$ updates $a_j$ as $a_j^{k+1} = \arg \min_{a_j} L'_\rho$.
3. PDC $j$ updates all $w_{lj}$ for $l \in P_j$: $w_{lj}^{k+1} = w_{lj}^k - \rho(a_l^{k+1} - a_j^{k+1})$. 
Resilient Distributed Prony using D-ADMM

1. Each PDC $j$ receives the update of $a_i^{k+1}$ for all $l \in P_j$.
2. PDC $j$ updates $a_j$ as $a_j^{k+1} = \arg \min_{a_j} L'_\rho$.
3. PDC $j$ updates all $w_{lj}$ for $l \in P_j$: $w_{lj}^{k+1} = w_{lj}^k - \rho(a_i^{k+1} - a_j^{k+1})$.
4. PDC $j$ sends $a_j^{k+1}$ to all $l \in P_j \cup S_j$, and receives $a_l^{k+1}$ from $l \in S_j$. 
Resilient Distributed Prony using D-ADMM

1. Each PDC \( j \) receives the update of \( a_i^{k+1} \) for all \( l \in P_j \).
2. PDC \( j \) updates \( a_j \) as \( a_j^{k+1} = \arg \min_{a_j} L'_\rho \).
3. PDC \( j \) updates all \( w_{lj} \) for \( l \in P_j \): \( w_{lj}^{k+1} = w_{lj}^k - \rho(a_i^{k+1} - a_j^{k+1}) \).
4. PDC \( j \) sends \( a_j^{k+1} \) to all \( l \in P_j \cup S_j \), and receives \( a_l^{k+1} \) from \( l \in S_j \).
5. PDC \( j \) updates all \( w_{jl} \) for \( l \in S_j \): \( w_{jl}^{k+1} = w_{jl}^k - \rho(a_i^{k+1} - a_j^{k+1}) \).
6. PDC \( j \) calculates \( E_j^k \) and \( E_{jl}^k \) for \( l \in P_j \cup S_j \).
Resilient Distributed Prony using D-ADMM

1. Each PDC $j$ receives the update of $a_i^{k+1}$ for all $l \in P_j$.

2. PDC $j$ updates $a_j$ as $a_j^{k+1} = \arg \min_{a_j} L'_\rho$.

3. PDC $j$ updates all $w_{lj}$ for $l \in P_j$: $w_{lj}^{k+1} = w_{lj}^k - \rho (a_i^{k+1} - a_j^{k+1})$.

4. PDC $j$ sends $a_i^{k+1}$ to all $l \in P_j \cup S_j$, and receives $a_l^{k+1}$ from $l \in S_j$.

5. PDC $j$ updates all $w_{jl}$ for $l \in S_j$: $w_{jl}^{k+1} = w_{jl}^k - \rho (a_i^{k+1} - a_l^{k+1})$.

6. PDC $j$ calculates $E_j^k$ and $E_{jl}^k$ for $l \in P_j \cup S_j$.

7. If $\log(E_{jl}^k) - \log(E_j^k) > E_T$ for any $l \in P_j \cup S_j$, PDC $j$ reports an alert about node $j$ to the ISO.
Resilient Distributed Prony using D-ADMM

1. Each PDC \( j \) receives the update of \( a_i^{k+1} \) for all \( i \in P_j \).
2. PDC \( j \) updates \( a_j \) as \( a_j^{k+1} = \arg \min_{a_j} L'_\rho \).
3. PDC \( j \) updates all \( w_{lj} \) for \( l \in P_j \): \( w_{lj}^{k+1} = w_{lj}^k - \rho(a_i^{k+1} - a_j^{k+1}) \).
4. PDC \( j \) sends \( a_j^{k+1} \) to all \( l \in P_j \cup S_j \), and receives \( a_i^{k+1} \) from \( l \in S_j \).
5. PDC \( j \) updates all \( w_{jl} \) for \( l \in S_j \): \( w_{jl}^{k+1} = w_{jl}^k - \rho(a_i^{k+1} - a_j^{k+1}) \).
6. PDC \( j \) calculates \( E_j^k \) and \( E_{ji}^k \) for \( l \in P_j \cup S_j \).
7. If \( \log(E_{ji}^k) - \log(E_j^k) > E_T \) for any \( l \in P_j \cup S_j \), PDC \( j \) reports an alert about node \( j \) to the ISO.
8. If the ISO gets an alert for PDC \( j \) from all PDCs belonging to \( P_j \cup S_j \) for \( K \) iterations, it removes PDC \( j \), rearranges a new communication graph \( G' \) with the remaining PDCs, and continues the iterations.
Simulation Results

A case study for the IEEE 68 bus model,

- 68 bus, 16 generators
- 5 computational areas
- A three-phase fault is considered occurring at the line connecting buses 1 and 2. The fault starts at $t = 0.1$ sec, clears at bus 1 at $t = 0.15$ sec and at bus 2 at $t = 0.20$ sec, $T_s=0.2$ seconds.
Simulation Results

Figure: S-ADMM

Figure: H-ADMM

Figure: A-ADMM
Simulation Results (D-ADMM) with Attack

$E_T = 8$, $K = 20.$

before detection

after detection
Conclusions

- Development of distributed algorithms is imperative considering the increasing number of PMUs in power systems.

- We consider the problem of estimating the frequencies and damping factors of oscillation modes using Prony method in a distributed way.

- We proposed three cyber-physical architecture for implementing the distributed Prony algorithm using several versions of ADMM.

- The results of the case studies verify that the distributed solution for the oscillation modes converges to the centralized solution.

- Using a heuristic cross verification method we showed how a malicious data manipulation can be detected and isolated.
Future Work

- Investigating the resiliency of the proposed algorithms under more complicated attack scenarios.

(Joint work with Jianhua Zhang)

- Incorporating the asynchronous wide-area communications considering the delay traffic models in both uplink and downlink:

\[
P(t) = \int_{-\infty}^{t} \phi(s) ds = \frac{1}{2} \left[ \text{erf} \left( \frac{\mu}{\sqrt{2}\sigma} \right) + \text{erf} \left( \frac{t - \mu}{\sqrt{2}\sigma} \right) \right] + \frac{(p - 1)}{2} e^{\frac{1}{2} \lambda^2 \sigma^2 + \mu \lambda} e^{-\lambda t} \left[ \text{erf} \left( \frac{\lambda \sigma^2 + \mu}{\sqrt{2}\sigma} \right) + \text{erf} \left( \frac{t - \lambda \sigma^2 - \mu}{\sqrt{2}\sigma} \right) \right].
\]

- Change the update strategy for downlink (needs convergence proof)

\[
w_i^k = w_i^{k-1} + \rho (a_i^k - (z^{k-1} + \gamma(z^{k-1} - z^{k-2}))), \quad i \notin S_2^k
\]
Part III—Graph-Theoretic Identifiability Analysis of Weighted Consensus Networks
Preliminaries

Consider the following single-input consensus model defined over a graph $G(V, E, W)$:

$$\dot{x}_i(t) = \sum_{j \in \mathcal{N}_i} w_{ij} (x_j(t) - x_i(t)) + b_i u(t), \quad i = 1, \ldots, n$$

Defining $x = [x_1 \ x_2 \ \cdots \ x_n]^T$

$$\dot{x}(t) = \mathcal{L}(W)x(t) + Bu(t), \quad y(t) =Cx(t), \quad x(0) = 0,$$

$x \in \mathbb{R}^n$, $\mathcal{L} = -L \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times 1}$, $W = \{w_{ij}, \forall \ i, j\}$

$$[\mathcal{L}]_{i,j} = \begin{cases} -w_{i,j} & i \sim j \\ \sum_{k \in \mathcal{N}_i} w_{i,k} & i = j \\ 0 & \text{otherwise} \end{cases}$$
Distinguishability/Identifiability

\[ \dot{x}(t) = \mathcal{L}(W)x(t) + Bu(t), \quad y(t) = Cx(t), \quad x(0) = 0 \]

Consider two distinct parameter sets \( W \) and \( W' \) (\( W \neq W' \)). These two sets are called *indistinguishable* if the respective models cannot produce different outputs \( y(t) \) for any given input, i.e., \( y(t, W) = y(t, W') \).
Distinguishability/Identifiability

\[ \dot{x}(t) = \mathcal{L}(W)x(t) + Bu(t), \quad y(t) = Cx(t), \quad x(0) = 0 \]

Consider two distinct parameter sets \( W \) and \( W' \) \((W \neq W')\). These two sets are called *indistinguishable* if the respective models cannot produce different outputs \( y(t) \) for any given input, i.e., \( y(t, W) = y(t, W') \).

\[
\frac{Y_1(s)}{U_1(s)} = \frac{Y_2(s)}{U_2(s)} = \frac{4.5}{s^4 + 12s^3 + 33s^2 + 18s}
\]
Distinguishability/Identifiability

\[
\dot{x}(t) = \mathcal{L}(W)x(t) + Bu(t), \quad y(t) = Cx(t), \quad x(0) = 0
\]

Consider two distinct parameter sets \( W \) and \( W' \) \((W \neq W')\). These two sets are called \textit{indistinguishable} if the respective models cannot produce different outputs \( y(t) \) for any given input, i.e., \( y(t, W) = y(t, W') \).

If \( W \) and \( W' \) are not indistinguishable, they are \textit{distinguishable}.

A parameter set \( W \) is said to be \textit{globally identifiable} if for all \( W' \neq W \), \( W \) and \( W' \) are distinguishable.
Identifiability Analysis

Identifiability in terms of Markov Parameters

\[
\dot{x}(t) = \mathcal{L}(W)x(t) + Bu(t), \quad y(t) = Cx(t), \quad x(0) = 0
\]

- **Lemma (Grewel, 1976):** The parameter sets $W$ and $W'$ are indistinguishable if and only if
  \[
  C\mathcal{L}^\ell(W)B = C\mathcal{L}^\ell(W')B, \quad \ell \geq 0.
  \]
- $W$ is identifiable if and only if the mapping from $W$ to the Markov parameters is injective (one-to-one).
Our Proposition

- There are analytical results for identifiability analysis of generic dynamic models (as early as 70’s and 80’s).

- Generally, investigating the parameter identifiability for medium and large-scale systems is a difficult and intractable task.

- We develop a simple sensor placement algorithm to guarantee identifiability of the edge-weights $W$ for consensus networks defined over a class of graphs.

- We integrate the results from the graph theory with these classical results of identifiability.
Preliminaries

- Let us consider a rooted graph $G$ with the root being the input node (node indexed by 1).
- Let us partition $V$ into the following sets:

$$S_i = \{v \in V : d(v, 1) = i\}, \quad i = 0, 1, \ldots, p.$$
The Studied Class of Graphs

**Assumption 1:** For a rooted graph $G$, nodes $v_l, v_q, v_s \in S_i$, $v_j \in S_{i+1}, \forall \ i \geq 1$ satisfy the following properties:

$$
\begin{align*}
&v_l \sim v_j, \ v_q \sim v_j \Rightarrow v_l = v_q, \\
&(v_l \in S^l_i) \sim (v_j \in S^j_i) \Rightarrow l = j, \\
&\dim(\{qv \in E(G) \mid q, v \in S^j_i\}) \leq 1, \forall \ i, j
\end{align*}
$$
The Studied Class of Graphs

- **Assumption 1:** For a rooted graph $G$, nodes $v_l, v_q, v_s \in S_i, v_j \in S_{i+1}, \forall \; i \geq 1$ satisfy the following properties:

  \[ v_l \sim v_j, \quad v_q \sim v_j \implies v_l = v_q, \]

  \[ (v_l \in S^l_i) \sim (v_j \in S^j_i) \implies l = j, \]

  \[ \dim(\{qv \in \mathcal{E}(G) \mid q, v \in S^j_i\}) \leq 1, \; \forall \; i, j \]

- **Assumption 2:** $W$ is identifiable if $C = I_n$.

- Parameter $b$ is not identifiable regardless of choice of $C$. 

![Graph Diagram](attachment:graph.png)
Two Supporting Lemmas

Lemma 1—The following holds for $\mathcal{L}$ of a $G$ satisfying Assumption 1:

$$[\mathcal{L}^k]_{v,1} = \begin{cases} 0 & 0 \leq k \leq d(v,1) - 1 \\
W(P_{v,1}) & k = d(v,1) \end{cases}$$

$P_{v,1}$ is the unique path of length $d(v,1)$ connecting nodes $v$ and 1. $W(P_{v,1})$ is the weight of path $P_{v,1}$:

$$W(P) = \prod_{e \in P} w_e$$

Proof: By strong induction on $k$. 
Two Supporting Lemmas

Lemma 2– Consider a node indexed as $v$ in $G$ and its neighboring nodes denoted by $v_1, \ldots, v_s$. Let $\mathcal{L} = -L$, where $L$ is the weighted Laplacian matrix of $G$. If $\mathcal{H}$ denotes a subgraph of $G$ induced by the set of all edges incident to $v$, and $\mathcal{V}_\mathcal{H}$ and $\mathcal{W}_\mathcal{H}$ denote the vertex set and the weights of all edges belonging to $\mathcal{H}$ respectively, then $[\mathcal{L}^i]_{v_s,1}$ can be uniquely computed from $\mathcal{W}_\mathcal{H}$ and $[\mathcal{L}^i]_{m,1}$, $(m \in \mathcal{V}_\mathcal{H} \setminus \{v_s\})$, $\forall i \geq 1$.

$v_s$ is called an available node.
The Proposed Algorithm

- Start with $S_0$ and place a sensor at this node.
The Proposed Algorithm

- Start with $S_0$ and place a sensor at this node.
- for $k = 1 : p$ for each set of siblings $S^i_k$
  - choose any $|S^i_k| - 1$ nodes belonging to $S^i_k$ and place sensors at them.
The Proposed Algorithm

- Start with $S_0$ and place a sensor at this node.
- for $k = 1 : p$ for each set of siblings $S_k^i$
  - choose any $|S_k^i| - 1$ nodes belonging to $S_k^i$ and place sensors at them.
  - for each neighboring siblings $q, v \in S_{k-1}^i$
    - if $S_k^q$ and $S_k^v$ are both non-empty
      - place an additional sensor in either $S_k^q$ or $S_k^v$.
The Main Theorem

Consider the following model with $G$ satisfying Assumptions 1 and 2.

$$\dot{x}(t) = \mathcal{L}(W)x(t) + Bu(t), \quad y(t) = Cx(t), \quad x(0) = 0$$

If $S \subset V$ is a set of sensor nodes determined by the proposed algorithm, $y(t)$ is the corresponding output measured by $S$, and $H(W)$ is the transfer function from $u(t)$ to $y(t)$, then the mapping from the $W$ to $H(W)$ is one-to-one.
The Main Theorem

Consider the following model with $G$ satisfying Assumptions 1 and 2.

$$\dot{x}(t) = \mathcal{L}(W)x(t) + Bu(t), \quad y(t) = Cx(t), \quad x(0) = 0$$

If $S \subset \mathcal{V}$ is a set of sensor nodes determined by the proposed algorithm, $y(t)$ is the corresponding output measured by $S$, and $H(W)$ is the transfer function from $u(t)$ to $y(t)$, then the mapping from the $W$ to $H(W)$ is one-to-one.

Proof:

$$W_{j-1,j} \triangleq \{w_{u,v} \in W \mid u \in S_{j-1}, \ v \in S_j\},$$

$$W_{j,j} \triangleq \{w_{u,v} \in W \mid u, v \in S_j\}, \quad j = 1, \ldots, p.Q_j \triangleq C\mathcal{L}^j B, \quad j = 1, 2, \ldots, p.$$

The proof follows from strong induction on $j$. In each step we proof the injective mapping of $(W_{j-1,j}, W_{j-1,j-1})$ to $\bigcup_{i=1}^{2n-1} Q_i$. 
More Results

- **Proposition 1:** If the proposed algorithm is applied to a rooted-tree $T$, then the number of placed sensors is equal to the number of non-input leaves of $T$, i.e., the set of leaves that are not the input node.

- **Proposition 2:** If $T$ is a star-graph, then the minimum number of sensors to identify $W$ is $(n - 2)$. 

![Network 1](image1)

![Network 2](image2)

![Network 3](image3)
Summary and Conclusions

- We investigate the identifiability problem in Laplacian consensus NDS.
- We translate the classical results of identifiability in terms of graph properties.
- We propose a sensor placement algorithm for a class of graphs.
- We prove that, our algorithm provides a sufficient condition of identifiability of the edge weights.
Publications (Published/Accepted)

Book Chapter:


Journal Articles:


Conference Proceedings:


Publications (Under review/ To be submitted)

Journal Articles:


Conference Proceedings (submitted):