山东大学电气工程学院
SCHOOL OF ELECTRICAL ENGINEERING OF SHANDONG UNIVERSITY

Stability and Stabilization of Large DCPPS Through Partial Spectral Discretization-based Eigen-Analysis

Hua Ye
Shandong University, China

Sept. 18, 2019

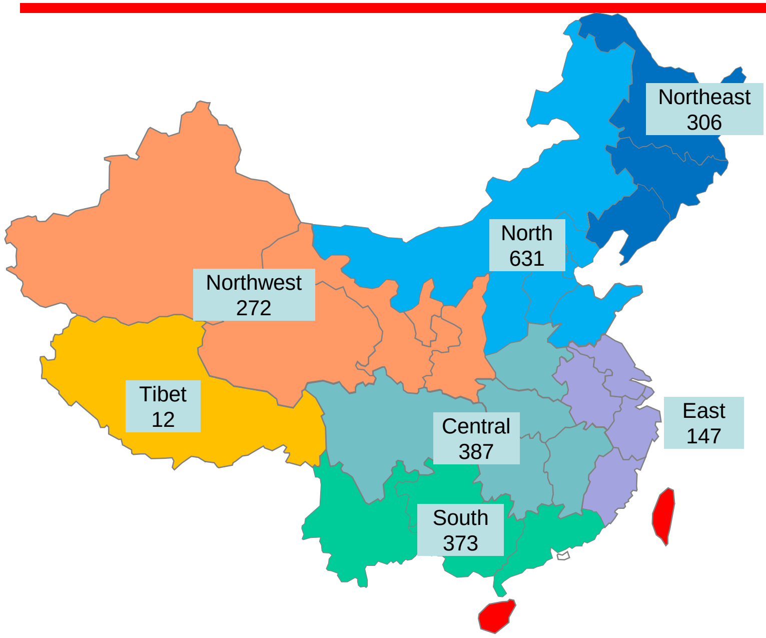
Outline

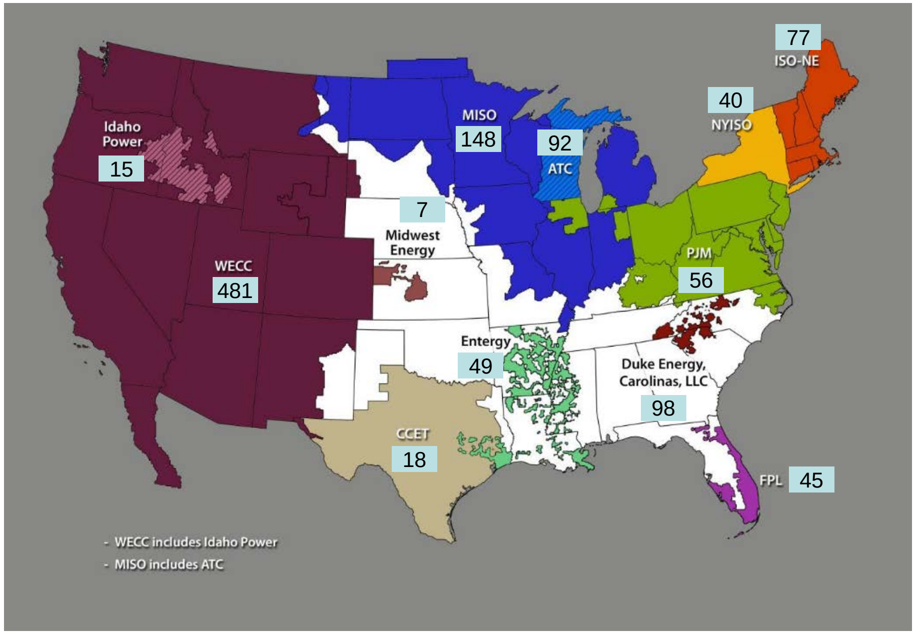
1. Introduction
2. Spectral discretization-based eigen-analysis
3. Partial spectral discretization-based eigenvalue computation method
4. Non-smooth optimization-based WADC tuning method
5. Conclusions

Outline

1. ***Introduction***
2. Spectral discretization-based eigen-analysis
3. Partial spectral discretization-based eigenvalue computation method
4. Non-smooth optimization-based WADC tuning method
5. Conclusions

1. Introduction



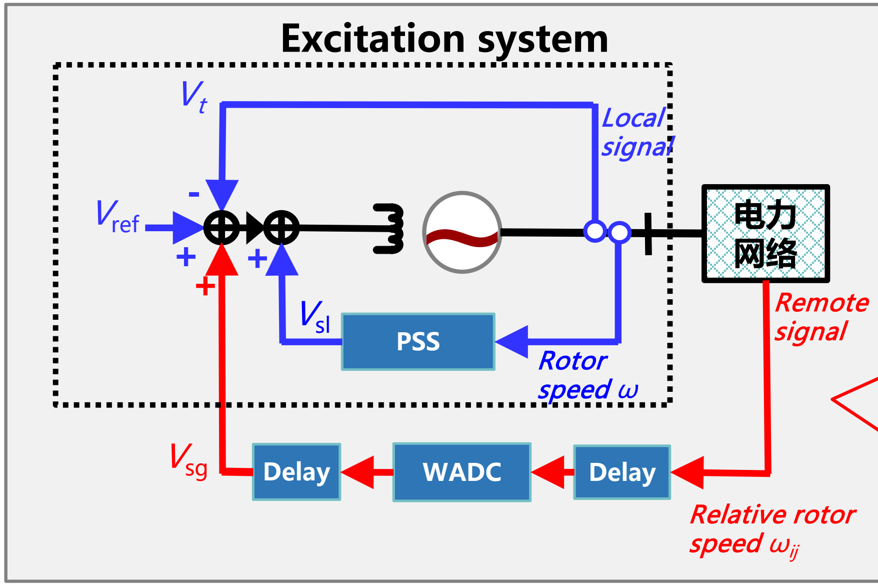


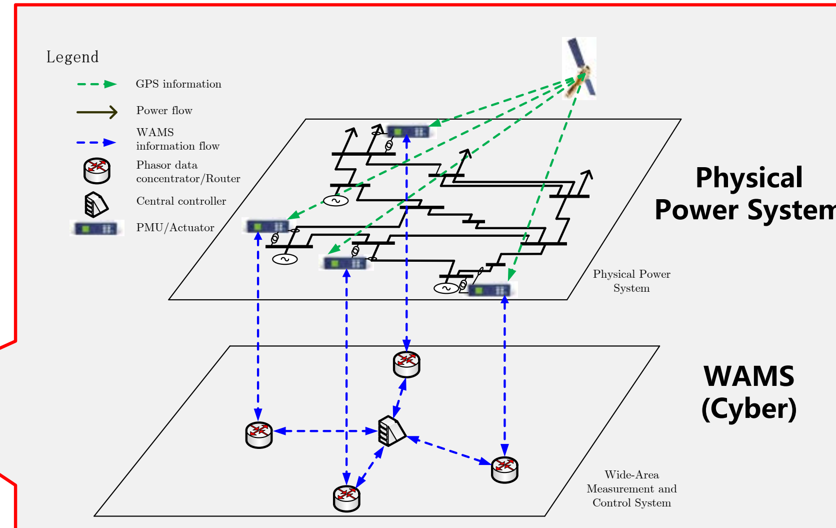
* As of 2012, over **2400** PMUs deployed in 500 kV and higher plants & substations of China.
* Source: Lu C, Shi B, Wu X, et al. Advancing China's smart grid: Phasor measurement units in a wide-area management system. *IEEE Power and Energy Magazine*, 2015, 13(5): 60-71.

* As of 03/31/2013, there were **1126** PMUs installed in US.
* Source: Department of Energy. Synchrophasor Technologies and their Deployment in Recovery Act Smart Grid Programs, August 2013.

- Thousands of phasor measurement units (PMUs) have been deployed in the transmission level all over the world, which provide a new measure for measuring, monitoring and control of the physical power system.
- Especially, wide-area damping controllers (WADCs) can effectively stabilize both local and interarea low frequency oscillations.

Background



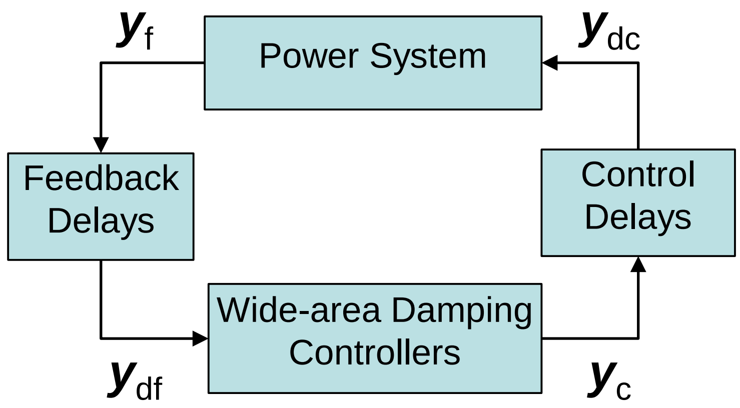


Delayed Cyber-Physical Power System (DCPPS)

- Since **remote signals** are employed as feedback signals, **time delay** in the range of tens to hundreds of millisecond emerges in transmission and process of wide-area measurements.
- With the consideration of time delay effects, the cyber-physical power system (CPPS) had been involved into a ***Delayed CPPS (DCPPS)***
- Time delays compromise the performance of wide-area control system and thus may jeopardize the stability of DCPPS.

DCPPS modeling

Structure of the DCPPS:



The dynamics of the DCPPS can be represented by the following linearized delayed differential equations (DDEs):

$$\begin{cases} \Delta\dot{\boldsymbol{x}}(t) = \tilde{\boldsymbol{A}}_0 \Delta\boldsymbol{x}(t) + \sum_{i=1}^{m} \tilde{\boldsymbol{A}}_i \Delta\boldsymbol{x}(t-\tau_i), \ \ t \geq 0 \\ \Delta\boldsymbol{x}(t) = \boldsymbol{\varphi}, \ \ t \in [-\tau_{\max}, 0] \end{cases}$$

where τ_i (i=1, …, m) are delay constants.

- The most popular methods to deal with DCPPS are: 1) delay-dependent stability criteria; 2) Padé approximation.
- Delay-dependent stability criteria
  - They are sufficient conditions for asymptotic stability and inherently *conservative*.
  - The *scalability* is constrained by the solution of the algebraic Riccati equation (ARE), where only system with less than 100 order can be dealt with.
- Padé approximation
  - Low order Padé approximation results in major phase estimating errors in the case of *large* time delay.

Inertial block vs. Exponential delay term

Inertial block: $\frac{1}{1+\tau s}$ Exponential delay term: $e^{-\tau s}$

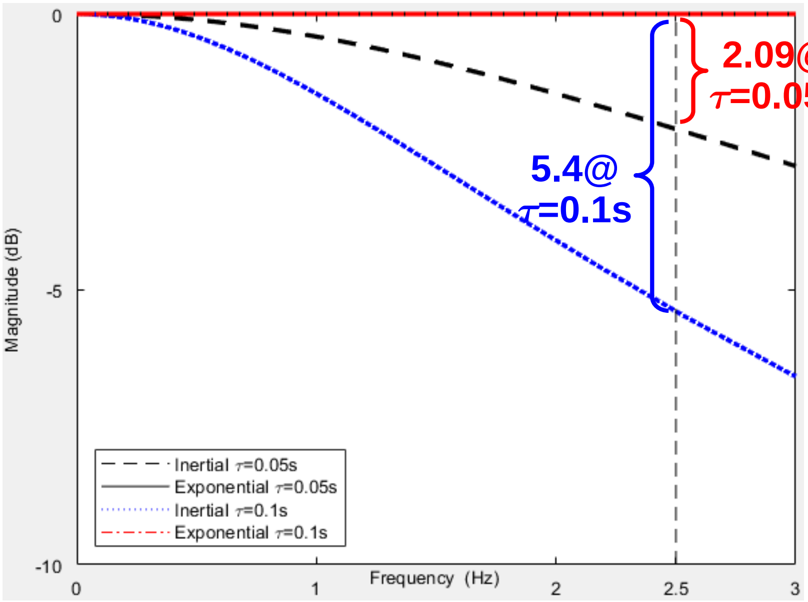


Figure (a) Magnitude responses for inertial block and exponential delay term when delays are 0.05 and 0.1s.

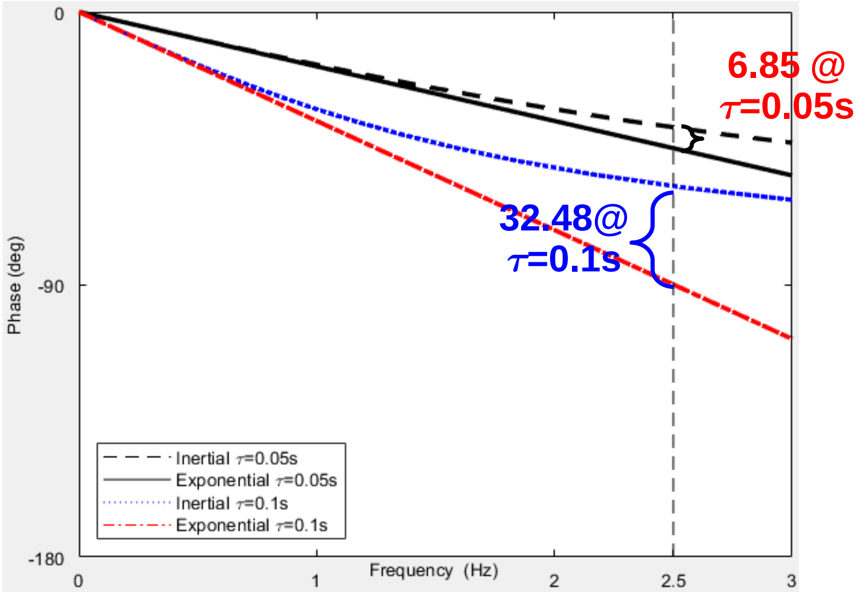


Figure (b) Phase responses for inertial block and exponential delay term when delays are 0.05 and 0.1s.

The inertial block leads to *major* magnitude and phase estimating errors.

1. The inertial block results in magnitude estimating errors of 2.09 and 5.4 db when delays are 0.05 and 0.1 s, respectively.
2. The inertial block results in phase estimating errors of 6.85 and 32.48 degrees when delays are 0.05 and 0.1 s, respectively.

Padé approximation vs. Exponential delay term

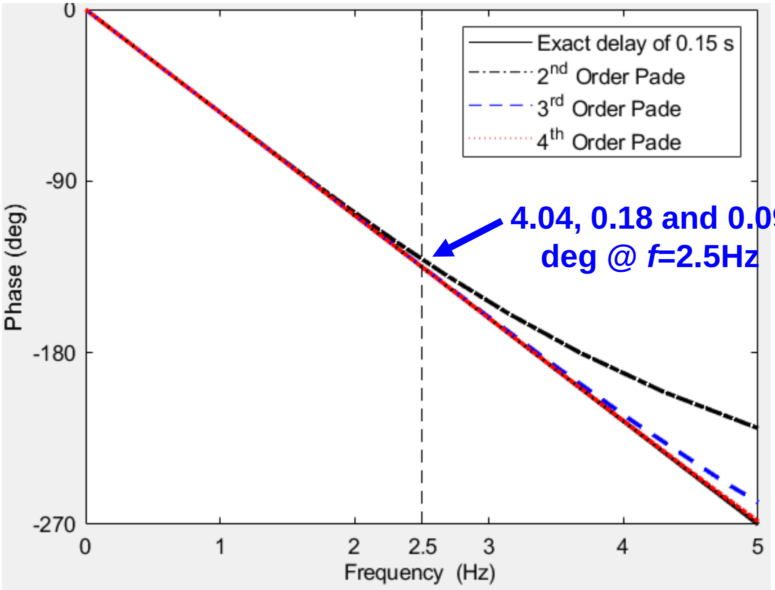


Figure (a) Phase responses of Padé approximation with orders of 2-4 when the delay is equal to 150ms.

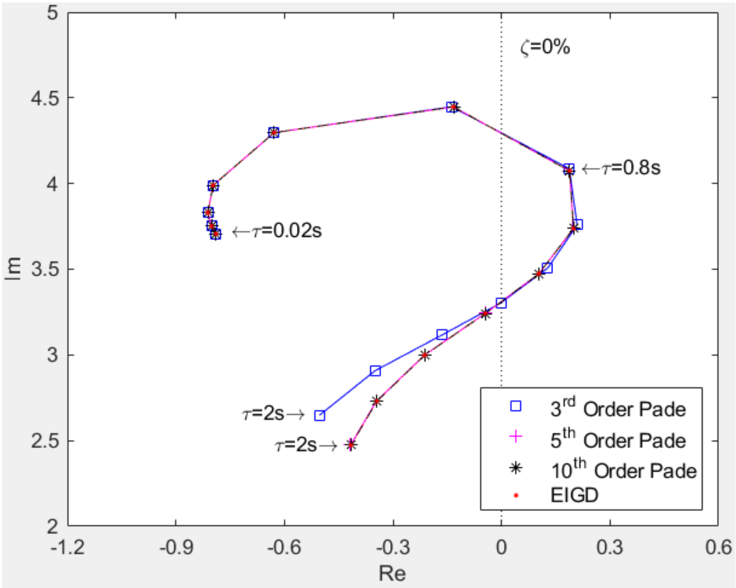


Figure (b) Trajectory of the inter-area oscillation mode of the two-area four-machine test system w.r.t. delay = 0.02, 0.05, 0.1, 0.2:0.2:2.0s.

Low order Padé approx. results in *major* phase estimating errors in the case of *large* time delays.

- When delay equal to 0.15s, the phase estimating errors for Padé approximation with orders of 2-4 are 4.04, 0.18 and 0.09 degrees, respectively.
- When delay increases to 0.5s, errors for Padé approximation with orders of 3 and 5 are 32.35 and 1.16 degrees, respectively.

Outline

1. Introduction

2. ***Spectral discretization-based eigen-analysis***

3. Partial spectral discretization-based eigenvalue computation method

4. Non-smooth optimization-based WADC tuning method

5. Conclusions

2. Spectral discretization-based eigen-analysis

The characteristic equation of DCPPS is:

$$\left(\tilde{A}_0 + \sum_{i=1}^{m} \tilde{A}_i e^{-\lambda\tau_i}\right)\boldsymbol{v} = \lambda\boldsymbol{v}$$

where λ and $\boldsymbol{v}$ are the eigenvalue and the corresponding right-eigenvector.

- **Challenges**
  - Since the exponential delay terms are involved, the characteristic equation is *transcendental* in nature.
  - The equation has *an infinite number* of eigenvalues, which is basically unsolvable by traditional eigenvalue methods.
- **Our solution**
  - To compute a set of the rightmost or the least damped eigenvalues of the system by using the *spectral discretization*-based eigen-analysis methods.

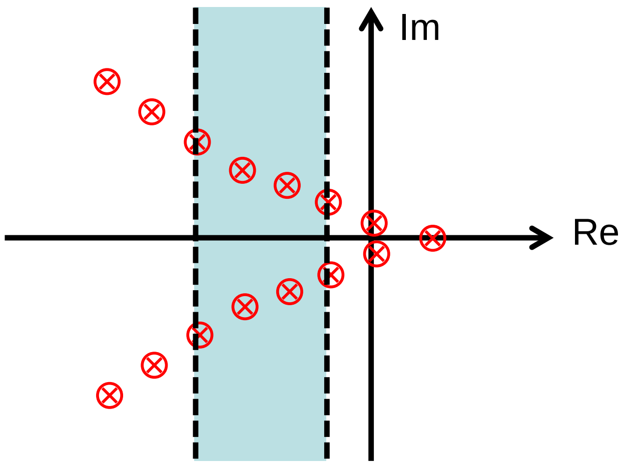
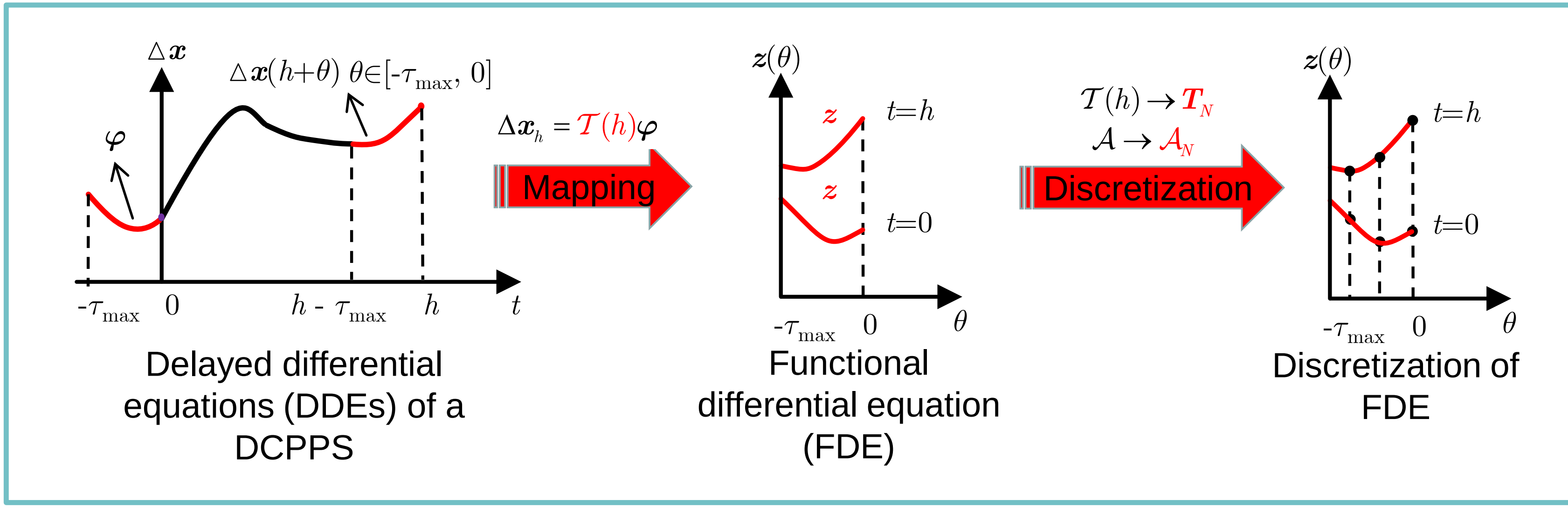


Figure. Illustration of the spectrum of DCPPS

- **Theoretical foundation**
  - There are only *a finite number* of eigenvalues in any vertical strip of the complex plane.
  - The number of eigenvalues lying in the right-half complex plane are at most in a finite number.

Principle of spectral discretization method

The spectral discretization-based eigenvalue method can compute a set of critical eigenvalues from the discretized matrices approximating the solution operator $\mathcal{T}(h)$ and the infinitesimal generator $\mathcal{A}$.



$$\begin{cases}\Delta\dot{\boldsymbol{x}}(t)=\tilde{\boldsymbol{A}}_0\Delta\boldsymbol{x}(t)+\sum_{i=1}^{m}\tilde{\boldsymbol{A}}_i\Delta\boldsymbol{x}(t-\tau_i), & t\geq 0\\ \Delta\boldsymbol{x}(t)=\varphi, & t\in[-\tau_{\max},0]\end{cases}$$

$$\begin{cases}\dot{\boldsymbol{z}}=\mathcal{A}\boldsymbol{z},\ \ \boldsymbol{z}=\Delta\boldsymbol{x}_t=\mathcal{T}(t)\varphi\\ \boldsymbol{z}(0)=\varphi(\theta),\ \theta\in[-\tau_{\max},0]\end{cases}$$

$$\begin{cases}\mathcal{T}(h)\rightarrow \boldsymbol{T}_N\\ \mathcal{A}\rightarrow \mathcal{A}_N\end{cases}$$

$$\left(\tilde{\boldsymbol{A}}_0+\sum_{i=1}^{m}\tilde{\boldsymbol{A}}_i e^{-\lambda\tau_i}\right)\boldsymbol{v}=\lambda\boldsymbol{v}$$

$$\begin{cases}\lambda\in\sigma(\mathcal{A})\\ \lambda=\dfrac{\ln\mu}{h},\mu\in\sigma(\mathcal{T}(h))\setminus\{0\}\end{cases}$$

$$\begin{cases}\hat{\lambda}\in\sigma(\mathcal{A}_N)\\ \hat{\lambda}=\dfrac{\ln\hat{\mu}}{h},\hat{\mu}\in\sigma(T_N)\setminus\{0\}\end{cases}$$

Framework of spectral discretization method

Delayed differential equations (DDE) of DCPPS

$$\Delta\dot{\boldsymbol{x}}(t) = \tilde{\boldsymbol{A}}_0 \Delta\boldsymbol{x}(t) + \sum_{i=1}^{m} \tilde{\boldsymbol{A}}_i \Delta\boldsymbol{x}(t-\tau_i)$$

Semi-group operators

1. Solution operator $\mathcal{T}(h)$
2. Infinitesimal generator $\mathcal{A}$

Functional equations (**ordinary functional equations, ODE**)

$$\Delta\boldsymbol{x}_h = \mathcal{T}(h)\boldsymbol{\varphi}$$
$$\Delta\dot{\boldsymbol{x}}_h = \mathcal{A}\Delta\boldsymbol{x}_h$$

(1) Spectral mapping

$$\lambda \in \sigma(\mathcal{A})$$
$$\lambda = \frac{\ln \mu}{h}, \mu \in \sigma(\mathcal{T}(h)) \setminus \{0\}$$

(2) Spectral discretization

$$(\Delta\boldsymbol{x}_h)^N = \mathcal{T}_N(\varphi)^N$$
$$(\Delta\dot{\boldsymbol{x}}_h)^N = \mathcal{A}_N(\Delta\boldsymbol{x}_h)^N$$

Linear multi-step
Implicit Runge-Kutta
Pseudo-spectral differencing

(3) Spectral estimation

$$\cup\hat{\lambda} = \sigma(\mathcal{A}_N), \ \cup\hat{\mu} = \sigma(\mathcal{T}_N)$$

(4) Spectral correction
(Newton iteration)

$$\hat{\lambda} \to \lambda, \ \hat{\mu} \to \mu$$

Spectral discretization-based eigenvalue algorithms

Lots of numerical methods, such as pseudo-spectral differencing (PS), linear multi-step (LMS) and implicit Runge-Kutta (IRK), can be utilized to discretize the solution operator $\mathcal{T}(h)$ and infinitesimal generator $\mathcal{A}$.

Table. Spectral discretization-based eigenvalue algorithms for large DCPPS

	Pseudo-spectral (PS)	**Linear multi-step (LMS)**	**Implicit Runge-Kutta (IRK)**
Infinitesimal generator ($\mathcal{A}$)	IGD-PS (IIGD) IGD-PS-II (EIGD)	IGD-LMS	IGD-IRK
Solution operator ($\mathcal{T}(h)$)	SOD-PS SOD-PS-II	SOD-LMS	SOD-IRK

Outline

1. Introduction
2. Spectral discretization-based eigen-analysis
3. ***Partial spectral discretization-based eigenvalue computation method***
4. Non-smooth optimization-based WADC tuning method
5. Conclusions

3. Partial spectral discretization-based eigenvalue computation method

- Motivation
  - The dimension of discretized matrices $\mathcal{A}_N$ and $\boldsymbol{T}_N$ approximating $\mathcal{A}$ and $\mathcal{T}(h)$ is $(N+1)n$, which is $N+1$ times of the number of system state variables, resulting in excessive CPU time consumption.
  - The computational burden can be reduced and the efficiency can be improved by reducing the dimension of $\mathcal{A}_N$ and $\boldsymbol{T}_N$.

- Basic idea of partial spectral discretization
  - Considering a few number of wide-area damping controllers (WADCs) are installed in power system to damp out inter-area oscillations, the state variables directly affected by communication delays are *very few*.
  - The dimensions of discretized matrices $\mathcal{A}_N$ and $\boldsymbol{T}_N$ can be significantly reduced through discretizing only the *delayed* state variables instead of *all* state variables.
  - We call this as *partial spectral discretization*.

State variable partitioning

The n system state variables are divided into two sets: n_1 non-delayed variables $\Delta\boldsymbol{x}^{(1)}$ and n_2 delayed variables $\Delta\boldsymbol{x}^{(2)}$. Note that $n_1 >> n_2$.

$$\Delta\boldsymbol{x} = \begin{bmatrix} \Delta\boldsymbol{x}^{(1)} \\ \Delta\boldsymbol{x}^{(2)} \end{bmatrix} \begin{matrix} \rightarrow n_1 \text{ non-delayed variables} \\ \rightarrow n_2 \text{ delayed variables} \end{matrix}$$

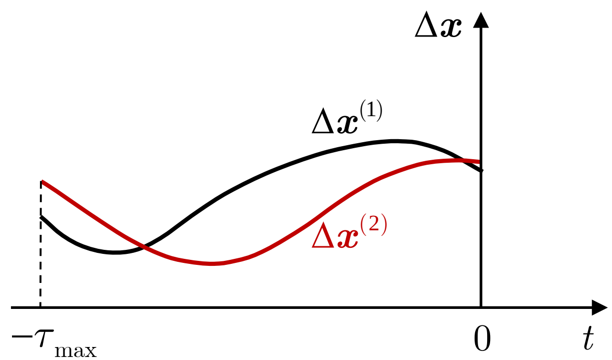


Figure. Partitioning of system state variables over the past time interval $[-\tau_{\max}, 0]$.

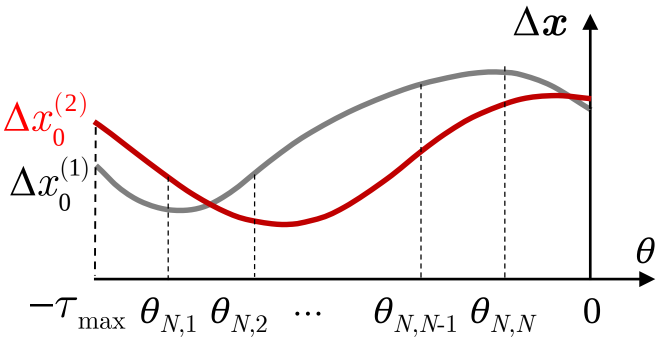
$$\begin{bmatrix} \Delta\dot{\boldsymbol{x}}^{(1)}(t) \\ \Delta\dot{\boldsymbol{x}}^{(2)}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_{11,0} & \boldsymbol{A}_{12,0} \\ \boldsymbol{A}_{21,0} & \boldsymbol{A}_{22,0} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{x}^{(1)}(t) \\ \Delta\boldsymbol{x}_2(t) \end{bmatrix} + \sum_{i=1}^{m} \begin{bmatrix} \boldsymbol{0}_{n_1\times n_1} & \boldsymbol{A}_{12,i} \\ \boldsymbol{0}_{n_2\times n_1} & \boldsymbol{A}_{22,i} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{x}^{(1)}(t-\tau_i) \\ \Delta\boldsymbol{x}^{(2)}(t-\tau_i) \end{bmatrix}$$

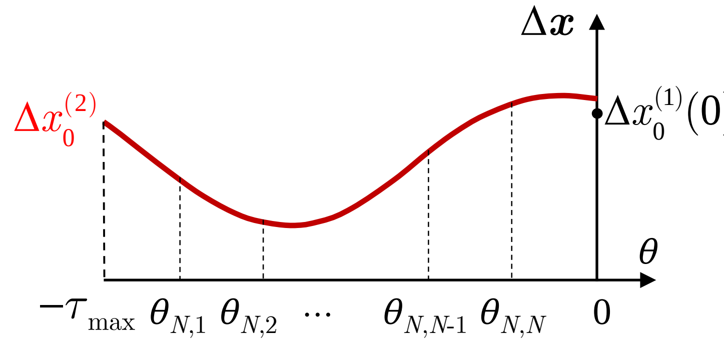
$$\begin{cases} \Delta\dot{\boldsymbol{x}}^{(1)}(t) = \boxed{\boldsymbol{A}_{11,0}\Delta\boldsymbol{x}^{(1)}(t)} + \boldsymbol{A}_{12,0}\Delta\boldsymbol{x}^{(2)}(t) + \sum_{i=1}^{m}\boldsymbol{A}_{12,i}\Delta\boldsymbol{x}^{(2)}(t-\tau_i) \\ \Delta\dot{\boldsymbol{x}}^{(2)}(t) = \boxed{\boldsymbol{A}_{21,0}\Delta\boldsymbol{x}^{(1)}(t)} + \boldsymbol{A}_{22,0}\Delta\boldsymbol{x}^{(2)}(t) + \sum_{i=1}^{m}\boldsymbol{A}_{22,i}\Delta\boldsymbol{x}^{(2)}(t-\tau_i) \end{cases}$$

The derivative of system states does not relate to $\Delta\boldsymbol{x}^{(1)}(\theta)$, $\theta\in[-\tau_{\max}, 0]$.

Principle of partial spectral discretization

At t=0, considering the derivative of system states does not relate to $\Delta \boldsymbol{x}^{(1)}(\theta)$, $\theta \in [-\tau_{max}, 0]$, the evaluation of $\boldsymbol{\varphi}^{(1)}(\theta)$ and $\boldsymbol{\varphi}^{(2)}(\theta)$ at each discrete points $\theta_{N,i}$ (i=0, …, N+1) on $[-\tau_{max}, 0]$ has been reduced to evaluate $\boldsymbol{\varphi}^{(1)}(0)$ at $\theta_{N,N+1}$=0 and $\boldsymbol{\varphi}^{(2)}(\theta)$ at $\theta_{N,i}$ (i=0, …, N+1).





$$\begin{cases} \dfrac{\mathrm{d}\boldsymbol{z}}{\mathrm{d}t} = \mathcal{A}\boldsymbol{z},\ \boldsymbol{z} = \Delta \boldsymbol{x}_t \\ \boldsymbol{z}(0) = \begin{bmatrix} \Delta \boldsymbol{x}_0^{(1)} \\ \Delta \boldsymbol{x}_0^{(2)} \end{bmatrix} = \begin{bmatrix} \varphi^{(1)}(\theta) \\ \varphi^{(2)}(\theta) \end{bmatrix},\ \theta \in [-\tau_{\max}, 0] \end{cases}$$

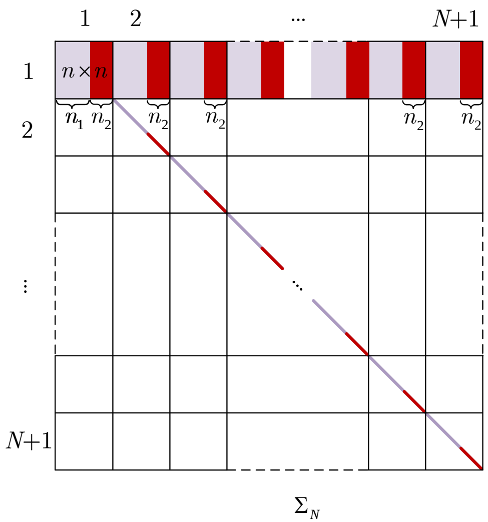
$$\begin{cases} \dfrac{\mathrm{d}\boldsymbol{z}}{\mathrm{d}t} = \mathcal{A}\boldsymbol{z},\ \boldsymbol{z} = \Delta \boldsymbol{x}_t \\ \boldsymbol{z}(0) = \begin{bmatrix} \varphi^{(1)}(0) \\ \varphi^{(2)}(\theta) \end{bmatrix},\ \theta \in [-\tau_{\max}, 0] \end{cases}$$

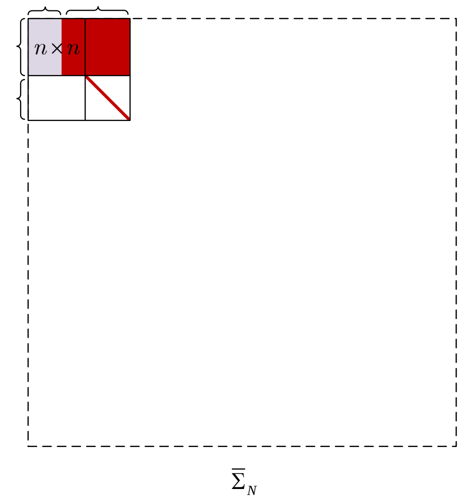
$$\mathcal{A}_N \Phi = \lambda \Phi$$

$$\bar{\mathcal{A}}_N \bar{\Phi} = \lambda \bar{\Phi}$$

Dimension reduction of the discretized matrix

$$\mathcal{A}_N = \prod_N^{-1} \Sigma_N \in \mathbb{R}^{(N+1)n\times(N+1)n} \quad \Rightarrow \quad \bar{\mathcal{A}}_N = \bar{\prod}_N^{-1} \bar{\Sigma}_N \mathbb{R}^{(n+Nn_2)\times(n+Nn_2)}$$

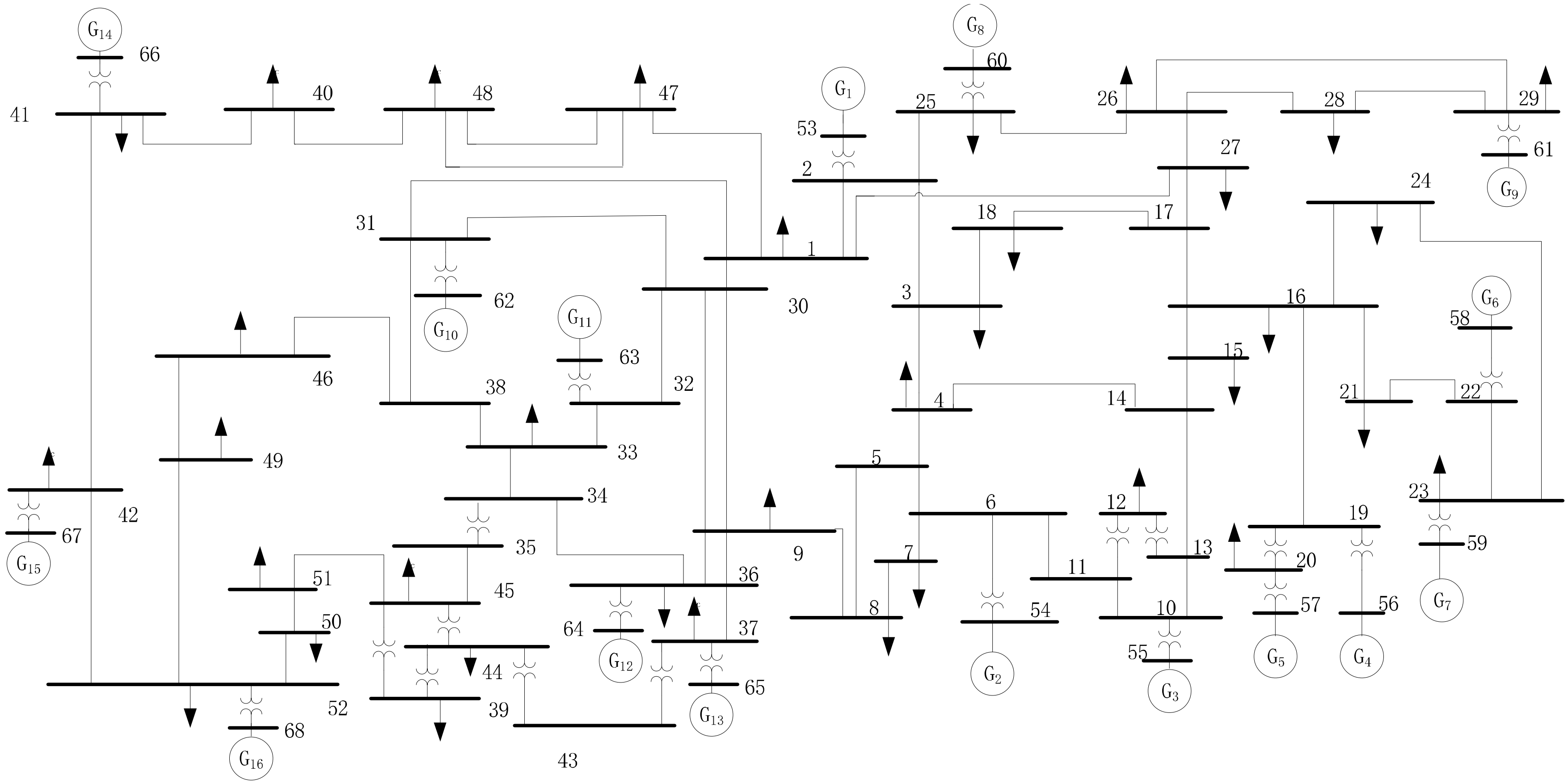




$$R_{\dim} = \frac{\dim(\bar{\mathcal{A}}_N)}{\dim(\mathcal{A}_N)} = \frac{n + Nn_2}{(N+1)n} \approx \frac{1}{N+1}$$

Since $n_1 >> n_2$, by applying the partial spectral discretization, the dimension of the matrix $\bar{\mathcal{A}}_N$ approximating $\mathcal{A}$ has been to $1/(N+1)$ times of $\mathcal{A}_N$.

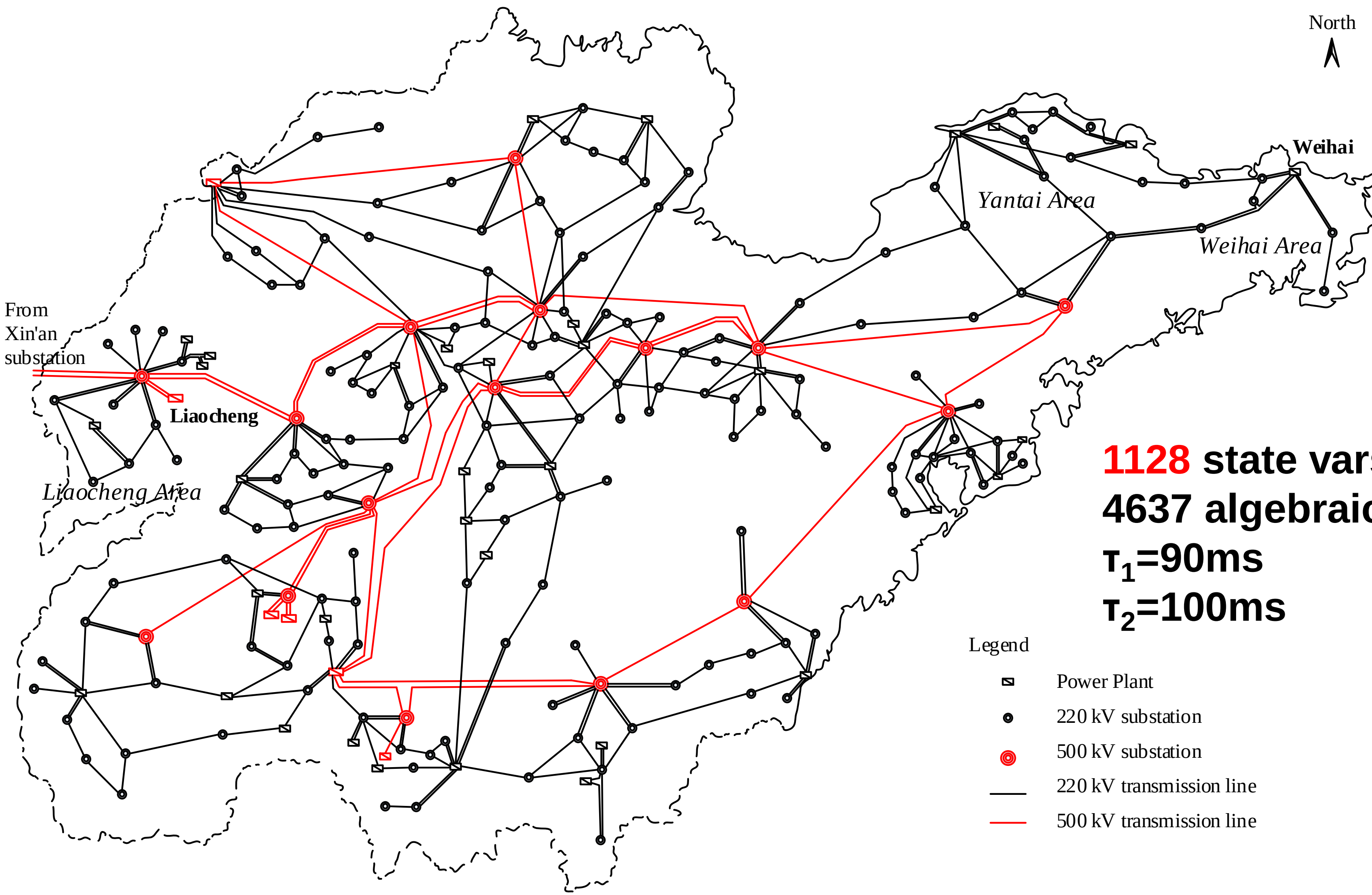
The 16-machine 68-bus test system



200 state variables **τ_{f1}=150 ms， τ_{c1}=90 ms**

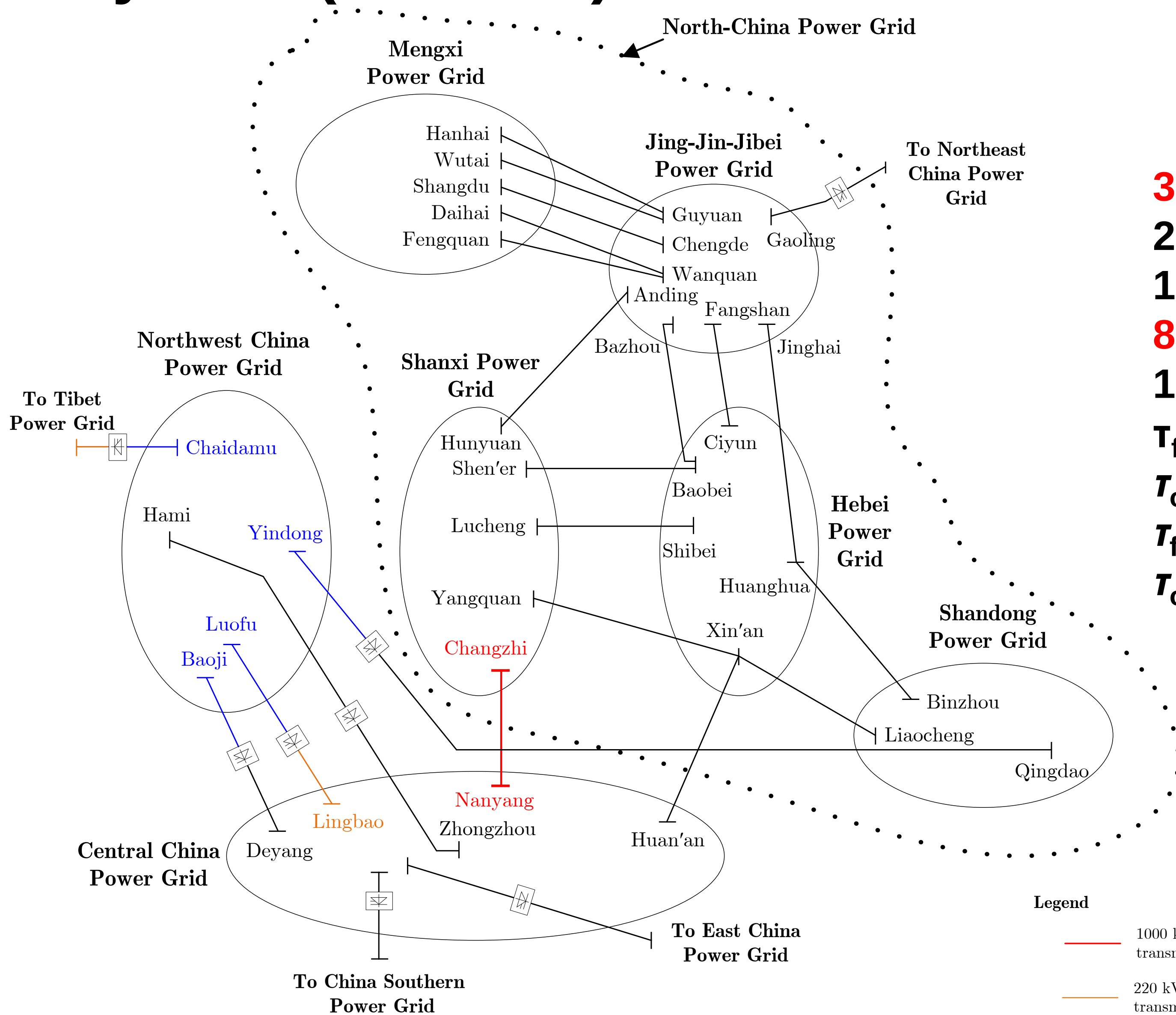
448 algebraic variables **τ_{f2}=70 ms， τ_{c2}=40 ms**

Shandong power grid (n=1128)

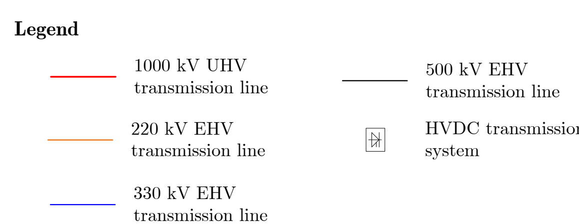


1128 state vars.
4637 algebraic vars.
τ_1=90ms
τ_2=100ms

The North China-Central China interconnected system (n=33028)



33028 buses
2405 generators
16 HVDC links
80577 state vars.
162718 algebraic vars.
τ_{f1}=120 ms
τ_{o1}= 100 ms
τ_{f2}=100 ms
τ_{o2}=80 ms



Numerical tests under constant delays

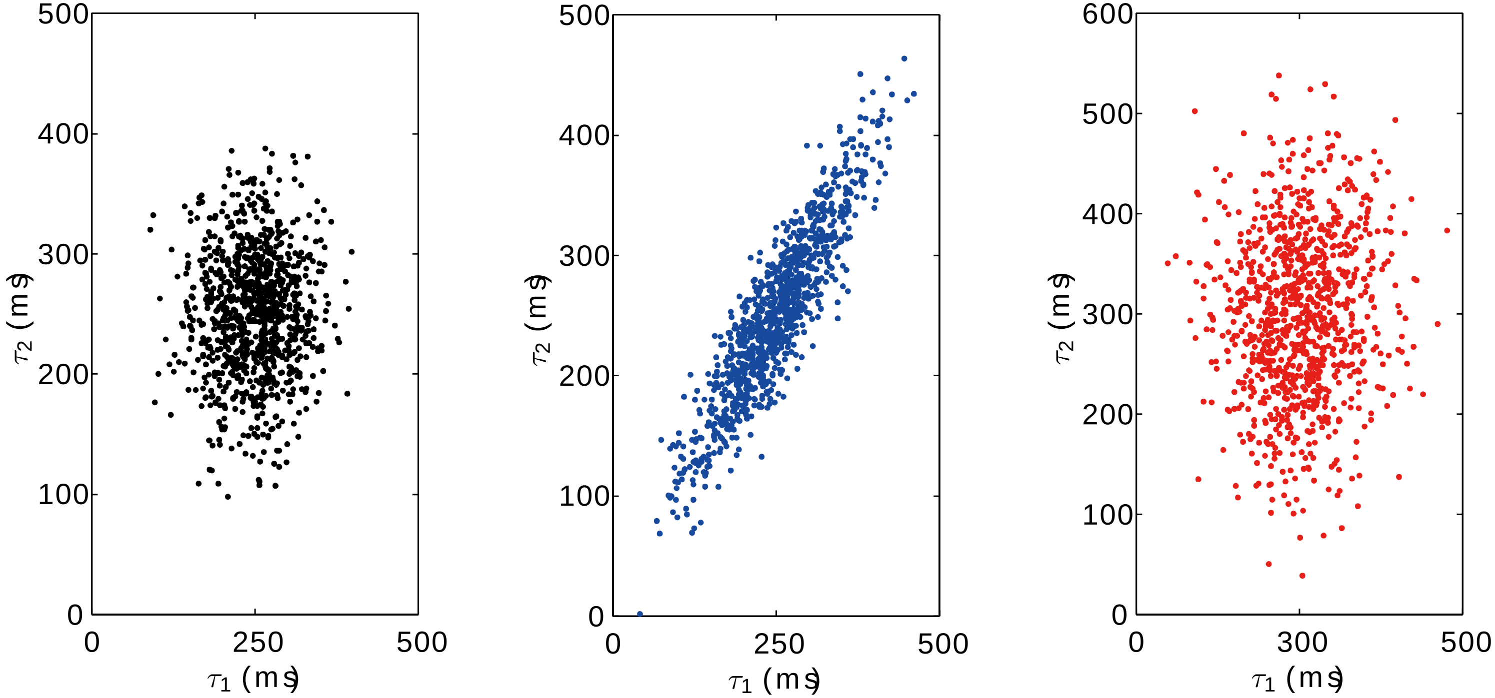
Test Syst.	*s*	*N*	*r*	EIGD		PEIGD		Speed up	w/o delay	
				Dim($\mathcal{A}_N$)	N_{IRA} / CPU Time (s)	Dim($\bar{\mathcal{A}}_N$)	N_{IRA} / CPU Time (s)		Dim	N_{IRA} / CPU Time (s)
System I (n=200, n_2=6)	7j	10	20	2200	9 / 0.33	260	**7 / 0.15**	1.71	200	**10 / 0.13**
	13j	10	20	2200	15 / 0.50	260	**13 / 0.27**	1.60	200	**17 / 0.19**
	7j	20	20	4200	9 / 0.74	320	**7 / 0.19**	3.03	200	**10 / 0.13**
	13j	20	20	4200	15 / 1.24	320	**13 / 0.35**	3.07	200	**17 / 0.19**
	7j	20	50	4200	9 / 4.32	320	**9 / 1.00**	4.32	200	**9 / 0.48**
	13j	20	50	4200	10/5.25	320	**10 / 1.19**	4.41	200	**11 / 0.79**
System II (n=1128, n_2=6)	7j	10	20	12408	5 / 1.39	1188	**5 / 0.19**	7.32	1128	**6 / 0.13**
	13j	10	20	12408	6 / 2.16	1188	**6 / 0.29**	7.45	1128	**8 / 0.21**
	7j	20	20	23688	5 / 2.55	1248	**5 / 0.22**	11.59	1128	**6 / 0.13**
	13j	20	20	23688	7 / 4.24	1248	**7 / 0.36**	11.78	1128	**8 / 0.21**
	7j	40	50	23688	7 / 17.32	1248	**7 / 1.11**	15.60	1128	**7 / 0.79**
	13j	40	50	23688	4 / 12.50	1248	**4 / 0.75**	15.67	1128	**4 / 0.50**
System III (n=80577, n_2=6)	7j	10	20	886347	6 / 203.10	80637	**6 / 23.78**	8.54	80577	**7 / 20.24**
	7j	20	20	1692117	6 / 381.47	80697	**6 / 23.82**	16.01	80577	**7 / 20.24**
	7j	20	50	1692117	5 / 1280.47	80697	**5 / 68.41**	18.72	80577	**6 / 66.44**

The dimension in PEIGD method has been reduced to about 1/(N+1) of that in EIGD method.

The speed up increases with the size of the power system.

CPU time for PEIGD is almost the same as the case without delay.

Numerical tests under random delays



(a) System I

$\tau_1, \tau_2 \sim \mathcal{N}$ (250, 70) ms

$\rho(\tau_1, \tau_2)$ =0

(b) System II

$\tau_1, \tau_2 \sim \mathcal{N}$ (250, 70) ms

$\rho(\tau_1, \tau_2)$ =0.9

(c) System III

$\tau_1, \tau_2 \sim \mathcal{N}$ (300, 80) ms

$\rho(\tau_1, \tau_2)$ =0

Figure. Distributions of two time delays in 1000 trials.

Table. Statistics of CPU time for one IRA iteration (N=20, s=7j, r=20)

Methods	CPU time (s)	System I	System II	System III
PEIGD	Max.	0.27	0.32	29.02
	Min.	0.17	0.19	22.07
	Avg.	**0.21**	**0.24**	**25.00**
EIGD	Max.	0.87	3.09	460.76
	Min.	0.55	2.22	330.93
	Avg.	**0.68**	**2.66**	**383.10**

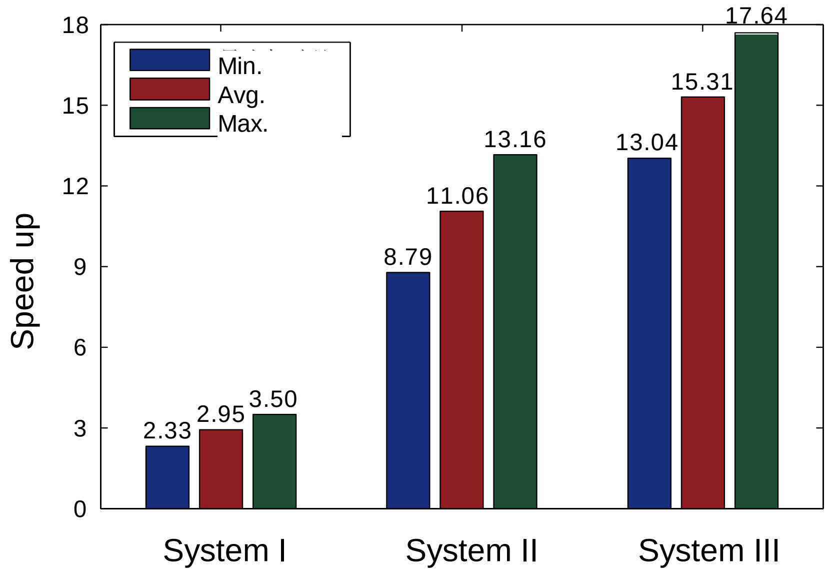


Figure. Histogram of speed up for 1000 trails

Compared with the EIGD method, the proposed PEIGD method can save 88%~94% CPU time for one implicitly restarted Arnoldi (IRA) iteration.

Outline

1. Introduction
2. Spectral discretization-based eigen-analysis
3. Partial spectral discretization-based eigenvalue computation method
4. ***Non-smooth optimization-based WADC tuning method***
5. Conclusions

4. Non-smooth optimization-based WADC tuning method

- Motivation
  - The existing method aims at maximizing the least damped oscillation mode rather than the targeted inter-area oscillation modes. The obtained WADC's parameters are absolutely not optimal.
  - To cope with the non-differentiable problem, intelligent optimization algorithms, such as PSO, are popular to pursuit the optimal solution, which are prone to fall into local extreme values and characterized by low computational efficiency and reliability, etc.

- Gradient sampling-based method
  - The objective function is determined as maximizing the least damping of the targeted inter-area oscillation modes associated with the WADC to be tuned.
  - The mathematical optimization methods instead of intelligent optimization algorithms are utilized to derive the optimal solutions, where the gradient at non-differentiable points are obtained by gradient sampling technique.

Objective function

The objective is to maximize the least damping of the targeted inter-area oscillation modes associated with the WADCs to be tuned.

$$\max_{p} \zeta_{\mathrm{I}}$$

$$\begin{aligned}
\text{s.t.}\ \ & \max\left(\mathrm{Re}(\lambda)\right) \le -\alpha,\ \ \lambda \in \mathbb{M}_{\mathrm{I}} \\
& \max\left(\zeta(\lambda)\right) \le \zeta_0^i,\ \ \lambda \in \mathbb{M}_{\mathrm{R}}^i, i = 1, \cdots, n_{\mathrm{p}} \\
& \max\left(\mathrm{Re}(\lambda)\right) \le -\alpha_0^i,\ \ \lambda \in \mathbb{M}_{\mathrm{R}}^i, i = 1, \cdots, n_{\mathrm{p}} \\
& K_{\mathrm{s}}^{\min} \le K_{\mathrm{s},j} \le K_{\mathrm{s}}^{\max},\ \ \lambda \in \mathbb{M}_{\mathrm{R}}^i, i = 1, \cdots, n_{\mathrm{c}} \\
& T_i^{\min} \le T_{i,j} \le T_i^{\max},\ \ \lambda \in \mathbb{M}_{\mathrm{R}}^i, i = 1, 2, 3, 4
\end{aligned}$$

Constraint on small signal stability

Upper/lower constraints on WADC's prameters

where,

$\zeta_{\mathrm{I}} := \min\{\zeta(\lambda) \mid \lambda \in \mathbb{M}_{\mathrm{I}}\}$ denotes the least damping ratio of targeted inter-area modes;

$\mathbb{M}_{\mathrm{I}}$ denotes the set of targeted inter-area oscillation modes under n_{p} operating conditions;

$\mathbb{M}_{\mathrm{R}}$ denotes the set of the remaining oscillation modes under n_{p} operating conditions.

Avoidance of “mode masking” problem

+: local oscillation mode   □: inter-area oscillation mode

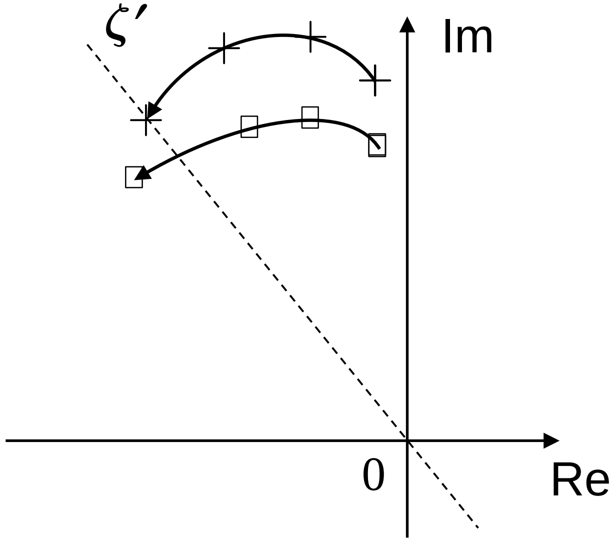


Figure (a) Illustration of optimal tuning of WADCs aiming at maximizing the smallest damping ratio among all oscillation modes.

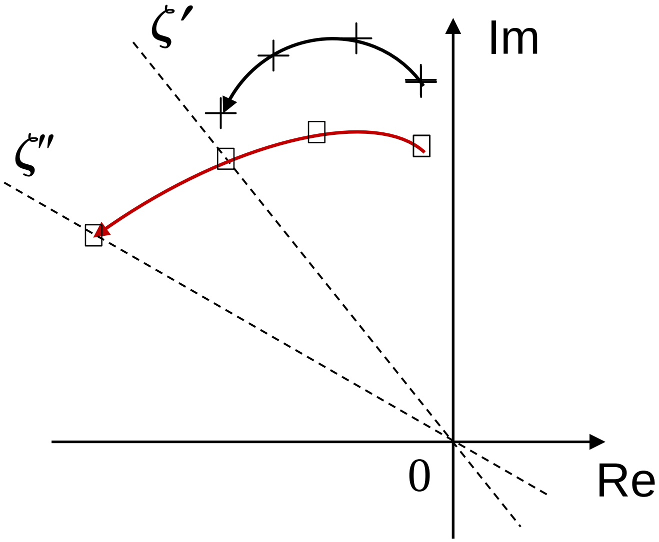


Figure (b) Illustration of optimal tuning of WADCs aiming at maximizing the smallest damping ratio among the target inter-area oscillation modes.

Tracking the targeted inter-area oscillation modes

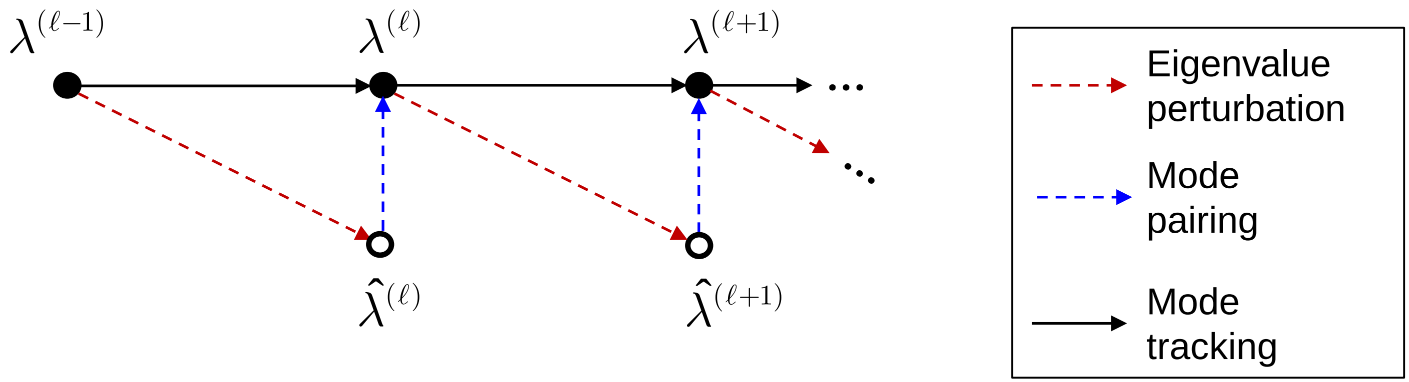


Figure. Principle of perturbation-based mode tracking

Let $\boldsymbol{p}^{(\ell)}$ be the parameters of WADCs and $\lambda^{(\ell)}$ be one mode in $\mathbb{M}_\mathrm{I}$ after the (ℓ-1)th parameter adjustment. At the next parameter tuning, $\lambda^{(\ell+1)}$ can be identified by the following three steps.

1) Estimating $\hat{\lambda}^{(\ell+1)}$: The estimation of $\lambda^{(\ell+1)}$ denoted by $\hat{\lambda}^{(\ell+1)}$ can be readily obtained by applying the matrix perturbation theory.

2) Paring $\lambda^{(\ell+1)}$ with $\hat{\lambda}^{(\ell+1)}$: It is achieved by screening the minimum Euclidean distance $\left\| \lambda^{(\ell+1)} - \hat{\lambda}^{(\ell+1)} \right\|$.

3) Determining inter-area modes in $\mathbb{M}_\mathrm{I}$: By taking $\hat{\lambda}^{(\ell+1)}$ as a bridge, $\lambda^{(\ell+1)}$ at the ℓth parameter tuning can be reliably track $\lambda^{(\ell)}$ at the last parameter tuning.

The objective function can be reformulated by using the penalty function method and by recasting the max-min problem in a dual form, which is a challenging *non-convex and non-smooth optimization problem*.

$$\max_{\boldsymbol{p}} J\left(\boldsymbol{p}\right),$$

$$\begin{aligned}
J\left(\boldsymbol{p}\right) = &-\zeta_{\mathrm{I}} + k_1 \max_{\lambda_k \in \mathbb{M}_{\mathrm{I}}} \left\{0, \mathrm{Re}\left(\lambda_k\right) + \alpha\right\} \\
&+ \sum_{i=1}^{n_{\mathrm{p}}} k_{i+1} \max_{\lambda_k \in \mathbb{M}_{\mathrm{R}}^i} \left\{0, \zeta_0^i - \zeta\left(\lambda_k\right)\right\} \\
&+ \sum_{i=n_{\mathrm{p}}+1}^{2n_{\mathrm{p}}} k_{i+1} \max_{\lambda_k \in \mathbb{M}_{\mathrm{R}}^i} \left\{0, \mathrm{Re}\left(\lambda_k\right) + \alpha^i\right\} \\
&+ \sum_{j=1}^{n_{\mathrm{c}}} \omega_j \max\left\{0, K_{\mathrm{s}}^{\min} - K_{\mathrm{s},j}, K_{\mathrm{s},j} - K_{\mathrm{s}}^{\max}\right\} \\
&+ \sum_{j=0}^{n_{\mathrm{c}}-1} \sum_{i=1}^{4} \omega_{4j+i+n_{\mathrm{c}}} \max\left\{0, T_i^{\min} - T_{i,j}, T_{i,j} - T_i^{\max}\right\}
\end{aligned}$$

Gradient sampling at non-differentiable points

- First, define *Clarke sub-differential* (i.e., generalized gradient) at a non-differentiable point $\boldsymbol{p}^{(\ell)}$:

$$\partial_{\mathrm{c}} J(\boldsymbol{p}^{(\ell)}) := \operatorname{conv}\left\{ \lim_{\boldsymbol{p}\to\boldsymbol{p}^{(\ell)}} \nabla J\left(\boldsymbol{p}\right) : \boldsymbol{p} \in \mathcal{N} \right\}$$

where "conv" denotes the convex hull and $\mathcal{N}$ is any full-measure subset of a neighborhood around $\boldsymbol{p}^{(\ell)}$ containing differentiable points.

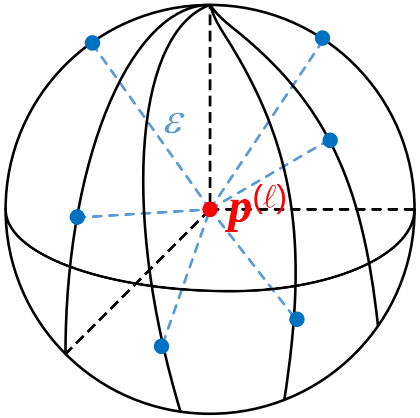


Figure. Point subset $\mathcal{N}$ around the non-differentiable point $\boldsymbol{p}^{(\ell)}$

- The *steepest descent direction* $\boldsymbol{d}^{(\ell)}$ is defined as the opposite of the vector $\boldsymbol{z}$ in the Clarke sub-differential whose norm is the smallest.

$$\boldsymbol{d}^{(\ell)} := -\arg \min_{\boldsymbol{z}\in\partial_{\mathrm{c}} J(\boldsymbol{p}^{(\ell)})} \left\| \boldsymbol{z} \right\|$$

Test results

Table. Performance comparison of optimal WADCs with different objective functions (Hz / %)

Operating condition	max{ζ_I} (The proposed method)		max{min{ζ_I , ζ_R }}	
	f_I / ζ_I	f_R / ζ_R	f_I / ζ_I	f_R / ζ_R
1	**0.48 / 13.09**	1.10 / 6.20	**0.49 / 6.34**	0.98 / 6.34
2	**0.43 / 16.95**	1.16 / 4.10	**0.46 / 8.69**	1.00 / 4.18
3	**0.48 / 13.41**	1.16 / 4.20	**0.39 / 7.52**	1.00 / 4.24

Since ζ_R is equal to or less than ζ_I, mode masking problem occurs and the optimization process is stagnated.

Table. Optimal parameters of WADC and CPU time

Operating conditions	The proposed method				PSO			
	K_s	T_1	T_3	Iterations / Time (s)	K_s	T_1	T_3	Iterations / Time (s)
1	56.77	4.00	0.69	82 / 47.64	56.85	4.00	0.69	200 / 217.23
2	100.0	4.00	0.78	43 / 57.13	100.0	4.00	0.78	200 / 220.30
3	100.0	4.00	0.75	50 / 83.63	100.0	4.00	0.75	200 / 235.82

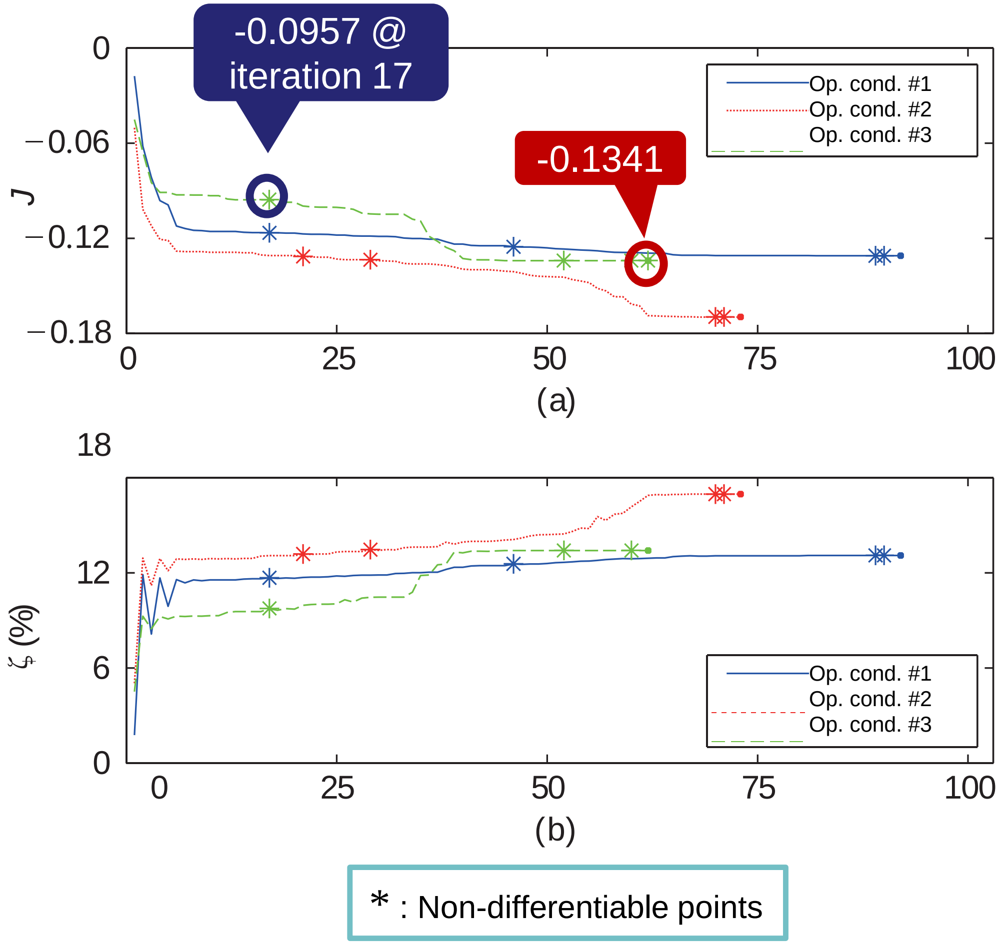


Figure. Iteration history of objective function *J* and the least damping ratio of the targeted inter-area oscillation modes ζ_I

Outline

5. Conclusion

- The spectral discretization-based eigenvalue computation method resolves a long-standing challenging problem for stability analysis of large power system with consideration of time delay impacts. A set of the rightmost or the least damped oscillation modes of the system can be efficiently computed.
- The partial spectral discretization-based method can significantly reduce the computational burden of original spectral discretization-based method by discretizing delayed state variables only. The computational efficiency can be enhanced by an order of magnitude.
- The parameters of WADC are optimally tuned by the gradient sampling-based non-smooth optimization method. Mode tracking method reliably tracks the targeted inter-area modes during the optimization process and the gradient sampling technique efficiently works out the steepest descent direction at the non-differentiable points. Since mode masking and stagnation problems are avoided and optimal solutions are obtained.

Publications

1. H. Ye, Y. Liu. Eigenvalue computation of large time-delayed power system. Beijing: **Science Press**, 2018. 365 pages.
2. Q. Mou, H. Ye, Y. Liu. Enabling highly efficient eigen-analysis of large delayed cyber-physical power systems by partial spectral discretization. ***IEEE Transactions on Power Systems***, 2019, DOI: 10.1109/TPWRS. 2019.2936488.
3. H. Ye, T. Li, Y. Liu. Time integration-based IGD methods for eigen-analysis of large delayed cyber-physical power system. ***IEEE Transactions on Power Systems**,* 2019, DOI: 10.1109/TPWRS.2019.2936871.
4. Q. Mou, H. Ye, Y. Liu. Nonsmooth optimization-based WADC tuning in large delayed cyber-physical power system by interarea modes tracking and gradient sampling. ***IEEE Transactions on Power Systems***, 2019, 34(1): 668-679.
5. Hua Ye, Kehan Liu, Qianying Mou, Yutian Liu. Modeling and formulation of DCPPS for small signal stability analysis and control. ***IEEE Transactions on Power Systems***, 2019, 34(3): 2419-2432.
6. H. Ye, Q. Mou, X. Wang, Y. Liu. Eigen-analysis of large delayed cyber-physical power system by time integration-based solution operator discretization methods. ***IEEE Transactions on Power Systems**,* 2018, 33(6): 5968-5978.

7. H. Ye, Q. Mou, Y. Liu. Calculation of critical oscillation modes for large delayed cyber-physical power system using pseudo-spectral discretization of solution operator. ***IEEE Transactions on Power Systems,*** 2017, 32(6): 4464-4476.

8. H. Ye, Q. Mou, Y. Liu. Enabling highly efficient spectral discretization-based eigen-analysis methods by Kronecker product. ***IEEE Transactions on Power Systems,*** 2017, 32(5): 4148-4150.

9. H. Ye, Y. Liu, P. Zhang. Efficient eigen-analysis for large delayed cyber-physical power system using explicit infinitesimal generator discretization. ***IEEE Transactions on Power Systems,*** 2016, 31(3): 2361-2370.

10. Q. Mou, H. Ye, Y. Liu, L. Gao. Applications of matrix perturbation theory to delayed cyber-physical power system. ***International Journal of Electrical Power & Energy Systems***, 2019, 107(1): 507-515.

11. H. Ye, W. Gao, Q. Mou, Y. Liu. Iterative infinitesimal generator discretization-based method for eigen-analysis of large delayed cyber-physical power system***. Electric Power Systems Research***, 2017, 143(1): 389-399.

Thanks for your attention !