Power System Simulation: From Numerical Methods to Semi-Analytical Methods

Kai Sun

Department of EECS

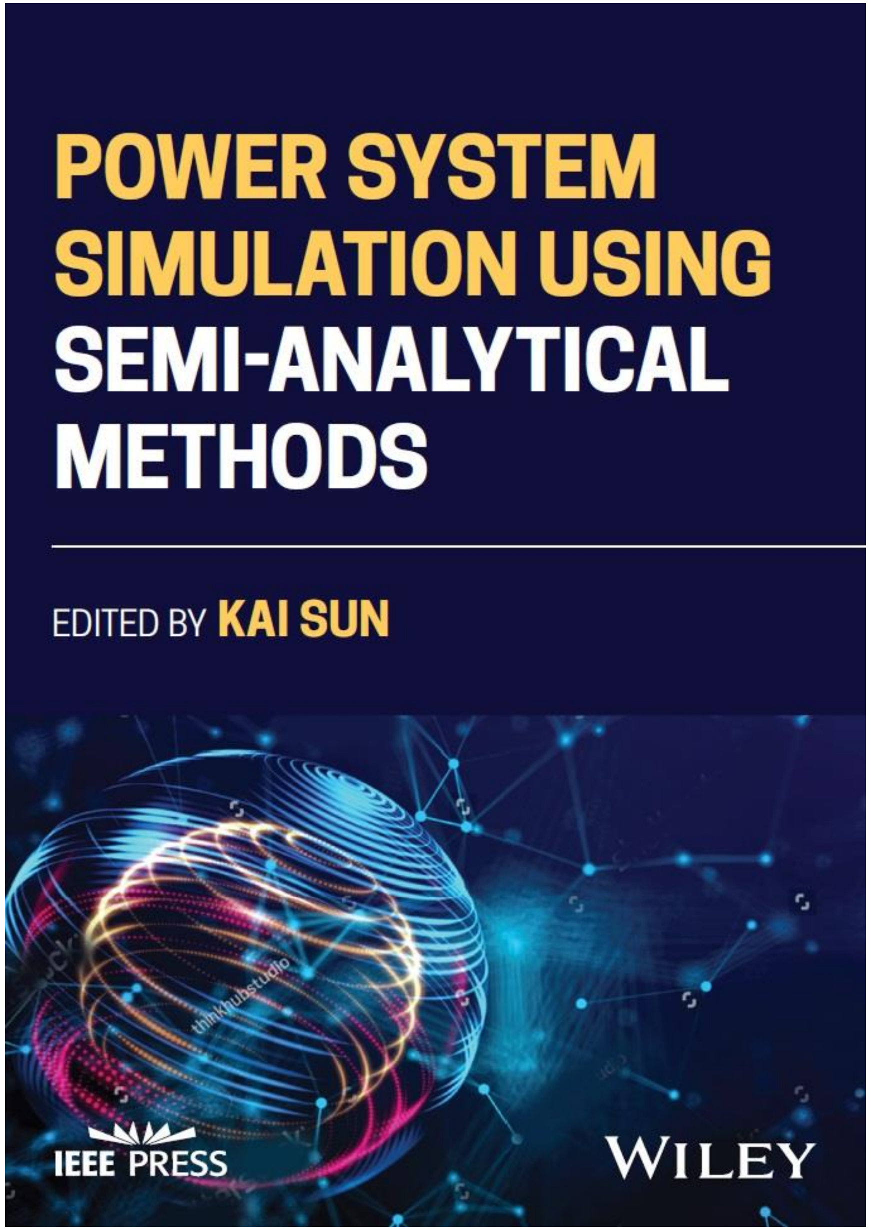
University of Tennessee, Knoxville

March 1st, 2024

kaisun@utk.edu

Content

- Background on power system simulation
- Ideas of semi-analytical solutions
- Introduction of semi-analytical methods and applications for power systems

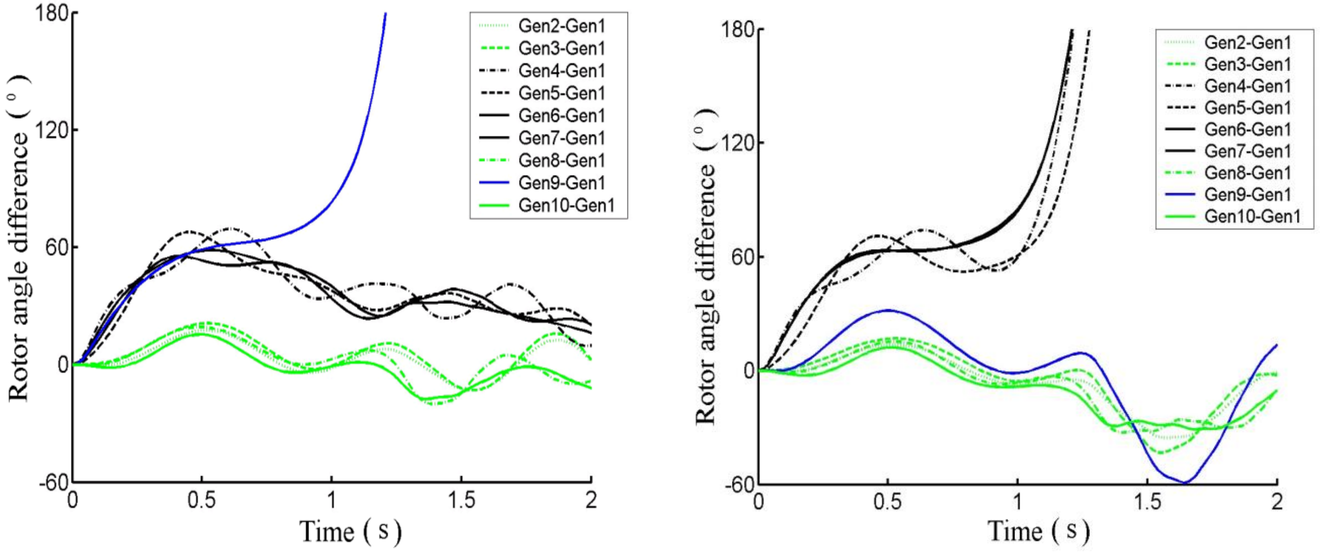


ISBN:978-1-119-98801-4

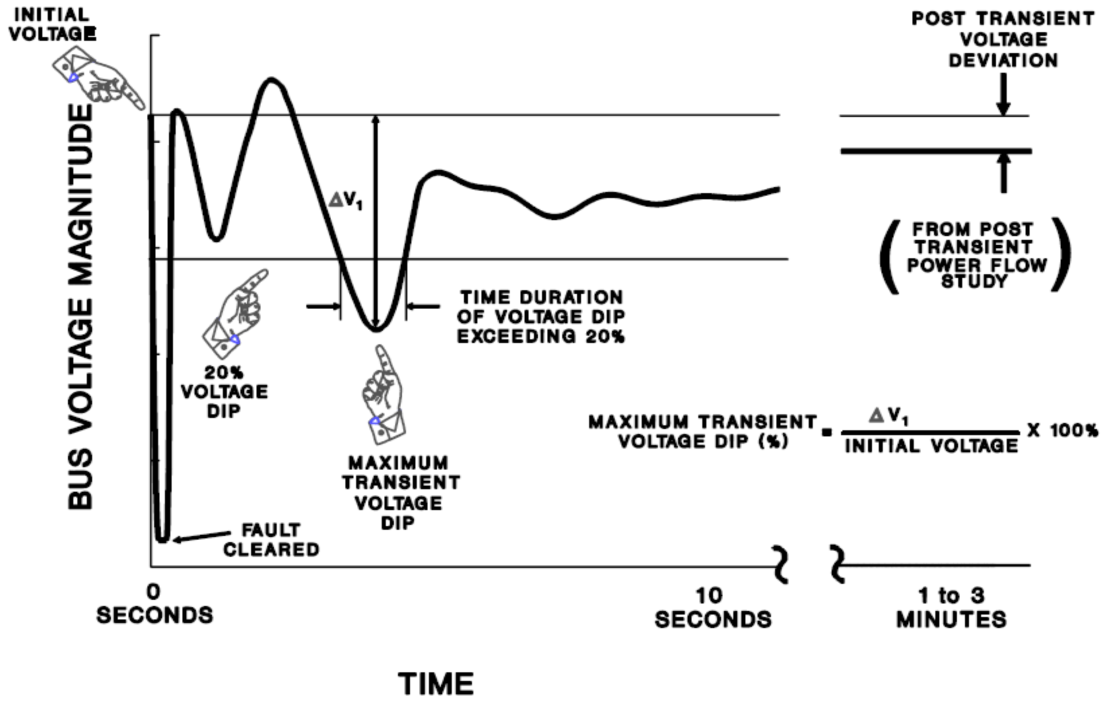
Why dynamic simulation

- In designing and operating a power system, its dynamic performance subjected to disturbances such as condition changes and contingencies needs to be assessed.
- It is important that when the changes are completed, the system safely settles to a new operating condition.
- In other words, not only should the new operating condition be acceptable (as revealed by steady-state analysis) but also the system must survive the transition to the new condition without violating any constraint or reliability criteria. This requires dynamic simulation.

TRANSIENT STABILITY OF GENERATORS



VOLTAGE PERFORMANCE PARAMETERS



Industrial Practices in Power System Simulation

- **Bulk power system model**:

  Differential-Algebraic-Equation (DAE) model:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{V}) \\ 0 = \mathbf{g}(\mathbf{x}, \mathbf{V}) = \mathbf{I}(\mathbf{x}, \mathbf{V}) - \mathbf{Y}_N \mathbf{V} \end{cases}$$

- **Simulation methods:**
  - Numerical integration methods such as Euler, Runge-Kutta and Trapezoidal-rule methods.
  - Linear or low-order approximation of nonlinear functions **f** and **g**
  - Slow due to small stepsizes (<1ms) for numerical stability or accuracy on large system models.

- **Industry practices**
  - For one contingency, commercial software typically requires 1-5 minutes to simulate 1 second of a detailed grid model such as a 70,000-bus Eastern Interconnection model (5,000-10,000 generators and 100,000 state variables).
  - Online simulations are performed on 1,000-3,000 critical contingencies every 10-15 min on a reduced model (~10,000 buses and 2,000 generators).

A partitioned (alternating) scheme solving power system DAEs

$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{V})$ DE

$\mathbf{I}(\mathbf{x}, \mathbf{V}) = \mathbf{Y}_N \mathbf{V}$ AE

- At each time step t_n:

Runge-Kutta method

$$\mathbf{I}(\mathbf{x}_n, \mathbf{V}_n) - \mathbf{Y}_N \mathbf{V}_n = 0$$

$\mathbf{x}_n$ Alternating $\mathbf{V}_n$

$$\dot{\mathbf{x}}_n = \mathbf{f}(\mathbf{x}_n, \mathbf{V}_n)$$

Newton-Raphson method

I_i

Generator DEs:

$$\frac{d\delta}{dt} = \Delta\omega_r$$

$$\frac{2H}{\omega_0}\frac{d\Delta\omega_r}{dt} = P_m - P_e - K_D\frac{\Delta\omega_r}{\omega_0}$$

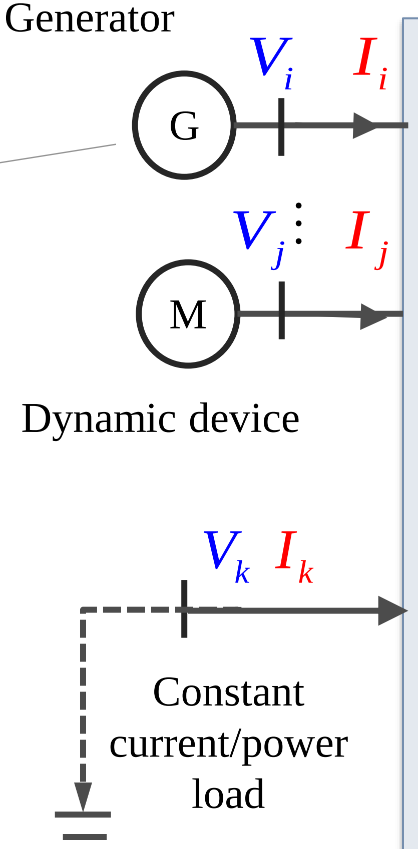
Generator AEs:

$$E_t = e^{-j(\delta-\pi/2)}V_i$$

$$E_t = E' - (R_a + jX'_d)I_t$$

$$P_e = \mathrm{Re}(E_t I_t^*)$$

$$I_i = e^{j(\delta-\pi/2)}I_t$$



Network AEs:

$$I_i = \frac{P_i - jQ_i}{V_i^*} = \sum_{j=1}^{N} Y_{ij}V_j$$

V_i

Numerical Integration Methods

- Solving an initial value problem starting from $\mathbf{x}=\mathbf{x_0}$ and $t=t_0$

$$\frac{d\mathbf{x}}{dt} = f(\mathbf{x},t) \quad \rightarrow \quad \Delta\mathbf{x} \approx f(\mathbf{x},t)\Delta t$$

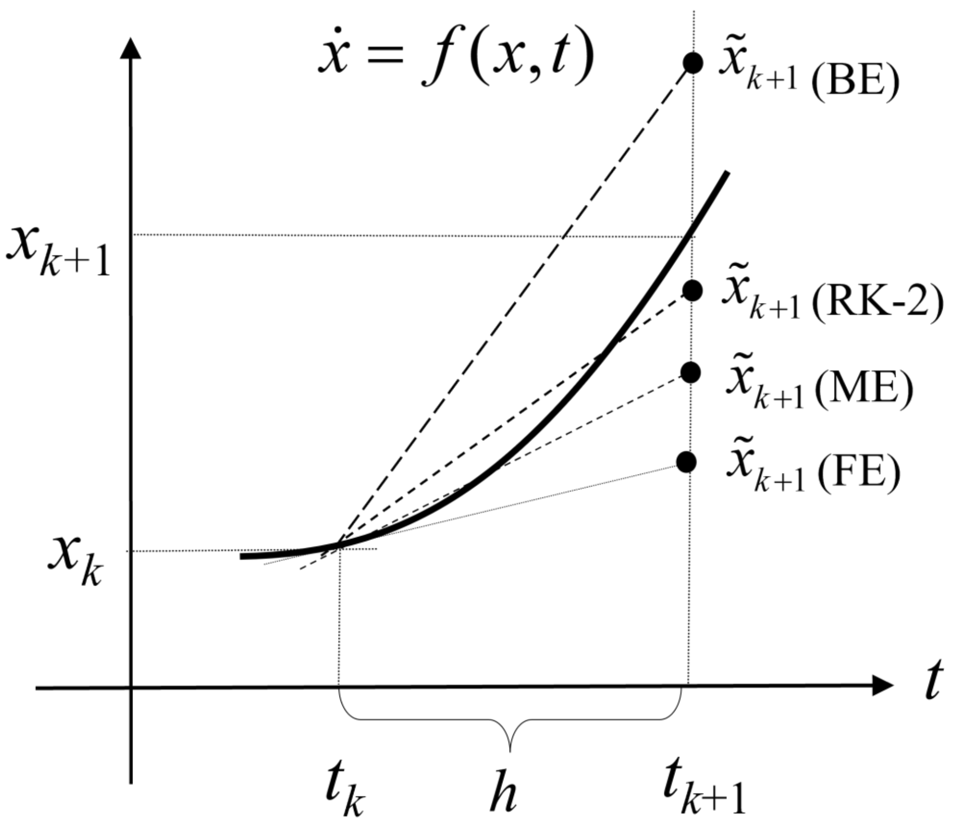
- **Explicit Methods**
  - **x** is computed using only its past values, e.g. Forward Euler and R-K methods

    Forward Euler method: $\mathbf{x}_{i+1} = \mathbf{x}_i + f(\mathbf{x}_i, t_i)\Delta t$

- **Implicit Methods**
  - **x** is computed also involving its future values, e.g. the Backward Euler and Trapezoidal-rule methods

    Backward Euler method: $\mathbf{x}_{i+1} = \mathbf{x}_i + f(\mathbf{x}_{i+1}, t_{i+1})\Delta t$

- For stiff systems with large $|\lambda_{max}/\lambda_{min}|$, if Δt is too large the explicit methods have poor numerical stability while implicit methods have low accuracy



Comparison of Explicit and Implicit Methods

When focusing on the fastest dynamics:

$$\dot{x} = f(x,t) \approx \lambda_{\max} x$$

Forward Euler Method (explicit)

$$x_i = x_{i-1} + f(x_{i-1}, t_{i-1})\Delta t$$

$$\approx x_{i-1} + \lambda_{\max} x_{i-1} \Delta t$$

$$= x_{i-1}(1 + \lambda_{\max}\Delta t)$$

$$x_i = x_0 (1 + \lambda_{\max}\Delta t)^i$$

The method is numerically stabile if

$$|1 + \lambda_{\max}\Delta t| < 1$$

$\Rightarrow \lambda_{\max}$ has a negative real part and

$$\Delta t < \frac{2}{|\lambda_{\max}|}$$

Backward Euler Method (implicit)

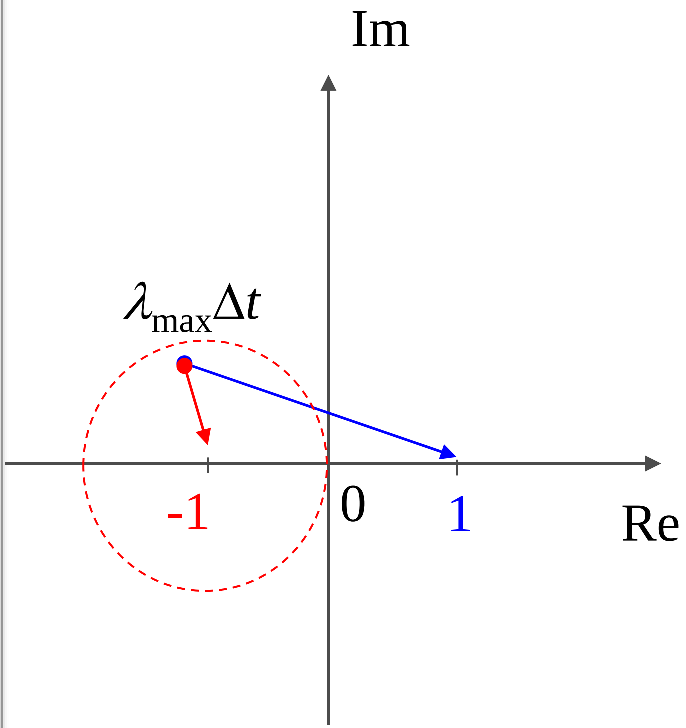
$$x_i = x_{i-1} + f(x_i, t_i)\Delta t$$

$$\approx x_{i-1} + \lambda_{\max} x_i \Delta t$$

$$x_i = x_{i-1} \frac{1}{1 - \lambda_{\max}\Delta t}$$

$$x_i = x_0 \left(\frac{1}{1 - \lambda_{\max}\Delta t} \right)^i$$

Δt can be arbitrarily large as long as $\lambda_{\max}$ has a negative real part (this method has **A-Stability**)



Finding an Analytical Solution of Nonlinear Differential Equations

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t))$$

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) & x_2(t) & \dots & x_M(t) \end{bmatrix}^T$$

$$\mathbf{f}(\cdot) = \begin{bmatrix} f_1(\cdot) & f_2(\cdot) & \dots & f_M(\cdot) \end{bmatrix}^T$$

$$\mathscr{L}[\mathbf{x}] = \frac{\mathbf{x}(0)}{s} + \frac{\mathscr{L}[\mathbf{f}(\mathbf{x})]}{s}$$

Adomian Polynomials

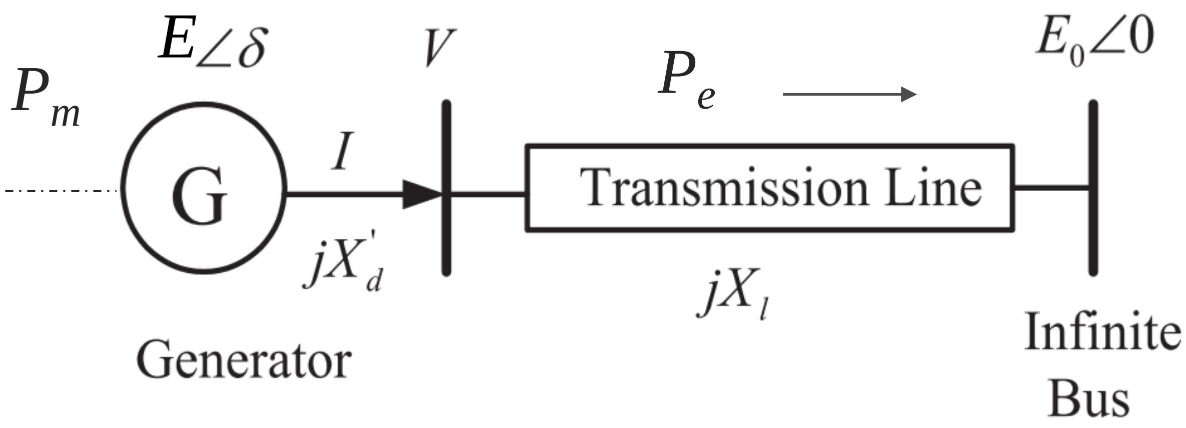
$$f_i(\mathbf{x}) = \sum_{n=0}^{\infty} A_{i,n} = \sum_{n=0}^{\infty} \frac{1}{n!} \left[\frac{\partial^n}{\partial \lambda^n} f_i\left(\sum_{i=0}^{n} \mathbf{x}_i \lambda^i \right) \right] \Bigg|_{\lambda=0}$$

$$\mathbf{x}(t) = \sum_{n=0}^{\infty} \mathbf{x}_n(t),$$

$$\mathscr{L}[\mathbf{x}_0] = \mathbf{x}(0)/s \quad \mathscr{L}[\mathbf{x}_{n+1}] = \mathscr{L}[\mathbf{A}_n]/s \qquad n \ge 0$$

$$\mathbf{x}(t) = \mathbf{x}_0(t) + \mathbf{x}_1(t) + \mathbf{x}_2(t) + \dots$$

Solving an SMIB System



$$\begin{cases} 2H\cdot\dot{\omega} = P_m - D\cdot(\omega-\omega_0) - P_e \\ \dot{\delta} = \omega \\ 0 = P_e - P_{\max}\sin\delta \end{cases}$$

Assume $\delta(0)=\alpha$ and $\omega(0)=\dot{\delta}(0)=\beta$ and solve: $\delta_{SAS}^{<3>}(t) = \delta_0(t)+\delta_1(t)+\delta_2(t)$

$$\delta_0 = \beta t + \alpha$$

$$\delta_1 = \frac{-D\beta + P_m}{4H}t^2 - \frac{P_{\max}\cos\alpha}{2H\beta}t + \frac{P_{\max}[-\sin\alpha+\sin(\beta t+\alpha)]}{2H\beta^2}$$

$$\delta_2 = \frac{D^2\beta - DP_m}{24H^2}t^3 + \frac{D\beta(\cos\alpha - \cos(\beta t+\alpha)) + P_m\cos(\beta t+\alpha)}{8H^2\beta^2}P_{\max}t^2$$
$$+\frac{P_{\max}[8D\beta(\sin\alpha+\sin(\beta t+\alpha)) - 4P_m(\sin\alpha+2\sin(\beta t+\alpha)) - P_{\max}(2\cos(\beta t+2\alpha)+2\cos\beta t+\cos 2\alpha+4)]}{16H^2b^3}t$$
$$-\frac{P_{\max}[32D\beta(\cos\alpha-\cos(\beta t+\alpha)) - 24P_m(\cos\alpha-2\cos(\beta t+\alpha)) - P_{\max}(4\sin(\beta t+2\alpha)+\sin 2(\beta t+\alpha)+12\sin\beta t-5\sin 2\alpha)]}{32H^2\beta^4}$$

Applicable to any operating condition and any disturbance!

Analytical Expansion

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}), \qquad t_0 = 0, \quad \mathbf{x}(0) = \mathbf{a}_0 \quad \xrightarrow{\text{Find}} \quad \mathbf{x}_{SAS}^{<K>}(t) \overset{\text{def}}{=} \mathbf{x}_0(t) + \mathbf{x}_1(t) + \mathbf{x}_2(t) + \cdots + \mathbf{x}_K(t)$$

$$\text{Assume} \quad \mathbf{x}(t) = \mathbf{a}_0 + \mathbf{a}_1 t + \mathbf{a}_2 t^2 + \ldots = \sum_{k=0}^{\infty} \mathbf{a}_k t^k \quad \Rightarrow \quad \frac{d\mathbf{x}(t)}{dt} = 0 + \mathbf{a}_1 + 2\mathbf{a}_2 t + 3\mathbf{a}_3 t^2 + \ldots = \sum_{k=0}^{\infty} (k+1)\mathbf{a}_{k+1} t^k$$

$$\mathbf{f}(\mathbf{x}(t)) = \mathbf{f}\left(\sum_{k=0}^{\infty} \mathbf{a}_k t^k\right) = \sum_{k=0}^{\infty} \mathbf{b}_k t^k$$

where $\mathbf{b}_0 = \mathbf{f}(\mathbf{a}_0)$,

$$\mathbf{b}_1 = \frac{1}{1!} \mathbf{J}_{\mathbf{f}}(\mathbf{a}_0)\mathbf{a}_1$$

$$\mathbf{b}_2 = \frac{1}{2!}\left(\begin{bmatrix} \vdots \\ \mathbf{a}_1^T \mathbf{H}_{f_i}(\mathbf{a}_0)\mathbf{a}_1 \\ \vdots \end{bmatrix} + 2\mathbf{J}_{\mathbf{f}}(\mathbf{a}_0)\mathbf{a}_2\right)$$

$\ldots$

$$0 + \mathbf{a}_1 + 2\mathbf{a}_2 t + 3\mathbf{a}_3 t^2 + \ldots = \mathbf{b}_0 + \mathbf{b}_1 t + \mathbf{b}_2 t^2 + \ldots$$

$$\mathbf{a}_1 = \mathbf{b}_0(\mathbf{a}_0)$$

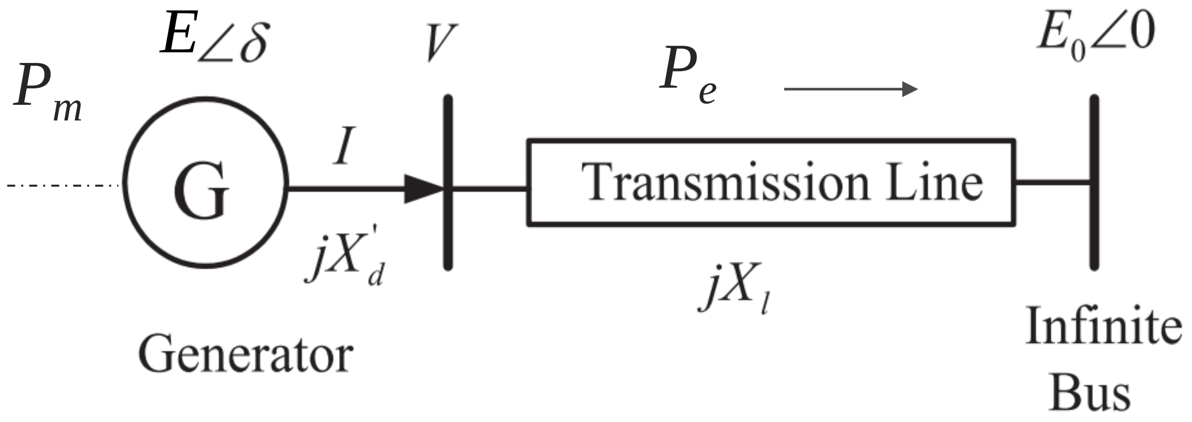
$$\mathbf{a}_2 = \frac{1}{2}\mathbf{b}_1(\mathbf{a}_0, \mathbf{a}_1)$$

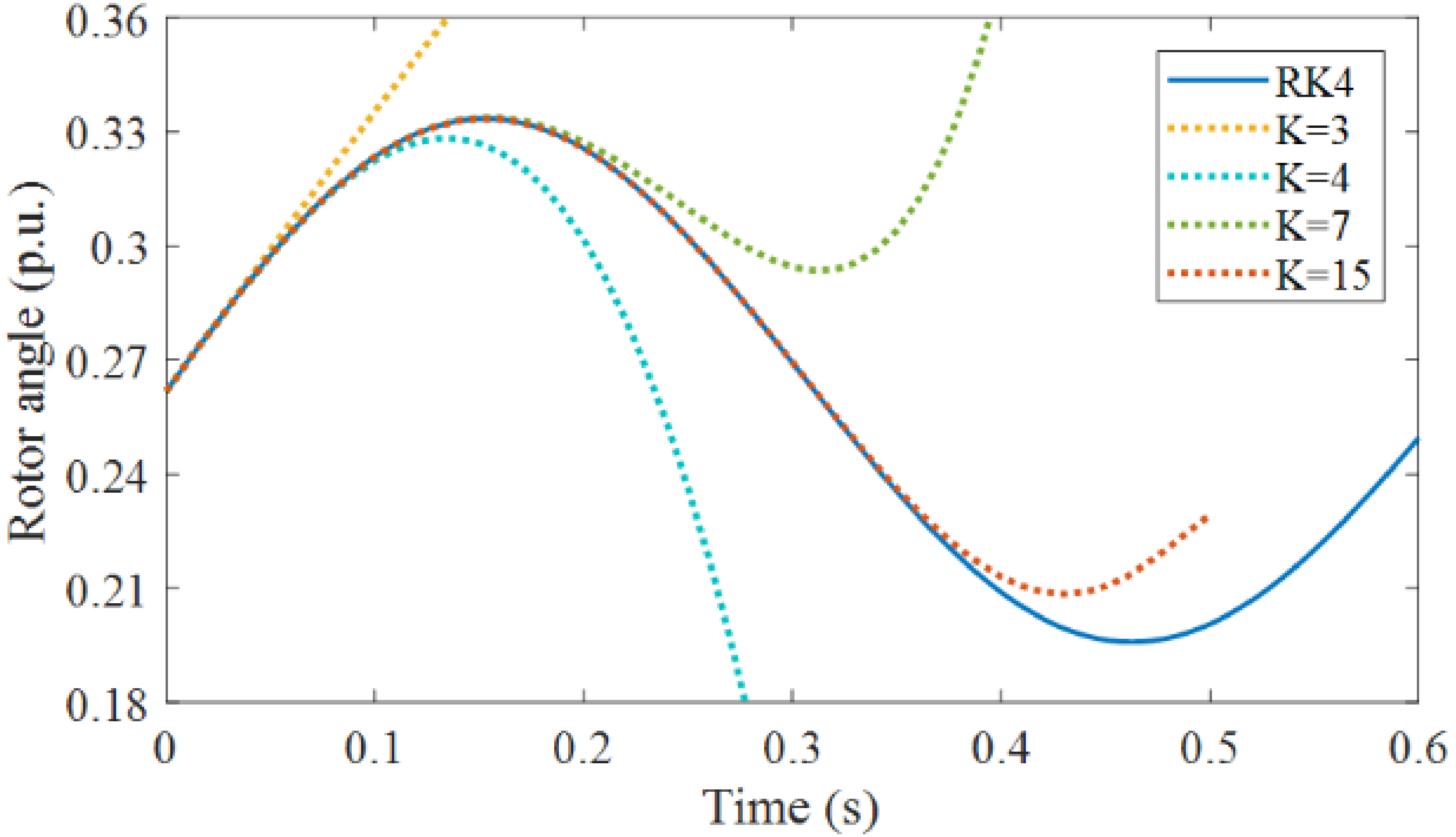
$$\Rightarrow \quad \mathbf{a}_3 = \frac{1}{3}\mathbf{b}_2(\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2)$$

$$\cdots$$

$$\mathbf{a}_K = \frac{1}{K}\mathbf{b}_{K-1}(\mathbf{a}_0, \mathbf{a}_1, \ldots, \mathbf{a}_{K-1})$$

Accuracy vs Order K





A semi-analytical approach for time-domain simulation

- A semi-analytical solution (SAS) is an approximate but analytical solution of the DAE model.

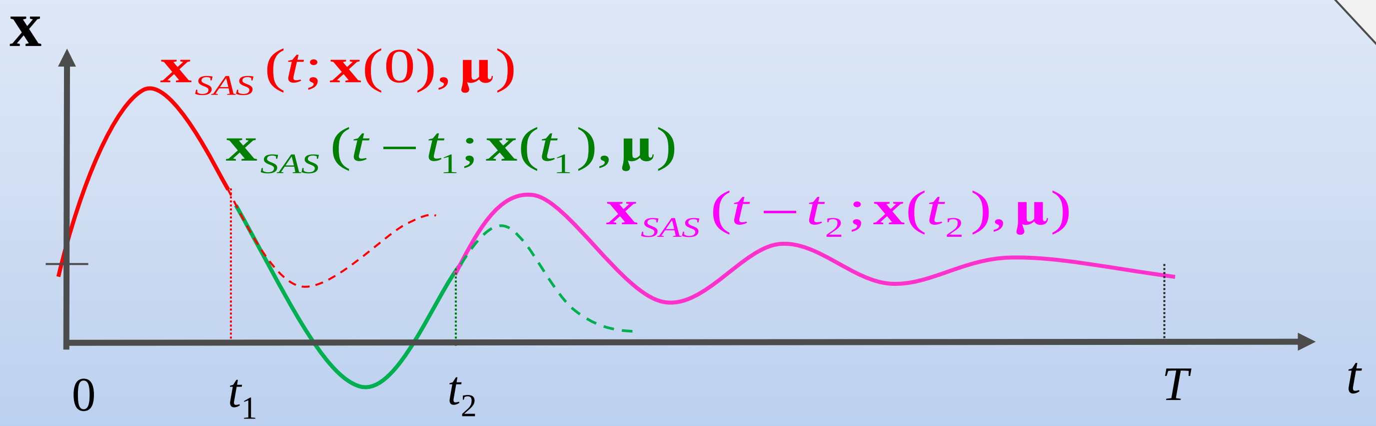
Stage 1 (offline): Deriving the SAS with symbolic variables on time t, initial state $\mathbf{x}_{ini}$ and selected parameters $\boldsymbol{\mu}$

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{y}, t) \\ 0 = \mathbf{g}(\mathbf{x}, \mathbf{y}, t) \end{cases}$$

$$\mathbf{x}(t; \mathbf{x}_{ini}, \boldsymbol{\mu}) = \sum_{k=0}^{\infty} \mathbf{x}_k(t; \mathbf{x}_{ini}, \boldsymbol{\mu})$$

$$\approx \sum_{k=0}^{K} \mathbf{x}_k(t; \mathbf{x}_{ini}, \boldsymbol{\mu}) \overset{\text{def}}{=} \mathbf{x}_{SAS}^{<K>}(t; \mathbf{x}_{ini}, \boldsymbol{\mu})$$

Stage 2 (online): Evaluating the SAS over consecutive time intervals until finishing the simulation period.



Forms and Mathematical Tools for Semi-analytical Solutions

$$\begin{cases} \dot{x} = f(x, v) \\ 0 = g(x, v) \end{cases} \qquad x(0) = x_0$$

$$x_{SAS}(x_0, \mu, t) = \begin{cases} \sum_{n=0}^{N} x_n(x_0, \mu, t) & [1] \\ \sum_{n=0}^{N} a_n(x_0, \mu) \times t^n & [1][3]\text{-}[5]\ [7]\text{-}[11] \\ \dfrac{\sum_{m=0}^{M} b_m(x_0, \mu) \times t^m}{\sum_{k=0}^{K} c_k(x_0, \mu) \times t^k} & [6] \\ x_0 + \cfrac{d_1(x_0, \mu) \times t}{1 + \cfrac{d_2(x_0, \mu) \times t}{1 + \cfrac{\ddots}{1 + d_L(x_0, \mu) \times t}}} & [2] \end{cases}$$

Related works:

[1] N. Duan, K. Sun, "Power System Simulation Using the Multi-stage **Adomian Decomposition**," *IEEE Trans. Power Systems*, 2017

[2] C. Liu, B. Wang, K. Sun, "Fast Power System Dynamic Simulation Using **Continued Fractions**," *IEEE Access*, 2018

[3] B. Wang, N. Duan, K. Sun, "A **Time-Power Series** Based Semi-Analytical Approach for Power System Simulation," *IEEE Trans. Power Systems*, 2019

[4] Y. Liu, K. Sun, R. Yao, B. Wang, "Power System Time Domain Simulation Using a **Differential Transformation** Method," *IEEE Trans. Power Systems*, 2019

[5] Y. Liu, K. Sun, "Solving Power System Differential Algebraic Equations Using **Differential Transformation**," *IEEE Trans. Power Systems*, 2020

[6] R. Yao, K. Sun, F. Qiu, "Vectorized Efficient Computation of **Pade Approximation** for Semi-Analytical Simulation of Large-Scale Power Systems," *IEEE Trans. Power Systems*, 2019

[7] R. Yao, K. Sun, et al, "Voltage Stability Analysis of Power Systems with Induction Motors Based on **Holomorphic Embedding**," *IEEE Trans. Power Systems*, 2019

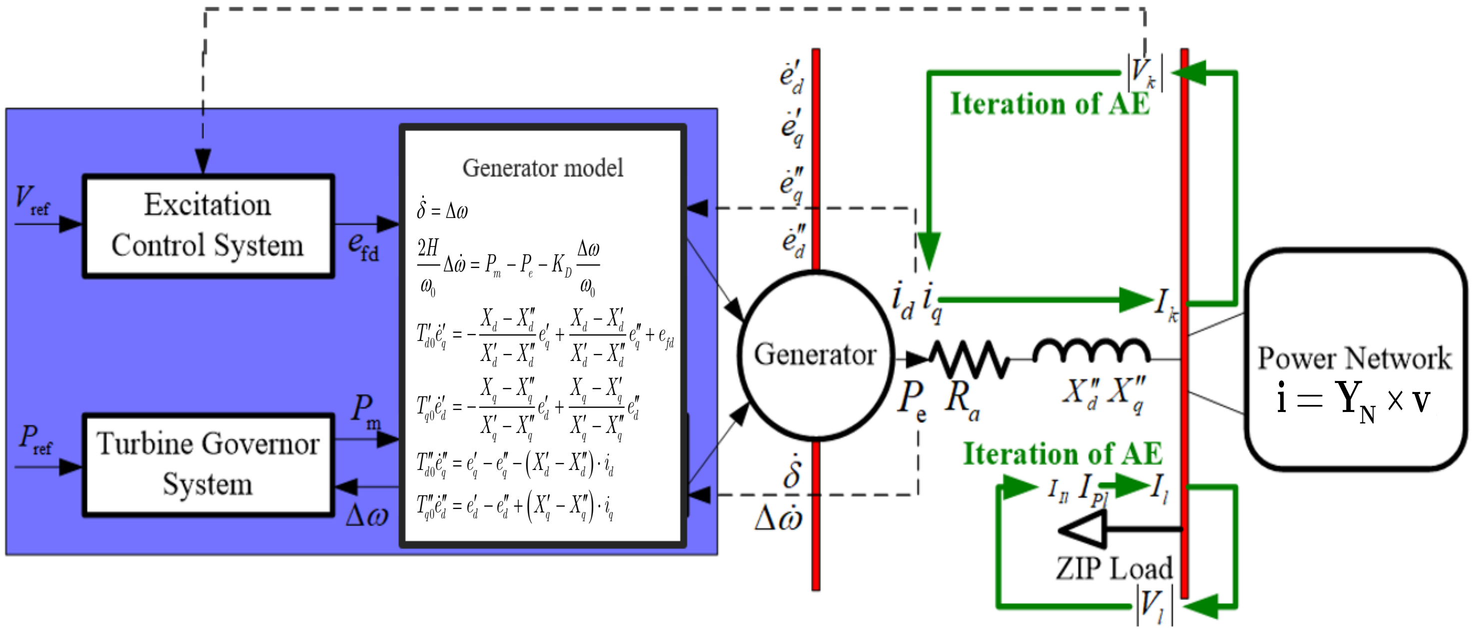
[8] R. Yao, Y. Liu, K. Sun, et al, "Efficient and Robust Dynamic Simulation of Power Systems with **Holomorphic Embedding**," *IEEE Trans. Power Systems*, 2020

[9] Y. Liu, K. Sun, J. Dong, "A Dynamized Power Flow Method based on **Differential Transformation**," *IEEE Access*, 2020

[10] B. Park, K. Sun, et al, "Examination of Semi-Analytical Solution Methods in the Coarse Operator of Parareal Algorithm for Power System Simulation," *IEEE Trans. Power Systems*, 2021 (**Adomian Decomposition** and **Homotopy Analysis**)

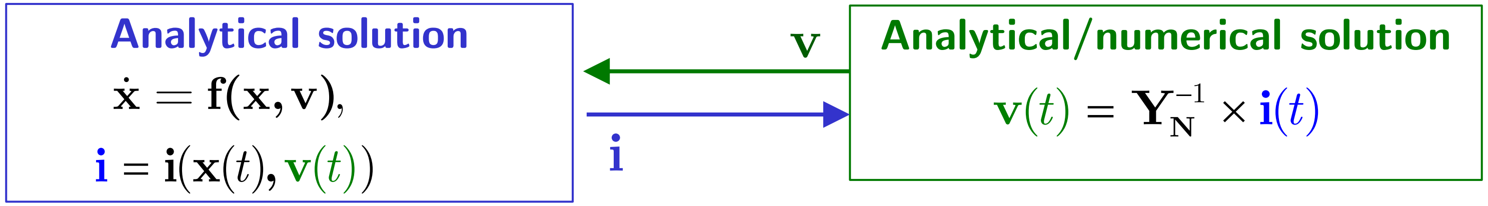
[11] Y. Liu, B. Park, K. Sun, et al, "Parallel-in-Time Power System Simulation Using a **Differential Transformation** based Adaptive Parareal Method," *IEEE Open Access Journal of Power and Energy*, 2022

Semi-analytical Simulation of a Large-scale Power Grid



Generator/Motor DEs

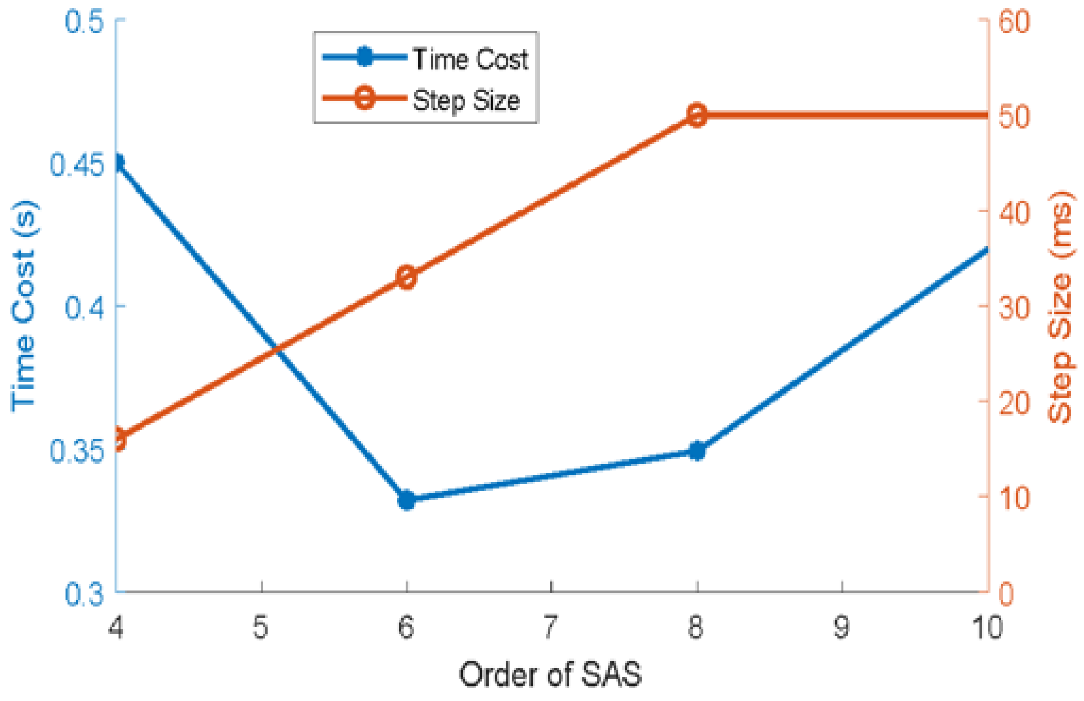
Generator/Motor/Network AEs



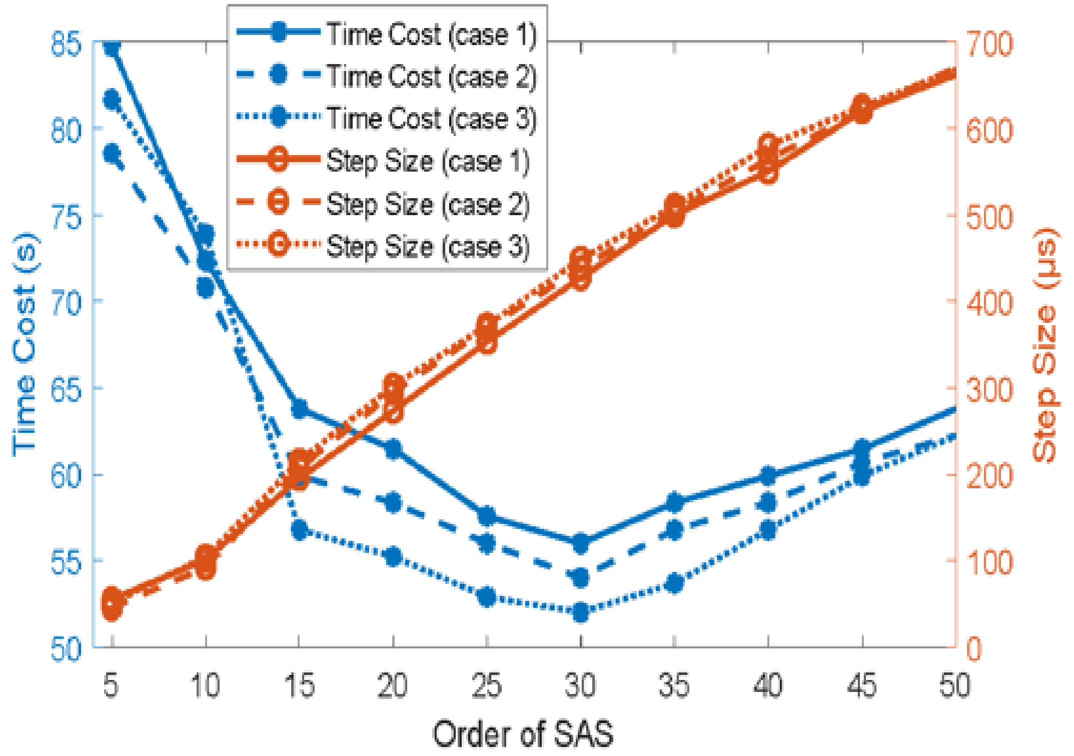
Partitioned solution scheme

- **1-stage analytical strategy**: Find and compute SAS while simulating.
  - Pro: Minimum storage for symbolic SAS
  - Con: All online work
- **2-stage analytical strategy** (integrated scheme): Offline find SAS and online compute SAS.
  - Pro: Minimum online computation
  - Con: Needs storage for complex SAS if symbolizing many parameters.
- **2-stage analytical-numerical strategy** (partitioned scheme replacing numerical DE solver by SAS): Offline find SAS of DEs, and online compute SAS together with numerical AE solution.
  - Pro. Simpler SAS
  - Con: Performance relies on numerical AE solver.

SAS Performances on IEEE 39-bus System



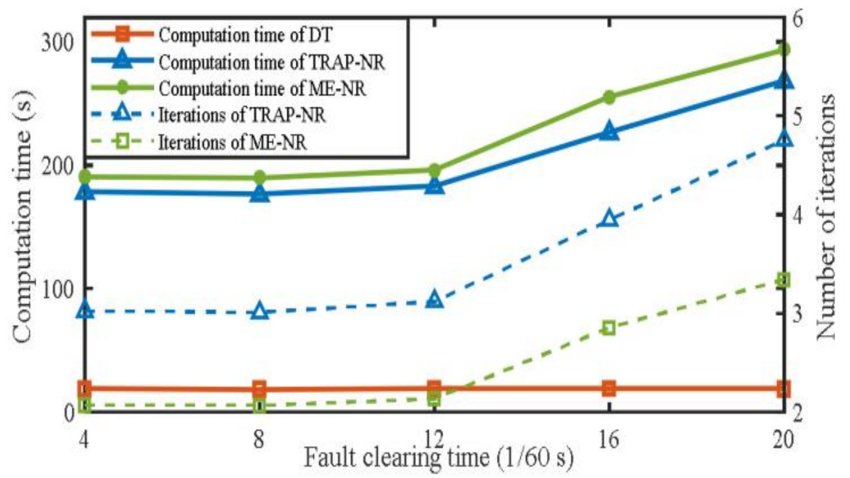
Transient Stability Simulation



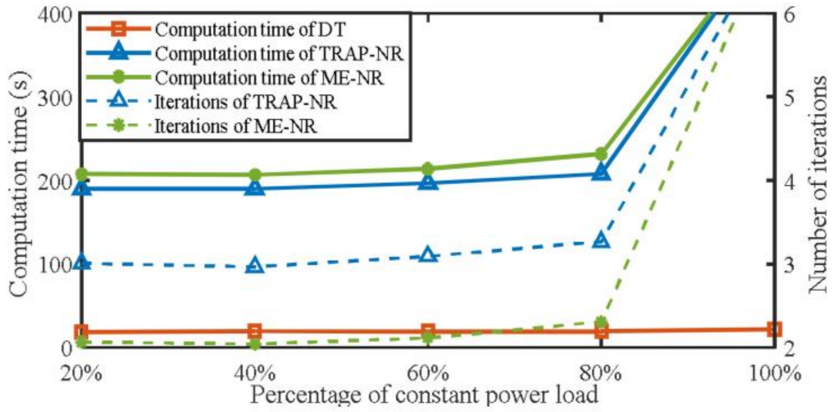
EMT Simulation

Performance of a Differential Transformation-based SAS (Polish 2383-bus system)

- Detailed models of generators, governors, turbines, exciters and ZIP loads
- Compare 3 methods (benchmark: TRAP-NR method with 0.1 ms time step)
  - TRAP-NR method in a simultaneous scheme, time step of 1ms
  - ME-NR method in a partitioned scheme, time step of 1ms
  - DT-based SAS method: order 8 and time step of 10ms
- Higher efficiency & better accuracy
  - Error is 1-2 orders lower, computation speed is 10x faster and the number of solved linear equations is reduced from 2000-3000 to 100.



Scenarios	Methods	Error of state variables (p.u.)	Error of bus voltages (p.u.)	Computation time (s)
Stable	**SAS**	**2.69×10^{-6}**	**3.33×10^{-7}**	**18.76**
	TRAR-NR	1.30×10^{-4}	1.10×10^{-6}	176.43
	ME-NR	2.63×10^{-4}	2.26×10^{-6}	191.40
Unstable	**SAS**	**1.89×10^{-6}**	**2.78×10^{-7}**	**18.85**
	TRAP-NR	1.41×10^{-4}	1.61×10^{-6}	182.76
	ME-NR	2.79×10^{-4}	2.93×10^{-6}	196.02



Variable-Step Optimal-Order Strategy (Polish 2383-bus System)

- A variable-step optimal-order strategy enables 5x speedup by using a 25x longer stepsize.

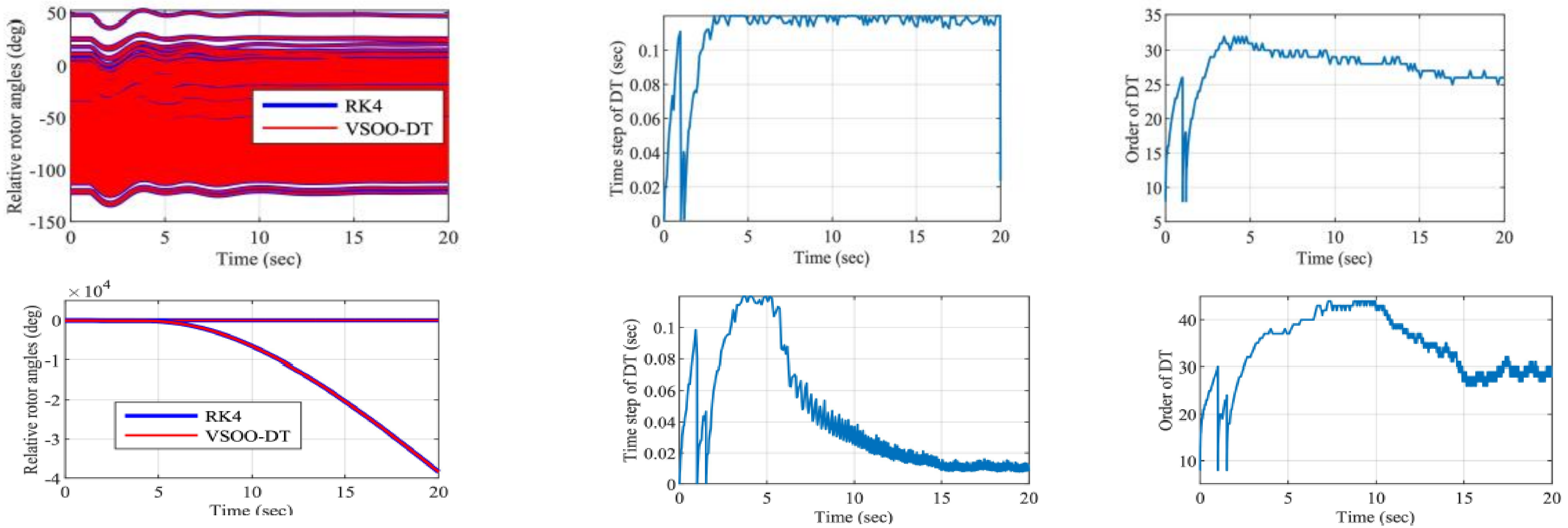


TABLE V
COMPUTATIONAL TIME BY DIFFERENT METHODS FOR 2383-BUS SYSTEMS

	RK4	DT	VS-DT	VSOO-DT
CPU Time (s)	35.5284	9.4555	12.3074	6.4822
Averaged Step (ms)	0.5556	5	4.6889	13.6597
Averaged Order	–	8	8	12.1957

SAS Performance on EMT Simulation (IEEE 39-bus System)

- Semi-analytical EMT simulation achieves 5-20x speed up by using 10-100x longer stepsizes.

TABLE II
COMPARING TIME STEPS AND TIME COSTS OF DIFFERENT METHODS

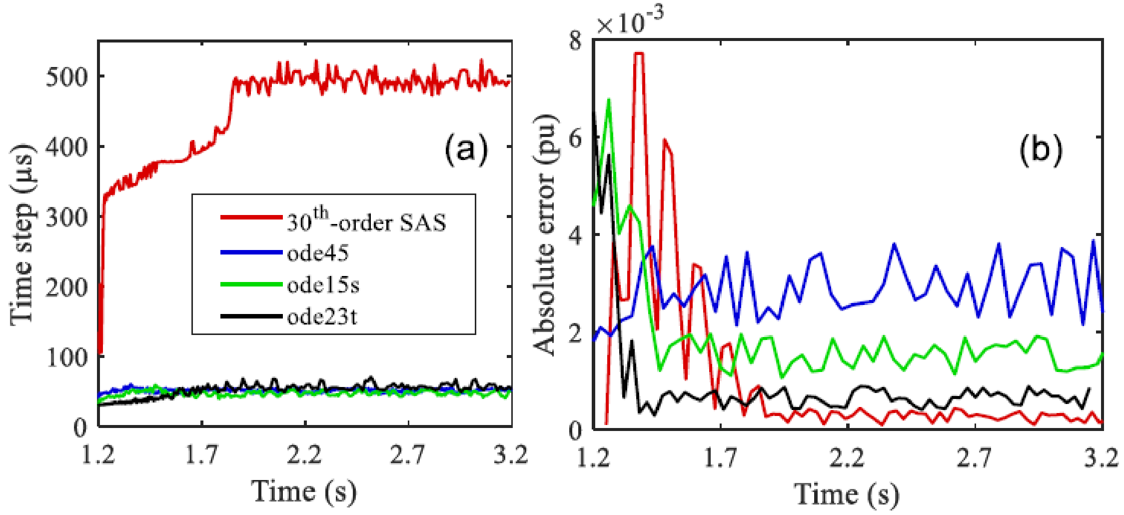
Case	NDF (ode15s)		Runge-Kutta (ode45)		SAS (30th-order)		Trapezoidal rule (ode23t)	
	Time step (μs)	Time cost (s)	Time step (μs)	Time cost (s)	Time step (μs)	Time cost (s)	Time step (μs)	Time cost (s)
Case 1	45.8	17.2	50.5	13.5	469	2.73	48.3	15.7
Case 2	46.2	17.4	50.3	13.6	464	2.71	48.6	15.5
Case 3	46.1	17.3	50.7	13.3	467	2.81	48.9	15.9

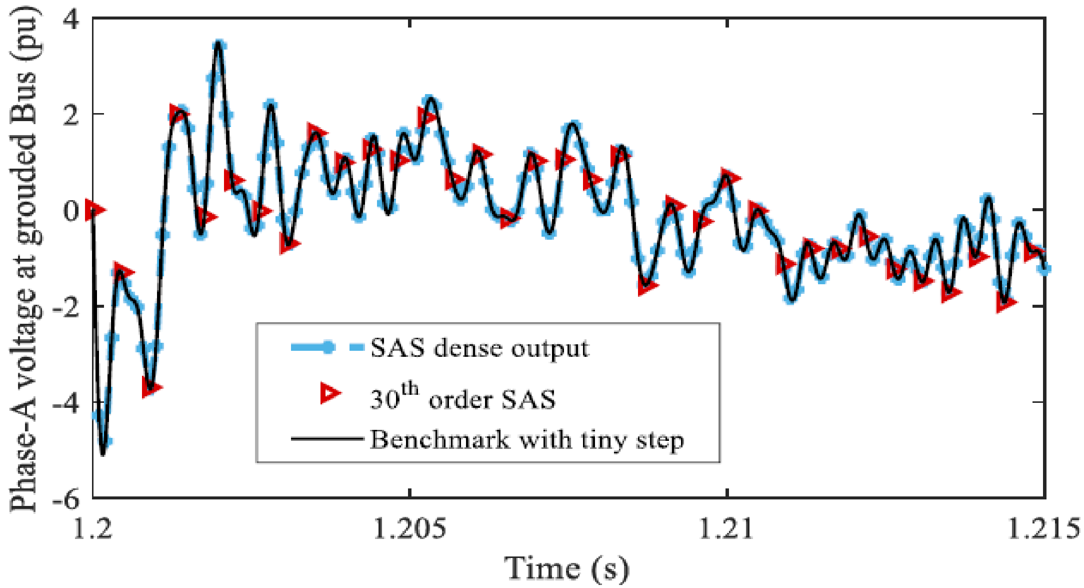
TABLE III
COMPARISON OF PERFORMANCE ON THE IEEE 39-BUS SYSTEM

Approach	PSCAD				SAS
Time step (μs)	5	50	75	100	467
Time cost (s)	56.2	5.62	3.75	2.81	2.71
Maximum error (pu)	1.10	5.52	6.11	5.95	7.7×10^{-3}
Average error ($\times10^{-3}$ pu)	0.15	1.73	2.80	4.11	0.032

TABLE IV
COMPARISON OF PERFORMANCE ON A 390-BUS SYSTEM

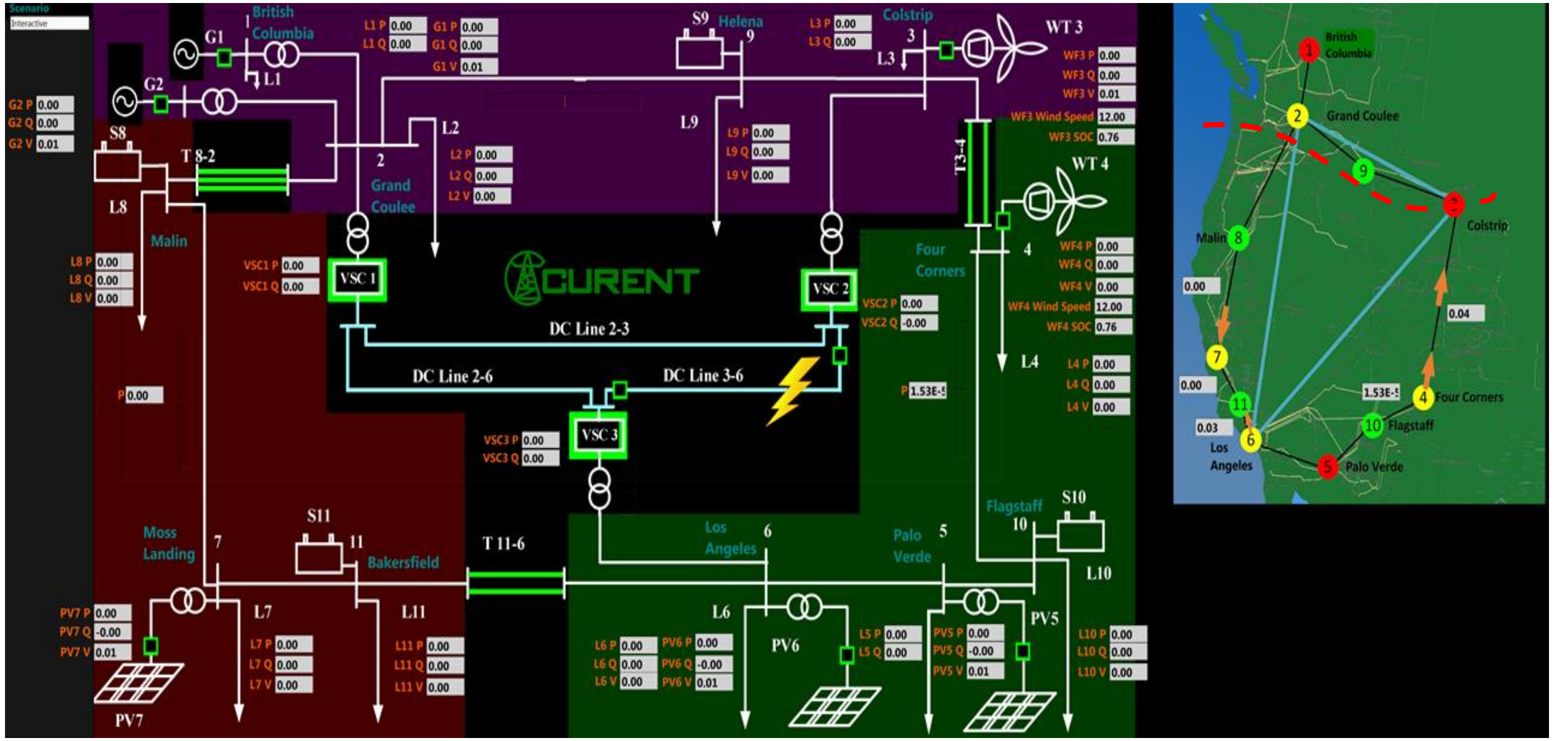
Approach	PSCAD				SAS
Time step (μs)	5	50	75	100	534
Time cost (s)	425	42.5	28.3	21.5	98
Maximum error (pu)	0.22	2.58	3.36	3.85	7×10^{-4}
Average error ($\times10^{-3}$ pu)	0.41	2.61	2.91	3.3	0.0012

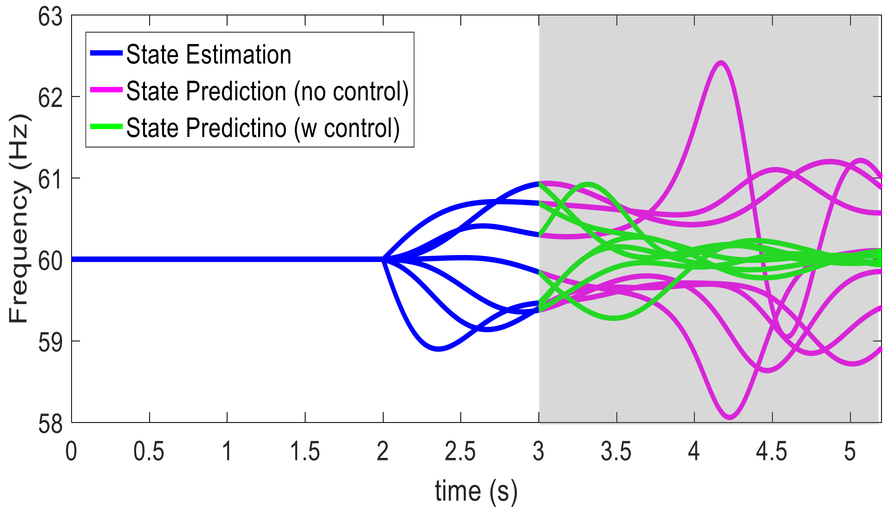


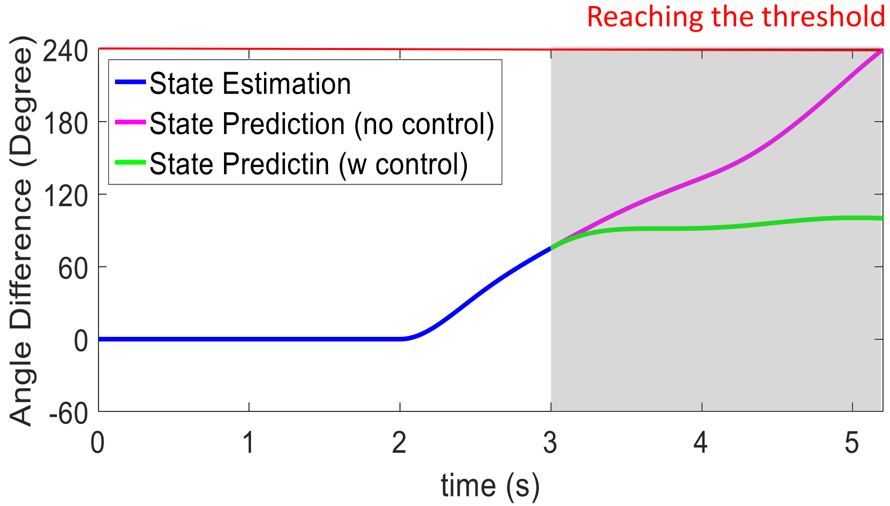


M. Xiong, K. Huang, Y. Liu, R. Yao, K. Sun, F. Qiu, "A Semi-Analytical Approach for State-Space Electromagnetic Transient Simulation Using the Differential Transformation,", submitted

Faster-Than-Real-Time Stability Assessment and Control (CURENT Hardware Testbed)







- Based on real-time state estimation, a SAS-based state predictor foresees instability in a next time window.
- After a line trip on the California-Oregon Intertie (2-8), once the N-S angle separation (between G2 and G7) is predicted to reach a threshold, separate the system and stabilize frequencies by HVDC.

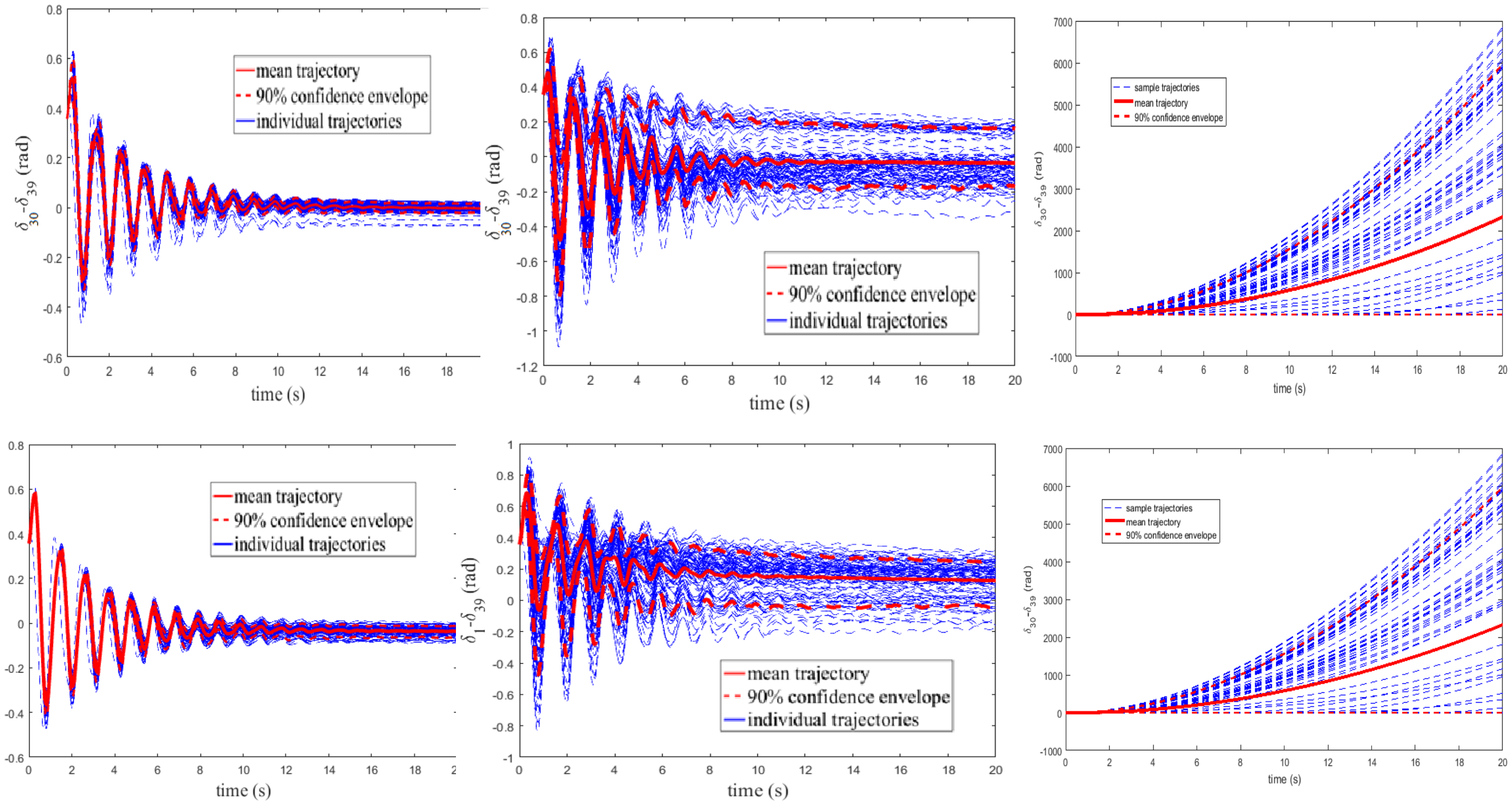
Stochastic Simulation with DERs and Stochastic Loads (IEEE 39-bus System)

LOW — uncertainty of load → **HIGH**

Case A	Case B	Case C

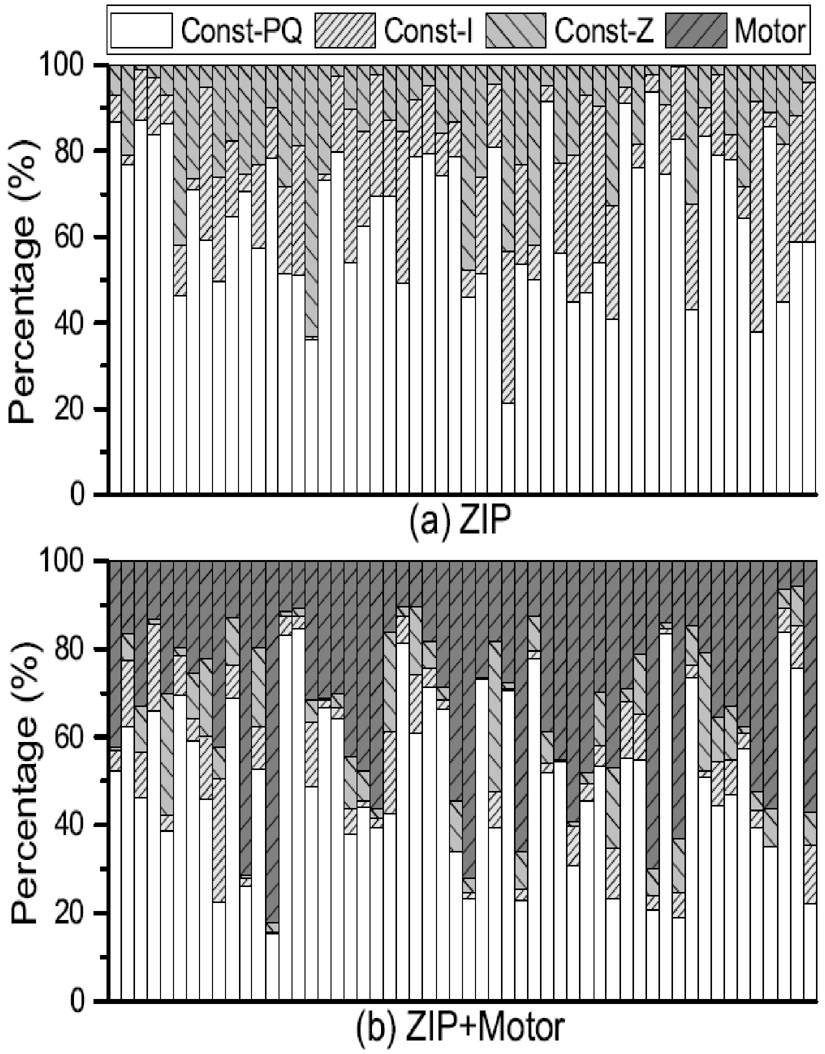
Euler-Maruyama

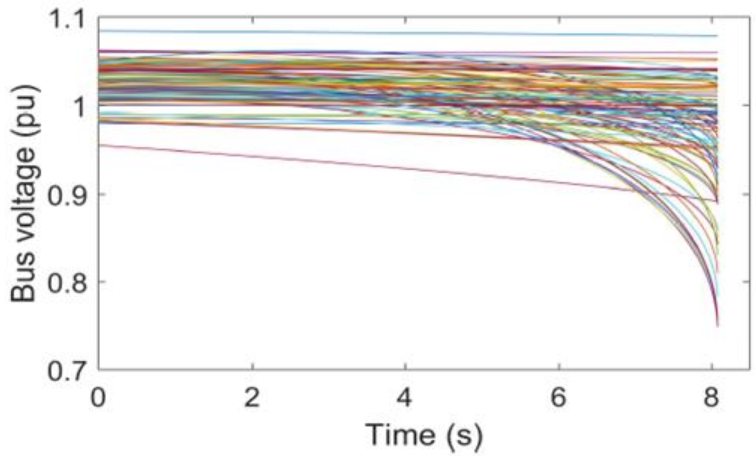
SAS



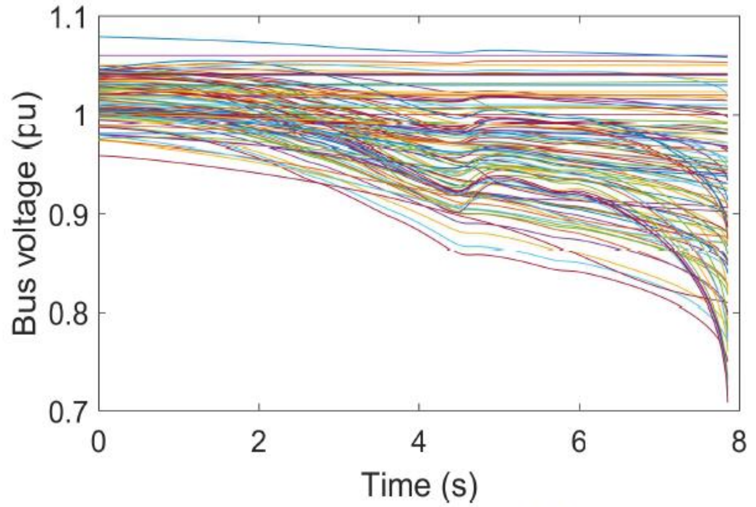
Time costs (s)	Cases B & C	Case A
Euler-Maruyama (single run)	11.6	11.4
Euler-Maruyama (100 runs)	1165.1	1142.4
SAS (single run)	5.1	5.1
SAS (100 runs)	511. 0	503.6

Simulation with Motor Loads (NPCC 140-bus system)





Bus voltage of NPCC system (ZIP load)



Bus voltage of NPCC system (ZIP+Motor load)

	Average stepsize	Time cost
Modified Euler	0.002 s	1150 s
SAS	0.196 s	12.09 s

Advantages of semi-analytical simulation

- May achieve 10-100x best time performance by a variable solution order and an adaptive step size.

- Promising for faster-than-real-time simulation stability assessment by shifting a majority of computation burdens to the offline stage

- Can be parallelized on high-performance computers thanks to the analytic nature of the SAS.

- Applicable to both deterministic and stochastic simulations.