Assessing the Effects of Uncertainty on the Dynamic Performance of Power Systems

Ian A. Hiskens

Vennema Professor of Engineering

Professor, Electrical Engineering and Computer Science

University of Michigan

Michigan**Engineering**

CDC Workshop: Uncertainty Management in Power System Dynamics

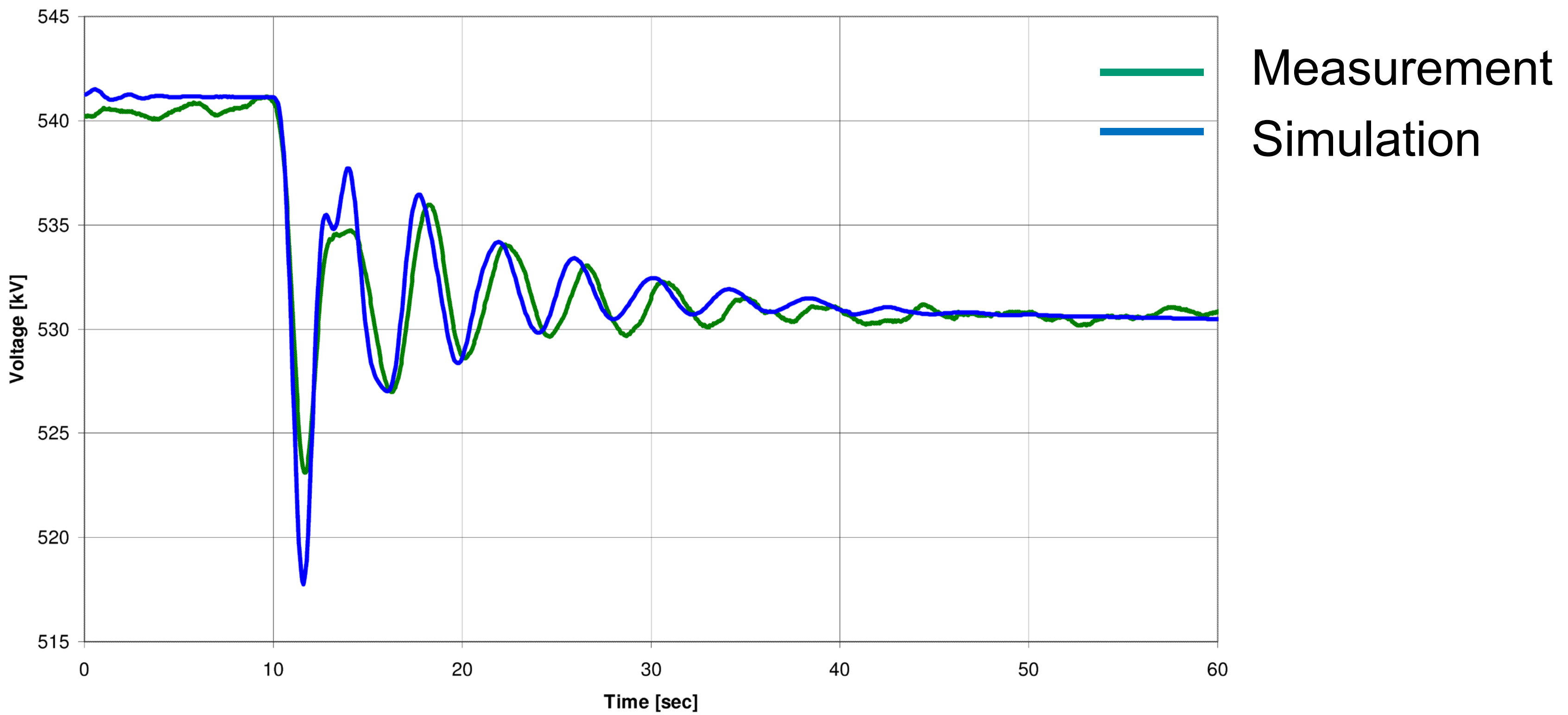
December 11, 2021

Motivation

- Dynamic performance assessment is vital in the design and operation of power systems (and other large-scale dynamical systems.)
- Post-mortem analysis of system events invariably reveals discrepancies between modelled and measured system behaviour.
  - Conclusion: models contain erroneous parameters.
- Sources of uncertainty:
  - Distribution networks.
  - Renewable generation.
  - Future: communications latency.

Example

- Voltage response to a generator trip.

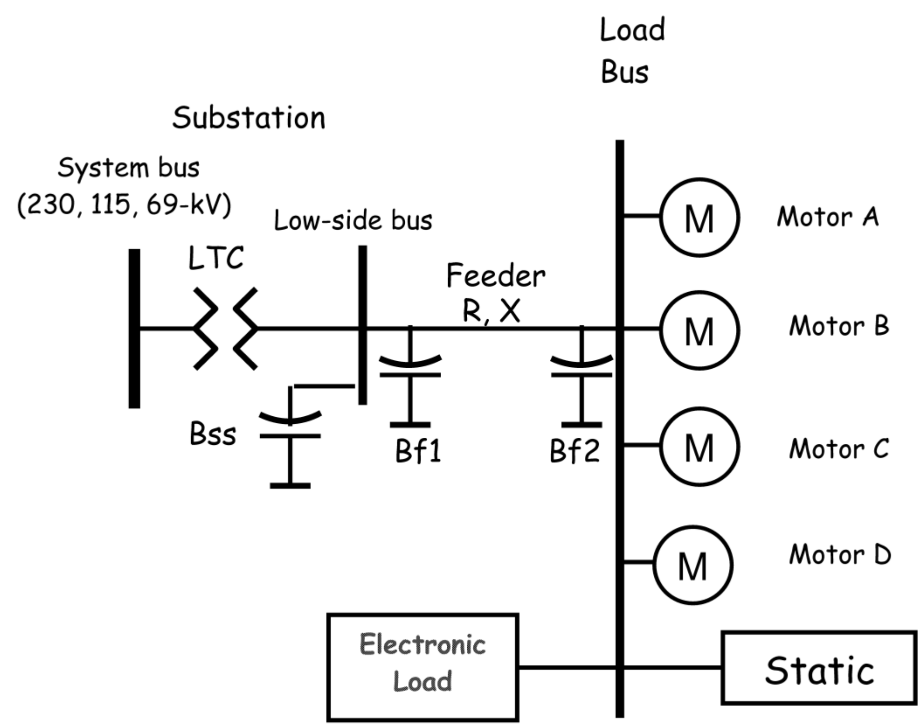


- Decisions based on simulations may lead to incorrect planning decisions and/or operating criteria.

Parameter uncertainty

- Large-scale renewable generation.
- Load model structures and parameters can never be known exactly.
  - Distribution network response will become even more uncertain with increased penetration of:
    - Distributed generation (e.g. solar PV).
    - Charging load of electric vehicles.

WECC load model



Uncertainty analysis

- Accounting for uncertainty in dynamic performance assessment is computationally challenging.
  - Monte Carlo: computationally infeasible.
- Probabilistic collocation method:
  - Assumes a polynomial relationship between uncertain parameters and quantities (outputs) of interest.
  - Computationally efficient when the number of uncertain parameters is relatively small.
- Trajectory approximation using first-order trajectory sensitivities.

Trajectory sensitivities

- Consider a trajectory (or flow) $x(t) = \phi(x_0, t)$ generated by simulation.
- Linearize the system around the *trajectory* rather than around the equilibrium point.

$$\begin{aligned}\Delta x(t) &= \frac{\partial \phi(x_0, t)}{\partial x_0} \Delta x_0 + \text{higher order terms} \\ &\approx \Phi(t) \Delta x_0\end{aligned}$$

- Trajectory sensitivities describe the change in the trajectory due to (small) changes in parameters and/or initial conditions.
  - Parameters incorporated via $\dot{\lambda} = 0, \qquad \lambda(0) = \lambda_0$
- Provide gradient information for iteratively solving inverse problems, such as parameter estimation.

Trajectory sensitivity evolution

- Along smooth sections of the trajectory

System evolution

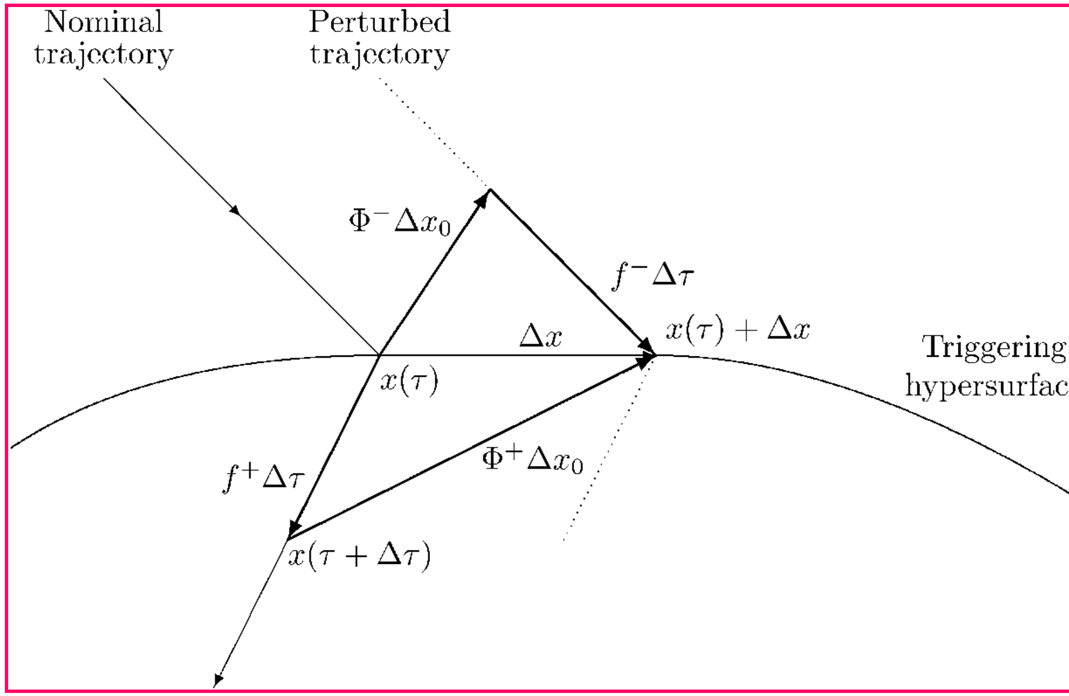
$$\dot{x} = f(x), \qquad x(0) = x_0$$

Sensitivity evolution

$$\dot{\Phi} = \left.\frac{\partial f}{\partial x}\right|_{x(t)} \Phi, \qquad \Phi(0) = I$$

- At an event

$$\Phi(\tau^+) = \Phi(\tau^-) - \left(f^+ - f^-\right) \frac{\partial \tau}{\partial x_0}$$



Trajectory sensitivity computation

Implicit numerical integration allows efficient computation of trajectory sensitivities.

System evolution

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

Trapezoidal integration

$$x^{k+1} = x^k + \tfrac{h}{2}\Big(f(x^k) + f(x^{k+1})\Big)$$

Each integration timestep involves a Newton solution process.

- The Jacobian $\underbrace{\left(\frac{h}{2}Df - I\right)}$ must be formed and factored.

Sensitivity evolution

$$\dot{\Phi} = Df(x(t))\Phi$$

Trapezoidal integration

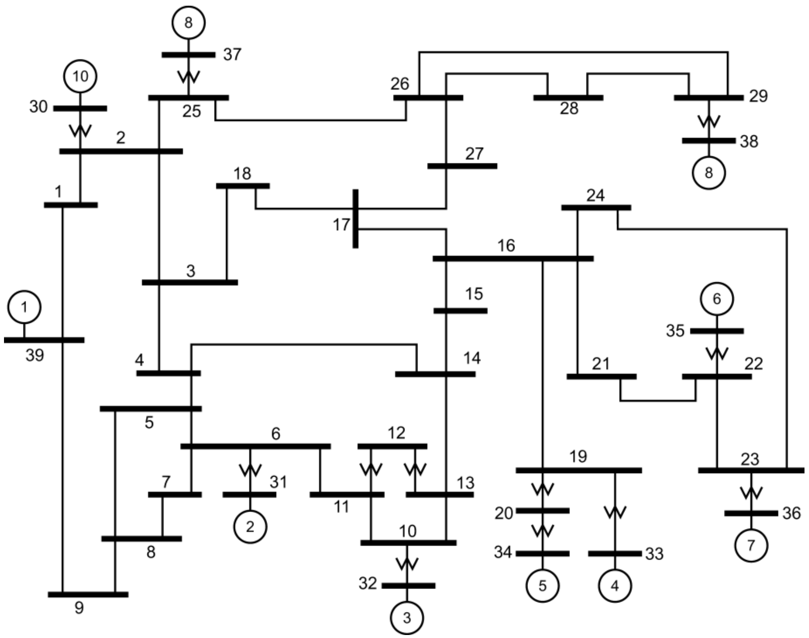
$$\Phi^{k+1} = \Phi^k + \tfrac{h}{2}\Big(Df(x^k)\Phi^k + Df(x^{k+1})\Phi^{k+1}\Big)$$

$$\Rightarrow \underbrace{\left(\frac{h}{2}Df(x^{k+1}) - I\right)}\Phi^{k+1} = -\Big(\tfrac{h}{2}Df(x^k) + I\Big)\Phi^k$$

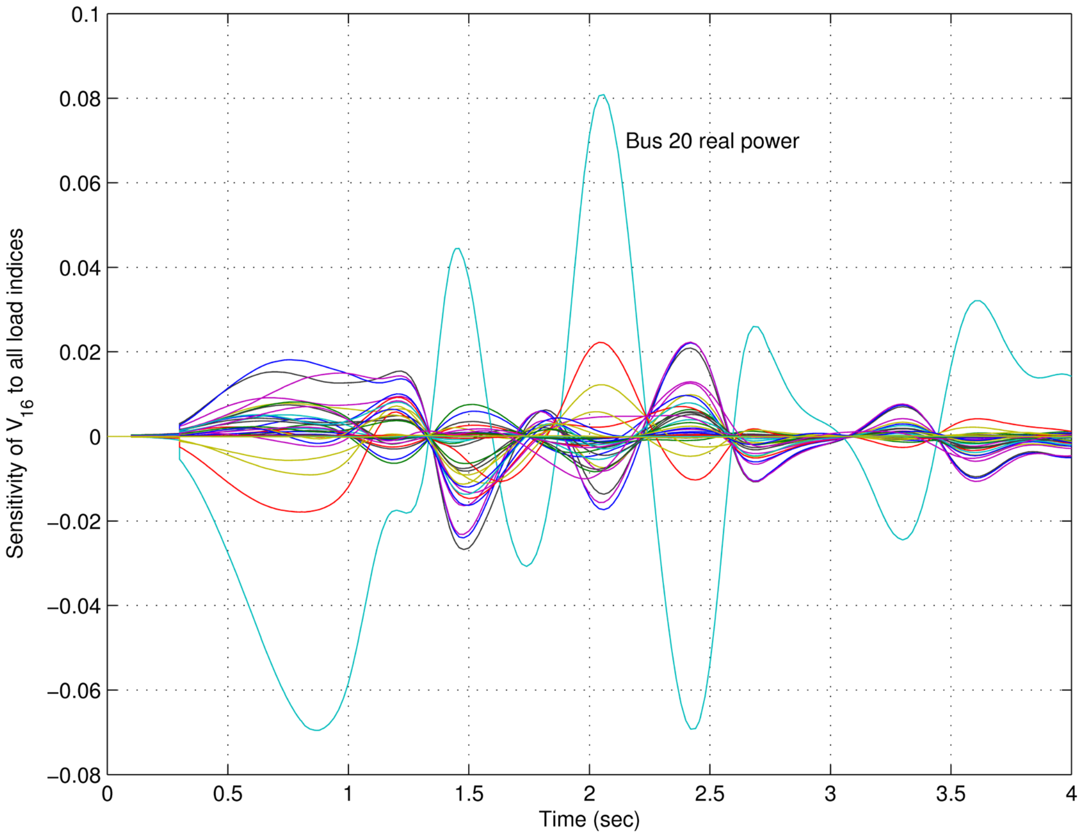
$\boxed{\text{Already factored}}$

Parameter identifiability

IEEE 39 bus system



Sensitivity of V_{16} to load parameters



Trajectory approximation

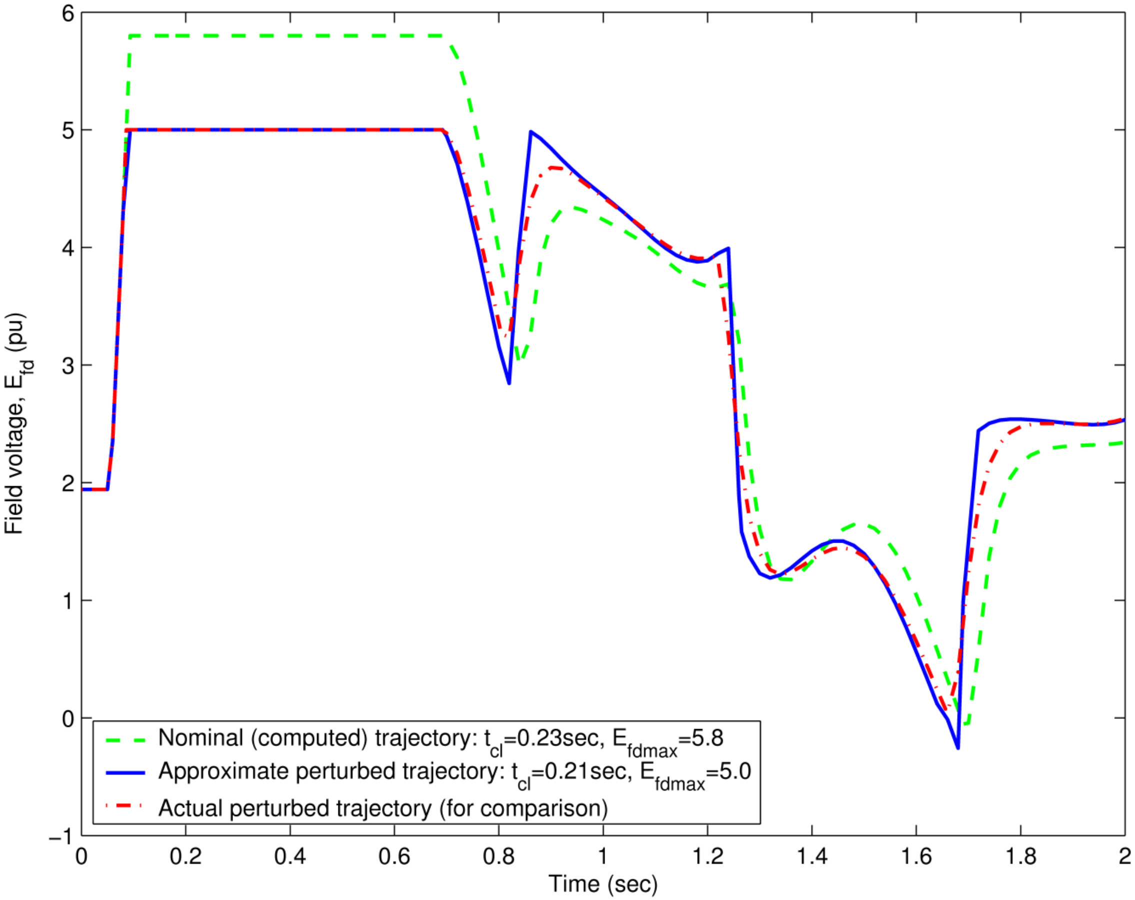
- Neglecting higher order terms of the Taylor series:

$$\phi(x_0 + \Delta x_0, t) \approx \phi(x_0, t) + \Phi(x_0, t)\Delta x_0$$

- Affine structure.

Example:

- Generator field voltage response to a fault.



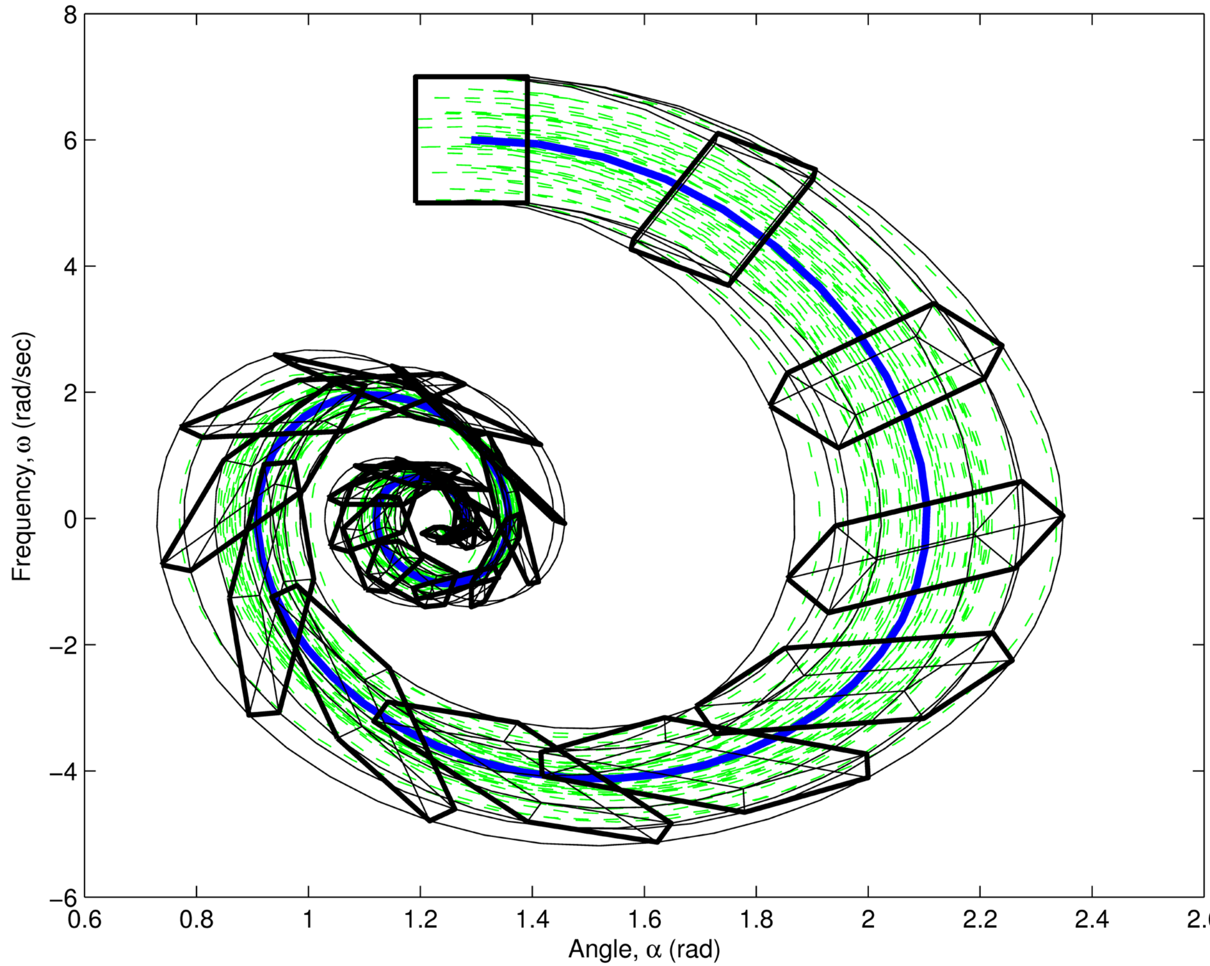
Parameter uncertainty

Worst-case analysis: Assume parameter uncertainty is uniformly distributed over an orthotope $\mathcal{B}$ (multi-dimensional rectangle.)

Assume all trajectories emanating from $x_0 + \mathcal{B}$ have the same order of events.

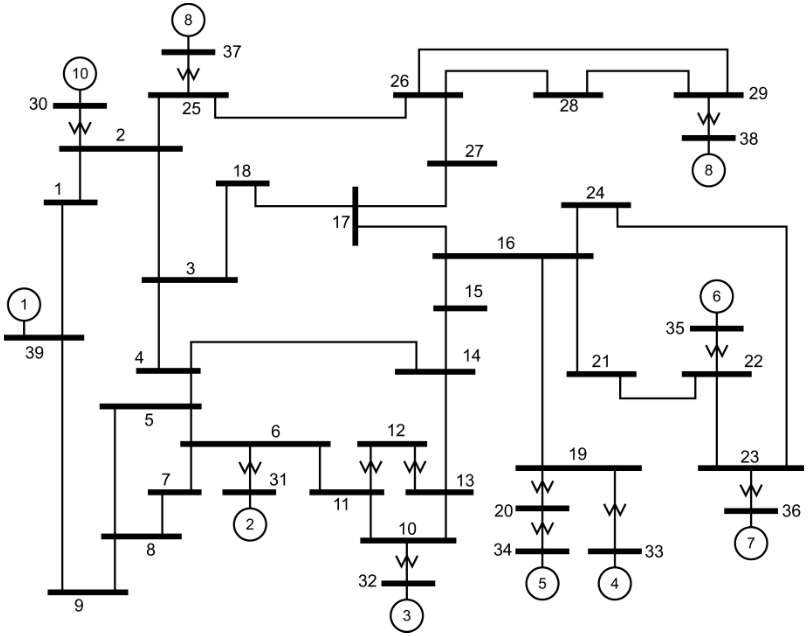
Propagation of uncertainty is described (approximately) by the time-varying parallelotope

$$\mathcal{P}(t) = \phi(x_0, t) + \Phi(x_0, t)\mathcal{B}$$

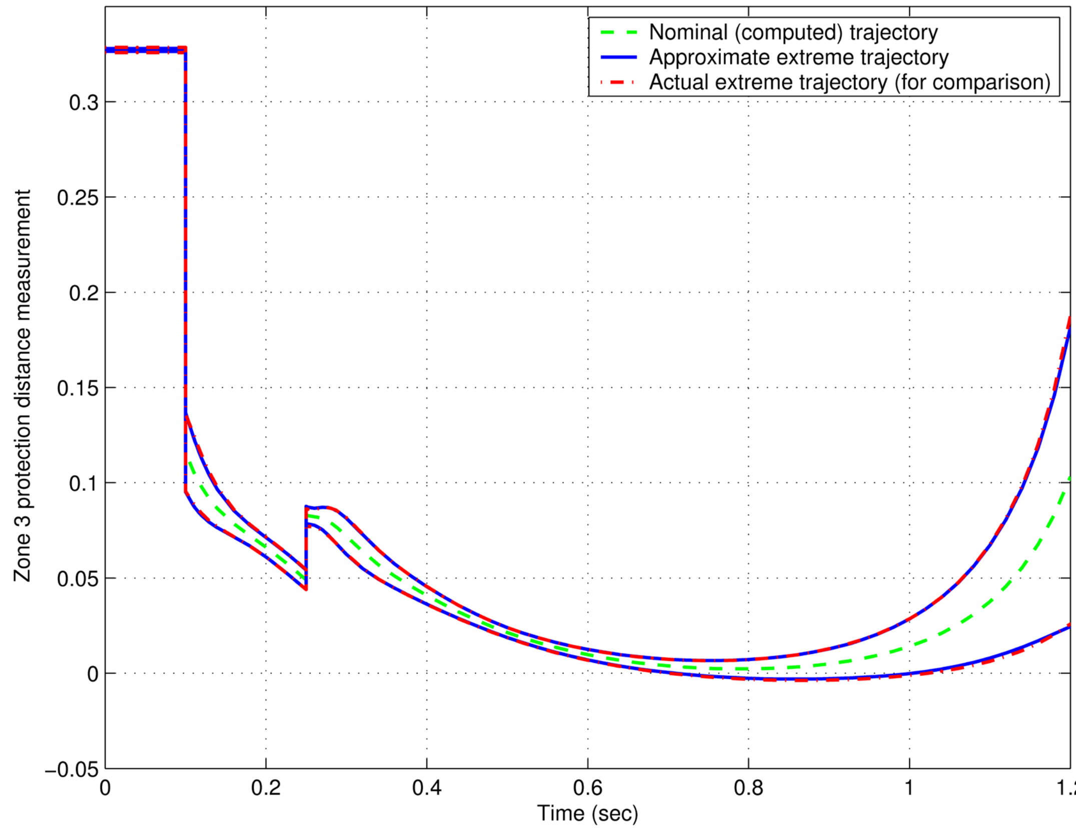


Example – worst case analysis

Uncertainty: $0.3 \leq \nu_{23}, \nu_{24} \leq 0.7$

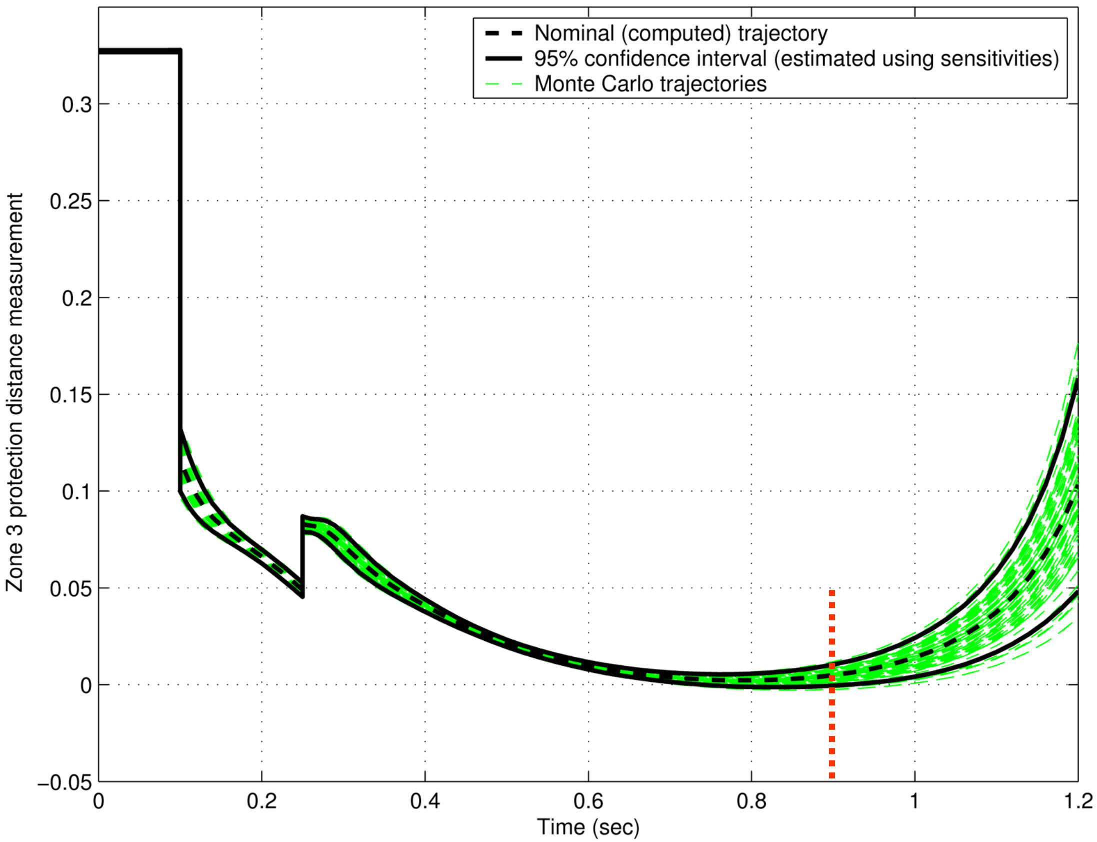


Protection signal (zero crossing indicates protection trip).

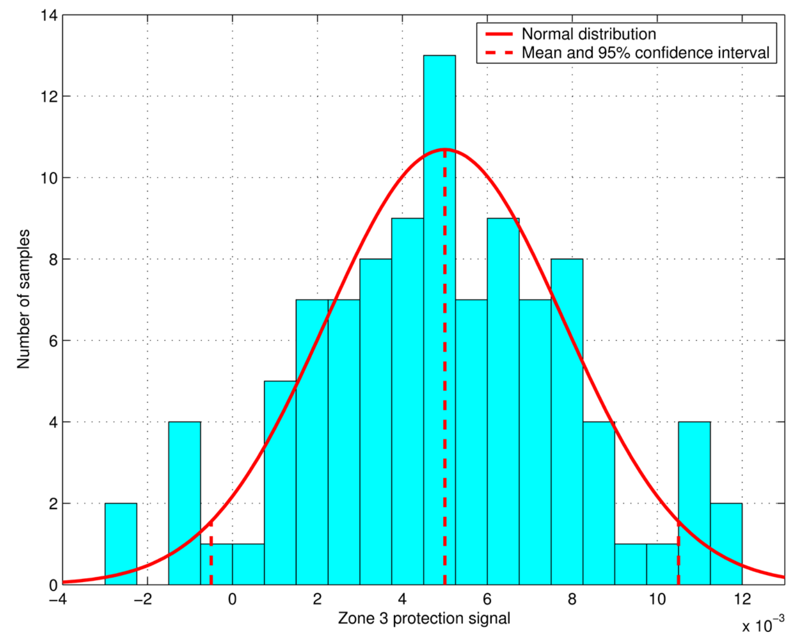


Example – probabilistic assessment

Uncertainty: $E[\nu_{23}] = E[\nu_{24}] = 0.5, \ \mathrm{Var}[\nu_{23}] = \mathrm{Var}[\nu_{24}] = 0.01$



Signal distribution at 0.9 sec.

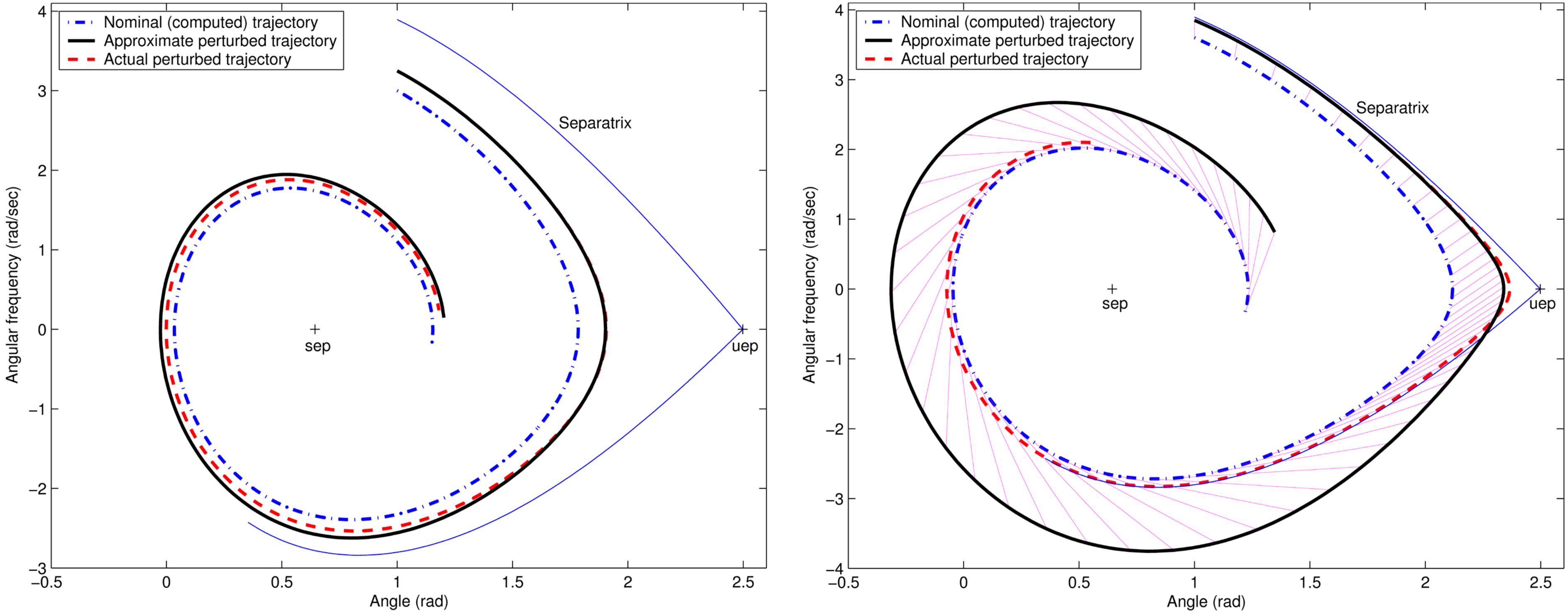


Approximation under stressed conditions

- Truncated Taylor series:

$$\phi(x_0 + \Delta x_0, t) \approx \phi(x_0, t) + \Phi(x_0, t)\Delta x_0$$

- Accuracy reduces as the trajectory approaches the stability boundary, where nonlinearities become dominant.



Improvement using time warping

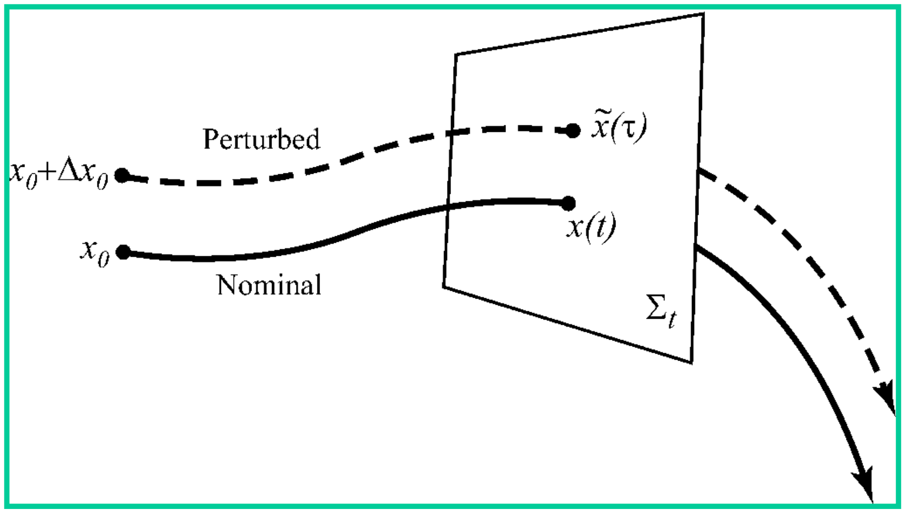
- Accuracy of approximations suffers when the nominal and perturbed trajectories proceed at substantially different rates.
  - Time synchronized points $x(t)$ and $\tilde{x}(t)$ on the nominal and perturbed trajectories will separate.
  - Even though the trajectories remain close in state space.
- This difficulty can be overcome by using a hyperplane Σ_t that "slides" along with the nominal trajectory.
- The point where the perturbed trajectory intersects the sliding hyperplane is given approximately by

$$\phi(x_0 + \Delta x_0, t + \Delta\tau)$$
$$\approx \phi(x_0, t) + \Phi(x_0, t)\Delta x_0 + f(x(t))\Delta\tau$$

where

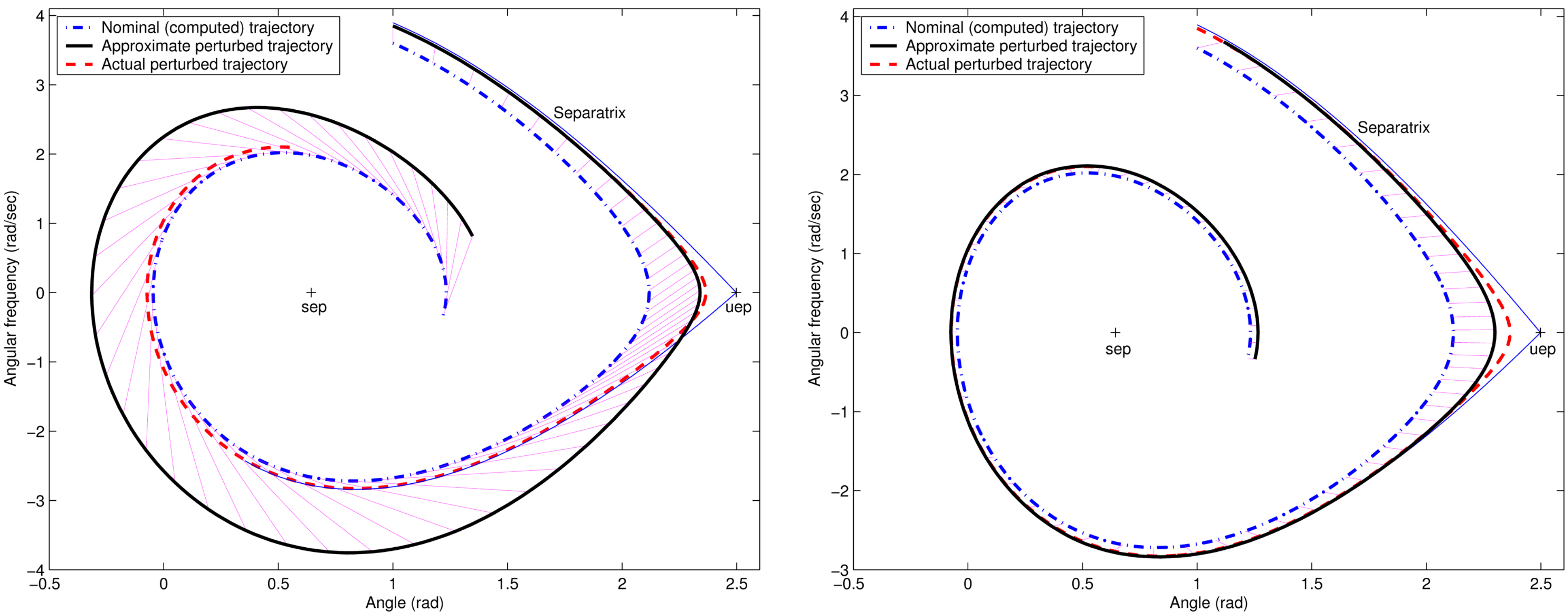
$$\Delta\tau = \frac{-1}{h_t^\top f(x(t))} h_t^\top \Phi(x_0, t)\Delta x_0$$

h_t is orthogonal to Σ_t.



Trajectory adjustment

- A useful choice for the time-varying hyperplane is given by: $h_t \equiv f(t)$
  - The hyperplane is always orthogonal to the nominal trajectory.



Second-order trajectory sensitivities

- The evolution of second-order sensitivities for a DAE system is described by,

$$
\begin{aligned}
\frac{d}{dt}\Big(\frac{\partial^2 x}{\partial x_0^2}\Big) &= \mathrm{diag}_n\big(\frac{\partial x}{\partial x_0}\big)^{\mathsf{T}}\Big(\frac{\partial^2 f}{\partial x^2}\mathrm{vec}_n\big(\frac{\partial x}{\partial x_0}\big) + \frac{\partial^2 f}{\partial x \partial y}\mathrm{vec}_n\big(\frac{\partial y}{\partial x_0}\big)\Big) \\
&\quad + \mathrm{diag}_n\big(\frac{\partial y}{\partial x_0}\big)^{\mathsf{T}}\Big(\frac{\partial^2 f}{\partial y \partial x}\mathrm{vec}_n\big(\frac{\partial x}{\partial x_0}\big) + \frac{\partial^2 f}{\partial y^2}\mathrm{vec}_n\big(\frac{\partial y}{\partial x_0}\big)\Big) \\
&\quad + \big(\frac{\partial f}{\partial x} \otimes I_n\big)\frac{\partial^2 x}{\partial x_0^2} + \big(\frac{\partial f}{\partial y} \otimes I_n\big)\frac{\partial^2 y}{\partial x_0^2}, \\
0_{nm\times n} &= \mathrm{diag}_m\big(\frac{\partial x}{\partial x_0}\big)^{\mathsf{T}}\Big(\frac{\partial^2 g}{\partial x^2}\mathrm{vec}_m\big(\frac{\partial x}{\partial x_0}\big) + \frac{\partial^2 g}{\partial x \partial y}\mathrm{vec}_m\big(\frac{\partial y}{\partial x_0}\big)\Big) \\
&\quad + \mathrm{diag}_m\big(\frac{\partial y}{\partial x_0}\big)^{\mathsf{T}}\Big(\frac{\partial^2 g}{\partial y \partial x}\mathrm{vec}_m\big(\frac{\partial x}{\partial x_0}\big) + \frac{\partial^2 g}{\partial y^2}\mathrm{vec}_m\big(\frac{\partial y}{\partial x_0}\big)\Big) \\
&\quad + \big(\frac{\partial g}{\partial x} \otimes I_n\big)\frac{\partial^2 x}{\partial x_0^2} + \big(\frac{\partial g}{\partial y} \otimes I_n\big)\frac{\partial^2 y}{\partial x_0^2}.
\end{aligned}
$$

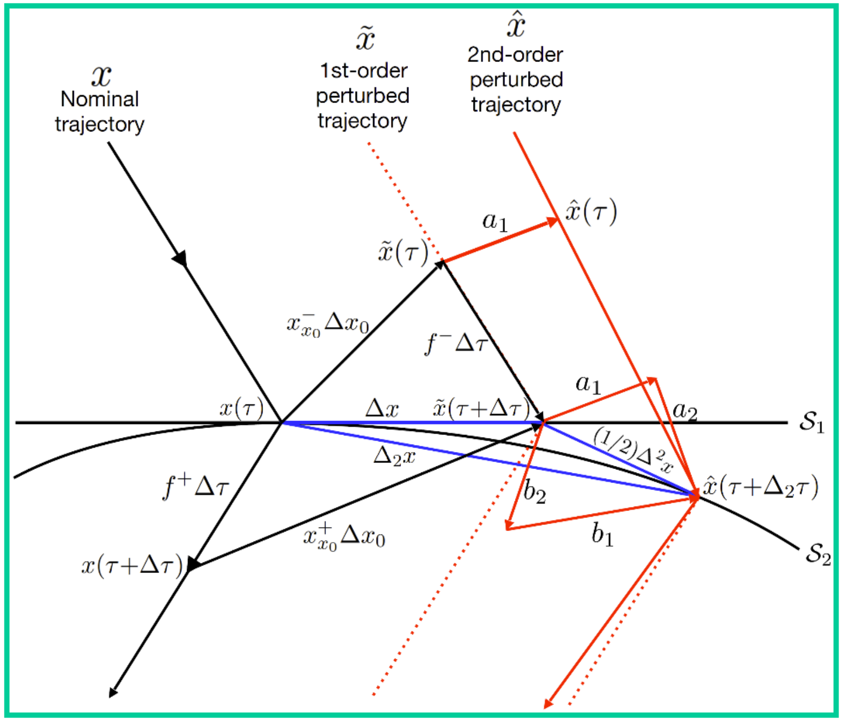
- Perhaps surprisingly, these sensitivities can be computed efficiently using LU factors that have previously been computed.

Jump conditions at events

- The evolution of second-order sensitivities at events is described by,

$$\frac{\partial^2 x_i^+}{\partial x_0^2} = \frac{\partial^2 x_i^-}{\partial x_0^2} + \frac{dx^-}{dx_0}^{\mathsf{T}} \frac{\partial^2 \tilde{h}_i}{\partial x^2} \frac{dx^-}{dx_0} + \sum_{j=1, j\neq i}^{n} \frac{\partial \tilde{h}_i}{\partial x_j} \frac{d^2 x_j^-}{dx_0^2} + \Big(\frac{\partial \tilde{h}_i}{\partial x_i} - 1\Big) \frac{d^2 x_i^-}{dx_0^2}$$
$$- \frac{\partial \tau}{\partial x_0}^{\mathsf{T}} \Big(\frac{\partial f_i^+}{\partial x_0} - \frac{\partial f_i^-}{\partial x_0}\Big) - \Big(\frac{\partial f_i^+}{\partial x_0} - \frac{\partial f_i^-}{\partial x_0}\Big)^{\mathsf{T}} \frac{\partial \tau}{\partial x_0}$$
$$- (f_i^+ - f_i^-) \frac{\partial^2 \tau}{\partial x_0^2} - \frac{\partial \tau}{\partial x_0}^{\mathsf{T}} \Big(\frac{\partial f_i^+}{\partial \tau} - \frac{\partial f_i^-}{\partial \tau}\Big) \frac{\partial \tau}{\partial x_0}$$

- These second-order sensitivities can be used to establish error bounds on first-order approximate trajectories.



Details can be found in Geng and Hiskens, "Second-order trajectory sensitivity analysis of hybrid systems", *IEEE TCAS-I*, Vol. 66, No. 5, May 2019, pp. 1922-1934.

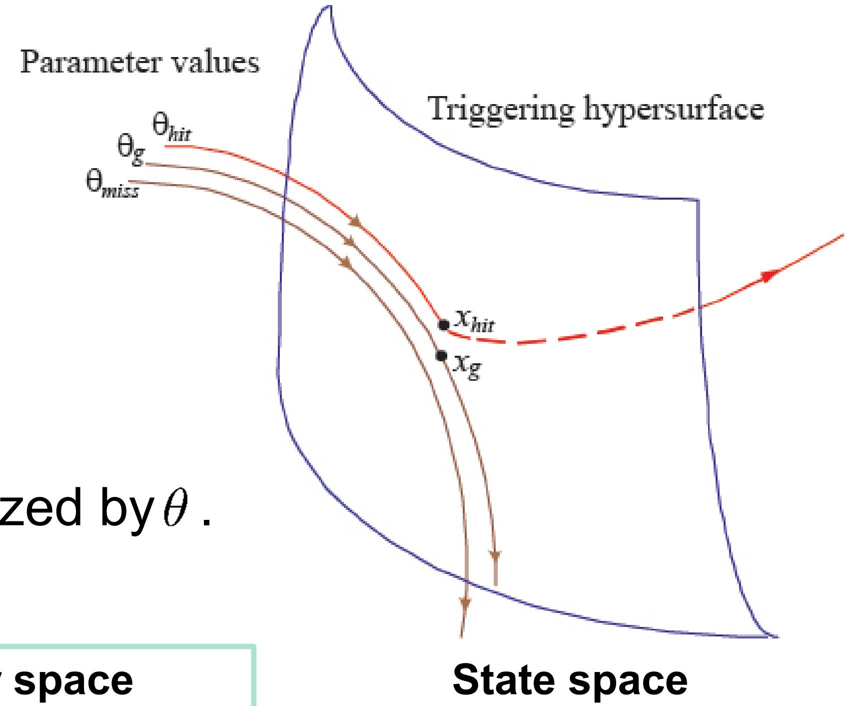
Grazing formulation for reachability

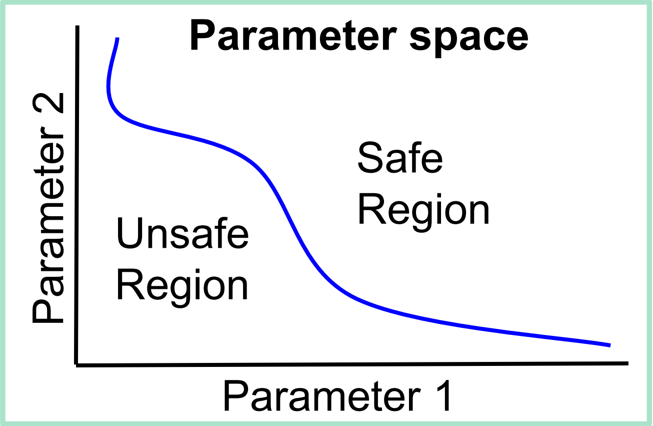
- At a grazing point the trajectory has a tangential encounter with the target hypersurface: $b(x) = 0$
- Tangency implies: $\nabla b^\top f(x) = 0$
- Grazing points are described by:

$$
\begin{aligned}
F_{g1}(x_g, \theta, t_g) &:= \phi(x_0(\theta), t_g) - x_g &= 0 \\
F_{g2}(x_g) &:= b(x_g) &= 0 \\
F_{g3}(x_g) &:= \nabla b(x_g)^\top f(x_g) &= 0
\end{aligned}
$$

where initial conditions are parameterized by θ .

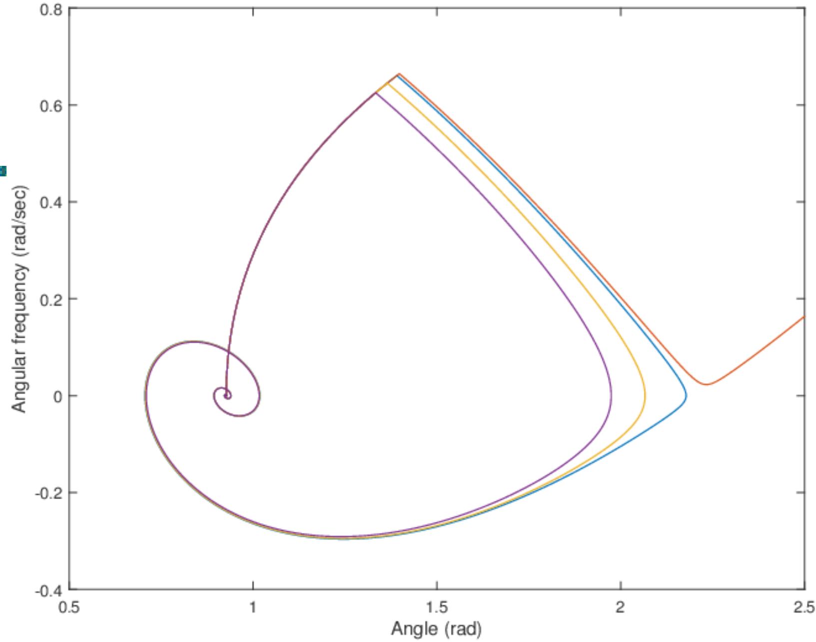
- This formulation extends naturally to a continuation setting.

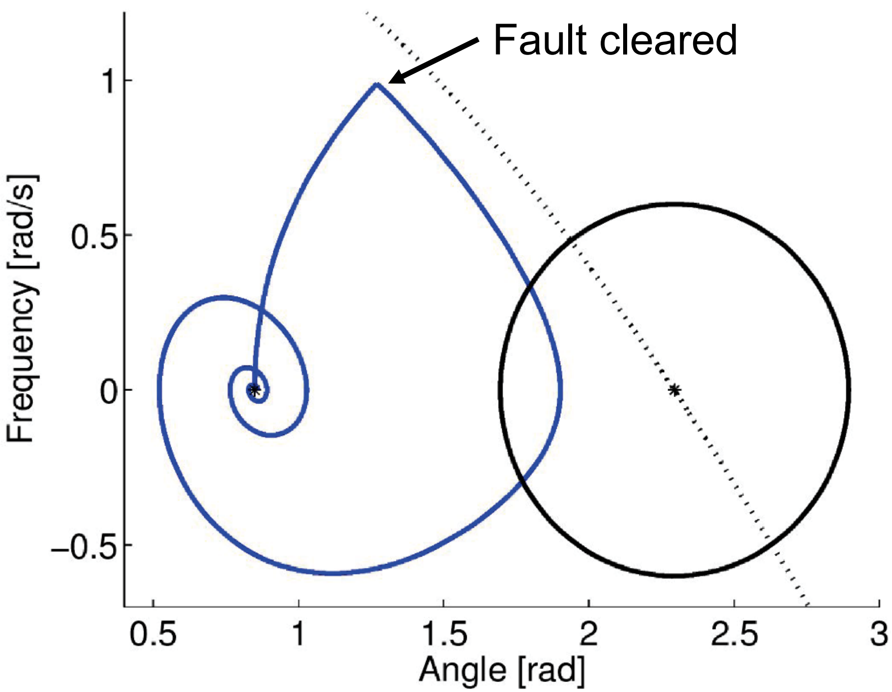


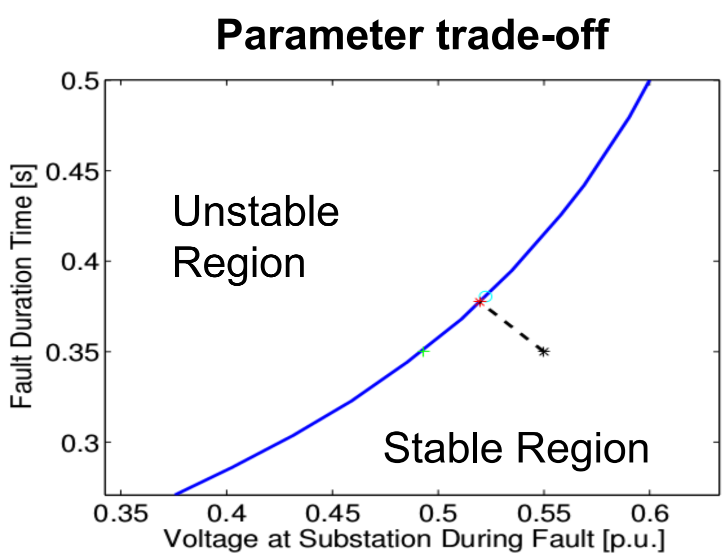


Stability assessment

- Approximate a trajectory on the stability boundary by a trajectory that spends a long time near the unstable equilibrium point.







Conclusions and future directions

- The level of uncertainty is certain to increase.
- It is not sufficient to know the nominal trajectory.
  - Need to know whether the reachable set remains safe.

- Load control offers enormous benefits, but will introduce inherently uncertain actuators.

- For large systems, trajectory sensitivities require a large amount of storage.
  - Explore the use of compression techniques for efficiently storing significant features.
- Further explore second-order sensitivities for bounding the error in first-order approximate trajectories.
- Incorporate uncertainty arising from the latency of wide-area controls.