Systematic Simulation using Sensitivity Analysis

Alexandre Donzé and Oded Maler

Verimag Laboratory, Grenoble

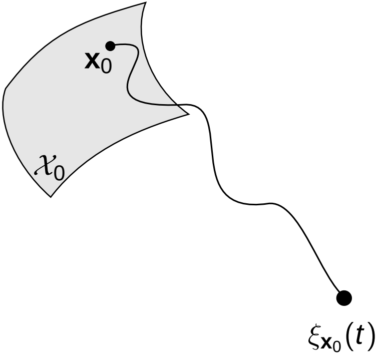
June 7, 2007

- Dynamical system of the form:

$$\dot{\mathbf{x}} = f(t, \mathbf{x})$$
$$\mathbf{x}(0) = \mathbf{x}_0 \in \mathcal{X}_0$$

- Unique trajectory:

$$\xi_{\mathbf{x}_0} : \mathbb{R}^+ \mapsto \mathcal{X}$$

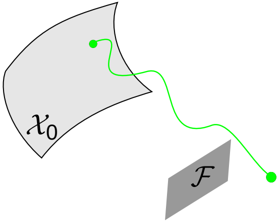


Let $\mathcal{F}$ be a bad set.

Safe

The system is safe if for all $\mathbf{x}_0 \in \mathcal{X}_0$, for all $t \in [0, T]$

$$\xi_{\mathbf{x}_0}(t) \cap \mathcal{F} = \emptyset$$

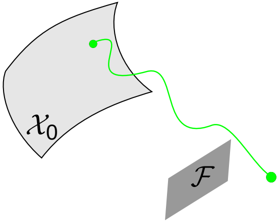


Let $\mathcal{F}$ be a bad set.

Safe

The system is safe if for all $\mathbf{x}_0 \in \mathcal{X}_0$, for all $t \in [0, T]$

$$\xi_{\mathbf{x}_0}(t) \cap \mathcal{F} = \emptyset$$



Goal

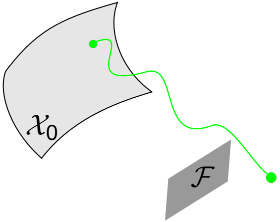
Using a finite number of trajectories,

Let $\mathcal{F}$ be a bad set.

Safe

The system is safe if for all $\mathbf{x}_0 \in \mathcal{X}_0$, for all $t \in [0, T]$

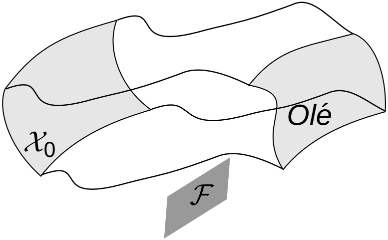
$$\xi_{\mathbf{x}_0}(t) \cap \mathcal{F} = \emptyset$$



Goal

Using a finite number of trajectories,

Prove that the system is safe

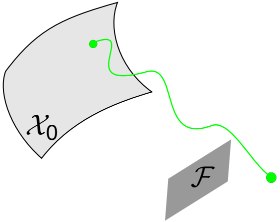


Let $\mathcal{F}$ be a bad set.

Safe

The system is safe if for all $\mathbf{x}_0 \in \mathcal{X}_0$, for all $t \in [0, T]$

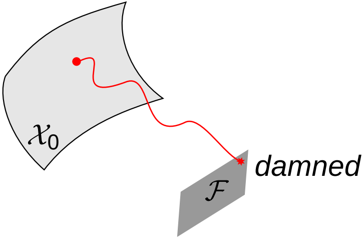
$$\xi_{\mathbf{x}_0}(t) \cap \mathcal{F} = \emptyset$$



Goal

Using a finite number of trajectories,

Prove that the system is safe
or
Find a counter-example.

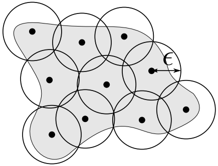


Outline

Definitions

- A sampling of $\mathcal{X}$ is a set $\mathcal{S} = \{\mathbf{x}_1, \ldots, \mathbf{x}_k\}$ of points in $\mathcal{X}$.

- The dispersion $\alpha_{\mathcal{X}}(\mathcal{S})$ is the smallest radius ϵ such that the union of all ϵ radius closed balls with their center in $\mathcal{S}$ covers $\mathcal{X}$.



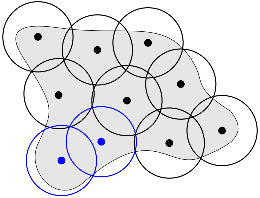
Refining

- Let $\mathcal{S}$ and $\mathcal{S}'$ be samplings of $\mathcal{X}$. We say that $\mathcal{S}'$ refines $\mathcal{S}$ if and only if $\mathcal{S}'$ has a smaller dispersion than $\mathcal{S}$.
  (E.g by adding points in the biggest “holes” of S)
- A refining process is complete if when applyied infinitely many times, it returns a sampling with 0 dispersion.
  (Intuitively, an infinite number of points with no “holes”)

Refining

- Let $\mathcal{S}$ and $\mathcal{S}'$ be samplings of $\mathcal{X}$. We say that $\mathcal{S}'$ refines $\mathcal{S}$ if and only if $\mathcal{S}'$ has a smaller dispersion than $\mathcal{S}$. (E.g by adding points in the biggest "holes" of S)
- A refining process is complete if when applyied infinitely many times, it returns a sampling with 0 dispersion. (Intuitively, an infinite number of points with no "holes")

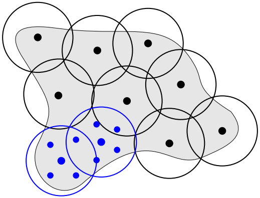
- If $\mathcal{S}$ is a sampling of $\mathcal{X}$ with dispersion ϵ. We say that $\mathcal{S}'$ refines locally $\mathcal{S}$ if there exist a subset $\mathcal{T}$ of $\mathcal{S}$ such that $\mathcal{S}' \cap \mathcal{B}_\epsilon(\mathcal{T})$ refines $\mathcal{T}$ in $\mathcal{B}_\epsilon(\mathcal{T})$.



Refining

- Let $\mathcal{S}$ and $\mathcal{S}'$ be samplings of $\mathcal{X}$. We say that $\mathcal{S}'$ refines $\mathcal{S}$ if and only if $\mathcal{S}'$ has a smaller dispersion than $\mathcal{S}$.
(E.g by adding points in the biggest “holes” of S)
- A refining process is complete if when applyied infinitely many times, it returns a sampling with 0 dispersion.
(Intuitively, an infinite number of points with no “holes”)

- If $\mathcal{S}$ is a sampling of $\mathcal{X}$ with dispersion ϵ. We say that $\mathcal{S}'$ refines locally $\mathcal{S}$ if there exist a subset $\mathcal{T}$ of $\mathcal{S}$ such that $\mathcal{S}' \cap \mathcal{B}_\epsilon(\mathcal{T})$ refines $\mathcal{T}$ in $\mathcal{B}_\epsilon(\mathcal{T})$.

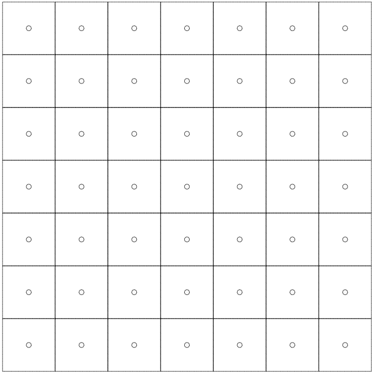


Application to Cubes

Assume that we want a sampling of $\mathcal{X}$ with a given dispersion ϵ.
What is the minimum number of points needed ?

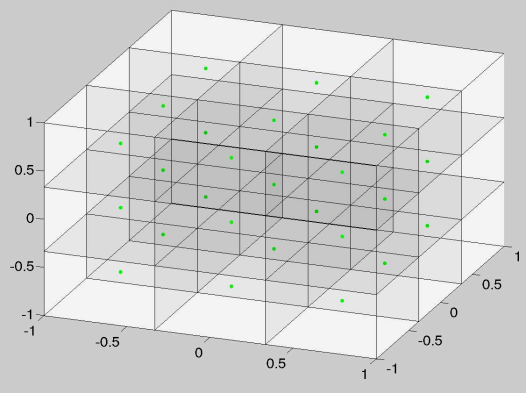
Application to Cubes

Assume that we want a sampling of $\mathcal{X}$ with a given dispersion ϵ.
What is the minimum number of points needed ?
If $\mathcal{X}$ is a cube and dispersion is defined with $\|.\|_\infty$, the solution is intuitive (known as Sukharev grids):



Application to Cubes

Assume that we want a sampling of $\mathcal{X}$ with a given dispersion ϵ.
What is the minimum number of points needed ?
If $\mathcal{X}$ is a cube and dispersion is defined with $\|.\|_\infty$, the solution is intuitive (known as Sukharev grids):

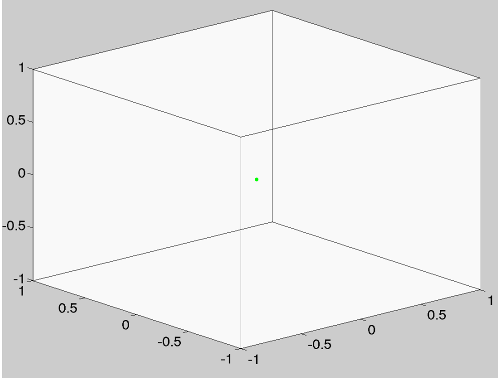


Incremental and Hierarchical Sampling

Assume that we distribute points sequentially. How do we decrease dispersion as fast as possible ?

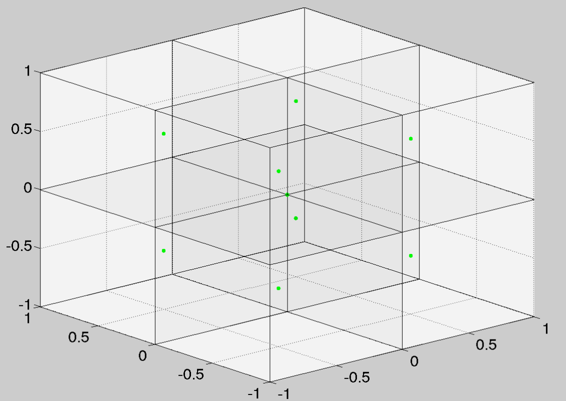
Incremental and Hierarchical Sampling

Assume that we distribute points sequentially. How do we decrease dispersion as fast as possible ?
A simple yet good solution is hierarchical sampling.



Incremental and Hierarchical Sampling

Assume that we distribute points sequentially. How do we decrease dispersion as fast as possible ?

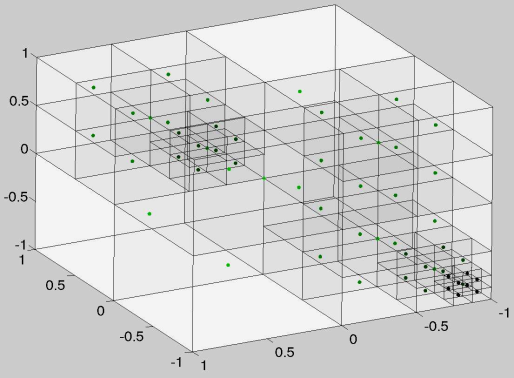


Local Hierarchical Sampling

Hierarchical sampling is clearly complete and also suitable for local refining.

Local Hierarchical Sampling

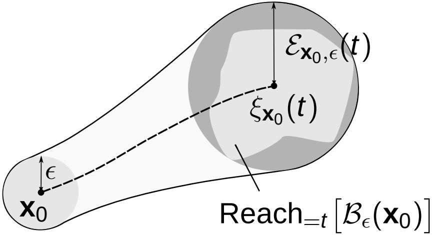
Hierarchical sampling is clearly complete and also suitable for local refining.



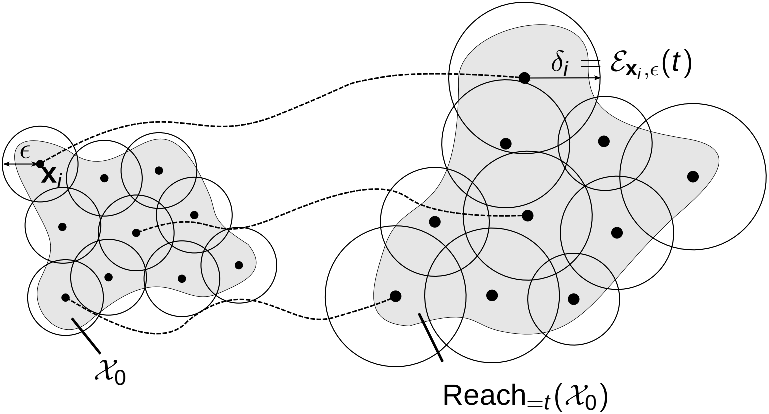
Definition

Given $\mathbf{x}_0 \in \mathcal{X}_0$, and $\epsilon > 0$, the expansion function of $\xi_{\mathbf{x}_0}$, denoted by $\mathcal{E}_{\mathbf{x}_0,\epsilon} : \mathbb{R}^+ \mapsto \mathbb{R}^+$ maps t to $\mathcal{E}_{\mathbf{x}_0,\epsilon}(t)$ such that:

$$\mathcal{E}_{\mathbf{x}_0,\epsilon}(t) = \sup_{d(\mathbf{x}_0,\mathbf{x})\leq\epsilon} d\left(\xi_{\mathbf{x}_0}(t), \xi_{\mathbf{x}}(t)\right)$$



If $\bigcup_{i=1}^{k} \mathcal{B}_\epsilon(\mathbf{x}_i)$ covers $\mathcal{X}_0$ then $\bigcup_{i=1}^{k} \mathcal{B}_{\delta_i}(\mathbf{x}_i)$ where $\delta_i = \mathcal{E}_{\mathbf{x}_0,\epsilon}(t)$ is a cover of $\text{Reach}_{=t}(\mathcal{X}_0)$.



Interest: if $\bigcup_{i=1}^{k} \mathcal{B}_{\delta_i}(\mathbf{x}_i)$ does not intersect $\mathcal{F}$ for all t, then clearly the system is safe.

Sensitivity to Initial Conditions

- We call sensitivity matrix the derivative of flow $\xi_{\mathbf{x}_0}$ w.r.t. $\mathbf{x}_0$:

$$\mathbf{s}_{\mathbf{x}_0}(t) \quad \triangleq \quad \frac{\partial \xi_{\mathbf{x}_0}}{\partial \mathbf{x}_0}(t).$$

Sensitivity to Initial Conditions

- We call sensitivity matrix the derivative of flow $\xi_{\mathbf{x}_0}$ w.r.t. $\mathbf{x}_0$:

$$\mathbf{s}_{\mathbf{x}_0}(t) \triangleq \frac{\partial \xi_{\mathbf{x}_0}}{\partial \mathbf{x}_0}(t).$$

- It satisfies:

$$\dot{\mathbf{s}}_{\mathbf{x}_0}(t) = J(f)_{|\mathbf{x}(t)}\ \mathbf{s}_{\mathbf{x}_0}(t)$$
$$\mathbf{s}_{\mathbf{x}_0}(0) = \mathbf{I}_n$$

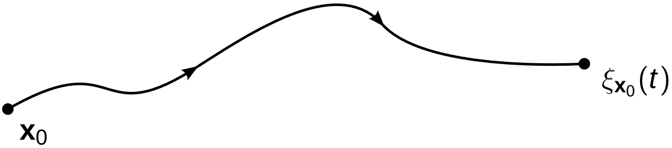
which is an ODE that can be solved together with the dynamics of the system.

Sensitivity to Initial Conditions (cont'd)

Say that $\mathbf{s}_{\mathbf{x}_0}(t) = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix}$

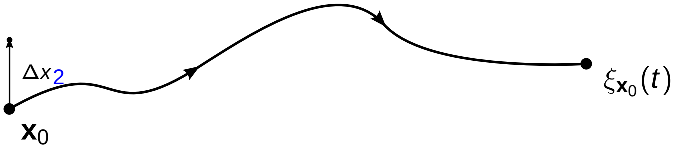
Sensitivity to Initial Conditions (cont'd)

$$\text{Say that } \mathbf{s}_{\mathbf{x}_0}(t) = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix}$$



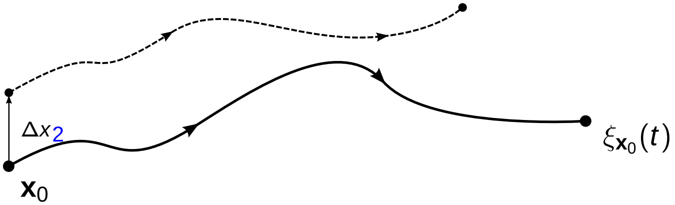
Sensitivity to Initial Conditions (cont'd)

$$\text{Say that } \mathbf{s}_{\mathbf{x}_0}(t) = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix}$$



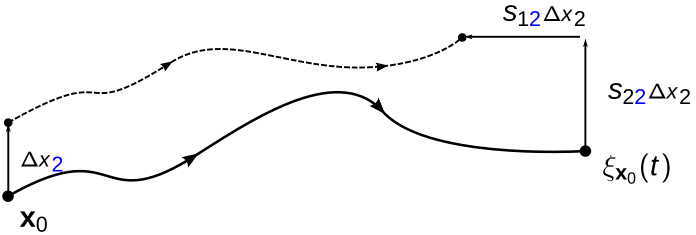
Sensitivity to Initial Conditions (cont'd)

Say that $\mathbf{s}_{\mathbf{x}_0}(t) = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix}$



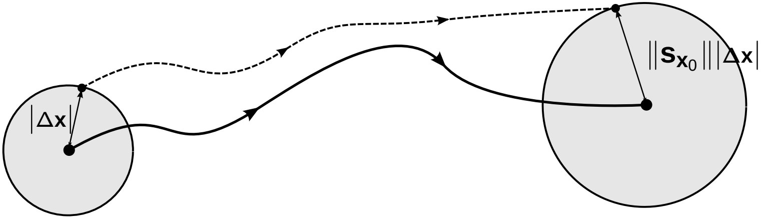
Sensitivity to Initial Conditions (cont'd)

$$\text{Say that } \mathbf{s}_{\mathbf{x}_0}(t) = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix}$$



Sensitivity to Initial Conditions (cont'd)

$$\text{Say that } \mathbf{s}_{\mathbf{x}_0}(t) = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix}$$



Main Result: Expansion and Sensitivity

Sensitivity provides a first order approximation of expansion function (w.r.t. initial dispersion ϵ):

Proposition

If f is C^2 and $\mathcal{X}_0$ compact, there exists a real $M > 0$ such that for all $\mathbf{x}_0 \in \mathcal{X}_0$, $t \in [0, T]$ and $\epsilon > 0$:

$$|\mathcal{E}_{\mathbf{x}_0,\epsilon}(t) - \|\mathbf{s}_{\mathbf{x}_0}(t)\| \, \epsilon \,| \quad \leq \quad M\epsilon^2$$

Main Result: Expansion and Sensitivity

Sensitivity provides a first order approximation of expansion function (w.r.t. initial dispersion ϵ):

Proposition

If f is C^2 and $\mathcal{X}_0$ compact, there exists a real $M > 0$ such that for all $\mathbf{x}_0 \in \mathcal{X}_0$, $t \in [0, T]$ and $\epsilon > 0$:

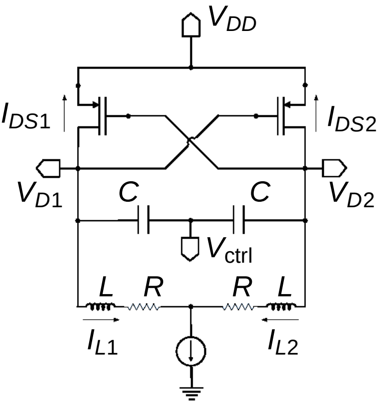
$$|\mathcal{E}_{\mathbf{x}_0,\epsilon}(t) - \|\mathbf{s}_{\mathbf{x}_0}(t)\| \, \epsilon \,| \quad \leq \quad M\epsilon^2$$

When f is affine, i.e. $f(t, \mathbf{x}) = A(t)\mathbf{x} + b(t)$, then expansion function can be computed exactly.

$$\mathcal{E}_{\mathbf{x}_0,\epsilon}(t) = \|\mathbf{s}_{\mathbf{x}_0}(t)\|\epsilon$$

First Application Example

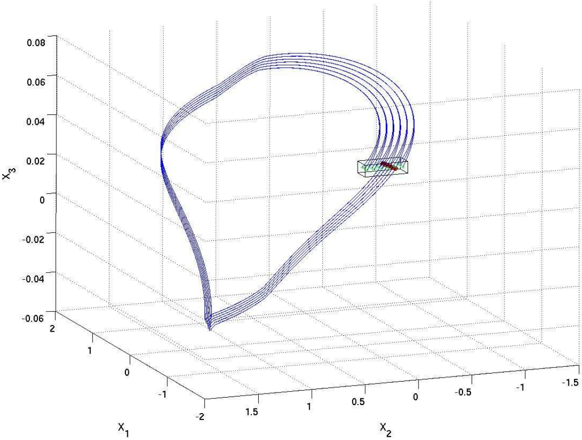
- We studyied the dynamics of an oscillating circuit: the Voltage Controlled Oscillator.
- Non linear circuit with three continuous state variables.



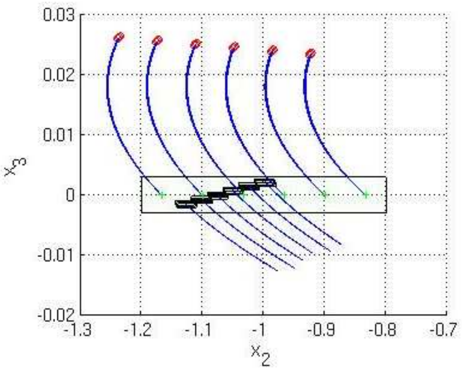
VCO schema

Verifying Oscillations

The goal is to analyse stability of oscillations



Verifying Oscillations



Estimation is experimentally good. A formal result requires further analysis.

⇒ Intermediate method between testing and pure formal verification

A Verification Algorithm

We come back to the case where a bad set $\mathcal{F}$ is present.

Input: $\delta > 0$, dispersion of sampling trajectories.
Output:

- `safe` if the system is found to be safe
- (`unsafe`, $\{\mathbf{x}\}$) if $\{\mathbf{x}\}$ is found a counter-example, i.e.: ξ_x intersects $\mathcal{F}$.
- (`uncertain`, $\mathcal{U}$) else. Then all the points in $\mathcal{U}$ induce uncertain trajectories: $\forall\, \mathbf{x} \in \mathcal{U},\ d(\xi_{\mathbf{x}}, \mathcal{F}) \leq \delta$.

A Verification Algorithm

We come back to the case where a bad set $\mathcal{F}$ is present.

Input: $\delta > 0$, dispersion of sampling trajectories.
Output:

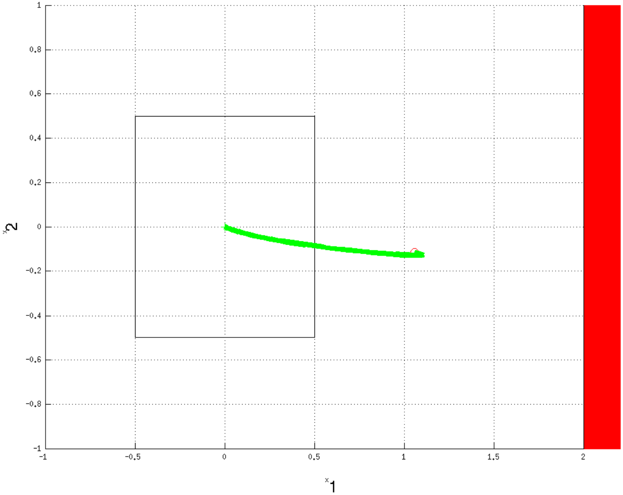
- `safe` if the system is found to be safe
- (`unsafe`, $\{\mathbf{x}\}$) if $\{\mathbf{x}\}$ is found a counter-example, i.e.: ξ_x intersects $\mathcal{F}$.
- (`uncertain`, $\mathcal{U}$) else. Then all the points in $\mathcal{U}$ induce uncertain trajectories: $\forall\, \mathbf{x} \in \mathcal{U},\ d(\xi_{\mathbf{x}}, \mathcal{F}) \leq \delta$.

In the last case, try again with smaller δ.

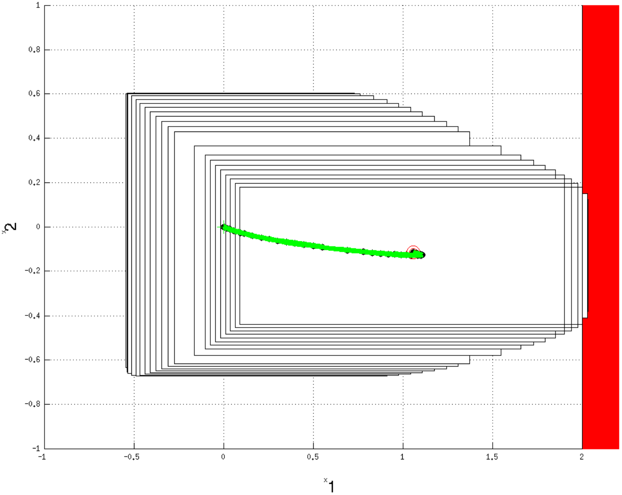
Theorem

If $d\left(Reach_{\leq T}(\mathcal{X}_0), \mathcal{F}\right) > 0$, then there exist a $\delta > 0$ for which the algorithm returns `safe`.

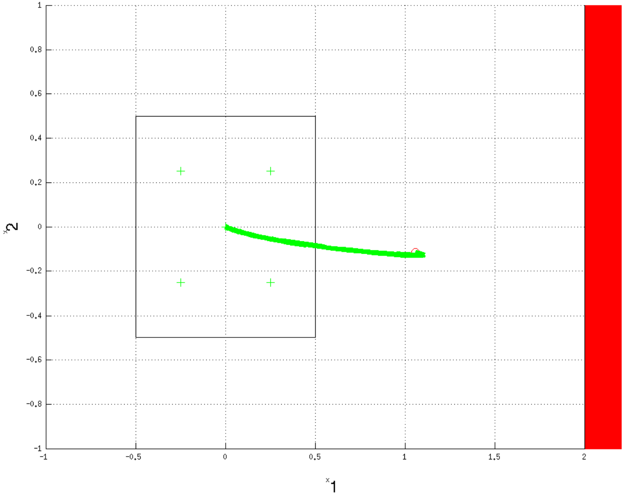
Algorithm steps



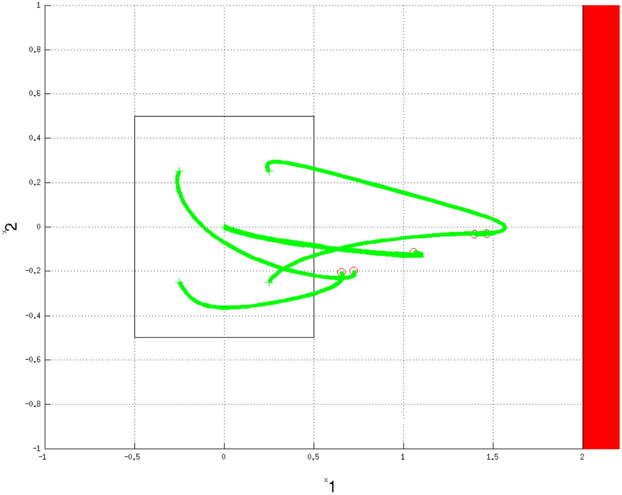
Algorithm steps



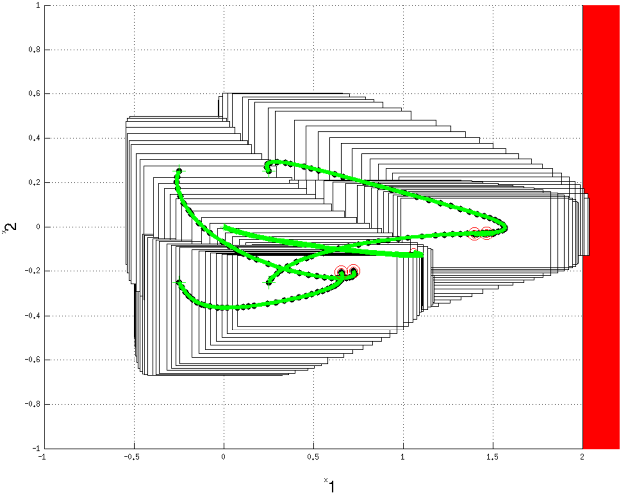
Algorithm steps



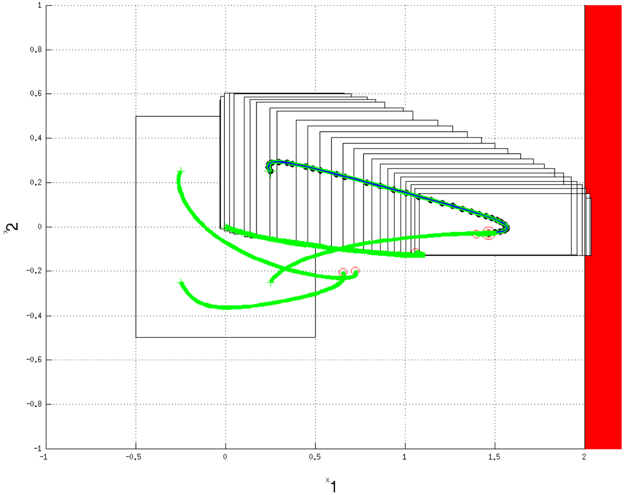
Algorithm steps



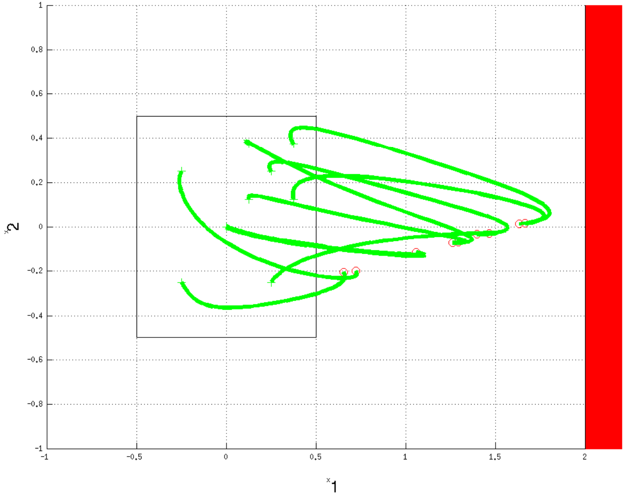
Algorithm steps



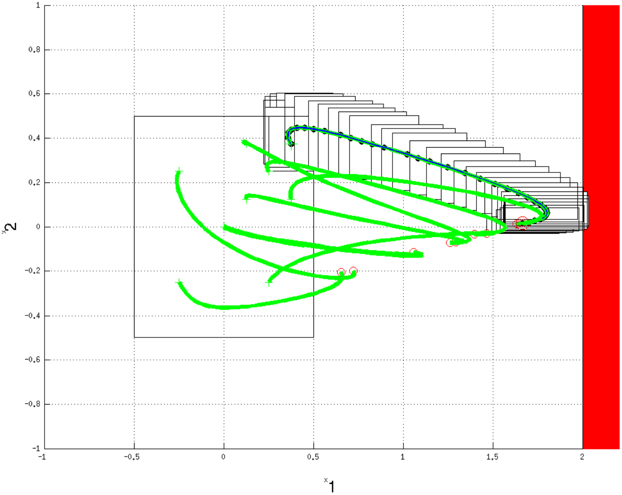
Algorithm steps



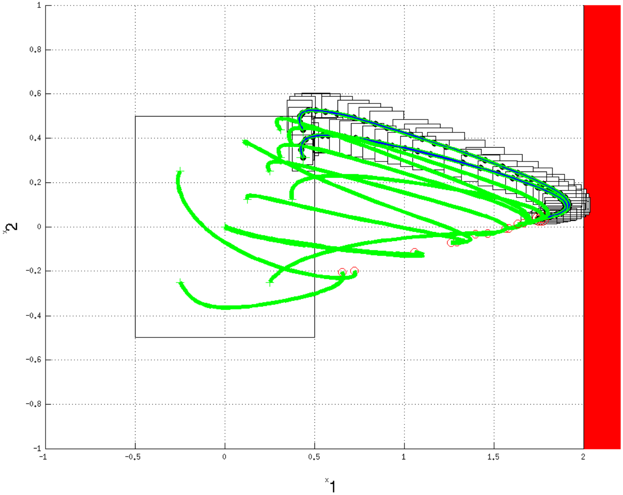
Algorithm steps



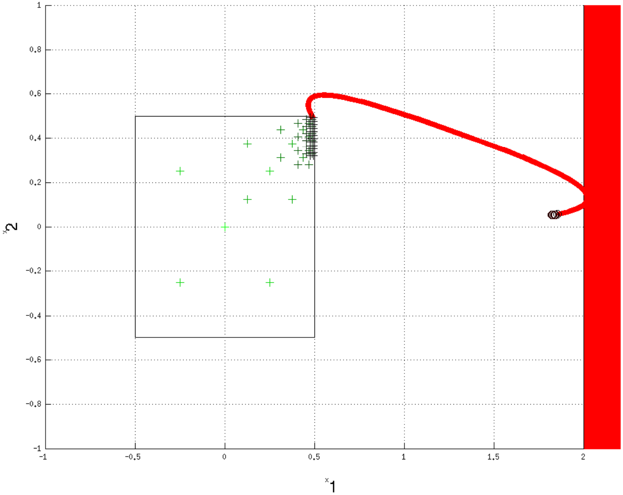
Algorithm steps



Algorithm steps



Algorithm steps



High Dimensional Affine Systems

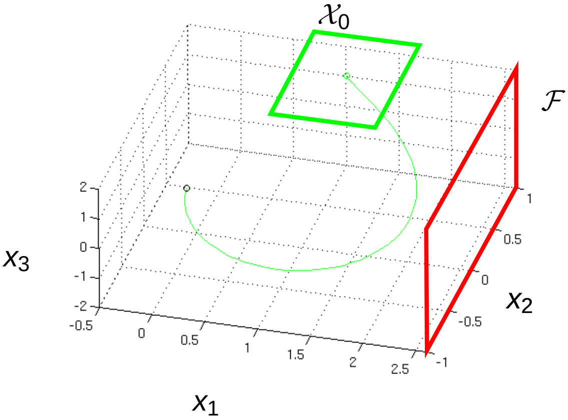
We considered the following affine, time varying dynamics:

$$\dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x} + \mathbf{b}(t)$$

$$\text{with } \mathbf{A}(t) = e^{-t}\mathbf{M} - \mathbf{I}_{50} \text{ and } \mathbf{b}(t) = \mathbf{b}_0 e^{-t} \sin t$$

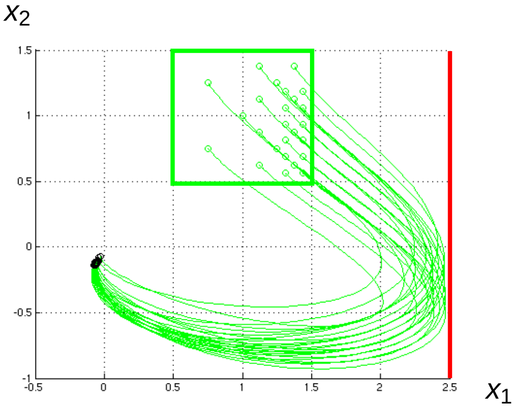
where $\mathbf{M}$ and $\mathbf{b}_0$ are respectively 50×50 and 50×1 matrices chosen randomly.
We used a 2-dimensional $\mathcal{X}_0$
The bad set $\mathcal{F}$ is the half plane given by an inequality of the form $x_1 \leq d$.

High Dimensional Affine Systems (cont'd)



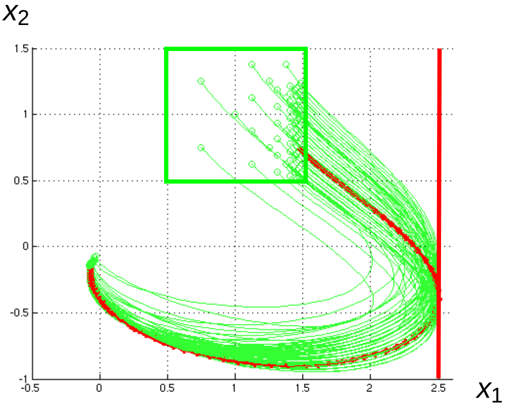
$d = 2.6$: One trajectory was enough to prove that the system is safe.

High Dimensional Affine Systems (cont'd)



$d = 2.5$: The system is declared uncertain using $\delta = 0.1$ after 25 trajectories.

High Dimensional Affine Systems (cont'd)



$d = 2.5$: The system was found unsafe with $\delta = 0.01$ after 63 trajectories.

Conclusion for Affine Time Varying Systems

- Computation time depends on the number of simulations, thus mainly on the dimension of $\mathcal{X}_0$
- We fixed the number of dimension of $\mathcal{X}_0$ to 3 and increased system dimension assuming that 64 simulations were needed[1]:

Nb of dimensions	50	100	150	200	250	300
Computation Time	3s	12s	30s	60s	100s	160s

[1] We used CVodes from Lawrence Livermore National Laboratory

Conclusion for Affine Time Varying Systems

- Computation time depends on the number of simulations, thus mainly on the dimension of $\mathcal{X}_0$
- We fixed the number of dimension of $\mathcal{X}_0$ to 3 and increased system dimension assuming that 64 simulations were needed[1]:

Nb of dimensions	50	100	150	200	250	300
Computation Time	3s	12s	30s	60s	100s	160s

[1]We used CVodes from Lawrence Livermore National Laboratory

Conclusion for Affine Time Varying Systems

- Computation time depends on the number of simulations, thus mainly on the dimension of $\mathcal{X}_0$
- We fixed the number of dimension of $\mathcal{X}_0$ to 3 and increased system dimension assuming that 64 simulations were needed[1]:

Nb of dimensions	50	100	150	200	250	300
Computation Time	3s	12s	30s	60s	100s	160s

- Further, a simulation with 2000 variables took 320s on a recent laptop.

Then we can argue that a 2000-dimensional LTV system can be verified in a few minutes.

[1] We used CVodes from Lawrence Livermore National Laboratory

Extension to Hybrid Systems

We consider a system with a switching hypersurface:

$$\dot{\mathbf{x}} = \begin{cases} f_1(t,\mathbf{x}) & \text{if} \quad g(\mathbf{x}) < 0 \\ f_2(t,\mathbf{x}) & \text{if} \quad g(\mathbf{x}) \geq 0 \end{cases}, \mathbf{x}(0) \in \mathcal{X}_0$$

At a non tangent crossing point, $\mathbf{s}_{\mathbf{x}_0}$ is discontinuous:

$$\mathbf{s}(\tau^+) - \mathbf{s}(\tau^-) = \frac{d\tau}{d\mathbf{x}_0}\Big(f_2(\tau, \xi_{\mathbf{x}_0}(\tau)) - f_1(\tau, \xi_{\mathbf{x}_0}(\tau))\Big)$$

$$\text{where } \frac{d\tau}{d\mathbf{x}_0} = \frac{\langle \nabla_{\mathbf{x}} g(\xi_{\mathbf{x}_0}(\tau)), \mathbf{s}_{\mathbf{x}_0}(\tau)\rangle}{\langle \nabla_{\mathbf{x}} g(\xi_{\mathbf{x}_0}(\tau)), f_1(\tau, \xi_{\mathbf{x}_0})\rangle}$$

Extension to Hybrid Systems

We consider a system with a switching hypersurface:

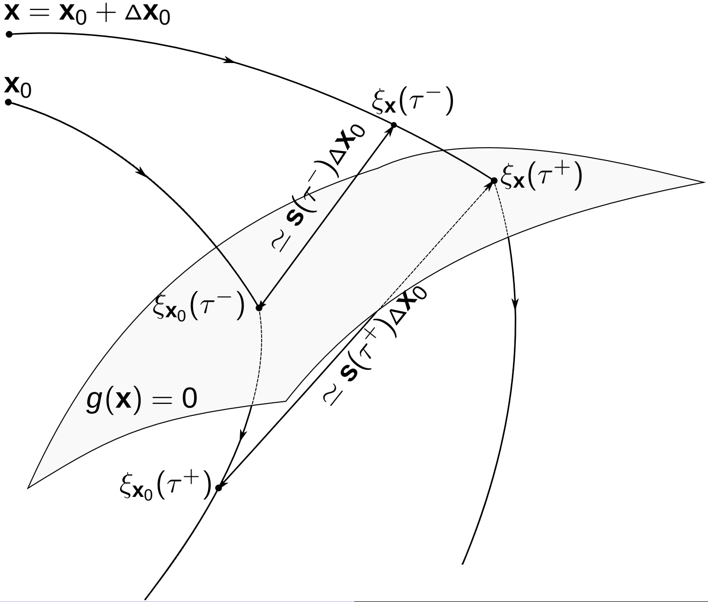
$$\dot{\mathbf{x}} = \begin{cases} f_1(t,\mathbf{x}) & \text{if} \quad g(\mathbf{x}) < 0 \\ f_2(t,\mathbf{x}) & \text{if} \quad g(\mathbf{x}) \geq 0 \end{cases}, \mathbf{x}(0) \in \mathcal{X}_0$$

At a non tangent crossing point, $\mathbf{s}_{\mathbf{x}_0}$ is discontinuous:

$$\begin{aligned} \mathbf{s}(\tau^+) - \mathbf{s}(\tau^-) &= \frac{d\tau}{d\mathbf{x}_0}\left(f_2(\tau,\xi_{\mathbf{x}_0}(\tau)) - f_1(\tau,\xi_{\mathbf{x}_0}(\tau))\right) \\ \text{where } \frac{d\tau}{d\mathbf{x}_0} &= \frac{\langle \nabla_{\mathbf{x}} g(\xi_{\mathbf{x}_0}(\tau)), \mathbf{s}_{\mathbf{x}_0}(\tau)\rangle}{\langle \nabla_{\mathbf{x}} g(\xi_{\mathbf{x}_0}(\tau)), f_1(\tau,\xi_{\mathbf{x}_0})\rangle} \end{aligned}$$

- Sensitivity can be computed for more general systems (work by P.I. Barton et al, I.A. Hiskens et al).
- An important issue is “grazing” (tangential crossing)

Sensitivity Jump at Hypersurface Crossing



Conclusion

- Work and easy to implement for general linear, nonlinear, and hybrid systems
- Intermediate method between testing and formal verification
- Formal for LTV systems (assuming exact simulation of the system)
- Potentially applicable for high dimension systems

On going work and dicussion:

- Subject to the curse of dimensionnality w.r.t $\mathcal{X}_0$ (even if hierarchicall sampling helps)
- Empirical tests with hybrid systems
- Extension to Systems with inputs

Thank You For Your Attention