Computing the Viability Kernel Using Maximal Reachable Sets

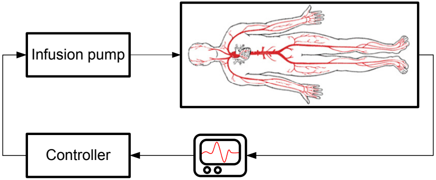
Shahab Kaynama, John Maidens, Meeko Oishi,
Ian M. Mitchell, Guy A. Dumont

University of British Columbia

www.ece.ubc.ca/∼kaynama

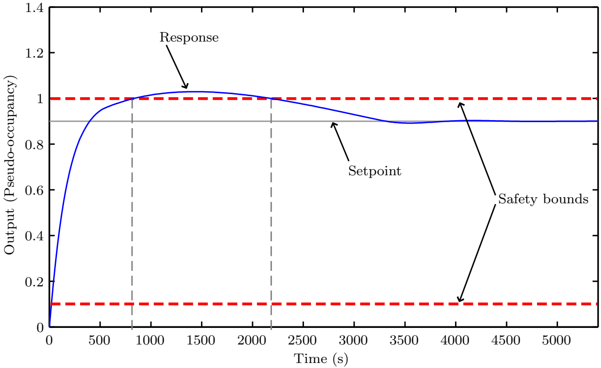
Motivation: Control of Anesthesia

- Control of depth of anesthesia
  - [Simanski, et al. 07; Ionescu, et al. 08; Syafiie, et al. 09; Dumont, et al. 09; Oliveira, et al. 09; Mendonca, et al. 09]
- Goal: closed-loop drug delivery system
  - Currently bolus-based open-loop system, 80 patients via clinical trials
- Key element for FDA/Health Canada: guarantees of **safety**



Motivation: Control of Anesthesia

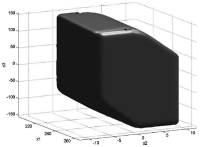
- Insufficient constraints for the receding horizon optimization



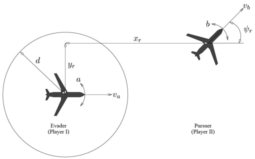
$$y \in [0.1, 1], \quad u \in [0, 0.8]$$

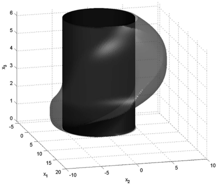
Motivation: Vehicle Safety, etc.

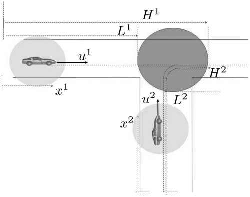
- Flight Envelope Protection [Image: Lygeros 04]



- Collision Avoidance [Images: Mitchell, et al. 05; Hafner and Del Vecchio 11]







Outline

Outline

Introduction

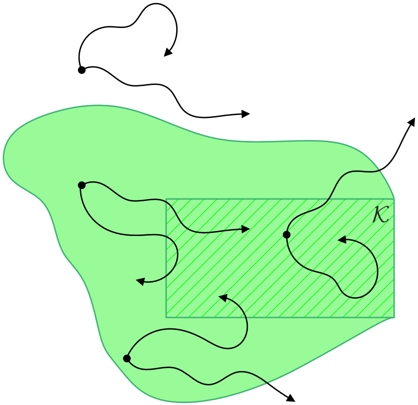
$$\begin{cases} \dot{x} = f(x,u), \quad x(0) = x_0, \quad t \in [0,\tau] =: \mathbb{T} \\ u(t) \in \mathcal{U} \qquad \text{(input constraint)} \\ \mathcal{K} \subset \mathcal{X} \qquad \text{(target set/state constraint)} \end{cases}$$

- Reachability analysis
  - [Tomlin, et al. 03; Aubin, et al. 11; Kurzhanski and Varaiya 00; Lygeros 04; Blanchini and Miani 08; ...]
  - Maximal vs. minimal reachability [Mitchell 07]

Introduction

- Maximal reach tube

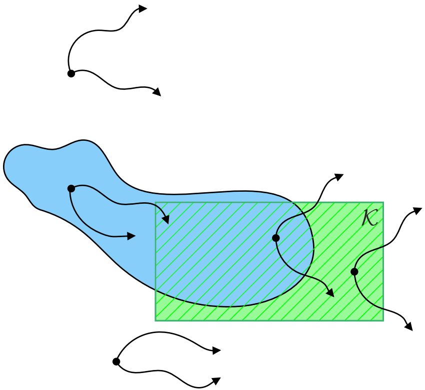
$$Reach^{\sharp}_{\mathbb{T}}(\mathcal{K},\mathcal{U}) := \{x_0 \in \mathcal{X} \mid \exists u(\cdot),\ \exists t,\ x(t) \in \mathcal{K}\}$$



Introduction

- Maximal reach set

$$Reach_t^{\sharp}(\mathcal{K},\mathcal{U}) := \{x_0 \in \mathcal{X} \mid \exists u(\cdot),\ x(t) \in \mathcal{K}\}$$



Introduction

- Maximal reach tube vs. set [Lygeros 04; Mitchell 07]

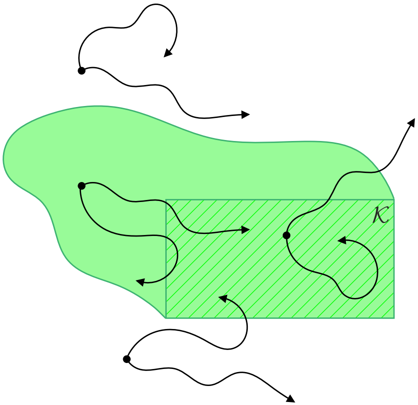
$$Reach^{\sharp}_{\mathbb{T}}(\mathcal{K},\mathcal{U}) = \bigcup_{t\in\mathbb{T}} Reach^{\sharp}_{t}(\mathcal{K},\mathcal{U})$$

- Lagrangian methods to approximate
  - e.g. [Frehse, et al. 11; Girard and Le Guernic 08; Girard, et al. 06; Han and Krogh 06; Kurzhanski and Varaiya 00; Kurzhanskiy and Varaiya 06]
  - Scalable and computationally efficient (polynomial)

Introduction

- Minimal reach tube

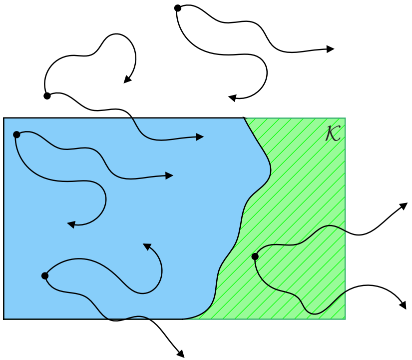
$$Reach^{\flat}_{\mathbb{T}}(\mathcal{K},\mathcal{U}) := \{x_0 \in \mathcal{X} \mid \forall u(\cdot),\ \exists t,\ x(t) \in \mathcal{K}\}$$



Introduction

- Viability kernel (finite horizon)

$$Viab_{\mathbb{T}}(\mathcal{K},\mathcal{U}) := \{x_0 \in \mathcal{X} \mid \exists u(\cdot),\ \forall t,\ x(t) \in \mathcal{K}\}$$



- Infinite horizon viab kernel $\equiv$ maximal controlled-invariant subset

Introduction

- Viability kernel vs. minimal reach tube [Cardaliaguet, et al. 99]

$$(Viab_{\mathbb{T}}(\mathcal{K},\mathcal{U}))^c = Reach^{\flat}_{\mathbb{T}}(\mathcal{K}^c,\mathcal{U})$$

- The **only** constructs to prove existence of safety control laws [Mitchell 07; Lygeros 04]
  - Applications: [Lygeros, et al. 98; Tomlin, et al. 03; Bayen, et al. 07; Gillula, et al. 10; Oishi, et al. 03; Aswani, et al. 11; Borrelli, et al. 10; Panagou, et al. 09; Del Vecchio, et al. 09; Ghaemi and Del Vecchio 11; ...]
- Eulerian methods to approximate
  - [Mitchell, et al. 05; Cardaliaguet, et al. 99; Gao, et al. 06; Saint-Pierre 94]
  - Computationally intensive (exponential) since grid-based

Problem Statement and Methodology

- Desirable to compute $Viab_{\mathbb{T}}(\mathcal{K}, \mathcal{U})$ (or $Reach^{\flat}_{\mathbb{T}}(\mathcal{K}, \mathcal{U})$) for high-dimensional systems for analysis and synthesis
- How to tackle the **"curse of dimensionality"**?

- Existing methods:
  - Hamilton-Jacobi projections [Mitchell and Tomlin 03]
  - Structure decomposition [Kaynama and Oishi IJC'11; Kaynama and Oishi TAC(ca); Mitchell 11; Stipanović, et al. 03]
- Proposed method:
  - A Lagrangian approach [this paper]

Problem Statement and Methodology

- Desirable to compute $Viab_{\mathbb{T}}(\mathcal{K}, \mathcal{U})$ (or $Reach^{\flat}_{\mathbb{T}}(\mathcal{K}, \mathcal{U})$) for high-dimensional systems for analysis and synthesis
- How to tackle the **"curse of dimensionality"**?

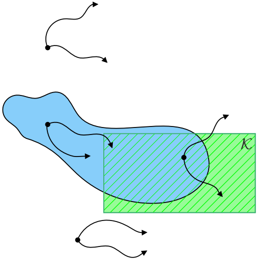
- Existing methods:
  - Hamilton-Jacobi projections [Mitchell and Tomlin 03]
  - Structure decomposition [Kaynama and Oishi IJC'11; Kaynama and Oishi TAC(ca); Mitchell 11; Stipanović, et al. 03]
- Proposed method:
  - A Lagrangian approach [this paper]

Outline

Lagrangian Approach

- Efficient techniques (Lagrangian) to compute maximal reach sets

$$Reach_t^\sharp(\mathcal{K},\mathcal{U}) := \{x_0 \in \mathcal{X} \mid \exists u(\cdot),\ x(t) \in \mathcal{K}\}$$



- Approximate $Viab_{\mathbb{T}}(\mathcal{K},\mathcal{U})$ via a nested sequence of sets reachable in small sub-time intervals of $\mathbb{T}$

Lagrangian Approach: Continuous-Time

- Start with an under-approximation $\mathcal{K}_{\downarrow}(P)$ of $\mathcal{K}$ (P: interval partition; M: uniform bound on f)

$$\mathcal{K}_{\downarrow}(P) := \{x \in \mathcal{K} \mid \mathrm{dist}(x, \mathcal{K}^c) \geq M\|P\|\}$$

- Recursively compute $K_0(P)$ from:

$$\begin{aligned} K_{|P|}(P) &= \mathcal{K}_{\downarrow}(P), \\ K_{k-1}(P) &= \mathcal{K}_{\downarrow}(P) \cap Reach^{\sharp}_{t_k - t_{k-1}}(K_k(P), \mathcal{U}) \\ &\text{for} \quad k \in \{1, \ldots, |P|\}. \end{aligned}$$

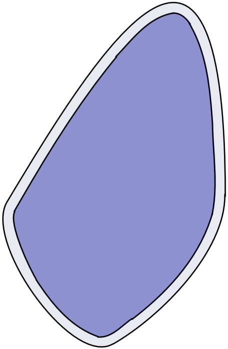
Lagrangian Approach: Continuous-Time

- Start with an under-approximation $\mathcal{K}_{\downarrow}(P)$ of $\mathcal{K}$ (P: interval partition; M: uniform bound on f)

$$\mathcal{K}_{\downarrow}(P) := \{x \in \mathcal{K} \mid \mathrm{dist}(x, \mathcal{K}^c) \geq M\|P\|\}$$

- Recursively compute $K_0(P)$ from:

$$K_{|P|}(P) = \mathcal{K}_{\downarrow}(P),$$
$$K_{k-1}(P) = \mathcal{K}_{\downarrow}(P) \cap Reach^{\sharp}_{t_k - t_{k-1}}(K_k(P), \mathcal{U})$$
$$\text{for} \quad k \in \{1, \ldots, |P|\}.$$



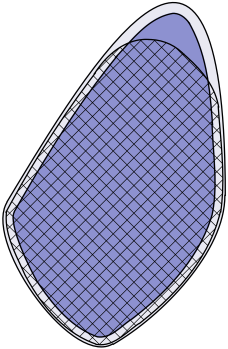
Lagrangian Approach: Continuous-Time

- Start with an under-approximation $\mathcal{K}_{\downarrow}(P)$ of $\mathcal{K}$ (P: interval partition; M: uniform bound on f)

$$\mathcal{K}_{\downarrow}(P) := \{x \in \mathcal{K} \mid \mathrm{dist}(x, \mathcal{K}^c) \geq M\|P\|\}$$

- Recursively compute $K_0(P)$ from:

$$K_{|P|}(P) = \mathcal{K}_{\downarrow}(P),$$
$$K_{k-1}(P) = \mathcal{K}_{\downarrow}(P) \cap Reach^{\sharp}_{t_k - t_{k-1}}(K_k(P), \mathcal{U})$$
$$\text{for} \quad k \in \{1, \ldots, |P|\}.$$



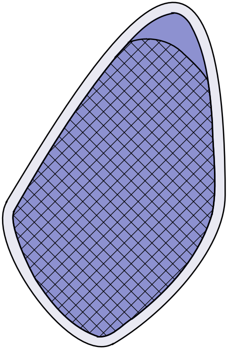
Lagrangian Approach: Continuous-Time

- Start with an under-approximation $\mathcal{K}_{\downarrow}(P)$ of $\mathcal{K}$ (P: interval partition; M: uniform bound on f)

$$\mathcal{K}_{\downarrow}(P) := \{x \in \mathcal{K} \mid \operatorname{dist}(x, \mathcal{K}^c) \geq M\|P\|\}$$

- Recursively compute $K_0(P)$ from:

$$K_{|P|}(P) = \mathcal{K}_{\downarrow}(P),$$
$$K_{k-1}(P) = \mathcal{K}_{\downarrow}(P) \cap Reach^{\sharp}_{t_k - t_{k-1}}(K_k(P), \mathcal{U})$$
$$\text{for} \quad k \in \{1, \dots, |P|\}.$$



Lagrangian Approach: Continuous-Time

- Start with an under-approximation $\mathcal{K}_{\downarrow}(P)$ of $\mathcal{K}$ (P: interval partition; M: uniform bound on f)

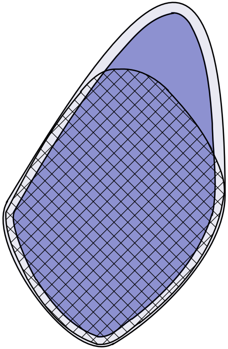
$$\mathcal{K}_{\downarrow}(P) := \{x \in \mathcal{K} \mid \mathrm{dist}(x, \mathcal{K}^c) \geq M\|P\|\}$$

- Recursively compute $K_0(P)$ from:

$$K_{|P|}(P) = \mathcal{K}_{\downarrow}(P),$$

$$K_{k-1}(P) = \mathcal{K}_{\downarrow}(P) \cap Reach^{\sharp}_{t_k - t_{k-1}}(K_k(P), \mathcal{U})$$

$$\text{for} \quad k \in \{1, \ldots, |P|\}.$$



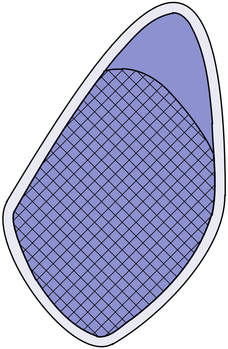
Lagrangian Approach: Continuous-Time

- Start with an under-approximation $\mathcal{K}_{\downarrow}(P)$ of $\mathcal{K}$ (P: interval partition; M: uniform bound on f)

$$\mathcal{K}_{\downarrow}(P) := \{x \in \mathcal{K} \mid \mathrm{dist}(x, \mathcal{K}^c) \geq M\|P\|\}$$

- Recursively compute $K_0(P)$ from:

$$K_{|P|}(P) = \mathcal{K}_{\downarrow}(P),$$
$$K_{k-1}(P) = \mathcal{K}_{\downarrow}(P) \cap Reach^{\sharp}_{t_k - t_{k-1}}(K_k(P), \mathcal{U})$$
$$\text{for} \quad k \in \{1, \ldots, |P|\}.$$



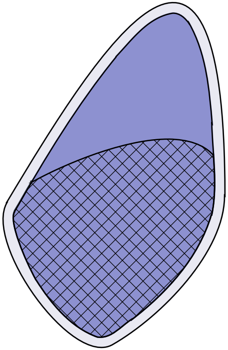
Lagrangian Approach: Continuous-Time

- Start with an under-approximation $\mathcal{K}_{\downarrow}(P)$ of $\mathcal{K}$ (P: interval partition; M: uniform bound on f)

$$\mathcal{K}_{\downarrow}(P) := \{x \in \mathcal{K} \mid \mathrm{dist}(x, \mathcal{K}^c) \geq M\|P\|\}$$

- Recursively compute $K_0(P)$ from:

$$K_{|P|}(P) = \mathcal{K}_{\downarrow}(P),$$
$$K_{k-1}(P) = \mathcal{K}_{\downarrow}(P) \cap Reach^{\sharp}_{t_k - t_{k-1}}(K_k(P), \mathcal{U})$$
$$\text{for} \quad k \in \{1, \ldots, |P|\}.$$



Lagrangian Approach: Continuous-Time

- Guaranteed under-approximation: $K_0(P) \subseteq Viab_{\mathbb{T}}(\mathcal{K}, \mathcal{U})$
- Arbitrarily precise by choosing a sufficiently fine partition:

$$Viab_{\mathbb{T}}(\mathring{\mathcal{K}}, \mathcal{U}) \subseteq \bigcup_{P \in \mathscr{P}(\mathbb{T})} K_0(P) \subseteq Viab_{\mathbb{T}}(\mathcal{K}, \mathcal{U})$$

Lagrangian Approach: Discrete-Time

- Particular form of the continuous-time case
- Recursively compute K_0 from:

$$K_n = \mathcal{K},$$
$$K_{k-1} = \mathcal{K} \cap Reach_1^\sharp(K_k, \mathcal{U})$$
$$\text{for} \quad k \in \{1, \ldots, n\}$$

- Compute exactly: $K_0 = Viab_{\mathbb{T} \cap \mathbb{Z}^+}(\mathcal{K}, \mathcal{U})$
- Closely related to discrete algorithms in e.g. [Saint-Pierre 94; Cardaliaguet, et al. 99; Blanchini and Miani 08]

Outline

Background and Introduction

Lagrangian Approach: Viability vs. Maximal Reachability

Computational Algorithms

Practical Examples

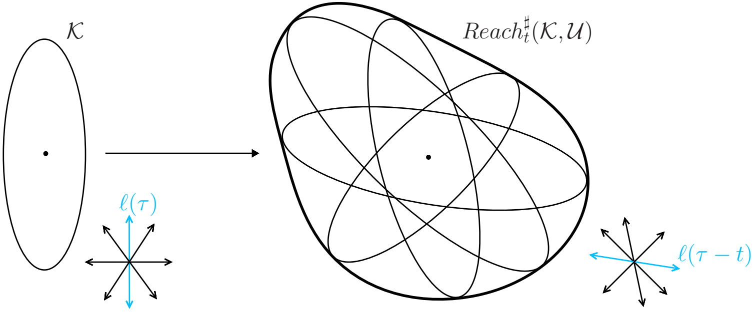
Conclusions and Future Work

Computational Algorithms

- Facilitates the use of scalable/efficient Lagrangian methods
- Ideally polytopes
  - But polytopic reach ([Kvasnica, et al. 04;...]) not scalable
  - Other techniques (e.g. zonotopes, ellipsoidal, support functions) not so easily convertible to polytopes (even if provide under-approximation)
- Piecewise ellipsoidal approach based on ellipsoidal techniques
  - Only deals with ellipsoids (fixed complexity)
  - Efficient, scalable, guaranteed under-approximation
  - Implementable in the Ellipsoidal Toolbox [Kurzhanskiy and Varaiya 06]
  - Generalizable to discriminating kernels
  - Safety-preserving control synthesis [in preparation]
  - Disadvantages: Loss of accuracy; Only LTI systems $\mathcal{L}(x) = Ax + Bu$

Computational Algorithms

Ellipsoidal techniques (under-)approximating the maximal reach set:



[Kurzhanski and Varaiya 00; Kurzhanski and Valyi 96]

Piecewise Ellipsoidal Algorithm (CT)

- Given $P \in \mathscr{P}(\mathbb{T})$ form an under-approximation $\mathcal{K}_{\downarrow}(P)$ of $\mathcal{K}$
- For a fixed terminal direction $\ell_\tau \in \mathcal{M}$ do the recursion

$$K_{k-1}^{*[\ell_\tau]} = \operatorname{maxvol}(K_{|P|}(P) \cap Reach_{t_k - t_{k-1}}^{\sharp[\ell_\tau]}(K_k^{*[\ell_\tau]}(P), \mathcal{U})) \quad \text{for} \quad k \in \{1, \ldots, |P|\}$$

with $K_{|P|}^{*[\ell_\tau]}(P) = K_{|P|}(P) = \mathcal{K}_{\downarrow}(P)$.

- Generates an **ellipsoidal** set $K_0^{*[\ell_\tau]}(P)$ such that

$$\bigcup_{\ell_\tau \in \mathcal{M}} K_0^{*[\ell_\tau]}(P) := K_0^*(P) \subseteq Viab_{\mathbb{T}}(\mathcal{K}, \mathcal{U})$$

Piecewise Ellipsoidal Algorithm (CT)

- Given $P \in \mathscr{P}(\mathbb{T})$ form an under-approximation $\mathcal{K}_{\downarrow}(P)$ of $\mathcal{K}$
- For a fixed terminal direction $\ell_\tau \in \mathcal{M}$ do the recursion

$$K_{k-1}^{*[\ell_\tau]} = \text{maxvol}(K_{|P|}(P) \cap Reach^{\sharp[\ell_\tau]}_{t_k - t_{k-1}}(K_k^{*[\ell_\tau]}(P), \mathcal{U}))$$

$$\text{for} \quad k \in \{1, \ldots, |P|\}$$

with $K_{|P|}^{*[\ell_\tau]}(P) = K_{|P|}(P) = \mathcal{K}_{\downarrow}(P)$.

- Generates an **ellipsoidal** set $K_0^{*[\ell_\tau]}(P)$ such that

$$\bigcup_{\ell_\tau \in \mathcal{M}} K_0^{*[\ell_\tau]}(P) := K_0^*(P) \subseteq Viab_{\mathbb{T}}(\mathcal{K}, \mathcal{U})$$

Piecewise Ellipsoidal Algorithm (CT): Loss of Accuracy

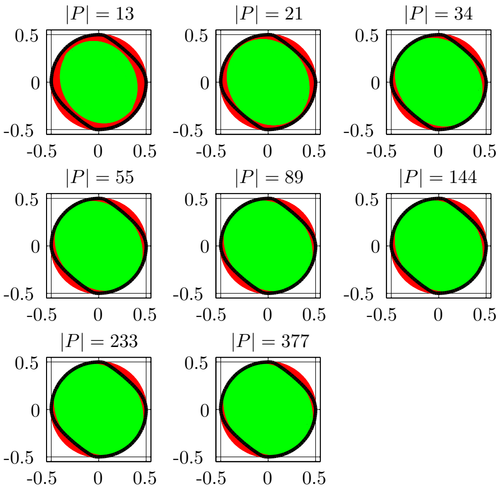
- Finer partition equals less accuracy loss (empirically)
- Simple example: The double integrator

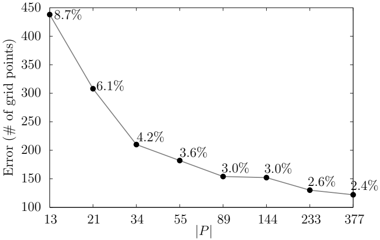
$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

subject to

$$u(t) \in \mathcal{U} := [-0.25, 0.25]$$
$$x(t) \in \mathcal{K} := \mathcal{E}(\mathbf{0}, \left[\begin{smallmatrix} 0.25 & 0 \\ 0 & 0.25 \end{smallmatrix}\right]), \quad \forall t \in [0, 1].$$

- Approximate with $|\mathcal{M}| = 10$ random directions





Error as a function of $|P|$.

[Produced via ET and LS Toolbox]

Outline

Applications: Flight Envelope Protection (CT, 4D)

- Longitudinal aircraft dynamics [Source: Bryson 94]

$$A = \begin{bmatrix} -0.003 & 0.039 & 0 & -0.322 \\ -0.065 & -0.319 & 7.740 & 0 \\ 0.020 & -0.101 & -0.429 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

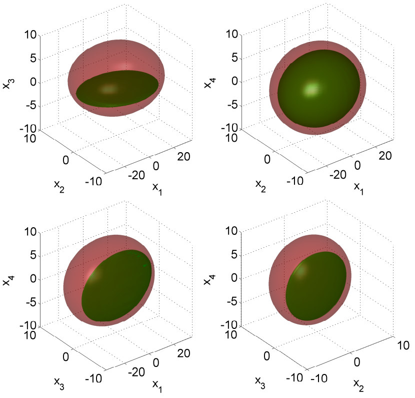
$$B = \begin{bmatrix} 0.010 & -0.180 & -1.160 & 0 \end{bmatrix}^T$$

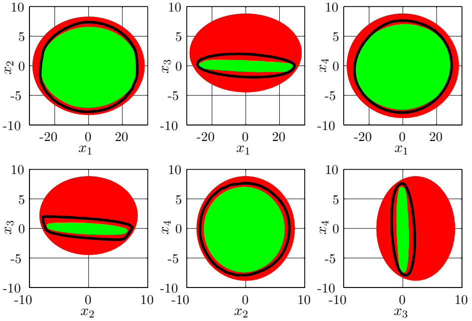
subject to

$$u(t) \in \mathcal{U} := [-13.3°, 13.3°],$$

$$x(t) \in \mathcal{K} := \mathcal{E}\left(\begin{bmatrix} 0 \\ 0 \\ 2.18 \\ 0 \end{bmatrix}, \begin{bmatrix} 1075.84 & 0 & 0 & 0 \\ 0 & 67.24 & 0 & 0 \\ 0 & 0 & 42.7716 & 0 \\ 0 & 0 & 0 & 76.0384 \end{bmatrix} \right), \quad \forall t \in [0, 2].$$

- We choose $|P| = 400$, $|\mathcal{M}| = 8$





Eulerian (Level-Set): 5.5 h
Piecewise Ellipsoidal: 10 m

[Produced via ET and LS Toolbox]

Applications: Safety in Anesthesia Automation (DT, 7D)

- Discrete-time Laguerre models (6D); patient's response to rocuronium
- Safety constraint: therapeutic bounds on output (pseudo-occupancy level), i.e. desired clinical effect
- Input constraint: actuator bounds (hard bounds on drug infusion rate)
- Compute viability kernel for a 30 min surgery (patient #80)

$$A = \begin{bmatrix} 0.9960 & 0 & 0 & 0 & 0 & 0 \\ 0.0080 & 0.9960 & 0 & 0 & 0 & 0 \\ -0.0080 & 0.0080 & 0.9960 & 0 & 0 & 0 \\ 0.0079 & -0.0080 & 0.0080 & 0.9960 & 0 & 0 \\ -0.0079 & 0.0079 & -0.0080 & 0.0080 & 0.9960 & 0 \\ 0.0079 & -0.0079 & 0.0079 & -0.0080 & 0.0080 & 0.9960 \end{bmatrix}$$

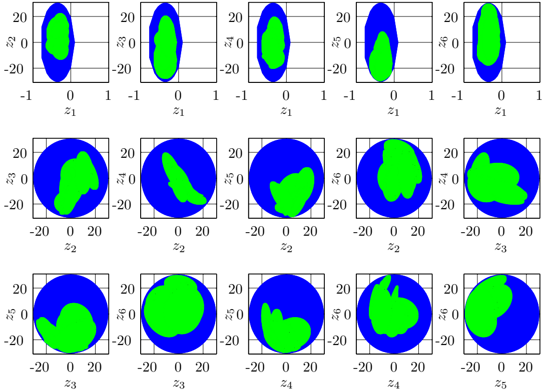
$$B = \begin{bmatrix} 0.0894 & -0.0890 & 0.0886 & -0.0883 & 0.0879 & -0.0876 \end{bmatrix}^T$$

$$C = \begin{bmatrix} 18.5000 & 8.2300 & 3.5300 & 4.3400 & 3.7000 & 3.0700 \end{bmatrix}$$

Applications: Safety in Anesthesia Automation (DT, 7D)

- Reformulate by projecting the output bounds onto the state space while making the control action regulatory.
- Dynamics are augmented and transformed to a coordinate system of dimension **seven**
- Compute for $|\mathcal{M}| = 30$ directions (15 resulted in non-empty ellipsoids)

Applications: Safety in Anesthesia Automation (DT, 7D)



Eulerian (Level-Set): ?
Piecewise Ellipsoidal: 15 m

[Produced via ET]

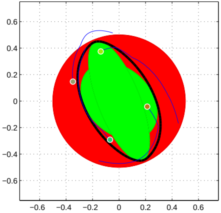
Outline

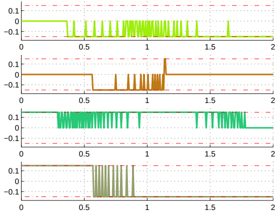
Conclusions

- Need to compute viability kernel for guarantees of safety
- Traditionally and exclusively computed using Eulerian methods (computationally intensive)
- Lagrangian methods (scalable/efficient) can compute maximal reach constructs
- Connection between viability kernel and maximal reach sets
- Enables the use of Lagrangian methods for viability kernel!
- Piecewise ellipsoidal algorithm based on ellipsoidal reach techniques
- Easily extendable to hybrid systems

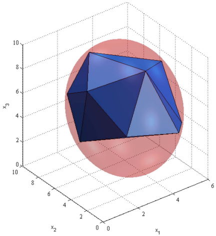
Future Work

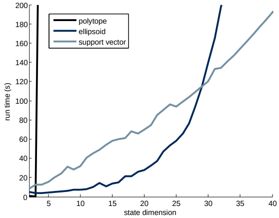
- Scalable synthesis of safety-preserving controllers





- Alternative algorithms w/o wrapping effect, e.g. **support vector**





Computing the Viability Kernel Using Maximal Reachable Sets

Shahab Kaynama, John Maidens, Meeko Oishi,
Ian M. Mitchell, Guy A. Dumont

University of British Columbia

Code available at:
www.ece.ubc.ca/∼kaynama/papers/HSCC12_matlab.zip