Published in *IEEE Control Systems Letters (L-CSS)*, doi: 10.1109/LCSYS.2021.3134101
2022 American Control Conference (ACC) **ThA06.2**

Fast Computation of Tight Funnels for Piecewise Polynomial Systems

Inkyu Jang[1], Hoseong Seo[2], and H. Jin Kim[1]*

[1] Department of Aerospace Engineering, Seoul National University, Seoul 08826, Korea
[2] Robot System Team, Samsung Research, Seoul 06765, Korea

Motivation

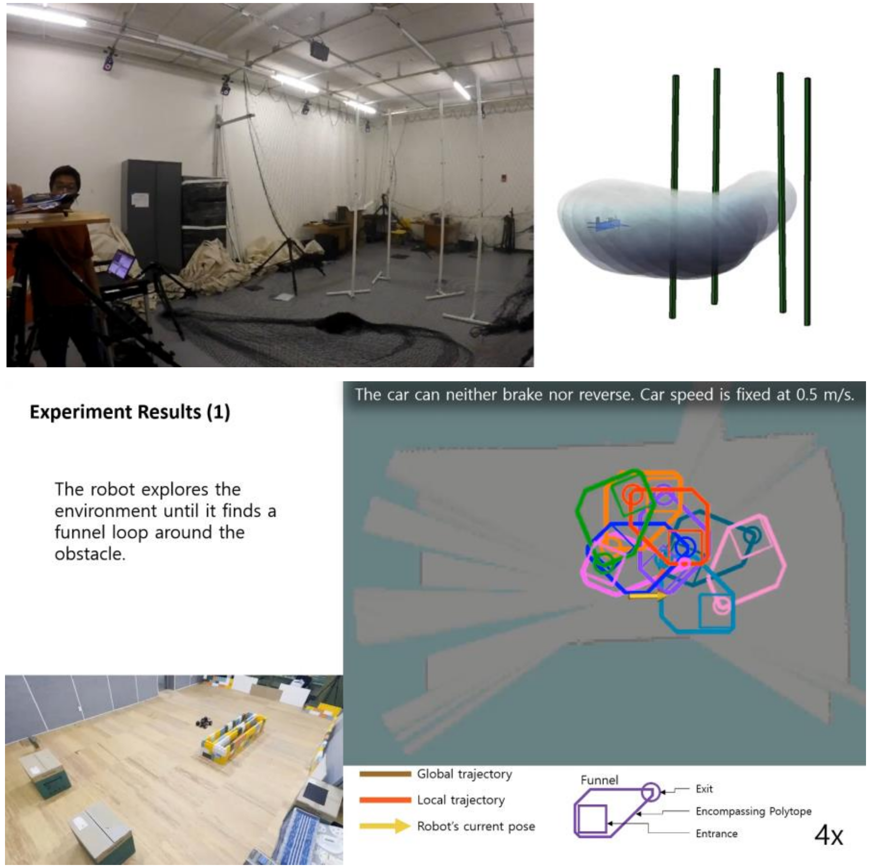


Fig. Funnel used in robust motion planning[1,2]

- In recent robust robotics, **funnels** (overapproximation of forward reachable sets) are often used as **conservative estimates of the bound of possible deviations** caused by **unknown but bounded disturbances**.

- The funnel finding problem is often computationally burdensome and requires hours of calculation.

- In this work, we propose a funnel-finding method for systems with piecewise polynomial dynamics, that is **tighter**, and more **computationally efficient** than existing works.

[1] Anirudha Majumdar, and Russ Tedrake. "Funnel libraries for real-time robust feedback motion planning." *The International Journal of Robotics Research* 36.8 (2017): 947-982.

[2] Inkyu Jang, et al. "Robust and Recursively Feasible Real-Time Trajectory Planning in Unknown Environments." *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021.

The Funnel-Finding Problem

Consider the following system dynamics:

$\dot{x} = f(t; x, u),$

where $u \in U$ is the unknown but bounded input of the system.

Definition (forward reachable set)

The *<u>forward reachable set (FRS)</u>* $FRS(T)$ given $x(0) \in X_0$ at $t = T$ is defined as

$$FRS(T) \coloneqq \left\{ x(T) \,\middle|\, \begin{array}{c} \dot{x} = f(t; x, u), u(t) \in U \ \ \forall t \in [0, T] \\ X(0) \in X_0 \end{array} \right\}.$$

$FRS(T)$ is the *<u>set of all reachable states</u>* to which the input trajectory $u(t)$ can drive the system.

Definition (funnel)

The set trajectory $X(t)$ is a valid *<u>funnel</u>* in time interval $[0, T]$ if

$$X(t) \supseteq FRS(t) \quad \forall t \in [0, T].$$

$X(t)$ is an *<u>overapproximation</u>* of all possible the state deviations by input $u(t)$.

Hamilton-Jacobi (HJ) Forward Reachability

The optimal control problem

minimize $J[T] = l_0\big(x(0)\big) + \int_0^T l\big(x(t), u(t)\big)\, dt$

subject to $\dot{x} = f(t; x, u)$

$$l(x,u) = \begin{cases} 0 & (u \in U) \\ \infty & (u \notin U) \end{cases}$$

Hamilton-Jacobi PDE associated with the optimal control problem

$$0 = \frac{\partial V}{\partial t} + \max_u \left[-l(x,u) + \frac{\partial V}{\partial t} \cdot f(t; x, u) \right]$$

$$= \frac{\partial V}{\partial t} + \max_{u \in U} \left[\frac{\partial V}{\partial t} \cdot f(t; x, u) \right],$$

$V(0, x) = l_0(x)$ (initial condition)

The value function $V(t,x) = \min_u J[t]$

- $V(t,x) \leq l_0(x_0)$ if state x is ***reachable*** at time t by $u(\cdot) \in U$ from the initial state x_0
- $FRS(t) = \{x \mid V(t,x) \leq 0\}$ is the ***forward reachable set (FRS)*** given $x(0) \in X_0 = \{x \mid l_0(x) \leq 0\}$

HJ Partial Differential Inequality for FRS Overapproximation

$$\frac{\partial V}{\partial t} + \max_{u \in U} \left[\frac{\partial V}{\partial t} \cdot f(t; x, u) \right] \leq 0,$$

or equivalently,

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial t} \cdot f(t; x, u) \leq 0, \qquad \forall u \in U$$

This ensures that the value function decreases along all possible trajectories $x(t)$, i.e.,

$$\frac{dV(t,x)}{dt} = \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} \cdot f(t; x, u) \leq 0$$

for any feasible input trajectory $u(t) \in U$.

Remark

If the above partial differential inequality holds,

$X(t) = \{x \mid V(t,x) \leq 0\}$ **is an overapproximation of the FRS given** $X(0) = \{x \mid l_0(x) \leq 0\}$.

Optimization Problem Formulation

Assumption

- U **is a full-dimensional bounded polytope, i.e.,** $U = \{u \mid A_u u \leq b_u\}$.
- **The dynamics is time invariant and control-affine, i.e.,** $f(t; x, u) = f(x) + g(x)u$.

Proposition (a variant of Farkas' Lemma)

$Ax \leq b$ **for all** x **such that** $Cx \leq d$, **i.e.,**

$\{x \mid Ax \leq b\} \subseteq \{x \mid Cx \leq d\}$,

<u>if and only if</u> $\exists \lambda \geq 0$ **such that** $b - Ax = \lambda^\top (d - Cx)$.

$x \in \mathbb{R}^n$,
$A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$,
$C \in \mathbb{R}^{l \times n}$, $d \in \mathbb{R}^l$,
$\lambda \in \mathbb{R}^{m \times l}$

Note : This is the positive polynomial theorem for the linear (order 1) case.

Optimization Problem Formulation

The size of the subzero level set of V is minimized to obtain a tight funnel.

The tight funnel-finding optimization problem

$$
\begin{aligned}
\underset{V(t,x),\ \lambda(t,x)}{\text{minimize}} \quad & \text{size}(\{x \mid V(T,x) \leq 0\}) \\
\text{subject to} \quad & \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} \cdot [f(x) + g(x)u] = \lambda(t,x)^{\top}(b_u - A_u u), \quad \forall u(t) \in U,\ \forall t \in [0,T] \quad \forall x \in ROI \\
& V(0,x) = l_0(x) \\
& \lambda(t,x) \geq 0
\end{aligned}
$$

Optimization Problem Formulation

The tight funnel-finding optimization problem

$$
\begin{aligned}
&\underset{V(t,x),\ \lambda(t,x)}{\text{minimize}} && \text{size}(\{x \mid V(T,x) \le 0\}) \\
&\text{subject to} && \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} \cdot f(x) = \lambda(t,x)^\top b_u, \qquad \forall t \in [0,T], \qquad \forall x \in ROI \\
& && \frac{\partial V}{\partial x} \cdot g(x) = -\lambda(t,x)^\top A_u, \qquad \forall t \in [0,T], \qquad \forall x \in ROI \\
& && V(0,x) = l_0(x) \\
& && \lambda(t,x) \ge 0
\end{aligned}
$$

Note : u is eliminated here!

Temporal Discretization

The original funnel-finding problem

minimize $\quad \text{size}(\{x \mid V(T,x) \le 0\})$

subject to
- HJ partial differential inequality
- Initial condition $V(0,x) = l_0(x)$

The constraints must be satisfied throughout the interval $t \in [0,T]$.

The sequential funnel-finding problem

1. Start with $V_0(x) = l_0(x)$.
2. For each $k \in \{0, 1, \cdots, K-1\}$,
   - Solve the funnel-finding problem for interval $t \in [t_k, t_{k+1}]$ with initial condition $V(t_k, x) = V_k(x)$.
   - Substitute $V_{k+1}(x) = V(t_{k+1}, x)$.

- **Global optimality (minimum funnel size) is partially lost.**
- **However, the funnel-finding problem can now be sequentially solved.**

More efficient computation!

Funnel-Certificate-Preserving Function Jumps

Assumption (jumps in V and λ)

For enhanced expressivity, we allow jumps at each time step t_k.

At $t = t_k$,

- $V(t,x)$ **jumps from** $V_k^-(x)$ **to** $V_k^+(x)$
- $\lambda(t,x)$ **jumps from** $\lambda_k^-(x)$ **to** $\lambda_k^+(x)$

To not break the funnel certificate,
the subzero level set of $V(t,x)$ should not shrink after the jumps, i.e.,

$$\{x \mid V_k^-(x) \leq 0\} \subseteq \{x \mid V_k^+(x) \leq 0\}$$

Or equivalently, $V_k^+(x) \leq \rho_k(x) \cdot V_k^-(x)$ **for some** $\rho_k \geq 0$.

Time Parametrization

Piecewise linear parametrization of V and λ

$$\beta(t,x) = \beta_k^+(x) \cdot \frac{t_{k+1} - t}{t_{k+1} - t_k} + \beta_{k+1}^-(x) \cdot \frac{t - t_k}{t_{k+1} - t_k} \coloneqq \mathrm{interp1}_{[t_k, t_{k+1}]}\left(t; \beta_k^+(x), \beta_{k+1}^-(x)\right)$$

$$\frac{\partial \beta}{\partial t} = \frac{\beta_{k+1}^-(x) - \beta_k^+(x)}{t_{k+1} - t_k} \qquad \forall t \in (t_k, t_{k+1})$$

$$\forall k \in \{0, 1, \cdots, K-1\}$$

$$\frac{\partial \beta}{\partial x} = \frac{\partial \beta_k^+}{\partial x} \cdot \frac{t_{k+1} - t}{t_{k+1} - t_k} + \frac{\partial \beta_{k+1}^-}{\partial x} \cdot \frac{t - t_k}{t_{k+1} - t_k} = \mathrm{interp1}_{[t_k, t_{k+1}]}\left(t; \frac{\partial \beta_k^+}{\partial x}, \frac{\partial \beta_{k+1}^-}{\partial x}\right)$$

Proposition

$\mathrm{interp1}_{[t_k, t_{k+1}]}\left(t; \alpha(x), \beta(x)\right) \geq 0 \ \ \forall x \in X$ **if and only if** $\alpha(x) \geq 0, \ \beta(x) \geq 0 \ \ \forall x \in X.$

Funnel Size Measure

Definition (funnel size)

$$\mathrm{size}(X) \coloneqq \inf_{\substack{x^\top S x \le 1 \ \ \forall x \in X \\ S \in \mathbb{S}^n_{++}}} \mathrm{trace}(S^{-1})$$

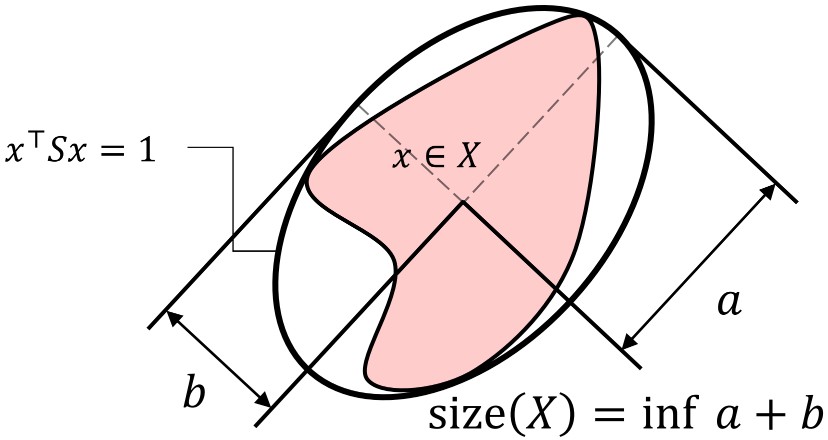


Fig. Illustration of the case when $n = 2$.

Remark 1

Minimization of $\mathrm{size}(X)$ is a SDP[1] w.r.t. S.

$$\begin{aligned} &\text{minimize} && \mathrm{trace}(R) \\ &\text{subject to} && \begin{bmatrix} S & I \\ I & R \end{bmatrix} \in \mathbb{S}^{2n}_{+} \end{aligned}$$

[1]Semidefinite Programming

Remark 2

If $X = \{x \mid l(x) \le 0\}$, $\mathrm{size}(X)$ is the optimal value for

$$\begin{aligned} &\underset{S,L}{\text{minimize}} && \mathrm{trace}(S^{-1}) \\ &\text{subject to} && x^\top S x - 1 \le L(x) l(x) \\ & && L(x) \ge 0 \qquad \forall x \end{aligned}$$

The Funnel-Finding Optimization Problem

$$\underset{\substack{V_k^+, V_{k+1}^-, \lambda_k^+, \lambda_{k+1}^-, \\ \rho_k, S_{k+1}, L_{k+1}}}{\text{minimize}} \quad \operatorname{trace}\left(S_{k+1}^{-1}\right)$$

subject to

$\frac{V_{k+1}^- - V_k^+}{t_{k+1} - t_k} + \frac{\partial V_k^+}{\partial x} \cdot f + (\lambda_k^+)^\top b_u = 0$ $\quad \frac{\partial V_k^+}{\partial x} \cdot g - (\lambda_k^+)^\top A_u = 0$		Feasibility at $t = t_k$
$\frac{V_{k+1}^- - V_k^+}{t_{k+1} - t_k} + \frac{\partial V_{k+1}^-}{\partial x} \cdot f + (\lambda_{k+1}^-)^\top b_u = 0$ $\quad \frac{\partial V_{k+1}^-}{\partial x} \cdot g - (\lambda_{k+1}^-)^\top A_u = 0$		Feasibility at $t = t_{k+1}$
$\lambda_k^+ \geq 0$		Dual feasibility at $t = t_k$
$\lambda_{k+1}^- \geq 0$		Dual feasibility at $t = t_{k+1}$
$V_k^+(x) \leq \rho_k(x) \cdot V_k^-(x)$	$\rho_k \geq 0$	Jump feasibility (V)
$x^\top S_{k+1} x - 1 \leq L_{k+1}(x) \cdot V_{k+1}^-(x)$	$S_{k+1} \in \mathbb{S}_{++}^n$	Size measure

Sum-of-Squares (SOS) Programming and Polynomial Positivity

Proposition

A polynomial $p(x)$ is nonnegative everywhere in $\mathbb{R}^n$
if it is a sum of squares of polynomials **$p_i(x)$, i.e.,**

$$p(x) = p_0(x)^2 + p_1(x)^2 + \cdots + p_N(x)^2 \in \Sigma^2[x] \qquad (x \in \mathbb{R}^n)$$

or equivalently,

$$p(x) = \frac{1}{2} m(x)^\top Q m(x)$$

where $Q \in \mathbb{S}_+^M$ is a positive semidefinite matrix,
and $m(x) = [m_0(x) \ \cdots \ m_M(x)]^\top$ is the column vector of lower-degree monomials.

Note : The converse does not hold in general.

Sum-of-Squares (SOS) Programming and Polynomial Positivity

Proposition (a variant of Handelman's Positivstellensatz[1])

A polynomial $p(x)$ is nonnegative everywhere in the set $X = \{x \mid c_i^\top x \leq d_i,\ i \in I\}$ if

$$p(x) = \sum_{j:I \to \{0,1\}} s_j(x) \cdot \prod_{i \in I} \left(d_i - c_i^\top x\right)^{j(i)}$$

where $s_j(x) \in \Sigma^2[x]$ for all j.

Note : The converse holds if X is a bounded polytope.[1]

[1] David Handelman. "Representing polynomials by positive linear functions on compact convex polyhedra." *Pacific Journal of Mathematics* 132.1 (1988): 35-62.

Optimizing with Piecewise Polynomial Systems

Assumption (piecewise polynomial system)

The system is assumed to be piecewise polynomial, whose *pieces* are convex polytopes.

$$f(x,u) = \begin{cases} f_0(x) + g_0(x)u & x \in \sigma_0 = \{x \mid A_{x\cdot 0}^{\top} x \leq b_{x\cdot 0}\} \\ \vdots & \vdots \\ f_N(x) + g_N(x)u & x \in \sigma_N = \{x \mid A_{x\cdot N}^{\top} x \leq b_{x\cdot N}\} \end{cases}$$

The pieces should completely cover the region of interest (ROI).

$$\mathrm{ROI} \subseteq \bigcap_{i \in \{0,\cdots,N\}} \sigma_i$$

$\Rightarrow$ **The constraints are applied to each piece σ_i.**

Remark : A wide range of systems can be accurately approximated in this form.

Results

System 1 : Damped pendulum

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ -\sin x_1 - x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$u \in [-0.1, 0.1]$

$X(0) = \{x \mid \|x\|_2 \leq 1\}$

$ROI = \{x \mid \|x\|_\infty \leq 1.3\}$

$t \in [0,5]$

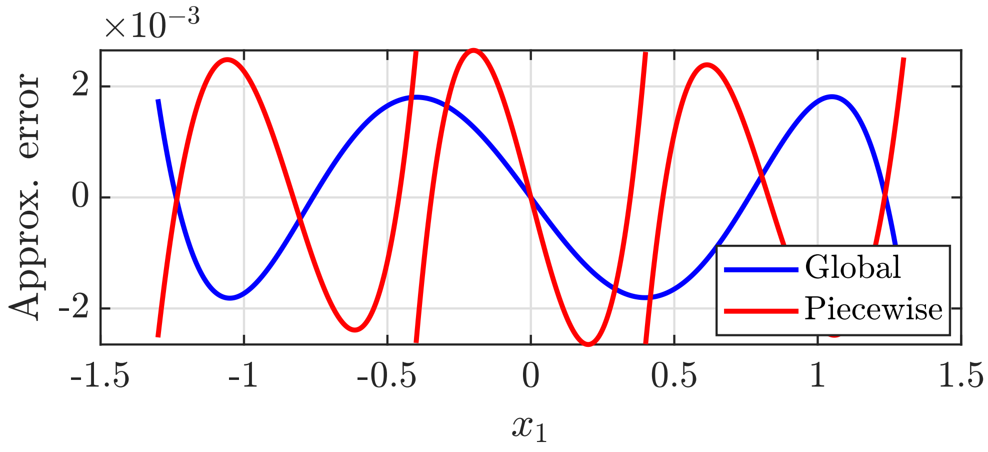


Fig. Error plots of the two polynomial approximations.

Approximation of the sine function

1. Global polynomial approximation

$$\sin x \approx 0.9930x - 0.1498x^3$$

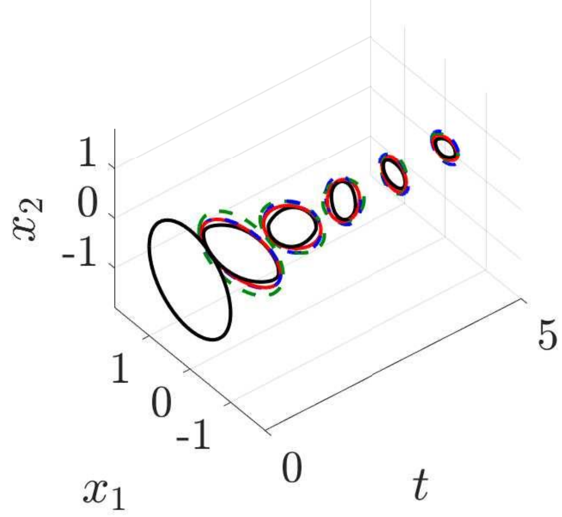
2. Piecewise polynomial approximation

$$\sin x \approx \begin{cases} 0.062 + 1.2691x + 0.3679x^2 & (x \in [-1.3, -0.4]) \\ 0.9801x & (x \in [-0.4, 0.4]) \\ -0.062 + 1.2691x - 0.3679x^2 & (x \in [0.4, 1.3]) \end{cases}$$

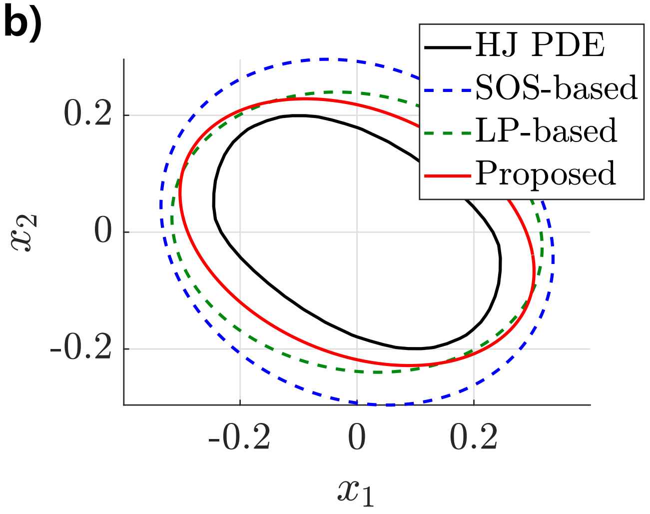
Results

Funnel computation result for the damped pendulum system

a)



b)



c)

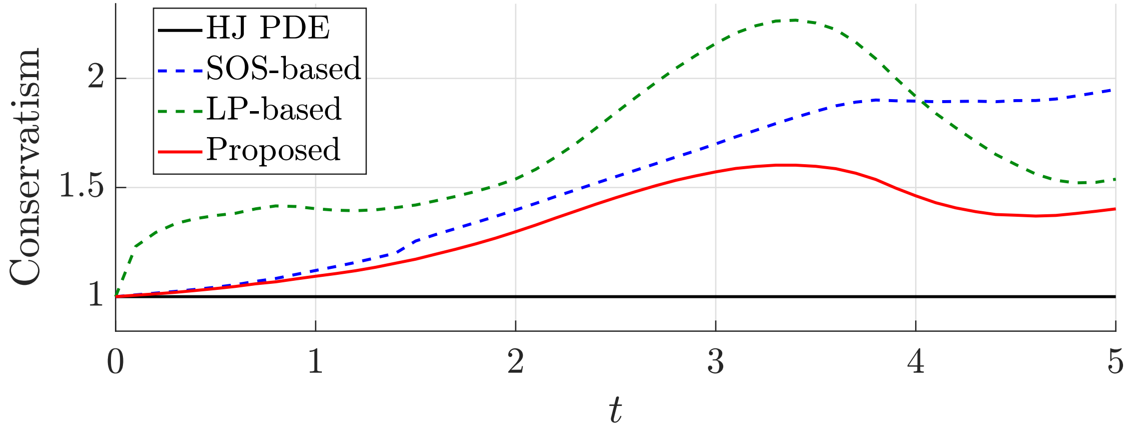


Fig. a) Funnel computation result
b) The resulting funnel at $t = 5$
c) Conservatism comparison of the four funnel-computing methods[1,2], where conservatism $= {}^{\text{size(Funnel)}}/_{\text{size(FRS)}}$.

[1] Anirudha Majumdar, and Russ Tedrake. "Funnel libraries for real-time robust feedback motion planning." *The International Journal of Robotics Research* 36.8 (2017): 947-982.

[2] Hoseong Seo, Clark Youngdong Son, and H. Jin Kim. "Fast funnel computation using multivariate Bernstein polynomial." *IEEE Robotics and Automation Letters* 6.2 (2021): 1351-1358.

Results

System 2 : Linear oscillator under Coulomb friction

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ -x_1 - \mathrm{sgn}(x_2) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

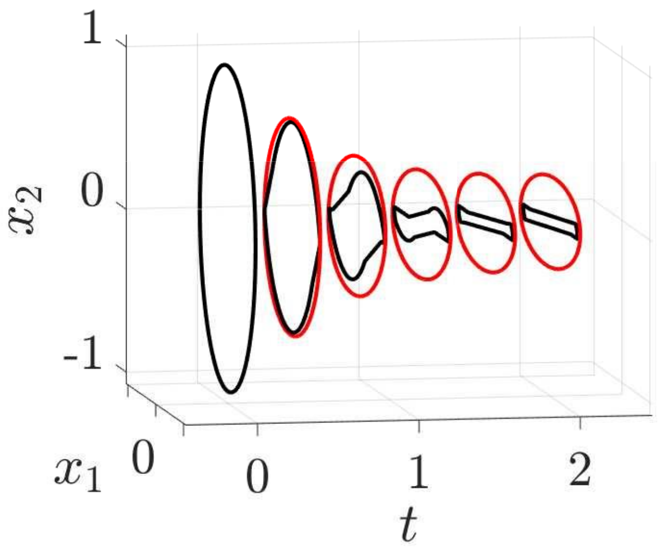
$u \in [-0.1, 0.1]$

$X(0) = \{x \mid \|x\|_2 \leq 1\}$

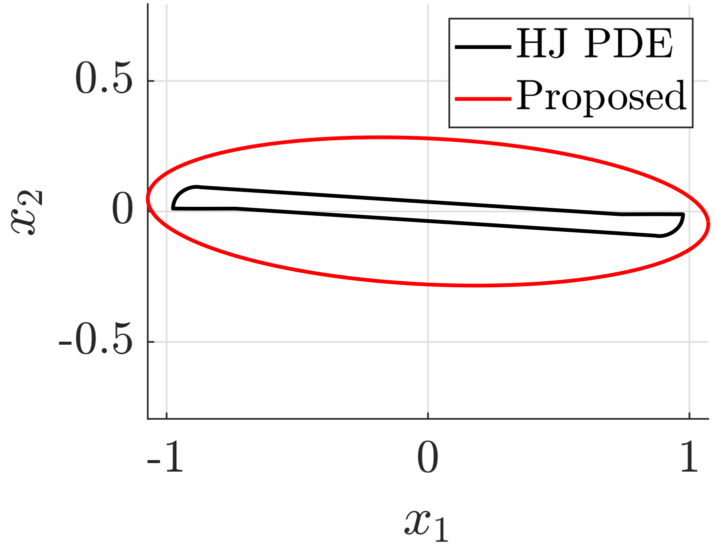
$\mathrm{ROI} = \{x \mid x\|x\|_\infty \leq 1.3\}$

$t \in [0,2]$

a)



b)



c)

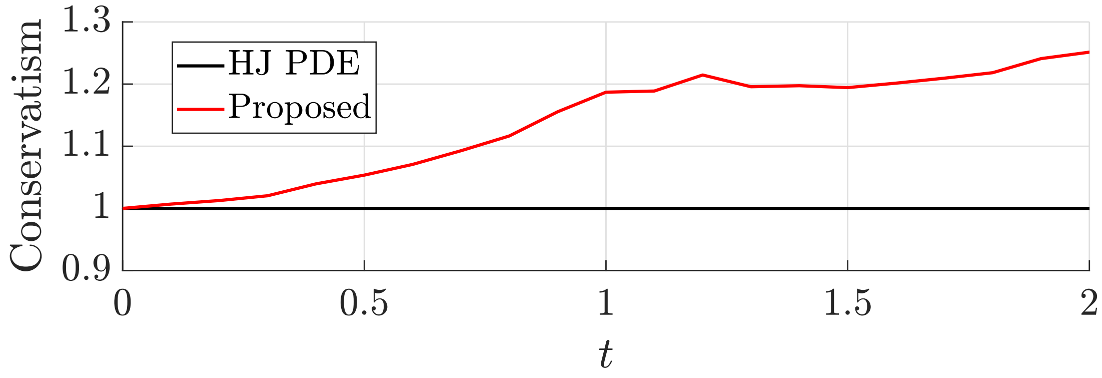


Fig. a) Funnel computation result **b)** The resulting funnel at $t = 2$
c) Conservatism comparison with the HJ-PDE solution (exact FRS)

Results

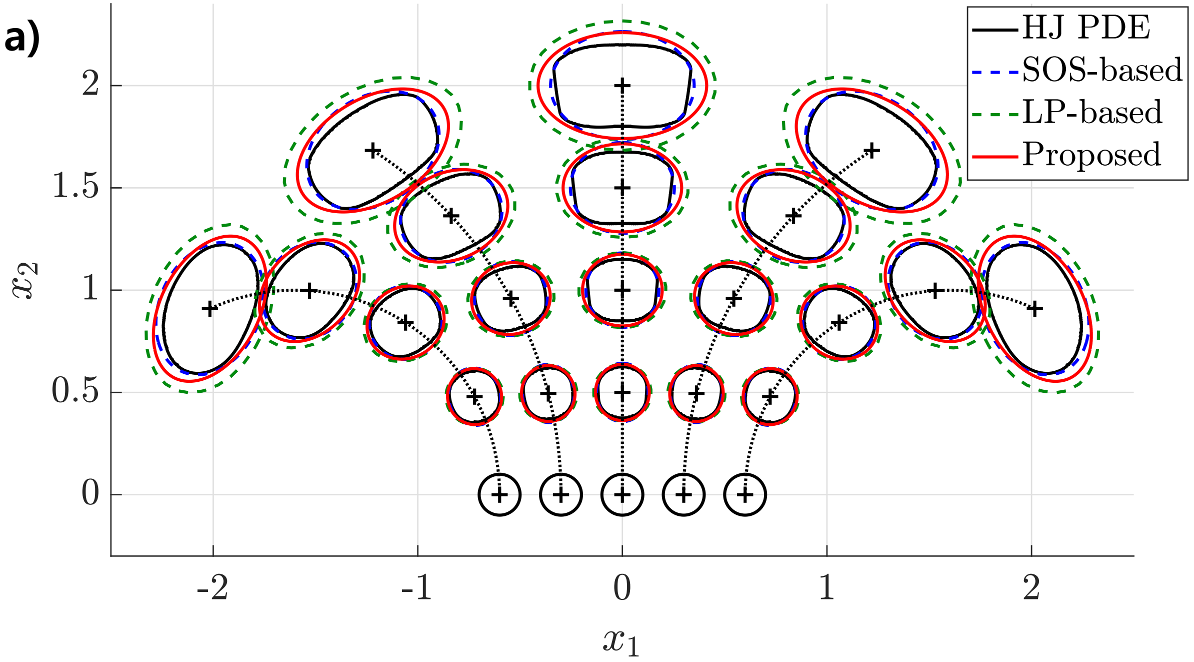
System 3 : Unicycle under uncertain speed and steering

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -\sin x_3 \\ \cos x_3 \\ \omega \end{bmatrix} + \begin{bmatrix} -\sin x_3 & 0 \\ \cos x_3 & 0 \\ 0 & 1 \end{bmatrix} u$$

$$U = \{u \in \mathbb{R}^2 \mid \|u\|_\infty \leq 0.05\}$$

$$X(0) = \{x \mid \|x\|_2 \leq 0.1\}$$

a)



b)

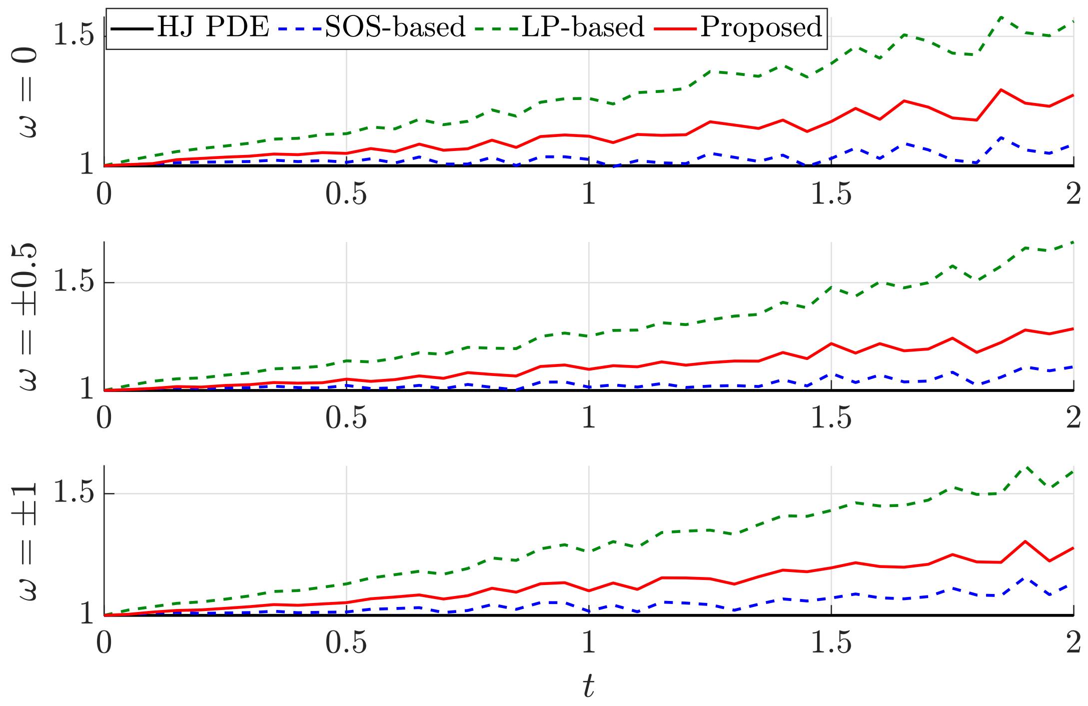


Fig. a) Funnel computation result for $\omega = 1, 0.5, 0, -0.5, -1$ (from left to right)
b) Conservatism comparison with four funnel-computing methods

Results (Calculation Efficiency)

Computation time measured in seconds

		HJ PDE (exact FRS)	SOS-based[1]	LP-based[2]	Proposed
Damped pendulum		10.754	19.936	**2.1084**	3.8986
Linear oscillator under Coulomb friction		60.175	-	-	**1.7003**
Unicycle	$\omega = 0$	505.52	358.76	12.714	**8.5910**
	$\omega = \pm 0.5$	1008.0	369.86	12.780	**9.0492**
	$\omega = \pm 1$	1389.1	380.44	12.761	**9.2288**

[1] Anirudha Majumdar, and Russ Tedrake. "Funnel libraries for real-time robust feedback motion planning." *The International Journal of Robotics Research* 36.8 (2017): 947-982.
[2] Hoseong Seo, Clark Youngdong Son, and H. Jin Kim. "Fast funnel computation using multivariate Bernstein polynomial." *IEEE Robotics and Automation Letters* 6.2 (2021): 1351-1358.

Conclusion and Future Work

In this work, we have

- proposed a **funnel-finding method** based on semidefinite programming that can be applied to systems with **piecewise polynomial dynamics**.
- showed that the proposed method computes **tighter funnels** in a relatively **fast computation time**, compared to other funnel-computing methods.

Possible future studies include

- funnel-computing methods that overcomes the *curse of dimensionality*.
- real-time algorithms that can adapt to changing input bounds.

Thank you!

Contact

Inkyu Jang
Lab for Autonomous Robotics Research (LARR)
Department of Aerospace Engineering, Seoul National University, Korea
leplusbon@snu.ac.kr

This research was supported by a grant to Theatre-Defense Research Center funded by Defense Acquisition Program Administration, and by Agency for Defense Development (UD200043CD_03).