Constrained Convex Generators: A Set Representation for Accurate Estimation

Interval Methods in Control

Daniel Silvestre

27th October 2023

Outline

Motivation for State Estimation

- Determine the remaining states for full state feedback;
- Prevent collisions;
- Detect loss of power in the rotors;
- Predict safety in future time instants.

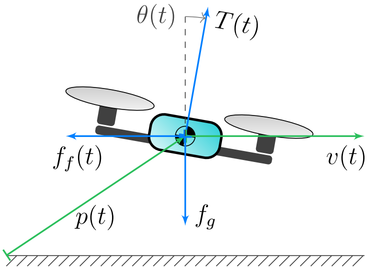


Figure: Drone schematic.

Motivation for State Estimation

- Determine the remaining states for full state feedback;
- Prevent collisions;
- Detect loss of power in the rotors;
- Predict safety in future time instants.

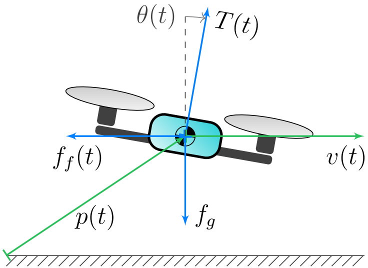


Figure: Drone schematic.

Motivation for State Estimation

- Determine the remaining states for full state feedback;
- Prevent collisions;
- Detect loss of power in the rotors;
- Predict safety in future time instants.

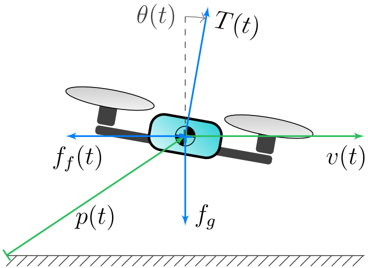


Figure: Drone schematic.

Motivation for State Estimation

- Determine the remaining states for full state feedback;
- Prevent collisions;
- Detect loss of power in the rotors;
- Predict safety in future time instants.

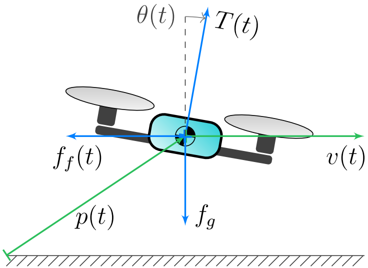
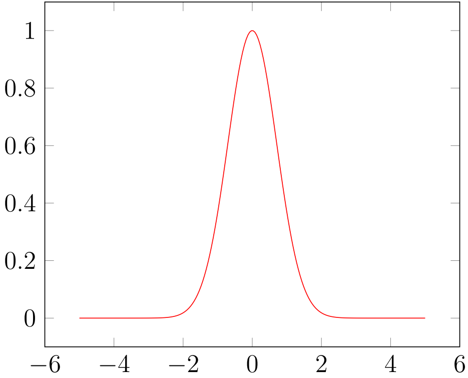


Figure: Drone schematic.

Stochastic State Estimation

Assumption: we know the probability density function for unknown signals.

- Start with prior knowledge
- Use the model to update the estimates with the elaspsed time.
- Improve the estimates by including the measurements.
- Repeat the procedure with new prior.



Stochastic State Estimation

Assumption: we know the probability density function for unknown signals.

- Start with prior knowledge
- Use the model to update the estimates with the elaspsed time.
- Improve the estimates by including the measurements.
- Repeat the procedure with new prior.

Stochastic State Estimation

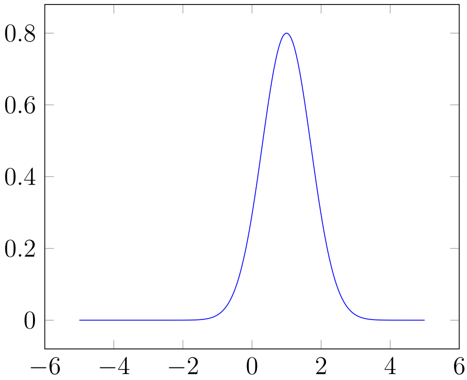
Assumption: we know the probability density function for unknown signals.

- Start with prior knowledge
- Use the model to update the estimates with the elaspsed time.
- Improve the estimates by including the measurements.
- Repeat the procedure with new prior.

Stochastic State Estimation

Assumption: we know the probability density function for unknown signals.

- Start with prior knowledge
- Use the model to update the estimates with the elaspsed time.
- Improve the estimates by including the measurements.
- Repeat the procedure with new prior.



Stochastic State Estimation

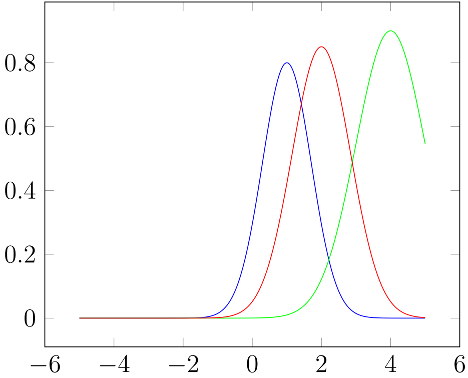
Assumption: we know the probability density function for unknown signals.

- Start with prior knowledge
- Use the model to update the estimates with the elaspsed time.
- Improve the estimates by including the measurements.
- Repeat the procedure with new prior.

Stochastic State Estimation

Assumption: we know the probability density function for unknown signals.

- Start with prior knowledge
- Use the model to update the estimates with the elaspsed time.
- Improve the estimates by including the measurements.
- Repeat the procedure with new prior.



Stochastic State Estimation

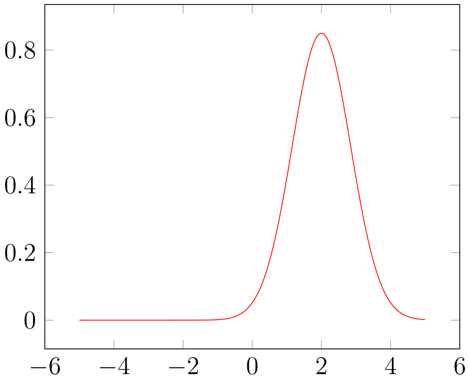
Assumption: we know the probability density function for unknown signals.

- Start with prior knowledge
- Use the model to update the estimates with the elaspsed time.
- Improve the estimates by including the measurements.
- Repeat the procedure with new prior.

Stochastic State Estimation

Assumption: we know the probability density function for unknown signals.

- Start with prior knowledge
- Use the model to update the estimates with the elaspsed time.
- Improve the estimates by including the measurements.
- Repeat the procedure with new prior.



How to perform Stochastic State Estimation

There are numerous methods depending on the set of assumptions:

- General case
  - Bayesian Filter (exact), Particle Filter (sampled version)
- Gaussian pdfs for all signals and linear dynamics
  - Kalman Filter
- Gaussian pdfs and nonlinear dynamics
  - Extended Kalman Filter, Cubature Kalman Filter, etc.

The above list is by no means exhaustive!

Problem

What if we do not know the entire probability density function but rather its support?

Set-valued State Estimation

- Set-valued State Estimation is suitable when:
  - We only know the support (i.e., all possible vector realizations of the random variables);
  - Distributions with multi-modes may *trick* a Kalman filter;
  - The application requires worst-case guarantees (like collision avoidance);
  - When we want to optimize over all possible states.

Definition (Set-valued state estimation)

Given the supports (admissible values) for the unknown signals, compute the set of all possible values for the state.

Outline

Set-Valued State Estimation for Linear Systems

- System dynamics are assumed linear and known:

$$\begin{aligned} x_{k+1} &= F_k x_k + B_k u_k + w_k \\ y_k &= C_k x_k + v_k \end{aligned}$$

Solution

Estimates for time $k+1$ are given by:

$$\mathcal{X}_{k+1} = (F_k \mathcal{X}_k \oplus \mathcal{W}_k + B_k u_k) \cap_{C_k} (y_k - \mathcal{V}_k),$$

where $\forall k \geq 0$ we have $x_k \in \mathcal{X}_k$, $w_k \in \mathcal{W}_k$, $v_k \in \mathcal{V}_k$ and $\oplus$ stands for the Minkowski sum and $\cap_C$ is the intersection after a linear map.

Required Set Operations (LPV case)

Linear Map $R\mathcal{X} + t$

Set obtained by applying the linear map to all points $\{Rx + t : x \in \mathcal{X}\}$

Minkowski sum $\mathcal{X} + \mathcal{Y}$

Set containing all sums of two vectors from the sets $\{x + y : x \in \mathcal{X}, y \in \mathcal{Y}\}$

Intersection through a Linear Map $\mathcal{X} \cap_R \mathcal{Y}$

All points in $\mathcal{X}$ that after the map R are also in $\mathcal{Y}$ $\{x : x \in \mathcal{X}, Rx \in \mathcal{Y}\}$

Set representations

Intervals

An Interval $\mathcal{I}$ can be represented by lower bounds l_b and upper bounds u_b such that:

$$\mathcal{I} = \{x : l_b \leq x \leq u_b\}$$

with element-wise inequalities.

Intervals (alternative definition)

An Interval $\mathcal{I}$ is a specific linear map of the hyper-cube:

$$\mathcal{I} = \{G\xi + c : \|\xi\|_\infty \leq 1\}$$

with G a diagonal matrix.

Set representations

Set	lin. map	lin. eq.	base set	$G\mathcal{X}+t$	$\mathcal{X}+\mathcal{Y}$	$\mathcal{X}\cap_R\mathcal{Y}$
$\mathcal{I}$	✓	✗	$\mathcal{B}_\infty$	✓	✓	✗

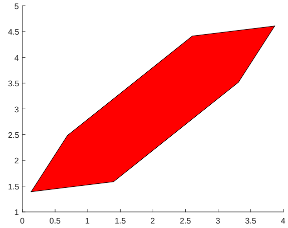
Set representations

Zonotopes

A Zonotope $\mathcal{Z}$ is a linear map of the hyper-cube:

$$\mathcal{Z} = \{G\xi + c : \|\xi\|_\infty \leq 1\}$$

for any matrix G.



Set representations

Set	lin. map	lin. eq.	base set	$G\mathcal{X}+t$	$\mathcal{X}+\mathcal{Y}$	$\mathcal{X}\cap_R\mathcal{Y}$
$\mathcal{I}$	✓	✗	$\mathcal{B}_\infty$	✓	✓	✗
$\mathcal{Z}$	✓	✗	$\mathcal{B}_\infty$	✓	✓	✗

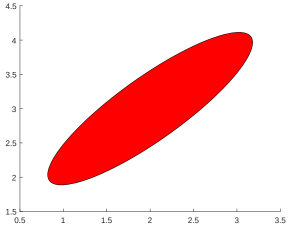
Set representations

Ellipsoids

An Ellipsoid $\mathcal{E}$ is a linear map of an ℓ_2 unit ball:

$$\mathcal{E} = \{G\xi + c : \|\xi\|_2 \leq 1\}$$

for any matrix G.



Set representations

Set	lin. map	lin. eq.	base set	$G\mathcal{X}+t$	$\mathcal{X}+\mathcal{Y}$	$\mathcal{X}\cap_R\mathcal{Y}$
$\mathcal{I}$	✓	✗	$\mathcal{B}_\infty$	✓	✓	✗
$\mathcal{Z}$	✓	✗	$\mathcal{B}_\infty$	✓	✓	✗
$\mathcal{E}$	✓	✗	$\mathcal{B}_2$	✓	✓	✗

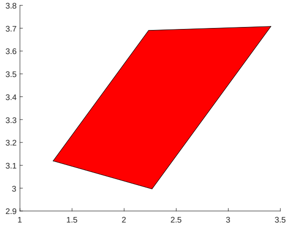
Set representations

Constrained Zonotopes

A Constrained Zonotope $\mathcal{CZ}$ is a linear map of an ℓ_∞ unit ball with an added linear constraint:

$$\mathcal{CZ} = \{G\xi + c : \|\xi\|_\infty \leq 1, A\xi = b\}$$

for any matrix G.



Set representations

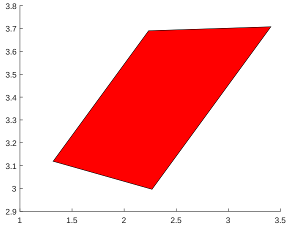
Set	lin. map	lin. eq.	base set	$G\mathcal{X}+t$	$\mathcal{X}+\mathcal{Y}$	$\mathcal{X}\cap_R\mathcal{Y}$
$\mathcal{I}$	✓	✗	$\mathcal{B}_\infty$	✓	✓	✗
$\mathcal{Z}$	✓	✗	$\mathcal{B}_\infty$	✓	✓	✗
$\mathcal{E}$	✓	✗	$\mathcal{B}_2$	✓	✓	✗
$\mathcal{CZ}$	✓	✓	$\mathcal{B}_\infty$	✓	✓	✓

Set representations

Polytope

A Polytope $\mathcal{P}$ is the explicit representation equivalent to the implicit way of writing $\mathcal{CZ}$:

$$\mathcal{P} = \{x : Ax \leq b\}$$



Set representations

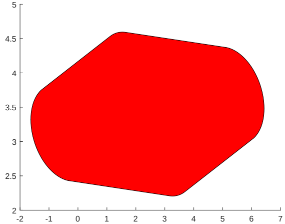
Set	lin. map	lin. eq.	base set	$G\mathcal{X}+t$	$\mathcal{X}+\mathcal{Y}$	$\mathcal{X}\cap_R\mathcal{Y}$
$\mathcal{I}$	✓	✗	$\mathcal{B}_\infty$	✓	✓	✗
$\mathcal{Z}$	✓	✗	$\mathcal{B}_\infty$	✓	✓	✗
$\mathcal{E}$	✓	✗	$\mathcal{B}_2$	✓	✓	✗
$\mathcal{CZ}$	✓	✓	$\mathcal{B}_\infty$	✓	✓	✓
$\mathcal{P}$	✓	✓	$\mathcal{B}_\infty$	✓	✓	✓

Set representations

Constrained Convex Generator [1]

A Constrained Convex Generator $\mathcal{CCG}$ is an implicit representation where the generator variables are constrained to some convex sets:

$$\mathcal{CCG} = \{G\xi + c : A\xi = b, \xi \in \mathcal{C}_1 \times \cdots \times \mathcal{C}_{n_p}\}.$$



Set representations

Set	lin. map	lin. eq.	base set	$G\mathcal{X}+t$	$\mathcal{X}+\mathcal{Y}$	$\mathcal{X}\cap_R\mathcal{Y}$
$\mathcal{I}$	✓	✗	$\mathcal{B}_\infty$	✓	✓	✗
$\mathcal{Z}$	✓	✗	$\mathcal{B}_\infty$	✓	✓	✗
$\mathcal{E}$	✓	✗	$\mathcal{B}_2$	✓	✓	✗
$\mathcal{CZ}$	✓	✓	$\mathcal{B}_\infty$	✓	✓	✓
$\mathcal{P}$	✓	✓	$\mathcal{B}_\infty$	✓	✓	✓
$\mathcal{CCG}$	✓	✓	any	✓	✓	✓

Outline

Set-Valued State Estimation for uncertain Linear Systems

- System dynamics are assumed linear with unknown parameters:

$$x_{k+1} = \left(F_k + \sum_{\ell=1}^{n_\Delta} \Delta_k^{(\ell)} U_\ell\right) x_k + B_k u_k + w_k$$

$$y_k = C_k x_k + v_k$$

Solution

Instead of $F_k \mathcal{X}_k$ *replace by:*

$$\text{cvxHull}\left(\bigcup_{\Delta \in \text{vertex}([-1,1]^{n_\Delta})} \left(F_k + \sum_{\ell=1}^{n_\Delta} \Delta_\ell(k) U_\ell\right) \mathcal{X}(k) \right).$$

Required Set Operations (uncertain LPV case)

Convex Hull $\mathrm{cvxHull}(\mathcal{X}, \mathcal{Y})$

Set with all points in line segments linking points in $\mathcal{X}$ and $\mathcal{Y}$,
$\mathrm{cvxHull}\left(\mathcal{X}, \mathcal{Y}\right) := \left\{z : z = \lambda x + (1-\lambda)y, \lambda \in [0,1], x \in \mathcal{X}, y \in \mathcal{Y}\right\}$.

- We can leverage the Balas formulation and write:

$$
\begin{aligned}
Z_h = \{p_h = G_x\xi_x + \lambda c_x + G_y\xi_y + (1-\lambda)c_y :\\
0 \leq \lambda \leq 1, A_x\xi_x = \lambda b_x, A_y\xi_y = (1-\lambda)b_y,\\
\|\xi_x\|_{\ell_x} \leq \lambda, \|\xi_y\|_{\ell_y} \leq (1-\lambda)\}
\end{aligned}
$$

- And rewrite it to be in standard CCG format.

Convex Hull for CCGs

- Consider two CCGs:

$$X = (G_x, c_x, A_x, b_x, \mathfrak{C}_x^{(\tau_x)}) \subset \mathbb{R}^n$$
$$Y = (G_y, c_y, A_y, b_y, \mathfrak{C}_y^{(\tau_y)}) \subset \mathbb{R}^n$$

- The exact convex hull is given by the CCGs $Z_h = (G_h, c_h, A_h, b_h, \mathfrak{C}_h)$:

$$G_h = \begin{bmatrix} G_x & G_y & c_x - c_y \end{bmatrix}, c_h = \frac{c_x + c_y}{2},$$
$$A_h = \begin{bmatrix} A_x & 0 & -b_x \\ 0 & A_y & b_y \end{bmatrix}, b_h = \begin{bmatrix} \frac{1}{2} b_x \\ \frac{1}{2} b_y \end{bmatrix}$$
$$\mathfrak{C}_h = \{\mathfrak{C}_x^{(\tau_x+1)}(\xi_x, \xi_\lambda, -1, 0.5), \mathfrak{C}_y^{(\tau_y+1)}(\xi_y, \xi_\lambda, 1, 0.5), \mathbb{R}\},$$

CCG vs CZ convex hull

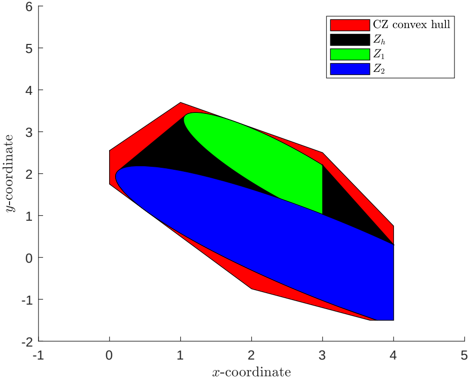
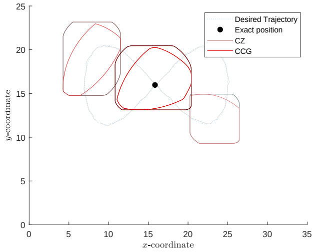


Figure: Comparison between the set Z_h and the convex hull that one would obtain if first converted both Z_1 and Z_2 to constrained zonotopes by overbounding all convex generators by the ℓ_∞ unit ball.

Simulation Results

Setup: Unicycle model with a digital compass that has a $\pm 5°$ error starting at position $(16, 16)$ with ± 2 uncertainty in the nominal initial position at $(15.5, 15.5)$. There are two beacons at $(5, 25)$ and $(23, 10)$ that can be detected in a radius of 5 and 2.

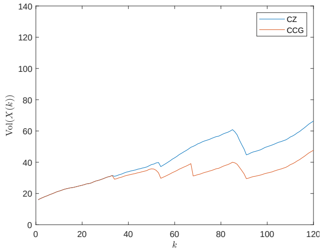
- Vehicle performing a figure 8;
- There is a clear advantage in using CCGs in terms of the hypervolume of the sets.



Simulation Results

Setup: Unicycle model with a digital compass that has a $\pm 5^\circ$ error starting at position $(16, 16)$ with ± 2 uncertainty in the nominal initial position at $(15.5, 15.5)$. There are two beacons at $(5, 25)$ and $(23, 10)$ that can be detected in a radius of 5 and 2.

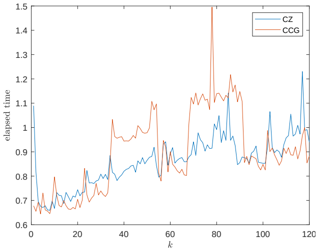
- CCGs hypervolume is decreased when the vehicle receives beacon range measurements;
- The exact representation of circle shapes allow for no conservatism;
- The wrapping effect of CZs can be observed from iteration 50 to 100.



Simulation Results

Setup: Unicycle model with a digital compass that has a $\pm 5^\circ$ error starting at position $(16, 16)$ with ± 2 uncertainty in the nominal initial position at $(15.5, 15.5)$. There are two beacons at $(5, 25)$ and $(23, 10)$ that can be detected in a radius of 5 and 2.

- Elapsed time for constructing the sets, order reduction and volume computation per iteration;
- Most time is spent in order reduction and volume computation (not needed in practice);
- There is a need for a better order reduction for CCGs.



Outline

Order Reduction for CCGs

- We can trivially use all order reduction methods from CZs if we upper bound all norm balls by ℓ_∞ balls;
- We can model the sets using Integral Quadratic Constraints (IQCs) much like in ReachSDP [2](currently under review);

This is quite an open topic for further research!

Outline

Concluding Remarks

Code for all set representations given in Github:

- `https://github.com/danielmsilvestre/ReachTool`
- Set-valued State Estimation is exact for the broad class of LPV (encompasses nonlinear models).
- Estimation in the presence of uncertainties returns the exact convex hull.
- Additional order reductions methods is still quite an open topic.
- It is possible to encode CCGs directly in Yalmip using the provided toolbox.

Main bibliography

[1] D. Silvestre, "Constrained convex generators: A tool suitable for set-based estimation with range and bearing measurements," *IEEE Control Systems Letters*, vol. 6, pp. 1610–1615, 2022. DOI: 10.1109/LCSYS.2021.3129729.

[2] H. Hu, M. Fazlyab, M. Morari, and G. J. Pappas, "Reach-sdp: Reachability analysis of closed-loop systems with neural network controllers via semidefinite programming," in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 5929–5934. DOI: 10.1109/CDC42340.2020.9304296.

Constrained Convex Generators: A Set Representation for Accurate Estimation

Interval Methods in Control

Daniel Silvestre

27th October 2023