# Explicit Computation of Guaranteed State Estimates using Constrained Convex Generators

Francisco Rego[1] and Daniel Silvestre[2]

*Abstract*—**One of the main challenges when performing set-based state estimation is the inherent trade-off between accuracy and computing time. When using accurate set representations like polytopes, even if written in Constrained Zonotopes (CZs) format, the data structures keep increasing in size which will lead to the need of some order reduction method that increases the computational load in the iterations when such a routine is run. Moreover, computing a vector estimate will amount to solving an optimization problem or a matrix inversion, which are expensive procedures if the state space is large. In this paper, we propose an efficient approach for the state estimation of discrete-time Linear Time-Invariant (LTI) systems based on Constrained Convex Generators (CCGs) that allows to write explicitly the set in terms of a fixed number of past inputs and measurements. In doing so, the whole estimation task amounts to performing a small number of multiplications with offline-computed matrices which makes the runtime computation significantly faster and removes the need for order reduction methods. Numerical results show the effectiveness of the proposed method.**

## I. Introduction

Set-based observers are algorithms that compute sets containing all possible state values for a given dynamical system. According to the surveys in [1], [2], the issue of designing set-based observers has received great attention. There are several useful applications ranging from state estimation with unknown distributions for the disturbances, position estimation of safety-critical equipment [3], fault detection [4], control of multiple-model systems [5] robust Model Predictive Control (MPC) [6], collision avoidance [7] and safe optimal control [8].

Since the early work in [9], which represents sets as ellipsoids, guaranteed state estimation methods have been extensively studied. A similar idea to [9] is used in [10], but sets are expressed by their minimum-volume bounding parallelotopes. In [11], a method with polytopic encoding is provided. Similar techniques are used in the work in [12], which employs a zonotope representation while reducing the size of the zonotope using either an analytical formula or solving a convex optimization problem. The work in [13] takes into account polytopic or ellipsoidal representations. Finally, [14] uses sub-pavings to improve accuracy at the expenses of computation.

This work was partially supported by the Portuguese Fundação para a Ciência e a Tecnologia (FCT) through project FirePuma (https://doi.org/10.54499/PCIF/MPG/0156/2019), through Institute for Systems and Robotics (ISR), under Laboratory for Robotics and Engineering Systems (LARSyS) project UIDB/50009/2020 (10.54499/UIDB/50009/2020), and through COPELABS, University Lusófona project UIDB/04111/2020 (10.54499/UIDB/04111/2020).

[1]Francisco Rego is with IPLuso, Lisbon, Portugal p7577@ipluso.pt

[2]Daniel Silvestre is with the School of Science and Technology, NOVA University of Lisbon (FCT/UNL), Caparica, Portugal dsilvestre@isr.tecnico.ulisboa.pt

The decision of how to depict the sets is crucial, which can range from ellipsoids [15] [16] to zonotopes, where [17]–[19] showed to have decreased wrapping effect. Interval representations such as those in [20], are also prone to wrapping effects. Constrained convex generators, which were suggested in [21], are a recent alternative that unifies these set representations.

Developing computationally efficient methods that do not excessively over-approximate the set of admissible states is one of the primary challenges concerning set-based observers. In comparison to CZs, using CCGs in state estimation can minimize conservatism, as shown in [21]. Nevertheless, there is an inherent trade-off between accuracy and computing time for set representations like Zonotopes, CZs and CCGs. To address this problem, we present a method that starts with a coarse estimate using an ellipsoidal method that can be written in closed-form (which can be swapped by other filters) and then improving its accuracy using a limited number of previous inputs and measurements. In doing so, the obtained set will be a mix of ellipsoidal and polytopic components (for which CCGs are exact) and the need for order reductions methods is eliminated. Moreover, most of the data structures can be computed offline, which greatly speeds up runtime computation. These features are critical if set-based methods are to be incorporated in MPCs [22].

Therefore, the main contributions of this work are as follows:

- A method for efficiently computing the CCG containing the state of an LTI system where most of the data structures can be pre-computed offline,
- A fast way to obtain a vector estimate by computing a center of the CCG without requiring the inversion of a matrix nor an optimization problem in runtime.

### A. Notation

Let $I_n$ be the identity matrix of size $n$, and let $\mathbf{0}_n$ stand for the n-dimensional array of zeros and $\mathbf{1}_n$ denote the n-dimensional array of ones. Dimensions are omitted when can be infered from context. For a vector $v$, its transpose is written as $v^\intercal$ and the Euclidean norm is denoted by $\|x\|_2 := \sqrt{x^\intercal x}$. Additionally, $\|x\|_\infty := \max_i |x(i)|$, where $x(i)$ is the $i$th element of $x$. The generalized intersection is represented by $\cap_R$ to mean $X \cap_R Y := \{x : x \in X, Rx \in Y\}$, the Minkowski sum of set $X$ and $Y$ by $\oplus$, i.e., $X \oplus Y := \{x + y : x \in X, y \in Y\}$, and the cartesian product by $\times$ as $X \times Y := \{\begin{bmatrix} x \\ y \end{bmatrix} : x \in X, y \in Y\}$.

## II. Problem Definition

The problem of guaranteed state estimation in discrete-time LTI systems can be formulated as the problem of finding a set of possible state values given measurements, disturbance, noise, and initial state bounds. The model is provided by:

$$x_{k+1} = Fx_k + Bu_k + w_k, \tag{1a}$$
$$y_k = Cx_k + v_k, \tag{1b}$$

where $x_k \in \mathbb{R}^n$, $u_k \in \mathbb{R}^{n_u}$, $w_k \in \mathbb{R}^n$, $y_k \in \mathbb{R}^{n_y}$, and $v_k \in \mathbb{R}^{n_y}$ represent the system state, input, disturbance signal, output, and noise, respectively. The problem this article addresses can be summed up as follows:

**Problem 1.** *How to calculate a set $X_k$ that ensures that $x_k \in X_k, \forall k \geq 0$, given $y_k$ measurements and compact convex sets $X_0$, $V$, and $W$, such that $x_0 \in X_0$, $v_k \in V$ and $w_k \in W$.*

### A. State estimation using CCGs

Before proceeding to the main results of this paper, in this section, we review the standard solution for guaranteed state estimation (Problem 1) with CCGs found in [3], [23].

The formal description of a CCG is given in Definition 1.

**Definition 1** (Constrained Convex Generators). *$\mathcal{Z} \subset \mathbb{R}^n$ is defined by the tuple $(G, c, A, b, \mathfrak{C})$ with $G \in \mathbb{R}^{n_c \times n_g}$, $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{n_c \times n_g}$, $b \in \mathbb{R}^{n_c}$, and $\mathfrak{C} := \{\mathcal{C}_1, \mathcal{C}_2 \ldots, \mathcal{C}_{n_p}\}$ such that:*

$$\mathcal{Z} = \{G\xi + c : A\xi = b, \xi \in \mathcal{C}_1 \times \ldots \times \mathcal{C}_{n_p}\}, \tag{2}$$

*where the sets $\mathcal{C}_1$ to $\mathcal{C}_{n_p}$ are the generator sets, $n_c$ is the number of constraints, $n_g$ is the sum of the size of the generators and $n_p$ is the number of generators.*

Intuitively, CCGs describe in an indirect form a set $X$ by describing it as a linear operator of a much larger state space of the generator variables $\xi$. CCGs are a very general form of representing sets, which does not require set approximations if we need to perform set operations between polytopes and ellipsoids, amoung many other sets. In particular, it generalizes constrained zonotopes or polytopes that correspond to

$$X = (G, c, A, b, \|\xi\|_\infty \leq 1) \tag{3}$$

or ellipsoids that can be written as

$$X = (G, c, [\,], [\,], \|\xi\|_2 \leq 1) \tag{4}$$

Other types of sets can also be described as CCGs such as ellipsotopes, intervals, or zonotopes. For more information on CCGs, the reader is referred to [3]. The usual operations such as linear maps, Minkowsky sum, and intersection are well-defined for CCGs and can be computed in closed-form as given in Definition 2.

**Definition 2.** *Consider three Constrained Convex Generators (CCGs) as in Definition 1:*

- $Z = (G_z, c_z, A_z, b_z, \mathfrak{C}_z) \subset \mathbb{R}^n$
- $W = (G_w, c_w, A_w, b_w, \mathfrak{C}_w) \subset \mathbb{R}^n$
- $Y = (G_y, c_y, A_y, b_y, \mathfrak{C}_y) \subset \mathbb{R}^m$

*and a matrix $R \in \mathbb{R}^{m \times n}$ and a vector $t \in \mathbb{R}^m$. The three set operations are defined as:*

$$RZ + t = (RG_z, Rc_z + t, A_z, b_z, \mathfrak{C}_z)$$
$$Z \oplus W = \left( \begin{bmatrix} G_z & G_w \end{bmatrix}, c_z + c_w, \begin{bmatrix} A_z & \mathbf{0} \\ \mathbf{0} & A_w \end{bmatrix}, \begin{bmatrix} b_z \\ b_w \end{bmatrix}, \{\mathfrak{C}_z, \mathfrak{C}_w\} \right)$$
$$Z \cap_R Y = \left( \begin{bmatrix} G_z & \mathbf{0} \end{bmatrix}, c_z, \begin{bmatrix} A_z & \mathbf{0} \\ \mathbf{0} & A_y \\ RG_z & -G_y \end{bmatrix}, \begin{bmatrix} b_z \\ b_y \\ c_y - Rc_z \end{bmatrix}, \{\mathfrak{C}_z, \mathfrak{C}_y\} \right)$$

Given these operations, one may solve Problem 1 recursively, since given a set $X_k \subset \mathbb{R}^n$ such that $x_k \in X_k$ and a measurement $y_k$, the set $X_{k+1} \subset \mathbb{R}^n$ such that $x_{k+1} \in X_{k+1}$ can be computed as

$$X_{k+1} = (FX_k \oplus W + Bu_k) \cap_C (y_k - V). \tag{5}$$

In this implementation, we assume that the sets $W$ and $V$ are represented as CCGs with a constant description for the disturbance and noise sets:

$$W := (G_w, c_w, [\,], [\,], \mathfrak{C}_w), \tag{6a}$$
$$V := (G_v, c_v, [\,], [\,], \mathfrak{C}_v). \tag{6b}$$

One major drawback with this approach is that, since a Minkowsky sum and a generalized intersection have to be performed at each time instant, the number of generator sets increases, rendering the problem computationally expensive after a certain number of iterations. Therefore, to address this issue in the next section, we will provide a more efficient solution with a fixed-length description.

## III. Explicit computation of guaranteed state estimates

In this section, we will resort to a conservative ellipsoidal estimate that can be exchanged by any other filter output as long as it is given by a closed-form expression (i.e., no iterative procedure). Other options could be found in the literature like in [24]–[26].

### A. Ellipsoidal observer

Before obtaining a state estimate with low conservatism, we start with a coarse ellipsoidal state estimate based on a Luenberger observer, which for systems like in (1), is given by

$$\hat{x}_{k+1} = F\hat{x}_k + Bu_k + L(y_k - C\hat{x}_k), \tag{7}$$

where $L$ is defined such that $\rho(F - LC) < 1$, where $\rho(\cdot)$ is the spectral radius. If the pair $(F, C)$ is detectable, such matrix $L$ always exists. Defining the estimation error as $e_k := x_k - \hat{x}_k$ from (1) we obtain

$$e_{k+1} = (F - LC)e_k + w_k - Gv_k. \tag{8}$$

Given that $\rho(F - LC) < 1$ there exists a symmetric matrix $P \in \mathbb{R}^{n \times n}$ such that

$$(F - LC)^\intercal P(F - LC) - P = -I_n, \tag{9}$$

and we may define a decrease rate as follows

$$a := \|P^{\frac{1}{2}}(F - LC)P^{-\frac{1}{2}}\|_2 = \sqrt{1 - \frac{1}{\sigma_{max}(P)}}, \tag{10}$$

where $\sigma_{max}(\cdot)$ is the maximum singular value. From (9), we have that $a < 1$. Therefore, defining

$$e_{init} := \max_{\xi \in X_0} \|P^{\frac{1}{2}}\xi\|_2 \tag{11}$$

$$e_{noise} := \max_{\xi \in W \oplus -LV} \|P^{\frac{1}{2}}\xi\|_2 \tag{12}$$

and applying Theorem 6 in [27], we have that $x_k \in \hat{X}_k$ for all $k \geq 0$, where

$$\hat{X}_k = \left\{ \hat{x}_k + \xi : \|P^{\frac{1}{2}}\xi\|_2 \leq a^k e_{init} + \frac{e_{noise}}{1-a} \right\}. \tag{13}$$

Given that the state estimate is an ellipsoid, it can be written in CCG format as follows

$$\hat{X}_k = \left( \left( a^k e_{init} + \frac{e_{noise}}{1-a} \right) P^{-\frac{1}{2}}, \hat{x}_k, [\,], [\,], \|\xi\|_2 \leq 1 \right). \tag{14}$$

We remark to the reader that the fact that CCGs allow for set operations between polytopes and ellipsoids, it is possible to use the conservative ellipsoidal estimate $\hat{X}_k$ and improve it by explicitly considering the exact iterations for some fixed number of time instants. Therefore, at some time $k$, the set $\hat{X}_{k-N}$ can be viewed as an implicit order reduction to the more accurate set $X_{k-N}$ that would be obtained by the direct recursion in (5). These two fact will be useful in the next section to provide the main contribution of this paper to have a set-valued observer that does not require order reduction methods and where most of the computations can be performed offline before the estimation procedure is run.

## B. Explicit finite-horizon observer

The state estimate given by (14) can serve as a conservative set that can be improved by $N$ iterations of the recursion (5). Specifically, defining for an integer $l$,

$$Y_k^l := \begin{bmatrix} y_{k-1} \\ \vdots \\ y_{k-l} \end{bmatrix}, \tag{15}$$

$$U_k^l := \begin{bmatrix} u_{k-1} \\ \vdots \\ u_{k-l} \end{bmatrix}, \tag{16}$$

we consider that the state estimate at time $k$ is expressed by

$$X_k^l = \left( G_{X,k}^l, c_{X,k}^l, A_{X,k}^l, b_{X,k}^l, \mathfrak{C}_X^l \right) \subset \mathbb{R}^n, \tag{17}$$

where

$$G_{X,k}^l = \left[ G^l \quad \left( a^{k-l} e_{init} + \tfrac{e_{noise}}{1-a} \right) G_0^l \right], \tag{18a}$$
$$c_{X,k}^l = c^l + c_U^l U_k^l + c_0^l \hat{x}_{k-l}, \tag{18b}$$
$$A_{X,k}^l = \left[ A^l \quad \left( a^{k-l} e_{init} + \tfrac{e_{noise}}{1-a} \right) A_0^l \right], \tag{18c}$$
$$b_{X,k}^l = Y_k^l + b^l + b_U^l U_k^l + b_0^l \hat{x}_{k-l}. \tag{18d}$$

For $l = 0$ one recovers the ellipsoidal observer considering that $G^0$, $c_U^0$, $A^0$,$A_0^0$, $b^0$, $b_U^0$ and $b_0^0$ are empty matrices,

$$\mathfrak{C}_X^0 = \{\xi : \|\xi\|_2 \leq 1\}, \tag{19}$$

and

$$G_0^0 := P^{-\frac{1}{2}}, \tag{20a}$$
$$c^0 := \mathbf{0}_n, \tag{20b}$$
$$c_0^0 := I_n. \tag{20c}$$

By applying (5) we obtain after simple computations the main result of this paper.

**Theorem 1.** *Given a state estimate $X_k^l$ such that $x_k \in X_k^l$, then $x_{k+1} \in X_{k+1}^{l+1}$ with*

$$\mathfrak{C}_X^{l+1} = \mathfrak{C}_w \times \mathfrak{C}_v \times \mathfrak{C}_X^l, \tag{21a}$$
$$G^{l+1} = \begin{bmatrix} G_w & \mathbf{0} & FG^l \end{bmatrix}, \tag{21b}$$
$$G_0^{l+1} = FG_0^l, \tag{21c}$$
$$c^{l+1} = Fc^l + c_w, \tag{21d}$$
$$c_U^{l+1} = \begin{bmatrix} B & Fc_U^l \end{bmatrix}, \tag{21e}$$
$$c_0^{l+1} = Fc_0^l, \tag{21f}$$
$$A^{l+1} = \begin{bmatrix} \mathbf{0} & G_v & CG^l \\ \mathbf{0} & \mathbf{0} & A^l \end{bmatrix}, \tag{21g}$$
$$A_0^{l+1} = \begin{bmatrix} CG_0^l \\ A_0^l \end{bmatrix}, \tag{21h}$$
$$b^{l+1} = \begin{bmatrix} -Cc^l - c_v \\ b^l \end{bmatrix}, \tag{21i}$$
$$b_U^{l+1} = \begin{bmatrix} \mathbf{0} & Cc_U^l \\ \mathbf{0} & b_U^l \end{bmatrix}, \tag{21j}$$
$$b_0^{l+1} = \begin{bmatrix} Cc_0^l \\ b_0^l \end{bmatrix}. \tag{21k}$$

*Proof.* We first consider that at time $k$ a state estimate is given by (17) and (18). The Theorem follows by applying (5) with the CCG operations defined in Definition 2, where $W$ and $V$ are given by (6). □

Based on Theorem 1, the observer proposed in this paper consists of selecting a fixed horizon $N$ and pre-computing the set $\mathfrak{C}_X^N$ and matrices $G^N$, $G_0^N$, $c^N$, $c_U^N$, $c_0^N$, $A^N$, $A_0^N$, $b^N$, $b_U^N$, and $b_0^N$, offline with Algorithm 1.

---

**Algorithm 1** Pre-computation of CCG parameters

---

**Require:** $G^0, c_U^0, A^0, A_0^0, b^0, b_U^0, b_0^0 = []$; $\mathfrak{C}_X^0$ is given by (19); $G_0^0, c^0, c_0^0$ are given by (20)
1: **for** $l \leftarrow 0$ to $N-1$ **do**
2: compute $\mathfrak{C}_X^{l+1}$, $G^{l+1}$, $G_0^{l+1}$, $c^{l+1}$, $c_U^{l+1}$, $c_0^{l+1}$, $A^{l+1}$, $A_0^{l+1}$, $b^{l+1}$, $b_U^{l+1}$, $b_0^{l+1}$ with (21)
3: **end for**
4: **return** $\mathfrak{C}_X^N$, $G^N$, $G_0^N$, $c^N$, $c_U^N$, $c_0^N$, $A^N$, $A_0^N$, $b^N$, $b_U^N$, $b_0^N$

---

After obtaining matrices $G^N$, $G_0^N$, $c^N$, $c_U^N$, $c_0^N$, $A^N$, $A_0^N$, $b^N$, $b_U^N$, and $b_0^N$ with Algorithm 1, at runtime, for $k \geq N$, the observer consists of the Algorithm 2.

**Algorithm 2** Explicit finite-horizon observer

**Require:** $\mathfrak{C}_X^N$, $G^N$, $G_0^N$, $c^N$, $c_U^N$, $c_0^N$, $A^N$, $A_0^N$, $b^N$, $b_U^N$, $b_0^N$, $L$, $a$, $e_{init}$ $e_{noise}$, $\hat{x}_0$

**for** $0 \leq k < N$ **do**

$\quad X_k = \left(\left(a^k e_{init} + \frac{e_{noise}}{1-a}\right) P^{-\frac{1}{2}}, \hat{x}_k, [\,], [\,], \|\xi\|_2 \leq 1\right)$

**end for**

**for** $k \geq N$ **do**

$\quad \hat{x}_{k-N+1} = F\hat{x}_{k-N} + Bu_{k-N} + L\left(y_{k-N} - C\hat{x}_{k-N}\right)$,

$\quad$ compute $Y_{k+1}^N$ by storing $y_k$ and discarding $y_{k-N}$

$\quad$ compute $U_{k+1}^N$ by storing $u_k$ and discarding $u_{k-N}$

$\quad G_{X,k+1}^N = \left[G^N \quad \left(a^{k+1-N} e_{init} + \frac{e_{noise}}{1-a}\right) G_0^N\right]$

$\quad c_{X,k+1}^N = c^N + c_U^N U_{k+1}^N + c_0^N \hat{x}_{k+1-N}$

$\quad A_{X,k+1}^N = \left[A^N \quad \left(a^{k+1-N} e_{init} + \frac{e_{noise}}{1-a}\right) A_0^N\right]$

$\quad b_{X,k+1}^N = Y_{k+1}^N + b^N + b_U^N U_{k+1}^N + b_0^N \hat{x}_{k+1-N}$

$\quad X_{k+1} = \left(G_{X,k+1}^N, c_{X,k+1}^N, A_{X,k+1}^N, b_{X,k+1}^N, \mathfrak{C}_X^N\right)$

**end for**

With Algorithm 2, to obtain the description of a guaranteed state estimate set we only have to perform a small number of computations proportional to the horizon length $N$, which may be significantly more efficient than performing (5) recursively. Given the finite-horizon nature of the algorithm, this approach is more conservative than applying (5) recursively. However, by increasing the horizon $N$ the introduced conservatism tends to disappear.

We have to remark that to apply this method for $k < N$ would imply storing in memory all the coefficients from $l = 1$ to $l = N-1$. However, it greatly increases the memory requirements for large $N$ and it would only have an effect in a small transient period. For that reason, we consider that for $k < N$ the state estimate is obtained with (14).

With the description of set $X_k^N$, an important operation is obtaining an estimate of the centre of the set. This can be done with an optimization algorithm by estimating the centre $x_k^{center}$ as

$$x_k^{center} = c_{X,k}^N + G_{X,k}^N \operatorname{argmin}_{A_{X,k}^N \xi = b_{X,k}^N} \|\xi\|_2 \tag{22}$$

Alternatively, this can be computed algebraically as follows

$$x_k^{center} = c_{X,k}^N + G_{X,k}^N A_{X,k}^{N,\intercal} \eta_k, \tag{23}$$

where $\eta_k$ is computed by solving the linear equation

$$A_{X,k}^N A_{X,k}^{N,\intercal} \eta_k = b_{X,k}^N. \tag{24}$$

Given that $a^k$ tends to zero, one may neglect the term $a^{k-N} e_{init}$ after some time. Therefore, we may consider that

$$G_{X,k}^N \approx G_X^N := \left[G^N \quad \tfrac{e_{noise}}{1-a} G_0^N\right], \tag{25a}$$
$$A_{X,k}^N \approx A_X^N := \left[A^N \quad \tfrac{e_{noise}}{1-a} A_0^N\right], \tag{25b}$$

and we can pre-compute the matrix

$$Z_X^N := G_X^N A_X^{N,\intercal} \left(A_X^N A_X^{N,\intercal}\right)^{-1}, \tag{26}$$

obtaining significant computational time savings in the computation of the CCG center as

$$x_k^{center} = c_{X,k}^N + Z_X^N b_{X,k}^N. \tag{27}$$

## IV. Numerical Results

To assess the performance of the proposed algorithm we consider a random system generated with the MATLAB function *drss* with dimension 15, an output of size 3 and input of size 5, that is, $x_k \in \mathbb{R}^{15}$, $u_k \in \mathbb{R}^5$ and $y_k \in \mathbb{R}^3$, for all $k \geq 0$. We consider that the initial state is drawn from an initial state which is a CCG given by

$$X_0 = (G_{X,0}, c_{X,0}, [\,], [\,], \mathfrak{C}_{X,0}) \subset \mathbb{R}^n, \tag{28}$$

where

$$\mathfrak{C}_{X,0} = \{\xi : \|\xi\|_\infty \leq 1\} \times \{\xi : \|\xi\|_2 \leq 1\}, \tag{29a}$$
$$G_{X,0} = \left[2I_{15} \quad I_{15}\right], \tag{29b}$$
$$c_{X,0} = \mathbf{0}_{15}. \tag{29c}$$

The disturbance and noise sets are expressed as (6) with parameters

$$\mathfrak{C}_W = \{\xi : \|\xi\|_\infty \leq 1\} \times \{\xi : \|\xi\|_2 \leq 1\}, \tag{30a}$$
$$G_W = \left[2I_{15} \quad I_{15}\right], \tag{30b}$$
$$c_W = \mathbf{0}_{15}, \tag{30c}$$
$$\mathfrak{C}_V = \{\xi : \|\xi\|_\infty \leq 1\} \times \{\xi : \|\xi\|_2 \leq 1\}, \tag{30d}$$
$$G_V = \left[I_3 \quad 2I_3\right], \tag{30e}$$
$$c_V = \mathbf{0}_3. \tag{30f}$$

The control input is constant and given by $u_k = 20\mathbf{1}_5$ for all $k \geq 0$.

Figure 1 shows the evolution in time of the projection of the first coordinate of the state estimate obtained with the ellipsoidal method of (14) (Ellipsoidal), the standard description obtained by applying recursively (5) (Standard), and the method proposed in this paper for various horizons $N$. From Figure 1, we observe that the performance of the algorithm approaches that of the standard case for large $N$.

In Figure 2, we plot the projection in the first two dimensions of the state estimate obtained with various methods and for different horizons $N$. As in Figure 1, we observe that the performance of the algorithm approaches that of the standard observer for large $N$. This fact can also be observed in Figure 3 which shows the size of the projection in the first dimension of the state estimate.

Figure 4 shows the time to compute the description of the set at runtime with an Intel Core i7-12700H processor at 2.70 GHz. From Figure 4, we can observe that the computation times are significantly more competitive with the method proposed in this paper since most of the matrix computations are done offline.

The most significant advantage of the method proposed in this paper is the fact that the set description size remains constant. Therefore, as shown in Figure 5 while with the standard method, the computation time increases at every iteration, with the method of Algorithm 2 the computation time remains constant.

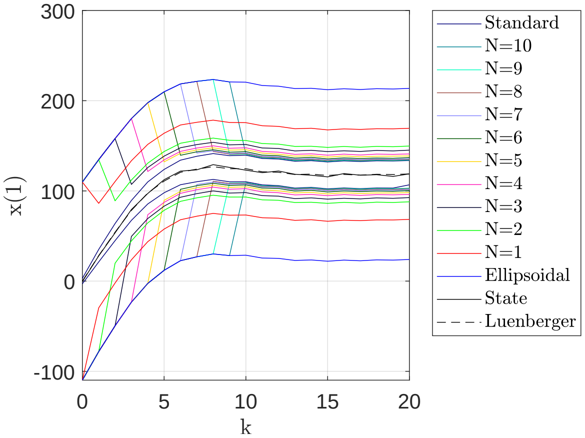


Fig. 1. Time plot of state enclosures for diverse state estimation methods for the first coordinate of the state. The black line in the middle represents the system's actual state, while the dashed line indicates the Luenberger state estimates $\hat{x}_k$.

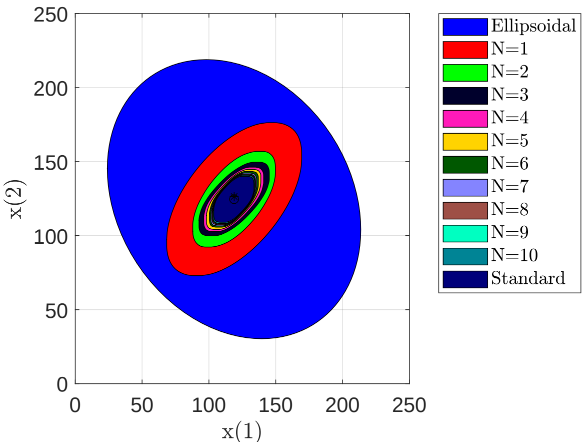


Fig. 2. State enclosures for various state estimation techniques when $k$ is set to 20. The asterisk ($*$) represents the Luenberger state estimate $\hat{x}_k$ and the circle ($\circ$) represents the true state of the system.

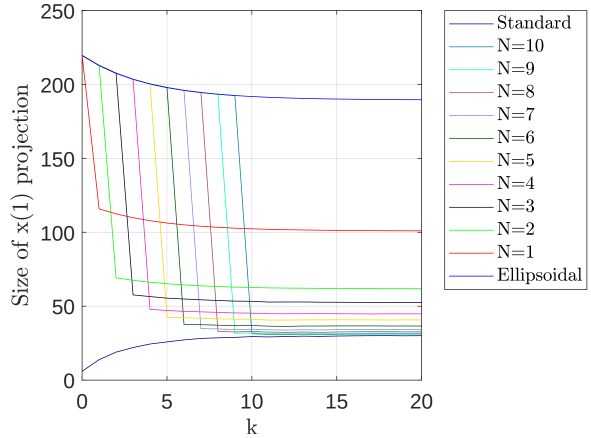


Fig. 3. Size of the projection of the first dimension of the state at various iterations for different state estimation methods and different horizons $N$.

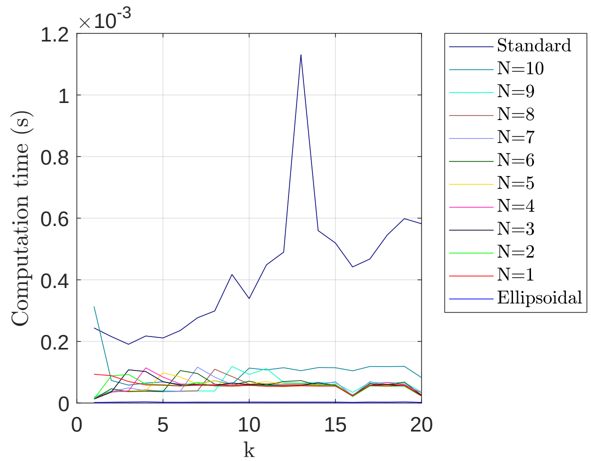


Fig. 4. Computation times for various state estimation methods.

Figure 6 shows the computation times for various centre computation methods for the standard observer and horizons $N = 1$ and $N = 10$. We tested the method of centre computation of solving (22) with YALMIP and the MOSEK solver [28] (Opt), the algebraic method of (23) (Alg), and the method with pre-computed matrices of (27) (Pre). For improved efficiency, for the optimization approach, we adopted the simplification (25) and used the function *optimizer* to pre-compile the optimization algorithm. From Figure 6 we observe that it is significantly more advantageous to compute the relevant matrices beforehand, instead of solving a linear equation at every time.

To highlight the main advantage of the proposed method, we used the same simulation for a larger number of iterations with the results being depicted in Figure 7. Since the description of the state estimate increases in size at each iteration, the computation of the center becomes more time-consuming, whereas, the proposed method benefits from the constant description and pre-computation of parts of the data structures being done offline. We remark that the presented method voids the need for an order reduction procedure, which is going to add conservatism and represent a time overhead.

## V. Conclusions

We proposed a novel, minimally conservative, and computationally efficient method for guaranteed state estimation of discrete-time linear time-invariant (LTI) systems, which uses CCGs. Additionally, we propose a method for computing the CCG centre using pre-calculated matrix inversions that is much quicker than alternative methods. The performance of the computations of the suggested technique is demonstrated by numerical results. In future work, we aim at developing a fast implementation using C code that can be used by the community and thus bridging one of the main drawbacks of guaranteed state estimation in comparison with Luenberger observers.

## References

[1] M. Pourasghar, V. Puig, and C. Ocampo-Martinez, "Interval observer versus set-membership approaches for fault detection in uncertain systems using zonotopes," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 10, pp. 2819–2843, 2019.

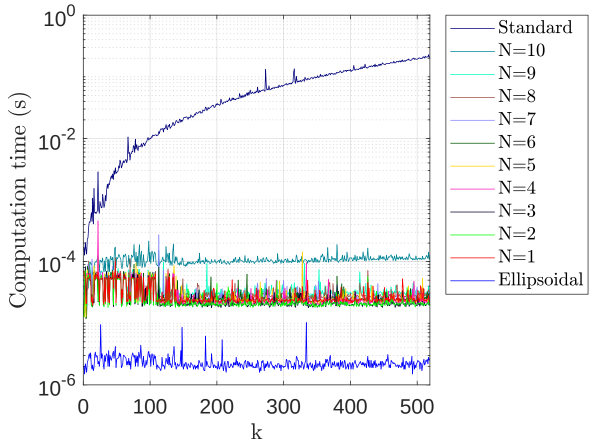


Fig. 5. Computation times until $k = 500$.

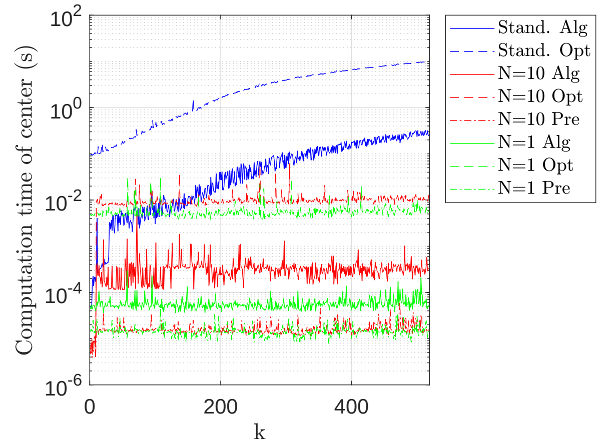


Fig. 7. Center computation times until $k = 500$.

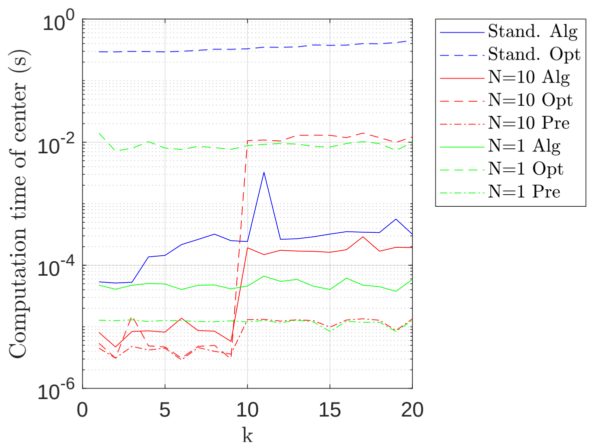


Fig. 6. Center computation times for various state estimation methods and various centre computation methods.


[2] M. Althoff and J. J. Rath, "Comparison of guaranteed state estimators for linear time-invariant systems," *Automatica*, vol. 130, p. 109662, 2021.

[3] D. Silvestre, "Accurate guaranteed state estimation for uncertain lpvs using constrained convex generators," in *IEEE 61st Conference on Decision and Control (CDC)*, 2022, pp. 4957–4962.

[4] D. Silvestre, P. Rosa, J. P. Hespanha, and C. Silvestre, "Stochastic and deterministic fault detection for randomized gossip algorithms," *Automatica*, vol. 78, pp. 46–60, 2017.

[5] P. Rosa, C. Silvestre, J. S. Shamma, and M. Athans, "Multiple-model adaptive control with set-valued observers," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 28th Chinese Control Conference*. IEEE, 2009, pp. 2441–2447.

[6] X. Ping, S. Yang, B. Ding, T. Raissi, and Z. Li, "Observer-based output feedback robust mpc via zonotopic set-membership state estimation for lpv systems with bounded disturbances and noises," *Journal of the Franklin Institute*, vol. 357, no. 11, pp. 7368–7398, 2020.

[7] J. Lin, H. Zhu, and J. Alonso-Mora, "Robust vision-based obstacle avoidance for micro aerial vehicles in dynamic environments," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 2682–2688.

[8] Z. Marvi and B. Kiumarsi, "Safe reinforcement learning: A control barrier function optimization approach," *International Journal of Robust and Nonlinear Control*, vol. 31, no. 6, pp. 1923–1940, 2021.

[9] F. Schweppe, "Recursive state estimation: Unknown but bounded errors and system inputs," *IEEE Transactions on Automatic Control*, vol. 13, no. 1, pp. 22–28, 1968.

[10] L. Chisci, A. Garulli, and G. Zappa, "Recursive state bounding by parallelotopes," *Automatica*, vol. 32, no. 7, pp. 1049–1055, 1996.

[11] J. S. Shamma and K.-Y. Tu, "Set-valued observers and optimal disturbance rejection," *IEEE Transactions on Automatic Control*, vol. 44, no. 2, pp. 253–264, 1999.

[12] T. Alamo, J. M. Bravo, and E. F. Camacho, "Guaranteed state estimation by zonotopes," *Automatica*, vol. 41, no. 6, pp. 1035–1043, 2005.

[13] F. Blanchini and S. Miani, *Set-theoretic methods in control*. Springer, 2008, vol. 78.

[14] L. Jaulin, "Robust set-membership state estimation; application to underwater robotics," *Automatica*, vol. 45, no. 1, pp. 202–206, 2009.

[15] A. Kurzhanski and P. Varaiya, "Ellipsoidal techniques for reachability analysis: internal approximation," *Systems & Control Letters*, vol. 41, no. 3, pp. 201–211, Oct. 2000.

[16] F. Chernousko, "Ellipsoidal state estimation for dynamical systems," *Nonlinear Analysis: Theory, Methods & Applications*, vol. 63, no. 5-7, pp. 872–879, Nov. 2005.

[17] T. Alamo, J. Bravo, and E. Camacho, "Guaranteed state estimation by zonotopes," *Automatica*, vol. 41, no. 6, pp. 1035–1043, Jun. 2005.

[18] C. Combastel, "Zonotopes and kalman observers: Gain optimality under distinct uncertainty paradigms and robust convergence," *Automatica*, vol. 55, pp. 265–273, 2015.

[19] S. Wang and W. Ren, "On the Convergence Conditions of Distributed Dynamic State Estimation Using Sensor Networks: A Unified Framework," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 4, pp. 1300–1316, Jul. 2018.

[20] F. Mazenc, T. N. Dinh, and S. I. Niculescu, "Interval observers for discrete-time systems," *International journal of robust and nonlinear control*, vol. 24, no. 17, pp. 2867–2890, 2014.

[21] 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.

[22] D. Silvestre and G. Ramos, "Model predictive control with collision avoidance for unknown environment," *IEEE Control Systems Letters*, vol. 7, pp. 2821–2826, 2023.

[23] D. Silvestre, "Exact set-valued estimation using constrained convex generators for uncertain linear systems," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 9461–9466, 2023, 22nd IFAC World Congress.

[24] D. Bertsekas and I. Rhodes, "Recursive state estimation for a set-membership description of uncertainty," *IEEE Transactions on Automatic Control*, vol. 16, no. 2, pp. 117–128, 1971.

[25] H. Witsenhausen, "Sets of possible states of linear systems given perturbed observations," *IEEE Transactions on Automatic Control*, vol. 13, no. 5, pp. 556–558, 1968.

[26] F. Chernous'ko, "Ellipsoidal estimates of a controlled system's attainability domain," *Journal of Applied Mathematics and Mechanics*, vol. 45, no. 1, pp. 7–12, 1981.

[27] F. F. C. Rego, "Distributed state estimation and cooperative path-following under communication constraints," Ph.D. dissertation, Ecole Polytechnique Fédérale de Lausanne, 2018.

[28] M. ApS, *The MOSEK optimization toolbox for MATLAB manual. Version 9.0.*, 2019. [Online]. Available: http://docs.mosek.com/9.0/toolbox/index.html