2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Zonotope Bundles for the Efficient Computation of Reachable Sets

Matthias Althoff and Bruce H. Krogh

Abstract— **We introduce zonotope bundles for computing the set of states reachable by a dynamical system, also known as the *reachable set*. Reachable set computations suffer from the curse of dimensionality, which has been successfully addressed by using zonotopes for linear systems. However, zonotopes are not closed under intersection leading to challenges when applying them to nonlinear and hybrid problems. We introduce zonotope bundles as the intersection of zonotopes (without explicitly computing the intersection). Zonotope bundles are closed under intersection, while inheriting many positive properties of zonotopes. This is demonstrated for linear, nonlinear, and hybrid systems. A further property of zonotope bundles is that their computation can be easily parallelized.**

I. Introduction

The reachable set for a dynamical system is the set of all states reached by all possible trajectories of a system starting from a specified set of initial states, under the influence of uncertain inputs and parameters. Safety verification is one of the most widely used applications of reachability analysis by demonstrating that no sets specified as unsafe can be reached. The avoidance of unsafe sets cannot be guaranteed by selected simulations, since the trajectory hitting an unsafe set might not be found. Besides the safety verification problem, reachability analysis is also useful for robustness analysis [1], abstraction of hybrid systems [2], invariant set computation [3], and state-bounding observers [4], [5].

Zonotopes are an efficient set representation for many reachability problems, but suffer from the fact that they are not closed under intersection (i.e. the intersection of zonotopes is not a zonotope in general), which is important for nonlinear and hybrid systems reachability analysis. This paper introduces zonotope bundles, which refer to the intersection of a collection of zonotopes, making them closed under intersection.

Several representations have been used for reachable sets. An important class of set representations is the class of polyhedra. When the continuous dynamics is described as $\dot{x} \in P$, where $x \in \mathbb{R}^n$ and P is a bounded convex polyhedron (i.e. a polytope), the reachable set can be exactly represented by polyhedra, which also holds for linear hybrid automata [6], [7]. For more complicated dynamics, such as linear time invariant (LTI) systems $\dot{x} = Ax + v(t)$, where $x \in \mathbb{R}^n$, $v(t) \in V \subset \mathbb{R}^n$, $A \in \mathbb{R}^{n\times n}$, the reachable set cannot be exactly computed in general [8], but always without the wrapping effect [9], where the wrapping effect is understood as the propagation of overapproximations through successive time steps. Possible set representations for the overapproximation from the class of polyhedra are polytopes [10], zonotopes [11], [12], oriented rectangular hulls [13], simplicies [14], and multidimensional intervals [15], [16].

Other representations, which are not from the class of polyhedra, are ellipsoids [17], support functions [18] (which can represent any convex set), and level sets [19] (which can represent any bounded set). Although ellipsoids offer a compact representation of reachable sets, they suffer from not being closed under Minkowski addition and intersection. The idea of computing with several instances of a set representation, similar to the concept in this paper, is described for ellipsoids in [20].

We will first recall *polytopes* and *zonotopes* and operations on them in Sec. II. Next, *zonotope bundles* are introduced in Sec. III. It will be shown that no required operation on zonotope bundles has complexity greater than $\mathcal{O}(n^3)$ with respect to the system dimension n. The gained accuracy when computing with zonotope bundles compared to zonotopes is demonstrated for linear systems (Sec. IV) when the initial set is not a zonotope. For nonlinear systems (Sec. V) it is shown that the capability of splitting zonotope bundles makes it possible to verify an evasive maneuver of a car while zonotopes and polytopes fail. For hybrid systems (Sec. VI) it is demonstrated that enclosures with guard sets can be much more accurately obtained with zonotope bundles compared to zonotopes, while polytopes are infeasible for dimensions greater than 4. Other advantages of zonotope bundles, such as parallelization, are summarized in Sec. VII.

II. Polytopes and Zonotopes

We first recall the definitions of polytopes and zonotopes and then discuss the complexity of the required operations for reachability analysis of linear systems. The most general set representation considered in this work are convex polytopes, for which two representations exist (Fig. 1).

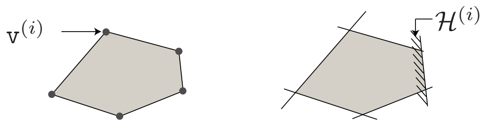
Definition 1 (H-Representation of a Polytope): A convex polytope $\mathcal{P}$ is the intersection of q halfspaces $\mathcal{H}^{(i)} = \{x \in \mathbb{R}^n | e_i\, x \leq d_i,\ e_i \in \mathbb{R}^{1\times n},\ d_i \in \mathbb{R}\}$, such that $\mathcal{P} = \{x \in \mathbb{R}^n \big| E\, x \leq d, \quad E \in \mathbb{R}^{q\times n}, d \in \mathbb{R}^q\}$. □

Definition 2 (V-Representation of a Polytope): Given r vertices $\mathtt{v}^{(i)} \in \mathbb{R}^n$, $\mathcal{P} = \mathtt{CH}(\mathtt{v}^{(1)}, \dots, \mathtt{v}^{(r)})$ is a convex polytope, where $\mathtt{CH}()$ is the convex hull operator. □

Zonotopes are a special case of polytopes. A polytope is a zonotope if it can also be represented by so-called generators (G-representation).

Matthias Althoff and Bruce H. Krogh are with the Department of Electrical and Computer Engineering, Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA 15213. (e-mail: malthoff@ece.cmu.edu, krogh@ece.cmu.edu).

978-1-61284-799-3/11/$26.00 ©2011 IEEE



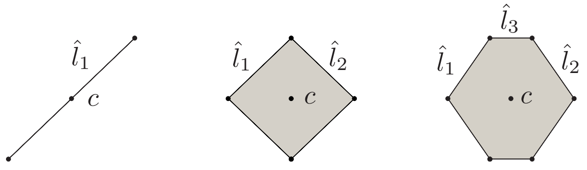
(a) V-repr. (b) H-repr.

Fig. 1. Possible representations of a polytope.

Definition 3 (G-Representation of a Zonotope): Denoting the center by $c \in \mathbb{R}^n$ (to which a zonotope is centrally symmetric) and the i^{th} generators by $g^{(i)} \in \mathbb{R}^n$, a zonotope is defined as

$$\mathcal{Z} = \Big\{ x \in \mathbb{R}^n \Big| x = c + \sum_{i=1}^{p} \beta_i \, g^{(i)}, \quad -1 \leq \beta_i \leq 1 \Big\}. \qquad \square$$

The definition can be interpreted as the Minkowski sum[1] of a finite set of line segments $\hat{l}_i = [-1,1]\, g^{(i)}$. Figure 2 illustrates how a zonotope is built step-by-step. The order of a zonotope is defined as $o = \frac{p}{n}$. If the order is less than one, the zonotope represents a set of lower dimension than n (Fig. 2(a)) and a zonotope of order one is a parallelotope (Fig. 2(b)). Zonotopes of order greater than one create sets with an increasing number of facets and vertices (Fig. 2(c)). A zonotope is denoted in a concise way by the list of its center and generators: $\mathcal{Z} = (c, g^{(1)}, \ldots, g^{(p)})$.



(a) $c \oplus \hat{l}_1$ (b) $c \oplus \hat{l}_1 \oplus \hat{l}_2$ (c) $c \oplus \hat{l}_1 \oplus \hat{l}_2 \oplus \hat{l}_3$

Fig. 2. Construction of a zonotope by Minkowski addition of line segments.

The representation of reachable sets does not only have to be compact, but more importantly, relevant operations have to be efficient with respect to the system dimension n, which are: linear transformation, Minkowski addition, box enclosure, and convex hull computation of linearly transformed sets [9]. For polytopes, Minkowski addition and convex hull computation, which is denoted by $opCH()$, are generally limited to problems with up to $4-6$ dimensions [21], [22] and tend to run into numerical problems unless infinite precision arithmetic is used [23]. Unlike polytopes, zonotopes are numerically stable and operations for reachability analysis have a maximum complexity of $\mathcal{O}(n^3)$. For $\mathcal{Z}_1 = (c_1, g^{(1)}, \ldots, g^{(p)})$, $Z_2 = (c_2, h^{(1)}, \ldots, h^{(u)})$, and $L \in \mathbb{R}^{n \times n}$, the required operations are computed according to [12] as:

$$\begin{aligned}
L\,\mathcal{Z}_1 :=& \{L\,x | x \in \mathcal{Z}_1\} = (Lc_1, Lg^{(1)}, \ldots, Lg^{(p)}), \\
\mathcal{Z}_1 \oplus \mathcal{Z}_2 :=& \{a + b | a \in \mathcal{Z}_1, b \in \mathcal{Z}_2\} \\
=& (c_1 + c_2, g^{(1)}, \ldots, g^{(p)}, h^{(1)}, \ldots, h^{(u)}), \\
\mathtt{CH}(\mathcal{Z}_1 \cup L\mathcal{Z}_1) :=& \{\alpha a + (1-\alpha) b | \\
& a \in \mathcal{Z}_1, b \in L\mathcal{Z}_1, \alpha \in [0,1]\} \\
\subseteq& 0.5(\tilde{L}c_1, \tilde{L}g^{(1)}, \ldots, \tilde{L}g^{(p)}, \\
& \hat{L}c_1, \hat{L}g^{(1)}, \ldots, \hat{L}g^{(p)}), \\
& \tilde{L} = (I + L), \quad \hat{L} = (I - L), \\
\mathtt{box}(\mathcal{Z}_1) =& [c_1 - \delta, c_2 + \delta], \quad \delta = \sum_{i=1}^{p} |g^{(i)}|,
\end{aligned} \tag{1}$$

where I is the identity matrix, the absolute value $|g^{(i)}|$ is computed elementwise, and $\mathtt{box}(\mathcal{Z}_1)$ is specified using interval notation. The convex hull operation has to be overapproximated since it results in a polytope in general. With respect to the system dimension, the complexity is only $\mathcal{O}(n)$ for $\mathcal{Z}_1 \oplus \mathcal{Z}_2$, $\mathtt{box}(\mathcal{Z}_1)$, and $\mathcal{O}(n^3)$ for $L\,\mathcal{Z}_1$, $\mathtt{CH}(\mathcal{Z}_1 \cup L\mathcal{Z}_1)$.

III. Zonotope Bundles

We define a zonotope bundle as the intersection of zonotopes:

Definition 4 (Zonotope Bundle): Given a finite set of zonotopes $\mathscr{Z}$, a zonotope bundle is $\mathscr{Z}^\cap = \{\bigcap_{i=1}^{s} \mathcal{Z}_i | \mathcal{Z}_i \in \mathscr{Z}\}$, i.e. the intersection of zonotopes $\mathcal{Z}_i$. Note that the intersection is not computed, but the zonotopes $\mathcal{Z}_i$ are stored in a list, which we write as $\mathscr{Z}^\cap = \{\mathcal{Z}_1, \ldots, \mathcal{Z}_s\}^\cap$. □

We show that operations in (1) can be performed in an exact or overapproximate way on zonotope bundles by separately performing them for each zonotope. Thus, the computational cost is the one for zonotopes times the number of zonotopes in the bundle, so that the maximum complexity of $\mathcal{O}(n^3)$ is inherited. In addition, we provide an upper bound for the overapproximation of the Minkowski addition.

Proposition 1 (Linear Transformation): The linear transformation of a zonotope bundle $L\,\mathscr{Z}^\cap$, where L has full rank is performed by

$$L\,\mathscr{Z}^\cap = \{L\,\mathcal{Z}_1, \ldots, L\,\mathcal{Z}_s\}^\cap.$$

When L does not have full rank, the computation is overapproximative: $L\,\mathscr{Z}^\cap \subseteq \{L\,\mathcal{Z}_1, \ldots, L\,\mathcal{Z}_s\}^\cap$. □

Proof: We want to show that $L \bigcap_{i=1}^{s} \mathcal{Z}_i = \bigcap_{i=1}^{s} L\,\mathcal{Z}_i$, for which it is sufficient to shown that $L(A \cap B) = L\,A \cap L\,B$ when L has full rank:

$$\begin{aligned}
x \in A \cap B \leftrightarrow& x \in A \text{ and } x \in B \\
\leftrightarrow& L\,x \in L\,A \text{ and } L\,x \in L\,B \\
\leftrightarrow& L\,x \in L\,A \cap L\,B.
\end{aligned}$$

When L does not have full rank, it only holds that if $x \in A$ and $x \in B \rightarrow L\,x \in L\,A$ and $L\,x \in L\,B$. □

[1] Minkowski sum of two sets: $A \oplus B = \{a + b | a \in A, b \in B\}$.

Proposition 2 (Minkowski Addition): The Minkowski addition of a zonotope bundle $\mathscr{Z}^\cap$ and a zonotope $\mathcal{Z}^{add}$ is overapproximated by

$$\mathscr{Z}^\cap \oplus \mathcal{Z}^{add} \subseteq \{\mathcal{Z}_1 \oplus \mathcal{Z}^{add}, \dots, \mathcal{Z}_s \oplus \mathcal{Z}^{add}\}^\cap. \qquad \square$$

Proof: We want to show that $\left(\bigcap_{i=1}^s \mathcal{Z}_i\right) \oplus \mathcal{Z}^{add} \subseteq \bigcap_{i=1}^s \left(\mathcal{Z}_i \oplus \mathcal{Z}^{add}\right)$, for which it is sufficient to shown that $(A \cap B) \oplus C \subseteq (A \oplus C) \cap (B \oplus C)$. After defining $x = y + c$, where $y \in A \cap B$, $c \in C$, we have:

$$\begin{aligned} y \in A \cap B \rightarrow & y \in A \text{ and } y \in B \\ \rightarrow & x \in (A \oplus C) \text{ and } x \in (B \oplus C) \\ \rightarrow & x \in (A \oplus C) \cap (B \oplus C). \end{aligned}$$

It is shown that $(A \cap B) \oplus C \neq (A \oplus C) \cap (B \oplus C)$ by a scalar counter-example: $A = [-1, -0.5]$, $B = [0.5, 1]$, $C = [-1, 1]$ so that $(A \cap B) \oplus C$ is undefined while $(A \oplus C) \cap (B \oplus C) = [-0.5, 0.5]$. □

In order to present an upper bound on the overapproximation of the Minkowski addition procedure, an alternative overapproximation is considered by pushing the facets of the set outwards. This is first considered for zonotopes and then for zonotope bundles:

Proposition 3 (Facet Lifting of Zonotopes): The addition of a zonotope in H-representation $\mathcal{Z} = \{x | Ex \leq d\}$ with a zonotope $\mathcal{Z}^{add}$ can be overapproximated by pushing the facets of $\mathcal{Z}$ outwards:

$$\mathcal{Z} \uparrow \mathcal{Z}^{add} := \{x | Ex \leq \tilde{d}\}, \quad \tilde{d}_i = d_i + \max(e_i \mathcal{Z}^{add}),$$

where e_i is the i^{th} row vector of E and $\mathcal{Z} \oplus \mathcal{Z}^{add} \subseteq \mathcal{Z} \uparrow \mathcal{Z}^{add}$. □

Proof: The Minkowski addition of a halfspace $\mathcal{H}^{(i)} = \{x \in \mathbb{R}^n | e_i x \leq d_i, e_i \in \mathbb{R}^{1 \times n}, d_i \in \mathbb{R}\}$ is computed exactly by pushing the dividing hyperplane outside: $\mathcal{H}^{(i)} \oplus \mathcal{Z}^{add} = \mathcal{H}^{(i)} \uparrow \mathcal{Z}^{add}$, see Fig. 3(a). The independent application results in an overapproximation:

$$\underbrace{\left(\bigcap_{i=1}^q \mathcal{H}^{(i)}\right)}_{=\mathcal{Z}} \oplus \mathcal{Z}^{add} \overset{\substack{\text{Proof of} \\ \text{Prop. 2}}}{\subseteq} \bigcap_{i=1}^q \underbrace{\left(\mathcal{H}^{(i)} \oplus \mathcal{Z}^{add}\right)}_{=\mathcal{H}^{(i)} \uparrow \mathcal{Z}^{add}} = \mathcal{Z} \uparrow \mathcal{Z}^{add}.$$

□

Proposition 4 (Facet Lifting of Zonotope Bundles): The facet lifting of a zonotope bundle $\mathscr{Z}^\cap$ by a zonotope $\mathcal{Z}^{add}$ is exactly obtained by

$$\mathscr{Z}^\cap \uparrow \mathcal{Z}^{add} = \{\mathcal{Z}_1 \uparrow \mathcal{Z}^{add}, \dots, \mathcal{Z}_s \uparrow \mathcal{Z}^{add}\}^\cap. \qquad \square$$

Proof: The proof is straightforward by rewriting the expressions using the halfspace notation:

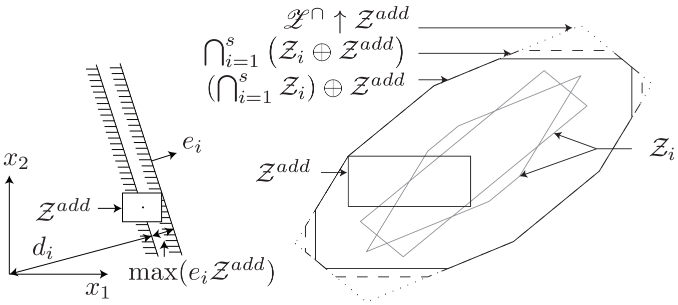
$$\bigcap_{i=1}^s \mathcal{Z}_i = \begin{array}{c} E^{(1)}x \leq d^{(1)} \\ E^{(2)}x \leq d^{(2)} \\ \vdots \\ E^{(s)}x \leq d^{(s)} \end{array} \xrightarrow{\uparrow \mathcal{Z}^{add}} \begin{array}{c} E^{(1)}x \leq \tilde{d}^{(1)} \\ E^{(2)}x \leq \tilde{d}^{(2)} \\ \vdots \\ E^{(s)}x \leq \tilde{d}^{(s)} \end{array}.$$

The right column is equivalent to $\{\mathcal{Z}_1 \uparrow \mathcal{Z}^{add}, \dots, \mathcal{Z}_s \uparrow \mathcal{Z}^{add}\}^\cap$. □

After combining Prop. 3 and Prop. 4 it follows that

$$\bigcap_{i=1}^s (\mathcal{Z}_i \oplus \mathcal{Z}^{add}) \overset{Prop.3}{\subseteq} \bigcap_{i=1}^s (\mathcal{Z}_i \uparrow \mathcal{Z}^{add}) \overset{Prop.4}{=} \mathscr{Z}^\cap \uparrow \mathcal{Z}^{add}$$

so that Prop. 2 is always better than facet lifting as is shown for an exemplary zonotope bundle in Fig. 3(b). Moreover, the Minkowski addition is computationally less expensive for zonotopes ($\mathcal{O}(n)$, see (1)), while face lifting requires projections for each normal vector ($\mathcal{O}(n^3)$).



x_2
$\mathcal{Z}^{add}$
e_i
d_i
x_1
$\max(e_i \mathcal{Z}^{add})$
$\mathscr{Z}^\cap \uparrow \mathcal{Z}^{add}$
$\bigcap_{i=1}^s \left(\mathcal{Z}_i \oplus \mathcal{Z}^{add}\right)$
$\left(\bigcap_{i=1}^s \mathcal{Z}_i\right) \oplus \mathcal{Z}^{add}$
$\mathcal{Z}^{add}$
$\mathcal{Z}_i$

(a) Halfspace translation.

(b) Comparison of Minkowski addition overapproximations.

Fig. 3. Minkowski addition of a zonotope bundle and a zonotope.

Proposition 5 (Convex Hull Computation): The convex hull of a zonotope bundle $\mathscr{Z}^\cap$ and its map $L\,\mathscr{Z}^\cap$ is overapproximated by

$$\mathtt{CH}(\mathscr{Z}^\cap \cup L\,\mathscr{Z}^\cap) \subseteq \{\mathtt{CH}(\mathcal{Z}_1 \cup L\,\mathcal{Z}_1), \dots, \mathtt{CH}(\mathcal{Z}_s \cup L\,\mathcal{Z}_s)\}^\cap.$$

□

Proof: It is sufficient to shown that

$$\mathtt{CH}((A \cap B) \cup L\,(A \cap B)) \subseteq \mathtt{CH}(A \cup L\,A) \cap \mathtt{CH}(B \cup L\,B)).$$

Define $y \in \mathtt{CH}((A \cap B) \cup L\,(A \cap B))$, then

$$\begin{aligned} \exists \alpha : y \in & \alpha\,(A \cap B) \oplus (1-\alpha)L(A \cap B) \\ \overset{\text{Prop. 1}}{\subseteq} & (\alpha\,A \cap \alpha\,B) \oplus ((1-\alpha)L\,A \cap (1-\alpha)L\,B) \\ \overset{\text{Prop. 2}}{\subseteq} & \underbrace{(\alpha\,A \oplus (1-\alpha)L\,A)}_{\subseteq \mathtt{CH}(A \cup L\,A)} \cap \underbrace{(\alpha B \oplus (1-\alpha)L\,B)}_{\subseteq \mathtt{CH}(B \cup L\,B)} \cap \\ & (\alpha\,B \oplus (1-\alpha)L\,A) \cap (\alpha A \oplus (1-\alpha)L\,B) \\ \subseteq & \mathtt{CH}(A \cup L\,A) \cap \mathtt{CH}(B \cup L\,B). \end{aligned}$$

□

Proposition 6 (Box Enclosure): The enclosing box of a zonotope bundle $\mathscr{Z}^\cap$ is overapproximated by

$$\mathtt{box}(\mathscr{Z}^\cap) \subseteq \{\mathtt{box}(\mathcal{Z}_1), \dots, \mathtt{box}(\mathcal{Z}_s)\}^\cap. \qquad \square$$

The results of the box enclosure are obvious and thus the proof is skipped. The intersection of two zonotope bundles is trivially done by concatenating the list of zonotopes: $\mathscr{Z}_A^\cap \cap \mathscr{Z}_B^\cap = \{\mathcal{Z}_{A,1}, \dots, \mathcal{Z}_{A,s_A}, \mathcal{Z}_{B,1}, \dots, \mathcal{Z}_{B,s_B}\}^\cap$. In order to detect if a zonotope bundle intersects an unsafe set, the following proposition is applied:

Proposition 7 (Intersection Detection): A zonotope bundle $\mathscr{Z}^\cap$ intersects a set $\mathcal{S}$ if and only if each zonotope intersects $\mathcal{S}$: $\mathscr{Z}^\cap \cap \mathcal{S} \neq \emptyset \leftrightarrow \forall i = 1 \dots s: \mathcal{Z}_i \cap \mathcal{S} \neq \emptyset$. □

This result directly follows from the associativity of intersection and is useful from a computational point of view for safety verification: Once a $\mathcal{Z}_i$ does not intersect an unsafe set, the check is completed, which is computationally cheaper than first computing the polytope representation of $\mathscr{Z}^\cap$.

A property explored further in Sec. V-A is that two zonotope bundles representing the same set might suffer more or less from overapproximations of operations, depending how tight the individual zonotopes are enclosing the intersection of all zonotopes in the bundle.

IV. Reachability Analysis of Linear Systems

We will evaluate the performance of zonotope bundles for linear systems given as $\dot{x} = Ax+Bu$, where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $x(0) \in \mathcal{R}(0) \subset \mathbb{R}^n$, $u(t) \in \mathcal{U} \subset \mathbb{R}^m$. Reachability analysis of linear systems does not require intersection of sets for which zonotope bundles are advantageous over zonotopes. Nevertheless, zonotope bundles provide tighter bounds compared to zonotopes when the initial set is not given as a zonotope. It will also be demonstrated that zonotope computations scale much better with the system dimension n compared to general polytopes.

The input set $\mathcal{U}$ is restricted to zonotopes. Typically, bounds for inputs/disturbances are given as intervals for each dimension (which is a special case of a zonotope), and if not, efficient methods exist for the enclosure of other sets by boxes (zonotope of order 1), parallelotopes (zonotope of order 1), or zonotopes.

A. Basic Procedure

Algorithm 1 shows the basic procedure for computing reachable sets for linear systems. The algorithm computes reachable sets of consecutive time intervals $[t_k, t_{k+1}]$, where $t_k := k\,r$, $k \in \mathbb{N}$ is the time step and $r \in \mathbb{R}^+$ is the time increment. The superposition principle is used to separately obtain the reachable set due to the homogeneous and input solution, denoted by $\mathcal{R}^h$ and $\mathcal{R}^i$, respectively. The computation of the first time interval of $\mathcal{R}^h([0,t_1])$ and $\mathcal{R}^i([0,t_1])$ involves linear transformation, convex hull computation, and Minkowski addition, see e.g. [24]. By arranging the computations for further time intervals according to [9], as shown in algorithm 1, the wrapping effect is avoided. The algorithm also obtains correct results when the system is unstable.

B. Numerical Results

We compare the performance of computing reachable sets with zonotope bundles to computations using zonotopes and polytopes by randomized examples. For dimensions $n = \{2,3\}$, 100 randomized systems are generated, while for $n = 4$ only the computational times are compared since polytope computations often did not terminate due to numerical problems[2].

Algorithm 1 Compute $\mathcal{R}_k := \mathcal{R}([t_k, t_{k+1}])$ for $k = 1 \dots N$

```
Input: Reachable sets $\mathcal{R}_0^h := \mathcal{R}^h([0,t_1])$, $\mathcal{R}_0^i := \mathcal{R}^i([0,t_1])$, system matrix $A$, time incr. $r$, time steps $N$.
Output: $\mathcal{R}_k$ for $k = 0 \dots N$
  $\mathcal{P}_0 = \texttt{box}(\mathcal{R}_0^i)$
  $\mathcal{R}_0 = \mathcal{R}_0^h \oplus \mathcal{P}_0$
  for $k = 1 \dots N$ do
    $\mathcal{R}_k^h = e^{Ar}\,\mathcal{R}_{k-1}^h, \quad \mathcal{R}_k^i = e^{Ar}\,\mathcal{R}_{k-1}^i$
    $\mathcal{P}_k = \mathcal{P}_{k-1} \oplus \texttt{box}(\mathcal{R}_k^i)$
    $\mathcal{R}_k = \mathcal{R}_k^h \oplus \mathcal{P}_k$
  end for
```

The state matrix A is randomly generated by uniformly sampling eigenvalues from a box in the complex plane with real values in $[-5,-0.2]$, and imaginary values in $[-5,5]$. Eigenvalues are chosen conjugate complex, unless the number of poles is uneven so that one is chosen real. The input matrix B is chosen as a $n \times n$ matrix whose entries are uniformly distributed in $[-1,1]$. The set of inputs $\mathcal{U}$ is a hypercube centered at the origin with edge length $\delta = 0.1$.

The initial set of states is randomly generated as a V-polytope with vertices $\mathtt{v}^{(i)} = d + l^{(i)}\,\varphi^{(i)}$, $d = [2,\dots,2]^T$, $\varphi^{(i)} \in \mathbb{R}^n$ is a point uniformly distributed on a unit-hypersphere, and $l^{(i)} \in \mathbb{R}$ is uniformly distributed within $[0,\mu]$, where $\mu = 1$. The number of sampled vertices is 2^n which equals the number for a hypercube in $\mathbb{R}^n$.

The time increment is chosen as $r = \|A^2\|_\infty^{-0.5}$ which is motivated by the required enlargement of reachable sets to ensure enclosure for time intervals (see [26]). The number of computed time steps $N \in \mathbb{N}$ is determined by the highest N fulfilling $\|e^{ANr}\|_2 \leq 0.1\|e^{Ar}\|_2$, which indicates a significant decay of the initial state solution.

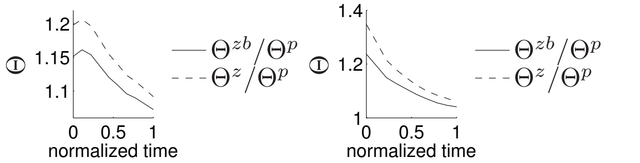
For the computation with zonotope bundles, the initial set is enclosed by a box and a parallelotope using principal component analysis (see [13] or Sec. V-A), while for zonotope computations only the parallelotope is used. The exactness of the computation is measured by a normalized volume $\Theta = V^{\frac{1}{n}}$ (V is the volume) of the reachable sets, making the results independent of n.

After normalizing the time of each example from $t = 0$ to $t = 1$, the mean value of the relative performance index Θ^z/Θ^p and Θ^{zb}/Θ^p is plotted in Fig. 4, where the superscripts refer to $\underline{z}$onotopes, $\underline{z}$onotope $\underline{b}$undles, and $\underline{p}$olytopes. It can be seen that the reachable sets are only marginally larger for zonotope computations and that the result is more accurate for zonotope bundles. Since the initial state solution becomes less dominant over time, the normalized volume ratio tends to 1.

The average computation times per time step $\bar{t}$ are presented in Table I showing that zonotope computations are much more efficient and scalable. There is a jump in the computation time from $n = 2$ to $n = 3$ for the polytope computation, since the MPT toolbox often has to search for alternative ways of computing the convex hull (the *analytical method* has been initially set). Note that the computation time

[2] used tool: MPT toolbox [25]

when using two zonotopes in the bundle is less than double since certain computations, such as the one of $\mathcal{R}_k^i$ and $\mathcal{P}_k$ (see Alg. 1) do not involve the bundle. The computations have been performed in MATLAB on an Intel i7 Processor with 1.6 GHz and 6 GB memory.



(a) Dimension $n = 2$.

(b) Dimension $n = 3$.

Fig. 4. Mean ratios of normalized volumes when comparing randomized zonotope bundle and zonotope computations with polytope computations.

TABLE I

COMPUTATION TIMES FOR RANDOMIZED LINEAR SYSTEMS.

dimension n	2	3	4	500	1000
$\bar{t}^p$ in [s]	0.0248	1.0423	14.836	—	—
$\bar{t}^z$ in [s]	0.0010	0.0011	0.0011	0.5158	5.0902
$\bar{t}^{zb}$ in [s]	0.0018	0.0019	0.0020	0.6796	7.0438

V. Reachability Analysis of Nonlinear Systems

In this section, the performance of zonotope bundles is demonstrated for nonlinear systems $\dot{x} = f(x,u)$, where $x(0) \in \mathcal{R}(0) \subset \mathbb{R}^n$, $u(t) \in \mathcal{U} \subset \mathbb{R}^{\hat{m}}$. Recently, it has been demonstrated that reachability techniques developed for linear systems can be efficiently extended to nonlinear systems by on-the-fly linearization [14], [27]. Overapproximation of the result is ensured by adding the set of linearization errors as an additional uncertain input. For many nonlinear problems, it is required to split the reachable set when the set of linearization errors is large [27]. Since zonotopes are not closed under intersection (equivalent to not being closed under splitting), they have to be overapproximated by parallelotopes (zonotopes of order 1) in order to obtain split sets in G-representation.

A. Splitting and Enclosing Zonotope Bundles

A zonotope, and thus also a zonotope bundle can be split by first overapproximating it by a parallelotope Ψ, which is obtained by defining the directions λ_i which span the parallelotope as columns in a matrix $\Lambda = [\lambda_1, \dots, \lambda_n]$:

$$\Psi := (c, w^{(1)}, \dots, w^{(n)}) = \Lambda\, \mathtt{box}(\Lambda^{-1}\mathscr{Z}^\cap). \qquad (2)$$

One of the column vectors λ_j has to be in direction of the normal vector of the dividing hyperplane, such that the splitting of the j^{th} generator $w^{(j)} = \gamma\, \lambda_j$ ($\gamma \in \mathbb{R}$) results into two parallelotopes Ψ_1, Ψ_2:

$$\Psi_1 = (c - \frac{1}{2}w^{(j)}, w^{(1)}, \dots, w^{(j-1)}, \frac{1}{2}w^{(j)}, w^{(j+1)}, \dots, w^{(n)})$$
$$\Psi_2 = (c + \frac{1}{2}w^{(j)}, w^{(1)}, \dots, w^{(j-1)}, \frac{1}{2}w^{(j)}, w^{(j+1)}, \dots, w^{(n)})$$

The split parallelotopes are used for the splitting of zonotope bundles:

$$\mathscr{Z}_1^\cap = \mathscr{Z}^\cap \cap \Psi_1, \quad \mathscr{Z}_2^\cap = \mathscr{Z}^\cap \cap \Psi_2.$$

Note that the number of zonotopes in the bundle is increased by one. In contrast to zonotopes, the result of the split is exact, i.e. $\mathscr{Z}_1^\cap \cap \mathscr{Z}_2^\cap = \emptyset$ and $\mathscr{Z}_1^\cap \cup \mathscr{Z}_2^\cap = \mathscr{Z}^\cap$, no matter how the other column vectors of Λ besides the j^{th} one are chosen and how tightly $\mathtt{box}()$ in (2) is obtained. However, subsequent operations such as Minkowski addition with zonotopes adds a larger error when Λ is chosen badly or $\mathtt{box}()$ returns a large overapproximation. Heuristics for computations of Λ are addressed next:

a) Box method (box): This method encloses the set by a box, such that $\Lambda = I$ and I is the identity matrix. Good results are obtained when the reachable set is not stretched out in certain directions, while being compact in others. This heuristics is especially useful when using several enclosing zonotopes since there are no numerical problems when computing Λ^{-1} such that a worst case enclosure is guaranteed.

b) PCA method (pca): The method returns directions with the greatest variance of the vertices of $\mathscr{Z}^\cap$ by a principal component analysis (see [13]) and thus requires vertex computation in contrast to the other heuristics.

c) Generator filtering (fil): This method is based on discarding generators of the zonotope bundle, such that only n generators remain for Λ. First, short generators are sorted out, next generator combinations are sorted out that do not span a large parallelogram, i.e. that are too much aligned. From this subset, the final n generators are picked providing the smallest volume of Ψ. This technique is equivalent to *method C* in [28], except that the union of generators of all zonotopes of the bundle is used.

d) Flow method (flow): This method is similar to the box method, but incorporates the flow of the dynamics. Starting from $\Lambda = I$, the column that is most aligned with the flow direction f, is replaced by f.

The presented methods are evaluated later in Sec. VI-B. In order to decrease overapproximative effects of operations (see Prop. 2-6) after the split, one can compute the enclosure of a zonotope bundle as $\mathtt{box}(\mathscr{Z}^\cap) = \mathtt{box}\Big(\bigcap_{i=1}^s \mathcal{Z}_i\Big)$ in contrast to Prop. 6. This requires the conversion of the zonotopes $\mathcal{Z}_i$ into a H-representation according to [28] which is linear in the number of generated halfspaces, but there might be up to $2\binom{p}{n-1}$ halfspaces. Thus, it is often required to reduce the number of generators of each $\mathcal{Z}_i$ in an overapproximative way using techniques from [11], [12], or the previously described parallelotope enclosures, which can be combined such that e.g. $\mathcal{Z}_i \subseteq \Psi^{box} \cap \Psi^{pca}$. Finally, the enclosing box of the H-representation can be obtained via linear programming.

B. Numerical Example

The effectiveness of the zonotope bundle computations is shown by an automated evasion maneuver of a car. The car is approaching a multi-lane crossing equipped with CICAS (Cooperative Intersection Collision Avoidance System) [29], where a standing vehicle is in its lane; see Fig. 5. The initial velocity is high enough that an evasive maneuver is more effective compared to a braking maneuver [30].

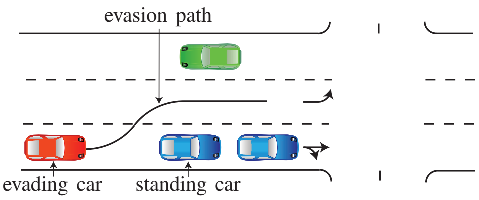


Fig. 5. Evasion maneuver.

The vehicle model consists of the lateral dynamics, the longitudinal dynamics, and a geometric model translating the heading and the velocity of the vehicle to positions on the road. The lateral dynamics is modeled by a so-called bicycle model, which neglects the roll and pitch dynamics; see [31]. The automated steering is performed by first generating a reference trajectory $y_d(t)$ of the lateral positions (perpendicular to the road direction) and the orientation $\Psi_d(t)$ considering the maximal available tire friction. The steering controller uses the measured lateral position $y(t)$ and orientation $\Psi(t)$ to update the steering input $\delta = k_1(\Psi_d(t) - \Psi(t)) + k_2(y_d(t) - y(t))$. The position and orientation is obtained by fusing an accurate differential GPS signal from the CICAS-equipped intersection with inertial accelerations from the vehicle. The state vector is $x = [\beta, \Psi, \dot{\Psi}, v, x, y]$, where all variables are measured at the center of mass: β is the side-slip angle, Ψ is the yaw angle, $\dot{\Psi}$ is the yaw rate, v is the velocity, x and y are the x- and y-position:

$$\begin{aligned}
\dot{x}_1 =& -\frac{1}{m x_4}\Big(\mu(c_r + c_f)x_1 + \mu(c_f l_f - c_r l_r)\frac{x_3}{x_4} + m x_3 x_4 \\
&\quad - c_f \mu \underbrace{\left(k_1(\Psi_d - x_2) + k_2(y_d - x_6)\right)}_{=\delta}\Big) \\
\dot{x}_2 =& x_3 \\
\dot{x}_3 =& \frac{1}{J}\Big(\mu(c_r l_r - c_f l_f)x_1 - \mu(c_r l_r^2 + c_f l_f^2)\frac{x_3}{x_4} \\
&\quad + \mu c_f l_f \underbrace{\left(k_1(\Psi_d - x_2) + k_2(y_d - x_6)\right)}_{=\delta}\Big) \\
\dot{x}_4 =& a_x \\
\dot{x}_5 =& \cos(x_1 + x_2)x_4 \\
\dot{x}_6 =& \sin(x_1 + x_2)x_4
\end{aligned}$$

The parameters of this model are listed in Table II, the set of initial states is $\mathcal{R}(0) = [-0.02, 0.02] \times [-0.05, 0.05] \times [-0.2, 0.2] \times [19.8, 20.2] \times [-0.2, 0.2] \times [-0.2, 0.2]$, and the set of disturbances is $a_x \in [-0.1, 0.1]$. The path of the maneuver consists of two arcs followed by a braking maneuver (see Fig. 5).

TABLE II

VEHICLE PARAMETERS PARTIALLY TAKEN OUT OF [32].

symb.	m	J	l_f	l_r	c_f,c_r	μ	k_1	k_2
value	1573	2873	1.1	1.58	$80e3$	0.7	2	0.5
unit	kg	kg m^2	m	m	N/rad	—	—	—

The different projections of the reachable set are shown in Fig. 6 when using zonotopes or zonotope bundles. The splitting of the reachable sets is performed as presented in Sec. V-A by choosing $\Lambda = I$ which is effective since the reachable set is not stretched out in certain directions. The box enclosure is computed using Prop. 6 instead of the tighter version discussed in Sec. V-A. The nonlinear effects are dominated by the change of the velocity x_4, which is more dominant for small velocities since $\dot{x}_1$, $\dot{x}_3$ are obtained by dividing by the velocity or its square. For $x_4 \to 0$ the model becomes singular, such that the verification is stopped for $x_4 < 2$ [m/s].

This nonlinear effect towards smaller velocities causes splitting, which results into a parallel computation of reachable sets shown in Fig. 7. When using zonotopes, the number of required splits explodes such that the verification fails with zonotopes, while it remains stable using zonotope bundles. It also fails for polytopes since certain operations such as the Minkowski addition are infeasible in 6 dimensions.

The system is considered as safe when the vehicle center (x_5, x_6) does not hit the forbidden region indicated in Fig. 6(c) which would cause a collision with the standing vehicle. In addition, the vehicle is not allowed to cross the lane boundary of the neighboring lane. Due to the reduced splitting, the computation time up to $t = 3.6$ s (when the reachable set starts exploding for the zonotope computation) is 32.8 s using zonotope bundles, which is less than the 115.9 s required for zonotopes. The example was implemented in MATLAB and executed on an Intel i7 Processor with 1.6 GHz and 6 GB memory.

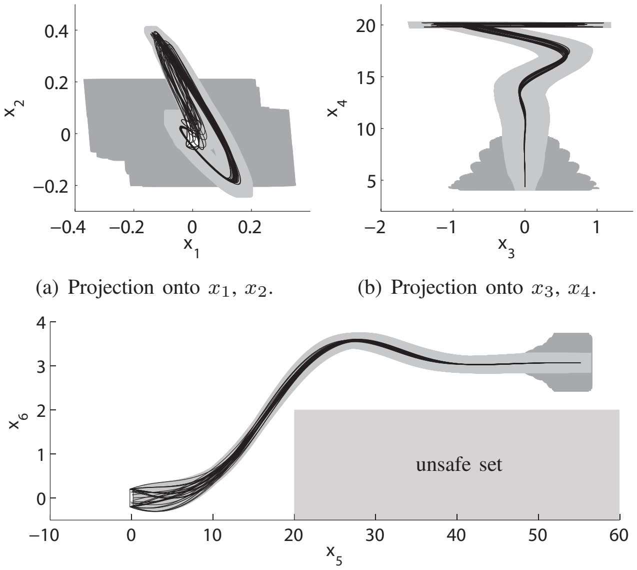
VI. REACHABILITY ANALYSIS OF HYBRID SYSTEMS

The reachability analysis of hybrid systems requires intersection with so-called guard sets. We will show that this intersection can be computed much more accurately with zonotope bundles compared to zonotopes, while the computation with polytopes typically becomes infeasible for dimensions greater than 4.

Hybrid systems are a combination of continuous and discrete dynamics, where to each discrete state an invariant region is assigned in which a continuous dynamics is valid. Once the continuous state is within a guard set, the discrete state may change according to a transition relation between discrete states, and has to change if it would leave the invariant. An additional jump of continuous state variables may be specified for the transition. The reachable set of a hybrid system is exemplarily illustrated in Fig. 8; a formal definition of the considered hybrid dynamics is given in [28].

A. Basic Procedure

We restrict ourselves to guard sets specified by polytopes, which can also be used to overapproximate arbitrarily shaped guard sets. When representing reachable sets by zonotope



(a) Projection onto x_1, x_2.

(b) Projection onto x_3, x_4.

(c) Projection onto x_5, x_6.

Fig. 6. Reachable set of the lane changing maneuver. Reachable sets are light gray when computing with zonotope bundles and dark gray when computing with zonotopes. Black lines show exemplary trajectories.

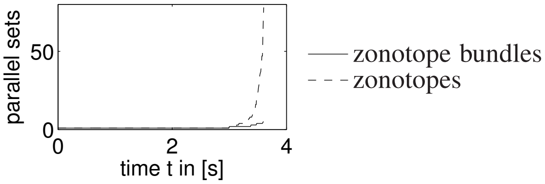


Fig. 7. Number of sets computed in parallel due to the splittings.

bundles, one has to convert them to a H-representation (which can be efficiently done by several parallelotope enclosures as presented in Sec. V-A) for intersection with guard sets using a standard toolbox[3]. In order to continue the computation with a G-representation, the vertices of the intersection are enclosed by parallelotopes as described in (2), except that the box of vertices instead of zonotopes is computed. These steps are illustrated in Fig. 9.

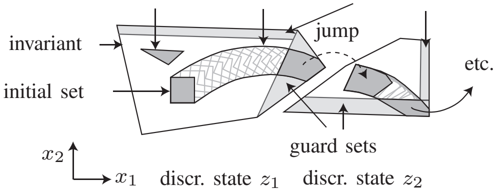
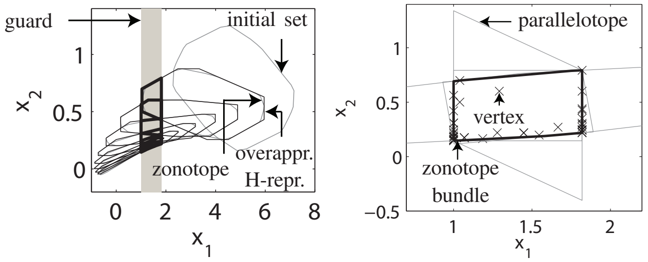


Fig. 8. Illustration of the reachable set of a hybrid automaton.



(a) Guard intersection.

(b) Enclosure of vertices.

Fig. 9. Reachable set intersection with a guard set by a zonotope bundle.

B. Numerical Examples

In analogy to linear systems, we randomly generate hybrid reachability problems consisting of a single guard set. Without loss of generality we fix the guard set while randomizing the initial set and the continuous dynamics. The reachable set is computed until no further set intersects the guard set.

The continuous dynamics is chosen as in Sec. IV-B and the guard set is a box $[1, 1+\epsilon] \times [-l, l] \times \ldots \times [-l, l]$, where ϵ is uniformly distributed within $[0, 1]$ and $l = 10^3$, see Fig. 9(a). The initial set is a zonotope with an order o uniformly distributed within $[1, 3]$, where the generators are randomly generated as the vertices in Sec. IV-B with $\mu = 1/o$, and the center is uniformly distributed within $[4, 8] \times [-4, 4] \times \ldots \times [-4, 4]$, forcing an intersection with the guard set before converging to the origin. The hypercube enclosing the uncertain input has maximum edge length $\delta = 0.01$.

The enclosure of vertices by a zonotope bundle has been performed using the methods *box*, *pca*, *fil*, and *flow* as described in Sec. V-A. The method combining these enclosures using a zonotope bundle is denoted by *comb*, and the enclosure by a polytope is denoted by *poly*. In addition, a randomized enclosure method *rand* is evaluated, which creates 100 randomly generated parallelotopes by randomizing Λ (see (2)). Each column of Λ is a vector uniformly distributed on a unit hypersphere, and the enclosure with the smallest volume is picked. The relative performance indices $\Theta^{\text{method}}/\Theta^{\text{poly}}$ (method $= \{\text{box}, \ldots, \text{poly}\}$) along with the computation times for the enclosure are presented in Table III. The computations are performed in MATLAB on an Intel i7 Processor with 1.6 GHz and 6 GB memory. It can be seen that the relative performance for zonotope bundles (method *comb*) is only marginally increasing with n, while the computation time is almost constant, whereas the computation for *poly* increases dramatically. Thus, for $n > 4$ only the computation time can be evaluated since Θ^{poly} is infeasible.

TABLE III

ENCLOSURE OF INTERSECTED REACHABLE SETS.

n	box	pca	fil	flow	rand	comb	poly
relative performance index $\Theta^{\text{method}}/\Theta^{\text{poly}}$							
2	1.362	1.155	2.068	2.186	1.112	1.018	1
3	1.803	1.339	4.686	3.544	1.945	1.064	1
4	2.246	1.486	8.981	5.628	3.487	1.135	1
computation times in [ms] for vertex enclosure							
2	0.385	0.524	0.731	1.214	92.27	2.854	7.778
3	0.359	0.520	0.698	1.240	90.85	2.817	30.42
4	0.400	1.610	0.813	1.255	96.81	4.078	2154
8	11.38	31.37	15.64	16.17	1580	74.57	–

[3] used tool: MPT toolbox [25]

VII. Conclusions

This paper introduces zonotope bundles, defined as the intersection of a set of zonotopes, which in contrast to zonotopes are closed under intersection. Operations on zonotope bundles for reachability analysis are computed storing a list of zonotopes and without performing any intersection. Safety verification can then be performed by checking for each zonotope in the bundle if a set of unsafe states is hit. Since all computations are performed on individual zonotopes, the computational effort is simply the effort of zonotope computation times the number of zonotopes in the bundle.

Since all computations are separately performed on zonotopes, it is straightforward to parallelize the presented operations. Furthermore, it has been shown that zonotope bundle computations scale better with the system dimension than polytopes, while improving the accuracy compared to single zonotope representations for linear, nonlinear, and hybrid systems. In the nonlinear example, only zonotope bundles could successfully verify the safety. Another property is that one can tune the accuracy by the number of zonotopes in the bundle. Measuring the improvement compared to zonotopes in higher dimensions is infeasible for many measures such as the volume of sets or their Hausdorff distance, such that introducing a fair and scalable measure is future work.

Acknowledgments

This work was supported in part by US NSF grants CNS 1035800, CCF-0926181 and AFOSR Grant FA9550-06-1-0312.

References

[1] J. D. Glover and F. C. Schweppe, “Control of linear dynamic systems with set constrained disturbances,” *IEEE Transactions on Automatic Control*, vol. 16, no. 5, pp. 411–423, 1971.

[2] E. Clarke, A. Fehnker, Z. Han, B. Krogh, O. Stursberg, and M. Theobald, *Tools and Algorithms for the Construction and Analysis of Systems*, ser. LNCS 2619. Springer, 2003, ch. Verification of Hybrid Systems based on Counterexample-Guided Abstraction Refinement, pp. 192–207.

[3] S. V. Raković, P. Grieder, M. Kvasnica, D. Q. Mayne, and M. Morari, “Computation of invariant sets for piecewise affine discrete time systems subject to bounded disturbances,” in *Proc. of the 43rd IEEE Conference on Decision and Control*, 2004, pp. 1418–1423.

[4] F. M. Schlaepfer and F. C. Schweppe, “Continuous-time state estimation under disturbances bounded by convex sets,” *IEEE Transactions on Automatic Control*, vol. 17, no. 2, pp. 197–205, 1972.

[5] T. Alamo, J. M. Bravo, and E. F. Camacho, “Guaranteed state estimation by zonotopes,” in *Proc. of the 42nd IEEE Conference on Decision and Control*, 2003, pp. 5831–5836.

[6] T. A. Henzinger, P.-H. Ho, and H. Wong-Toi, “Algorithmic analysis of nonlinear hybrid systems,” *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 540–554, 1998.

[7] G. Frehse, “PHAVer: Algorithmic verification of hybrid systems past HyTech.” in *Hybrid Systems: Computation and Control*, ser. LNCS 3413. Springer, 2005, pp. 258–273.

[8] G. Lafferriere, G. J. Pappas, and S. Yovine, “Symbolic reachability computation for families of linear vector fields,” *Symbolic Computation*, vol. 32, pp. 231–253, 2001.

[9] A. Girard, C. Le Guernic, and O. Maler, “Efficient computation of reachable sets of linear time-invariant systems with inputs,” in *Hybrid Systems: Computation and Control*, ser. LNCS 3927. Springer, 2006, pp. 257–271.

[10] A. Chutinan and B. H. Krogh, “Computational techniques for hybrid system verification,” *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 64–75, 2003.

[11] W. Kühn, “Rigorously computed orbits of dynamical systems without the wrapping effect,” *Computing*, vol. 61, pp. 47–67, 1998.

[12] A. Girard, “Reachability of uncertain linear systems using zonotopes,” in *Hybrid Systems: Computation and Control*, ser. LNCS 3414. Springer, 2005, pp. 291–305.

[13] O. Stursberg and B. H. Krogh, “Efficient representation and computation of reachable sets for hybrid systems,” in *Hybrid Systems: Computation and Control*, ser. LNCS 2623. Springer, 2003, pp. 482–497.

[14] T. Dang, O. Maler, and R. Testylier, “Accurate hybridization of nonlinear systems,” in *Hybrid Systems: Computation and Control*, 2010, pp. 11–19.

[15] N. Ramdani, N. Meslem, and Y. Candau, “Reachability analysis of uncertain nonlinear systems using guaranteed set integration,” in *Proc. of the 17th IFAC World Congress*, 2008, pp. 8972–8977.

[16] N. S. Nedialkov and K. R. Jackson, *Perspectives on Enclosure Methods*. Springer-Verlag, 2001, ch. A New Perspective on the Wrapping Effect in Interval Methods for Initial Value Problems for Ordinary Differential Equations, pp. 219–264.

[17] A. B. Kurzhanski and P. Varaiya, “Ellipsoidal techniques for reachability analysis,” in *Hybrid Systems: Computation and Control*, ser. LNCS 1790. Springer, 2000, pp. 202–214.

[18] A. Girard and C. Le Guernic, “Efficient reachability analysis for linear systems using support functions,” in *Proc. of the 17th IFAC World Congress*, 2008, pp. 8966–8971.

[19] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, “A time-dependent Hamilton–Jacobi formulation of reachable sets for continuous dynamic games,” *IEEE Transactions on Automatic Control*, vol. 50, pp. 947–957, 2005.

[20] A. A. Kurzhanskiy and P. Varaiya, “Ellipsoidal techniques for reachability analysis of discrete-time linear systems,” *IEEE Transactions on Automatic Control*, vol. 52, no. 1, pp. 26–38, 2007.

[21] V. Kaibel and M. E. Pfetsch, *Algebra, Geometry and Software Systems*. Springer, 2003, ch. Some Algorithmic Problems in Polytope Theory, pp. 23–47.

[22] K. Fukuda, “From the zonotope construction to the Minkowski addition of convex polytopes,” *Journal of Symbolic Computation*, vol. 38, no. 4, pp. 1261–1272, October 2004.

[23] R. Bagnara, P. M. Hill, and E. Zaffanella, “The Parma Polyhedra Library: Toward a complete set of numerical abstractions for the analysis and verification of hardware and software systems,” *Science of Computer Programming*, vol. 72, pp. 3–21, 2008.

[24] M. Althoff, “Reachability analysis and its application to the safety assessment of autonomous cars,” Dissertation, TU München, 2010, URL: http://nbn-resolving.de/urn/resolver.pl?urn:nbn:de:bvb:91-diss-20100715-963752-1-4.

[25] M. Kvasnica, P. Grieder, and M. Baotić, “Multi-Parametric Toolbox (MPT),” 2004. [Online]. Available: http://control.ee.ethz.ch/~mpt/

[26] M. Althoff, C. Le Guernic, and B. H. Krogh, “Reachable set computation for uncertain time-varying linear systems,” in *Hybrid Systems: Computation and Control*, 2011.

[27] M. Althoff, O. Stursberg, and M. Buss, “Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization,” in *Proc. of the 47th IEEE Conference on Decision and Control*, 2008, pp. 4042–4048.

[28] ——, “Computing reachable sets of hybrid systems using a combination of zonotopes and polytopes,” *Nonlinear Analysis: Hybrid Systems*, vol. 4, no. 2, pp. 233–249, 2010.

[29] M. Maile and L. Delgrossi, “Cooperative intersection collision avoidance system for violations (CICAS-V) for avoidance of violation-based intersection crashes,” National Highway Traffic Safety Administration, Tech. Rep. 09-0118, 2009.

[30] C. Schmidt, F. Oechsle, and W. Branz, “Research on trajectory planning in emergency situations with multiple objects,” in *Proc. of the IEEE Intelligent Transportation Systems Conference*, 2006, pp. 988–992.

[31] R. Rajamani, *Vehicle Dynamics and Control*, F. F. Ling and E. F. Gloyna, Eds. Springer, 2006.

[32] J. Guldner, W. Sienel, H.-S. Tan, J. Ackermann, S. Patwardhan, and T. Bünte, “Robust automatic steering control for look-down reference systems with front and rear sensors,” *IEEE Transactions on Control Systems Technology*, vol. 7, pp. 2–11, 1999.