



# Computing reachable sets of hybrid systems using a combination of zonotopes and polytopes

Matthias Althoff<sup>a,\*</sup>, Olaf Stursberg<sup>b</sup>, Martin Buss<sup>a</sup>

<sup>a</sup> Institute of Automatic Control Engineering (LSR), Technische Universität München, 80290 München, Germany

<sup>b</sup> Institute of Control and System Theory, Department of Electrical Eng., University of Kassel, Germany

## ARTICLE INFO

### Article history:

Received 25 October 2008

Accepted 26 March 2009

### Keywords:

Reachability analysis

Hybrid systems

Safety verification

Zonotopes

Polytopes

## ABSTRACT

The computation of reachable sets for hybrid systems with linear continuous dynamics is addressed. Zonotopes are used for the representation of reachable sets, resulting in an algorithm with low computational complexity with respect to the dimension of the considered system. However, zonotopes have drawbacks when being intersected with transition guards which determine the discrete behavior of the hybrid system. For this reason, in the proposed approach, reachable sets are represented by polytopes within guard sets as an intermediate step in order to enclose them by zonotopes afterwards. Different methods for the conservative conversion from zonotopes to polytopes and vice versa are proposed and numerically evaluated.

© 2009 Elsevier Ltd. All rights reserved.

## 1. Introduction

Due to continuous technological progress, the complexity of technical systems is steadily increasing. In order to ensure safe operation of more and more complex systems, new verification methods have to be developed for large systems with respect to the number of continuous state variables and discrete states. For purely discrete systems, powerful verification methods already exist, allowing to prove the safe operation of fairly complex systems, see e.g. [1]. For hybrid systems with combined discrete and continuous dynamics, safety assessment is an open problem for large instances of many system classes. A straightforward way to check the safe operation of a hybrid system, is to apply Monte Carlo simulation as described, for instance, in [2]. While Monte Carlo simulation focuses on computing the probability that the system fails, there are methods which generate test cases in such a way, that the test coverage is maximized in order to find unsafe executions [3]. The advantage of simulations is that they can be applied for a wide class of system dynamics and the disadvantage is that they cannot prove the safety of a system. This is because infinitely many possible executions of a system exist when the continuous initial state or continuous input/disturbance can take values within specified sets with infinitely many elements.

A possibility to use simulation techniques for the safety verification of hybrid systems, is to guarantee that system trajectories stay within certain regions around exemplary trajectories, allowing to cover all possible behaviors by a finite number of simulations [4,5]. Another method to prove safe operation, is to compute barrier certificates [6]. If the barrier can separate the sets of unsafe states from the sets of reachable states, the system is safe. Instead of computing such barriers, one can directly compute the set of states, that the system can reach in (in-)finite time. If the reachable set does not intersect unsafe regions, the system is proven to be safe. In the past, numerous approaches have been developed which use different representations for the reachable sets: level sets [7], ellipsoids [8], polytopes [9], oriented rectangular hulls [10] and zonotopes [11]. A problem that has to be faced in reachability analysis is the curse of dimension. The different types of

\* Corresponding author. Tel.: +49 89 289 23434.

E-mail addresses: [althoff@tum.de](mailto:althoff@tum.de) (M. Althoff), [stursberg@uni-kassel.de](mailto:stursberg@uni-kassel.de) (O. Stursberg), [mb@tum.de](mailto:mb@tum.de) (M. Buss).

representation suffer more or less from the problem that the propagation of the reachable sets becomes intractable in high dimensions. For hybrid systems with linear continuous dynamics, zonotopes have low time and memory complexity [12]. The disadvantage of zonotopes is, that they are not closed under intersection, i.e. the intersection of two zonotopes is not a zonotope in general. In hybrid systems, intersections with sets that enable discrete transitions (guard sets) are inevitable, such that the intersection of zonotopes with guard sets has to be over-approximated. In [12], this problem has been addressed for guard sets that are modeled as hyperplanes. The previous work of the authors [13] proposes an algorithm for guard sets that are modeled as polytopes, which include the special case of hyperplanes. The main differences between the works in [13] and [12] is, that [13] can handle more general guard sets, whereas [12] focuses on hyperplanes as guard sets what leads to probably more accurate results due to the specialization. Besides reachability analysis, zonotopes are also used e.g. for guaranteed state estimation [14] and collision detection [15].

The efficient implementation of zonotopes is due to their special representation by the so-called generators, which one cannot use to represent polytopes. A common representation for zonotopes and polytopes is the so-called halfspace representation, which allows for intersection with guard sets. For this reason, one major aspect of this work is to provide methods for the transformation of zonotopes from the generator representation to the halfspace representation. The second major aspect is the over-approximation of polytopes, resulting from the intersection with guard sets, by zonotopes which allow us to continue the computation with zonotopes in newly reached locations.

## 2. Problem statement

The considered problem is to compute the reachable set of continuous states of a hybrid system for a time span  $[0, t_f]$ . The hybrid system is modeled as a hybrid automaton which is similarly defined as in [10]:

**Definition 1** (*Hybrid Automaton*). A hybrid automaton  $HA = (Z, z^0, X, X^0, U, inv, T, g, h, f)$  consists of:

- the finite set of locations  $Z = \{z_1, \dots, z_\xi\}$  with an initial location  $z^0 \in Z$ .
- the continuous state space  $X \subseteq \mathbb{R}^n$  and the set of initial continuous states  $X^0$  such that  $X^0 \subseteq inv(z^0)$ .
- the continuous input space  $U \subseteq \mathbb{R}^n$ .
- the mapping<sup>1</sup>  $inv: Z \rightarrow 2^X$  which assigns an invariant  $inv(z) \subseteq X$  to each location  $z$ .
- the set of discrete transitions  $T \subseteq Z \times Z$ . A transition from  $z_i \in Z$  to  $z_j \in Z$  is denoted by  $(z_i, z_j)$ .
- the guard function  $g: T \rightarrow 2^X$  that associates a guard set  $g((z_i, z_j))$  for each transition from  $z_i$  to  $z_j$ , where  $g((z_i, z_j)) \cap inv(z_i) \neq \emptyset$ .
- the jump function  $h: T \times X \rightarrow X$  which returns the next continuous state when a transition is taken.
- the flow function  $f: Z \times X \times U \rightarrow \mathbb{R}^n$  that defines a continuous vector field for the time derivative of  $x$ :  $\dot{x} = f(z, x, u)$ .

The invariants  $inv(z)$  and the guard sets  $g((z_i, z_j))$  are modeled by polytopes. The jump function as well as the flow function are linear:

$$\begin{aligned} \dot{x} &= A_z x + u, \quad u \in U_z \subseteq U, \\ x' &= K_z x + l_z, \end{aligned} \tag{1}$$

where  $x'$  denotes the state after the transition is taken and  $A_z, K_z \in \mathbb{R}^{n \times n}, l_z \in \mathbb{R}^n$  are specific for location  $z \in Z$ . The sets  $U_z$  are modeled by zonotopes and are also dependent on the location  $z$ .

The evolution of the hybrid automaton is described informally: Starting from an initial location  $z(0) = z^0$  and an initial state  $x(0) \in X^0$ , the continuous state evolves according to the flow function that is assigned to each location  $z$ . If the continuous state is within a guard set, the corresponding transition can be taken and has to be taken if the state would leave the invariant  $inv(z)$ . When the transition from the previous location  $z_i$  to the next location  $z_j$  is taken, the system state is updated according to the jump function and the continuous evolution within the next invariant.

The definition of a hybrid automaton  $HA$  now allows us to define the set of continuous reachable states of  $HA$ :

**Definition 2** (*Exact Reachable Set*). The continuous reachable set of  $HA$  for a given initial location  $z^0$  and a set of initial states  $X^0$  at time  $t = r$  is:

$$R^e(r) = \{x(r) \mid (z(t), x(t)) \text{ is a solution of } HA \forall t \in [0, r], z(0) = z^0, x(0) \in X^0\}$$

where  $z(t)$  and  $x(t)$  denote the discrete and continuous state at time  $t$ .

In this work, over-approximations of reachable sets are computed which are denoted by  $R(r) \supseteq R^e(r)$ . The over-approximated reachable set for a time interval is defined as  $R([0, r]) = \bigcup_{t \in [0, r]} R(t)$ . Note that over-approximations of reachable sets still allow us to prove the safety of a system, because the original system does not intersect an unsafe set if the over-approximation does not. However, due to the over-approximation, one cannot decide if the system is unsafe if an unsafe set is intersected. An illustration of a reachable set for a hybrid automaton is given in Fig. 1.

<sup>1</sup>  $2^X$  is the powerset of  $X$ .

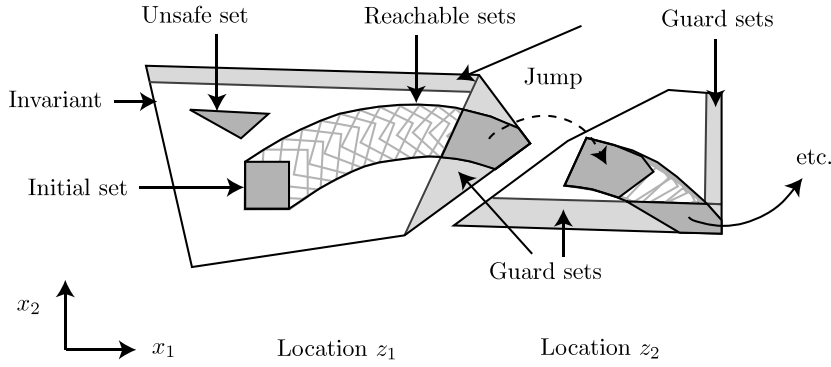


Fig. 1. Illustration of the reachable set of a hybrid automaton.

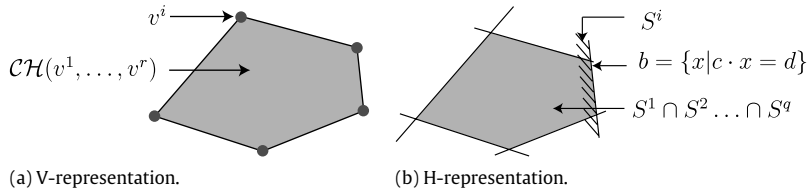


Fig. 2. Possible representations of a polytope.

### 3. Preliminaries

Before the computation of reachable sets is addressed, the definitions of their representation by convex polytopes and zonotopes are recalled. For polytopes, there exist two representations: the halfspace representation (H-representation) and the vertex representation (V-representation). The halfspace representation of a hyperplane is  $S := \{x | c \cdot x \leq d, c \in \mathbb{R}^{1 \times n}, d \in \mathbb{R}\}$ . For a given  $S$ , let  $b := \{x | c \cdot x = d\}$  denote the corresponding bounding hyperplane which divides the  $n$ -dimensional space into two halfspaces. The vector  $c$  is the normal vector of the hyperplane and  $d$  the scalar product of any point on the hyperplane with the normal vector. A polytope  $P$  is the nonempty intersection of a finite set  $S := \{S^1, \dots, S^q\}$  of halfspaces:

**Definition 3** (*H-Representation of a Polytope*). For  $q$  halfspaces, a convex polytope  $P$  is the set:

$$P = \{x \in \mathbb{R}^n | C \cdot x \leq d, C \in \mathbb{R}^{q \times n}, d \in \mathbb{R}^{q \times 1}\}.$$

A polytope with vertex representation is defined as the convex hull of a finite set of vertices  $v^i \in \mathbb{R}^n$ :

**Definition 4** (*V-Representation of a Polytope*). For  $r$  vertices  $v^i \in \mathbb{R}^n$ , a convex polytope  $P$  is the set  $P = \mathcal{CH}(v^1, \dots, v^r)$ , with

$$\mathcal{CH}(v^1, \dots, v^r) = \left\{ \sum_{i=1}^r \alpha^i v^i \mid v^i \in \mathbb{R}^n, \alpha^i \in \mathbb{R}, \alpha^i \geq 0, \sum_{i=1}^r \alpha^i = 1 \right\}.$$

The halfspace and the vertex representation are illustrated in Fig. 2.

Zonotopes are a special case of polytopes, i.e. they can also be represented by the halfspace or the vertex representation. A further possibility for the representation of zonotopes is the use of the so-called *generators*  $g^{(i)} \in \mathbb{R}^n$ . The representation via generators (*G-representation*) is defined as (see e.g. [16]):

**Definition 5** (*G-Representation of a Zonotope*). A zonotope is a set:

$$Z = \left\{ x \in \mathbb{R}^n \mid x = c + \sum_{i=1}^p \beta^i \cdot g^{(i)}, -1 \leq \beta^i \leq 1 \right\}$$

with  $c, g^{(1)}, \dots, g^{(p)} \in \mathbb{R}^n$ . The order of a zonotope is defined as  $o = \frac{p}{n}$  and the short notation is  $Z = (c, g^{(1)}, \dots, g^{(p)})$ .

The vector  $c$  is the center of the zonotope, to which the latter is always centrally symmetric. An alternative definition of a zonotope is the Minkowski sum<sup>2</sup> of a finite set of line segments  $l_i = [-1, 1]g^{(i)}$ , which allows us to illustrate how a

<sup>2</sup> Minkowski sum of two sets  $A, B$ :  $A + B = \{a + b | a \in A, b \in B\}$

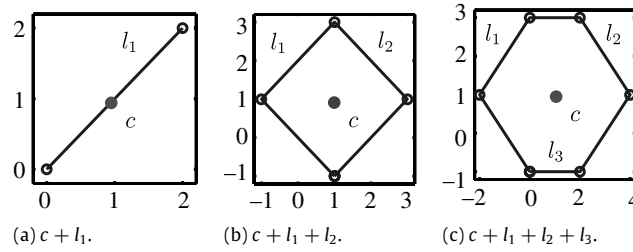


Fig. 3. Construction of a zonotope.

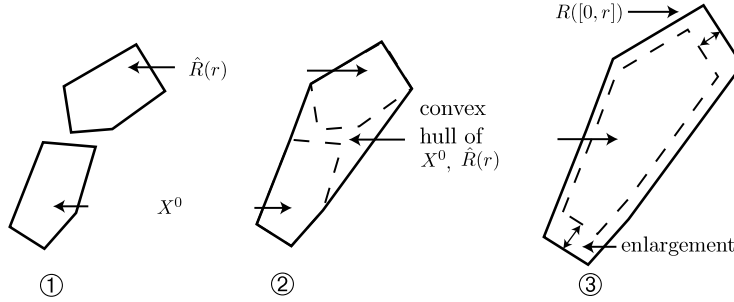


Fig. 4. Computation of the reachable set for a time interval.

zonotope is built step-by-step by adding further generators. This is shown in Fig. 3, where from the left to the right further two-dimensional generators are added.

#### 4. Computing over-approximated reachable sets

First, the over-approximation of reachable sets is discussed for purely linear continuous dynamics, and in the second part of this section, the intersection with guard sets is addressed. Reachable sets of linear systems with uncertain parameters and inputs have been computed in an earlier work of the authors [17]. The basic steps that are undertaken in order to compute the reachable sets are recalled in the following. These steps are in common with other approaches, such as e.g. [11,9,10]. The reachable set of a time interval  $t \in [0, r]$ , ( $r \in \mathbb{R}^+$ ), is obtained by

- (1) computation of the reachable set  $\hat{R}$  without input ( $u = 0$ ) for  $t = r$  based on the initial set  $X^0$ ,
- (2) generation of the convex hull of the time point solutions,
- (3) enlargement of the convex hull to ensure enclosure of all trajectories for the time interval  $t \in [0, r]$  under all possible inputs.

These basic steps are illustrated in Fig. 4.

The reachable set for a time interval  $[t_k, t_{k+1}]$  starting from the one for  $[t_{k-1}, t_k]$  is obtained as  $R([t_k, t_{k+1}]) = e^{Ar}R([t_{k-1}, t_k]) + \hat{R}([0, r])$ , where  $t_k = k \cdot r$  ( $k \in \mathbb{N}^+$ ),  $e^{Ar}$  is the matrix exponential of the system matrix  $A$  and  $\hat{R}([0, r])$  is the reachable set due to the input, see [17]. The reachable set for  $[0, t_f]$ , where  $t_f = k_f \cdot r$  is the final time, is obtained by the union of the reachable sets for the time intervals indexed by  $k$ :  $R([0, t_f]) = \bigcup_{k=1}^{k_f} R([t_{k-1}, t_k])$ .

The procedure for linear continuous systems is now extended to hybrid systems with linear continuous dynamics, which has to account for the intersection with invariant sets and guard sets. The reachable set within a location is computed as presented above, until it has completely left the invariant at  $t_i$ , such that  $R(t_{i-1}) \cap \text{Inv} \neq \emptyset$  and  $R(t_i) \cap \text{Inv} = \emptyset$  or until the final time is reached ( $t = t_f$ ). In this work, the reachable sets are represented by zonotopes and the guard sets  $g((z_i, z_j))$  by polytopes. Thus, their intersection cannot be computed exactly in general. In order to continue the computation with zonotopes, the intersection with guard sets and the subsequent mapping according to the jump function has to be over-approximated by a zonotope. The first and last time step when the guard set  $g((z_i, z_j))$  is intersected by the reachable set are denoted by  $t_e^j, t_l^j$  respectively. The next steps to be taken in order to obtain an over-approximation after a transition are:

- (1) The zonotopes which intersect with a guard set, i.e. those establishing  $R([t_e^j, t_l^j])$ , are converted from generator to halfspace representation.
- (2) This conversion enables the computation of the intersection of the reachable set with the guard set  $g((z_i, z_j))$  using standard algorithms.

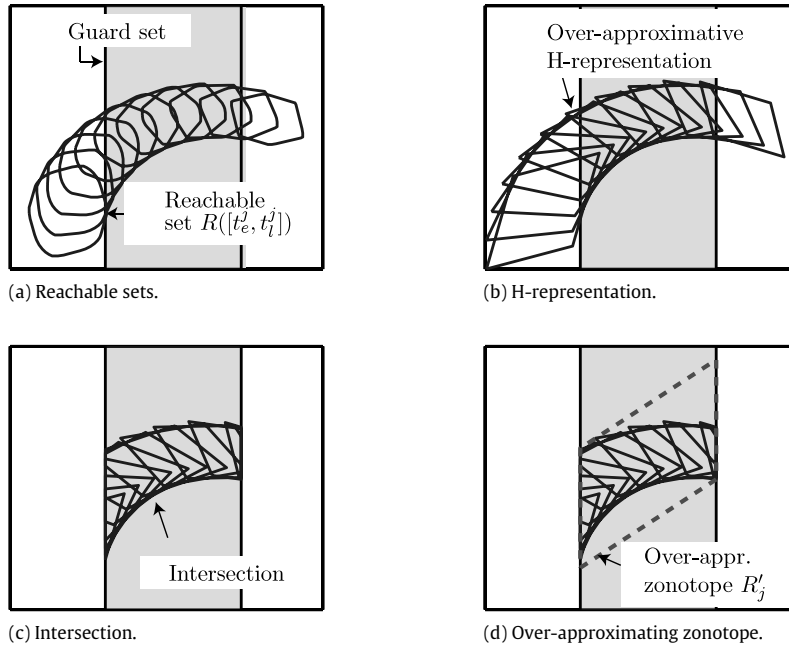


Fig. 5. Over-approximation of the intersection of zonotopes with a polytope by a single zonotope.

- (3) The intersected reachable sets are mapped according to the jump function  $h$ . As the jump function is restricted to a linear mapping as defined in (1), the halfspace representations are exactly mapped to new halfspace representations.
- (4) The union of the mapped reachable sets is enclosed by a zonotope  $R_j'$ .

These steps are illustrated in Fig. 5 for the case when the jump function is an identity mapping, i.e.  $h((z_i, z_j), x) = x$ . The time interval  $[t_e^j, t_l^j]$  when the reachable set intersects the guard  $g((z_i, z_j))$  is over-approximated by checking the intersection of the guard set with axis-oriented boxes enclosing the reachable sets in order to save computational time. In order to formulate the overall procedure for the computation of reachable sets within one location in Algorithm 1, the following operators are introduced: The operator over-approximating the time interval of guard intersection is denoted by `intersectionTimes()` and the operators for the conversion to H- and G-representation are denoted by `hConv()` and `gConv()` respectively. The superscript  $h$  of the reachable sets  $R^h$  in Algorithm 1 indicates that the reachable set has a halfspace representation.

---

**Algorithm 1** Compute  $R_j'$  within a location  $z_i$

---

**Input:** Reachable sets  $R([0, r])$ ,  $\bar{R}([0, r])$ , invariant  $inv$ , guard set  $g((z_i, z_j))$

**Output:**  $R^h[0, t_l], R_j'$

$k := 1, \quad R^h([0, r]) = \text{hConv}(R([0, r])) \cap inv$

**while**  $R^h([t_{k-1}, t_k]) \neq \emptyset$  or  $t < t_f$  **do**

$R([t_k, t_{k+1}]) = e^{Ar} R([t_{k-1}, t_k]) + \bar{R}([0, r])$

$R^h([t_k, t_{k+1}]) = \text{hConv}(R([t_k, t_{k+1}])) \cap inv$

$k := k + 1$

**end while**

$t_l = (k - 1) \cdot r$

$R^h([0, t_l]) = \bigcup_{k=1}^{t_l/r} R^h([t_{k-1}, t_k])$

$[t_e^j, t_l^j] = \text{intersectionTimes}(R^h([0, t_l]), g((z_i, z_j)))$

$R_j' = \text{gConv}(R^h([t_e^j, t_l^j]) \cap g((z_i, z_j)))$

---

It should be mentioned that the proposed procedure also allows us to verify hybrid systems with nonlinear continuous dynamics, when computing the continuous reachable set as presented in a previous work of the authors [18]. However, the main focus of this paper is to investigate the use of zonotopes for hybrid systems w.r.t. the discrete transitions, such that the continuous dynamics plays only a limited role. In the next section, the conversion from generator to halfspace representation of zonotopes is presented in detail.

## 5. Over-approximative halfspace representations of zonotopes

Since zonotopes are a special case of polytopes, there exists an exact halfspace representation of zonotopes. However, the conversion is computationally expensive, as there might exist up to  $2 \binom{p}{n-1}$  facets, where  $p$  is the number of generators and  $n$  is the dimension, see e.g. [19]. The formula for the number of facets becomes evident later in this section. Due to the large number of facets, zonotopes are over-approximated by a halfspace representation, which uses less facets. In a first step, the zonotopes are over-approximated by parallelotopes, which are zonotopes of order  $o = 1$ , i.e. the number of generators  $p$  equals the system dimension  $n$  (see Definition 5), resulting in  $2n$  facets. In a second step, the zonotopes are over-approximated by zonotopes of smaller order (but of order greater than one) – the tradeoff between accuracy and computational effort will be discussed. In order to assess the quality of the over-approximations, the zonotopes to be transformed are randomly generated and assessed according to an over-approximation measure.

### 5.1. Exact halfspace conversion of zonotopes

In this subsection, the exact conversion of zonotopes from G- to H-representation is presented. The halfspace representation  $C \cdot x \leq d$  contains the normal vectors of the  $i$ th halfspace as the  $i$ th row vector of  $C$  and the  $i$ th element of  $d$  is the scalar product of any point on the  $i$ th separating hyperplane with the corresponding normal vector. Thus, the absolute value of  $d$  is the distance from the origin to the halfspace. For the computation of the normal vectors, the  $n$ -dimensional cross product is introduced as in e.g. [20]. Given are  $n - 1$  vectors  $h^i \in \mathbb{R}^n$  which are stored in a matrix  $H = [h^1, \dots, h^{n-1}] \in \mathbb{R}^{n \times n-1}$ , and if the  $i$ th row is removed, the matrix is denoted by  $H^{[i]} \in \mathbb{R}^{n-1 \times n-1}$ . The cross product  $nX(H)$  of the vectors stored in  $H$  is defined as:

$$y = nX(H) = [\dots, (-1)^{i+1} \det(H^{[i]}), \dots]^T$$

The H-representation is computed for parallelotopes (zonotopes of order  $o = 1$ ) first, and then for general zonotopes.

#### 5.1.1. Parallelotopes

Parallelotopes have  $2n$  facets where  $n$  of them are non-parallel, and each of the facets is spanned by  $n - 1$  generators  $g^{(i)} \in \mathbb{R}^n$ . In order to obtain a concise notation, the matrix of generators is introduced as  $G = [g^{(1)}, \dots, g^{(n)}]$  and  $G^{(i)} \in \mathbb{R}^{n \times n-1}$  is defined as a matrix in which the  $i$ th generator is removed, such that  $G^{(i)}$ <sup>3</sup> contains the generators of the  $i$ th facet. The normal vectors and distances of the facets to the origin are computed as follows:

**Lemma 6** (Halfspace Representation of Parallelotopes). *The halfspace representation  $C \cdot x \leq d$  of a parallelotope with  $n$  independent generators is*

$$C = \begin{bmatrix} C^+ \\ -C^+ \end{bmatrix}, \quad d = \begin{bmatrix} d^+ \\ d^- \end{bmatrix}$$

with:

$$C_i^+ = nX(G^{(i)})^T / \|nX(G^{(i)})\|_2, \\ d_i^+ = C_i^+ \cdot c + \Delta d_i, \quad d_i^- = -C_i^+ \cdot c + \Delta d_i, \quad \Delta d_i = |C_i^+ \cdot g^{(i)}|.$$

$C_i^+$  denotes the  $i$ th row of  $C^+$  and  $d_i$  the  $i$ th element of  $d$ . The computational complexity of computing the H-representation for a given  $G$  is  $\mathcal{O}(n^5)$ .

**Proof.** The  $i$ th facet of a parallelotope can be reached by addition of a single generator to the center  $c$  (see Fig. 6). As there are only  $n$  generators, the facet must be spanned by the matrix of remaining generators  $G^{(i)}$ . Thus, the normal vectors are computed as  $C_i^+ = nX(G^{(i)})^T / \|nX(G^{(i)})\|_2$ . It is sufficient to compute  $n$  normal vectors denoted by a superscript ‘+’, as the remaining  $n$  normal vectors denoted by a superscript ‘−’, differ only in sign due to the central symmetry of zonotopes.

The elements  $d_i^+$  are the scalar products of any point on the  $i$ th halfspace with its normal vector. A possible point on the  $i$ th halfspace is  $c + g^{(i)}$ . Thus, the values of  $d_i^+$  can be computed as  $d_i^+ = C_i^+ \cdot c + \Delta d_i$  and  $d_i^- = -C_i^+ \cdot c + \Delta d_i$  with  $\Delta d_i = |C_i^+ \cdot g^{(i)}|$ .

The computational complexity is derived as follows: The computation of the determinants is  $\mathcal{O}((n - 1)^3) = \mathcal{O}(n^3)$  when applying LU decomposition, and thus the computation of the  $n$ -dimensional cross product is  $n \cdot \mathcal{O}(n^3) = \mathcal{O}(n^4)$ . As  $n$  non-parallel hyperplanes have to be computed, the overall complexity is  $\mathcal{O}(n^5)$ .  $\square$

#### 5.1.2. Zonotopes

The extension to the conversion of general zonotopes from G- to H- representation is straightforward. For a general zonotope, the G-matrix is of dimension  $n \times p$ . Because  $n - 1$  generators have to be selected from  $p$  generators for each non-parallel facet, the result are  $2 \binom{p}{n-1}$  facets. The generators that span a facet are obtained by cancelling  $p - n + 1$  generators

<sup>3</sup> Here, the removal is denoted by  $\langle \rangle$  and not by  $[]$  because a column instead of a row is removed.

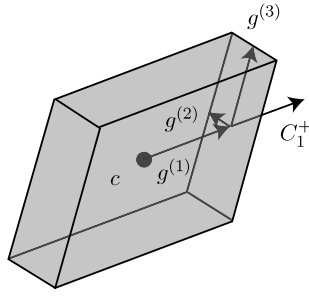


Fig. 6. Generators and normal vector of a parallelotope.

from the  $G$ -matrix. This is denoted by  $G^{(\gamma, \dots, \eta)}$ , where  $\gamma, \dots, \eta$  are the  $p - n + 1$  indices of the generators that are taken out of  $G$ . The extended computation for the halfspace representation is:

**Theorem 7** (Halfspace Representation of Zonotopes). *The halfspace representation  $C \cdot x \leq d$  of a zonotope with  $p$  independent generators is*

$$C = \begin{bmatrix} C^+ \\ -C^+ \end{bmatrix}, \quad d = \begin{bmatrix} d^+ \\ d^- \end{bmatrix}$$

with:

$$C_i^+ = nX(G^{(\gamma, \dots, \eta)})^T / \|nX(G^{(\gamma, \dots, \eta)})\|_2,$$

$$d_i^+ = C_i^+ \cdot c + \Delta d_i, \quad d_i^- = -C_i^+ \cdot c + \Delta d_i, \quad \Delta d_i = \sum_{v=1}^p |C_i^+ \cdot g^{(v)}|.$$

The index  $i$  varies from 1 to  $\binom{p}{n-1}$ . The complexity of the computation with respect to the number  $p$  of generators is  $\mathcal{O}(\binom{p}{n-1} \cdot p)$ , which is linear in the number of facets.

**Proof.** The computation of  $C_i^+$ ,  $d_i^+$ ,  $d_i^-$  is analog to Lemma 6. The difference for the computation of  $\Delta d_i$  is that  $p$  generators contribute to pushing the facets outwards.

The complexity result is obtained as follows: The computational complexity for computing the normal vectors is  $\mathcal{O}(\binom{p}{n-1} \cdot n^4)$  since the complexity for a single facet is  $\mathcal{O}(n^4)$  (see Lemma 6). The computational complexity for the computation of  $\Delta d$  is  $\mathcal{O}(\binom{p}{n-1} \cdot p)$  as  $p$  generators have to be considered for each element  $\Delta d_i$ . The overall complexity with respect to the number of generators is  $\mathcal{O}(\binom{p}{n-1} \cdot n^4) + \mathcal{O}(\binom{p}{n-1} \cdot p) = \mathcal{O}(\binom{p}{n-1} \cdot p)$ .  $\square$

The authors do not know any reference where an algorithm of similar complexity has been proposed. As the number of halfspaces is  $2 \binom{p}{n-1}$ , order reduction techniques for zonotopes are proposed in the following.

## 5.2. Order reduction of zonotopes

The order of zonotopes  $Z^{\text{orig}}$  is reduced in an over-approximating way, i.e. the reduced zonotope  $Z^{\text{red}}$  is described by less generators ( $r < p$ ) while ensuring that  $Z^{\text{red}} \supseteq Z^{\text{orig}}$ . In this work, the  $r$  generators are selected from the  $p$  original generators, which are then stretched in order to ensure  $Z^{\text{red}} \supseteq Z^{\text{orig}}$ . The order reduction of zonotopes is motivated by saving computational time, what is achieved when using  $r < p$  generators for the change of the zonotopes to halfspace representation. Computational time is also reduced for the enclosure of polytopes obtained from the intersection of zonotopes with guard sets (see Fig. 5(d)). The enclosure is based on the vertex representation of the polytopes, which requires less vertices when the zonotopes are represented by less generators before the intersection.

First, reduction techniques are assessed for the reduction of zonotopes to parallelotopes, and later for the reduction to zonotopes of order greater than one. For the reduction of the zonotope order, it is useful to introduce how to over-approximate a zonotope  $Z = (c, g^{(1)}, \dots, g^{(p)})$  by an axis-aligned box in  $G$ -representation, as presented e.g. in [11]:

$$\mathcal{H} = \text{box}(Z) = (c, b^{(1)}, \dots, b^{(n)}) \quad (2)$$

$$\text{with } b_j^{(i)} = \begin{cases} \sum_{v=1}^p |g_j^{(v)}|, & \text{for } i = j \\ 0, & \text{for } i \neq j. \end{cases}$$

The subscript  $j$  of  $b_j^{(i)}$  denotes the  $j$ th element of  $b^{(i)}$ . The  $\text{box}()$  operator allows us to compute the over-approximation of a zonotope by a parallelotope:



**Proposition 8** (Order Reduction to a Parallelotope). An over-approximating parallelotope  $\Psi$  is obtained from a zonotope  $Z$  by:

$$\Psi = \Gamma \cdot \text{box}(\Gamma^{-1}Z)$$

where  $\Gamma \in \mathbb{R}^{n \times n}$  is a matrix of  $n$  generators  $g^{(i)}$  taken out of the set of all  $p$  generators.  $\Gamma$  has to have full rank.

**Sketch of Proof.** This approach first transforms the coordinates of  $Z$  by the linear map  $\Gamma^{-1}$ , where the new coordinate axes are the  $n$  chosen generators within  $\Gamma$ . Note that this coordinate system is not orthogonal in general. Within the transformed coordinate system, the zonotope is over-approximated by a box ( $\text{box}(\Gamma^{-1}Z)$ ) aligned to the coordinate axes. As a final step, the zonotope is transformed back to the original coordinate system. The over-approximation is guaranteed by the fact, that the zonotope is over-approximated in the transformed coordinate system by the  $\text{box}()$  operator, such that it is also over-approximated after the transformation to the original coordinate system.  $\square$

For the reduction to zonotopes of order greater than 1, the following scheme is applied:

**Proposition 9** (Order Reduction of a Zonotope). An over-approximating zonotope with reduced number of generators  $Z^{\text{red}}$  is obtained by splitting the zonotope  $Z = \check{Z} + \tilde{Z}$  into  $\check{Z}$  and  $\tilde{Z}$ , and by reducing  $\tilde{Z}$  to a parallelotope afterwards:

$$Z^{\text{red}} = \check{Z} + \Psi, \quad \Psi = \Gamma \cdot \text{box}(\Gamma^{-1} \cdot \tilde{Z}).$$

The split into  $(\check{Z}, \tilde{Z})$  and the selection of  $\Gamma$  (which has to have full rank) has to be decided by an order reduction scheme.

Note that the addition of zonotopes  $\check{Z} = (\check{c}, \check{g}^{(1)}, \dots, \check{g}^{(\check{p})})$  and  $\tilde{Z} = (\tilde{c}, \tilde{g}^{(1)}, \dots, \tilde{g}^{(\tilde{p})})$  is simply computed by adding the centers and concatenating the generators:  $\check{Z} + \tilde{Z} = (\check{c} + \tilde{c}, \check{g}^{(1)}, \dots, \check{g}^{(\check{p})}, \tilde{g}^{(1)}, \dots, \tilde{g}^{(\tilde{p})})$ . This is because of the nature of zonotopes which are defined by the Minkowski addition of generators. It remains to find heuristics that select the generators of  $\Gamma$  in Propositions 8 and 9 as well as generators of  $\check{Z}$  in Proposition 9 (which remain unchanged).

### 5.3. Generator selection

This subsection presents heuristics that allow us to efficiently select generators for the order reduction of zonotopes. First, the generator selection for the over-approximation to parallelotopes (Proposition 8) is addressed. The generalization for the abstraction to zonotopes of order greater than one (Proposition 9) is considered afterwards. In order to assess the quality of the order reduction, an over-approximation index is introduced next.

#### 5.3.1. Over-approximation index

The proposed over-approximation measure  $\Theta$  is based on the ratio of the volume of the over-approximating set  $S^{\text{red}}$  compared to the original set  $S^{\text{orig}} : V(S^{\text{red}})/V(S^{\text{orig}})$  and  $V()$  is the operator returning the volume of a set. It is additionally considered, that each dimension represents a variable whose values may differ in scale, such that they have to be normalized. For this reason, the considered sets are weighted by some weighting matrix  $W = \text{diag}(w)$ ,  $w \in \mathbb{R}^n$ , resulting in the ratio  $V(W \cdot S^{\text{red}})/V(W \cdot S^{\text{orig}})$ . Another aspect of the over-approximation measure is that the measure should allow us to compare the results of over-approximations obtained in vector spaces of different dimension. For this reason, the  $n$ th root of the volume ratio is taken, which is equivalent to the ratio of the edge length of  $n$ -dimensional cubes containing the corresponding volumes. Combining the mentioned aspects, the over-approximation measure is computed as:

$$\Theta = \left( \frac{V(W \cdot S^{\text{red}})}{V(W \cdot S^{\text{orig}})} \right)^{\frac{1}{n}}, \quad W = \text{diag}(w), w \in \mathbb{R}^n.$$

In order to simplify the notation, it is assumed from now on, that all sets  $S$  are represented in a normalized vector space, where the original sets have been mapped to  $S^{\text{orig}} := W \cdot S^{\text{orig}}$  beforehand, such that the weighting matrix in the normalized state space is the identity matrix ( $W = I$ ).

#### 5.3.2. Generator selection for parallelotopes

Given is the task to select  $n$  out of  $p$  generators for the generator matrix  $\Gamma$  in order to compute over-approximating parallelotopes as shown in Proposition 8. The selection should minimize the over-approximation index  $\Theta$  of the resulting parallelotopes  $\Psi$ . A straightforward approach is an exhaustive search by computing the over-approximation value  $\Theta$  for all possible combinations of  $n$  out of  $p$  generators:

**Proposition 10** (Exhaustive Search). The exhaustive search for the best combination of  $n$  out of  $p$  generators for the over-approximation of a zonotope by a parallelotope (Proposition 8) has complexity  $\mathcal{O}(\binom{p}{n} \cdot p)$  with respect to the number of generators  $p$ . Instead of searching generators which minimize the over-approximation measure  $\Theta$ , one can alternatively evaluate the over-approximation measure  $\Theta^* = V(S^{\text{red}})$  which is computationally less expensive.



**Proof.** The exhaustive search allows us to generate  $\binom{p}{n}$  different parallelotopes. The complexity for the computation of the parallelotopes with respect to the number of generators according to Proposition 8 is  $\mathcal{O}(p)$ . The complexity for the sorting of the  $\Theta$  values of all parallelotopes is at worst  $\mathcal{O}(i \cdot \log(i))$  when applying e.g. a merge-sort algorithm, where  $i$  is the number of parallelotopes. Thus, the overall complexity of the exhaustive search is  $\mathcal{O}(\binom{p}{n} \cdot p) + \mathcal{O}(\binom{p}{n} \log \binom{p}{n}) = \mathcal{O}(\binom{p}{n} \cdot p)$ .

As all parallelotopes are over-approximations of the same zonotope, the volume  $V(S^{\text{orig}})$  and the dimension  $n$  are equal for all parallelotopes. Thus, the over-approximation index  $\Theta^* = V(S^{\text{red}})$  is minimal if  $\Theta = (V(S^{\text{red}})/V(S^{\text{orig}}))^{1/n}$  is minimal.  $\square$

Clearly, the exhaustive search is infeasible in high dimensions and for zonotopes of high order. For this reason, the number of possible combinations  $\binom{p}{n}$  for the exhaustive search has to be reduced. Two approaches are discussed, where one approach reduces the number of generators  $\tilde{p} < p$  for the exhaustive search. The other approach passes only a subset of reasonable combinations of  $n$  generators, which is then evaluated by  $\Theta^*$ .

*Reduction of the number of generators.* A method which has proven effective for extracting  $\tilde{p}$  promising generators  $g^{(i_1)}, \dots, g^{(i_{\tilde{p}})}$  is to select the longest generators, such that  $\|g^{(i_1)}\|_2 \geq \dots \geq \|g^{(i_{\tilde{p}})}\|_2 \geq \|g^{(i_{\tilde{p}+1})}\|_2 \geq \dots \geq \|g^{(i_p)}\|_2$ . The importance for selecting promising generators is apparent when investigating the complexity for  $\tilde{p} = n + \kappa$  ( $\kappa \in \mathbb{N}^+$ ) generators:

**Proposition 11.** If  $\tilde{p}$  is chosen as  $\tilde{p} = n + \kappa$ , the complexity with respect to  $\kappa$  is  $\mathcal{O}(\kappa^{n+1})$ , i.e. polynomial in  $\kappa$  for fixed  $n$ .

**Proof.** The complexity of the exhaustive search is according to Proposition 10

$$\mathcal{O}\left(\binom{n+\kappa}{n} \cdot (n+\kappa)\right) = \mathcal{O}\left((n+\kappa)^n \cdot (n+\kappa)\right) = \mathcal{O}(\kappa^{n+1}). \quad \square$$

Besides extracting the longest generators, clustering methods like k-means have also been investigated. The data points of the clusters have been defined as the end points of the generators starting in the origin. Next, the vectors from the origin to the cluster centers have been chosen as the reduced set of generators. However, in numerical experiments this approach did not perform better than the selection based on the length of the generators.

*Reduction of the number of combinations of generators.* The second mentioned possibility to accelerate the generator selection is to compute the over-approximation index  $\Theta^*$  on a subset of possible combinations of generators. The subset of combinations of generators is selected by an alternative over-approximation index  $\tilde{\Theta}$  which is computationally less expensive and is computed on all  $\binom{p}{n}$  combinations of generators. In this work, the alternative measure is computed as:  $\tilde{\Theta} = |\det[g^{(i_1)}, \dots, g^{(i_n)}]|^{-1}$ , where  $|\det[g^{(i_1)}, \dots, g^{(i_n)}]|$  is the volume spanned by the  $n$  selected generators. By picking the combination with the lowest over-approximation index, the generator combination spanning the largest volume is selected. This is motivated by the observation that a volume maximization of the parallelotope spanned by the selected generators tends to minimize the volume of the overall obtained parallelotope  $\Psi$ .

It is remarked that the proposed order reduction techniques are not stable if the representation of the zonotope is changed. Consider a zonotope with longest generator  $g^{\text{max}}$ . After replacing this generator by  $\frac{1}{2}g^{\text{max}}$  and  $\frac{1}{2}g^{\text{max}}$ , the zonotope is unchanged (only its representation is changed). However, it is likely that the generators  $\frac{1}{2}g^{\text{max}}$  will not be selected such that the over-approximation computed by the algorithm will not be the same as before. This problem will be the subject of future research.

### 5.3.3. Generator selection for zonotopes

In this subsection, the generator selection for parallelotopes is extended to zonotopes of order greater than one, so that Proposition 9 can be applied. According to Proposition 9, the generator selection can be split into two subtasks: the selection of  $u$  generators for  $\tilde{Z}$  which are not reduced and the selection of generators which are over-approximated by the parallelotope  $\Psi$ . The center of the original zonotope can be assigned to either  $\tilde{Z}$  or  $\Psi$ . The generator selection for  $\Psi$  is performed as previously shown in Section 5.3.2. The remaining generators of  $\tilde{Z}$  are chosen as the longest generators, what has also been applied in Section 5.3.2 for the preselection of generators.

## 5.4. Numerical results

This subsection combines techniques for the order reduction of zonotopes with the conversion from generator to halfspace representation and shows numerical results. The halfspace representation can be obtained in a direct way or by intersection of different halfspace representations. In order to evaluate the proposed methods, one has to randomly generate zonotopes, which is described next.

### 5.4.1. Randomly generated zonotopes

For the evaluation of the over-approximating methods, zonotopes are generated by independent randomized generators  $g^{(i)}$ . In a first step, randomized points are obtained, which are uniformly distributed on a unit hypersphere. These random points are generated by computing  $x/\|x\|_2$ , where each element of  $x \in \mathbb{R}^n$  is a Gaussian random variable, see [21]. Next, the generator is defined as the vector from the origin to a point on the hypersphere, which is stretched by the length of the generators  $\hat{r}^i = \|g^{(i)}\|_2$  and  $\hat{r}^i$  has a uniform distribution within the interval  $0 < \hat{r}^i \leq r^{\text{max}}$ .

**Table 1**

Results for conversion to halfspace representation via parallelotopes.

Meth.	Mean of $t^{comp}$ (s):	Mean of $\Theta$ :	[min, max] of $\Theta$ :	Variance of $\Theta$ :
Dimension $n = 2$ , zonotope order $o = 2$				
A, B, C:	0.056	1.0582	[1.0030, 1.1349]	0.0011
Dimension $n = 2$ , zonotope order $o = 6$				
A:	0.0289	1.0858	[1.0362, 1.1164]	0.0003
B:	0.0204	1.0859	[1.0362, 1.1164]	0.0003
C:	0.0056	1.0908	[1.0369, 1.1522]	0.0005
Dimension $n = 4$ , zonotope order $o = 2$				
A, B:	0.0325	1.1550	[1.0343, 1.2606]	0.0022
C:	0.0078	1.1560	[1.0343, 1.2899]	0.0024
Dimension $n = 4$ , zonotope order $o = 6$				
A:	5.0297	1.2687	[1.2101, 1.3096]	0.0004
B:	0.2019	1.2838	[1.2101, 1.3743]	0.0010
C:	0.0060	1.2967	[1.2143, 1.3995]	0.0015
Dimension $n = 6$ , zonotope order $o = 2$				
A, B:	0.3876	1.2556	[1.0779, 1.3868]	0.0037
C:	0.0221	1.2574	[1.0779, 1.4088]	0.0039
Dimension $n = 8$ , zonotope order $o = 2$				
A:	> 10 min	—	[—, —]	—
B:	7.1025	1.3641	[1.1845, 1.5247]	0.0041
C:	1.1140	1.3670	[1.1845, 1.5756]	0.0046

#### 5.4.2. Direct conversion

The direct conversion of zonotopes, which over-approximates the zonotopes by a halfspace representation, is performed the following:

- (1) Reduce the order of the zonotope as shown in Proposition 9.
- (2) Apply the conversion from generator to halfspace representation as presented in Theorem 7.

The numerical results for the evaluation of the direct conversion are separately discussed for the case when the zonotopes are reduced to parallelotopes or to zonotopes of order greater than one. For the reduction to parallelotopes, three methods are evaluated:

- **Method A.** This method denotes the exhaustive search.
- **Method B.** The exhaustive search is performed on the  $\tilde{p} = n + \kappa$  longest generators, where  $n$  is the system dimension and  $\kappa$  is chosen to  $\kappa = 8$ .
- **Method C.** Method C is based on method B. Instead of computing the over-approximation index  $\Theta^*$  on all combinations of the reduced set of  $\tilde{p} = n + \kappa$  generators ( $\kappa = 8$ ), the alternative measure  $\Theta$  is computed on the combinations of  $\tilde{p} = n + \kappa$  generators, from which the best  $\lambda = n + \tilde{\kappa}$  combinations are chosen and  $\tilde{\kappa} = 3$ . Next, the over-approximation measure  $\Theta^*$  is computed only on those  $\lambda$  combinations.

The results of the evaluation of the introduced methods are presented in Table 1. The over-approximation index  $\Theta$  as well as the overall computational time  $t^{comp}$  are obtained from 100 randomized zonotopes, where the randomization is performed as described previously in Section 5.4.1. The computations were performed on a 3.7 GHz single core desktop computer in Matlab. Due to the computational complexity for the computation of the volume of the original zonotopes, which is necessary to obtain the over-approximation index  $\Theta$ , the evaluation for dimensions greater than 8 is intractable. It can be observed that the performance limitation from method A to C is small compared to the considerable reduction of computational time. For this reason, method C is suggested for the over-approximative halfspace conversion using parallelotopes.

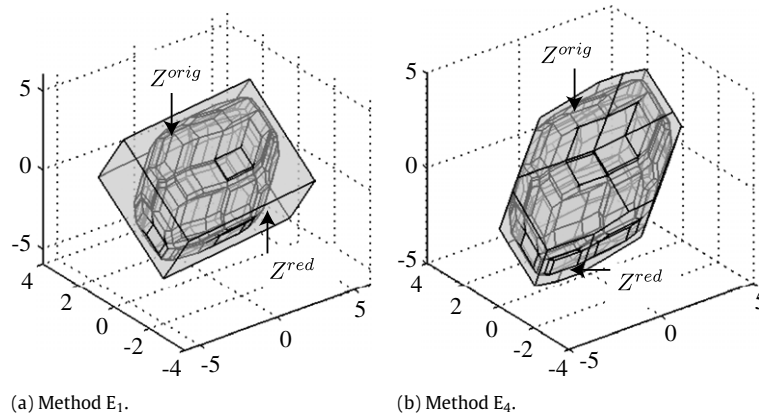
The extension to the halfspace conversion of zonotopes with order greater than one is based on method C, which provides the parallelotope  $\Psi$  of Proposition 9. The extended method is denoted by  $D_u$ , where  $u$  denotes the number of unreduced generators in  $\tilde{Z}$ .

Analogously, to the previous results for parallelotopes, 100 randomized zonotopes have been generated. The results for the computational time  $t^{comp}$  and the over-approximation index  $\Theta$  are displayed in Table 2. Note that method  $D_2$  is faster computed compared to  $D_0$  for 8 dimensions, because the computation for  $\Psi$  saves more time (due to less combinations for  $\Gamma$ ) than the more complicated halfspace conversion adds. Compared to the previous results in Table 1, the computational costs are much higher due to the halfspace conversion of zonotopes of greater order. On the other hand, the increase in accuracy is marginal, hence the reduction methods  $D_u$  are not favored.

**Table 2**

Results for conversion to halfspace representation via reduced zonotopes.

Meth.	Mean of $t^{comp}$ (s):	Mean of $\Theta$ :	[min, max] of $\Theta$ :	Variance of $\Theta$ :
Dimension $n = 2$ , zonotope order $o = 6$				
$D_0$ :	0.0063	1.0888	[1.0433, 1.1346]	0.0004
$D_2$ :	0.0062	1.0853	[1.0291, 1.1846]	0.0007
$D_4$ :	0.0067	1.0727	[1.0174, 1.2506]	0.0012
Dimension $n = 4$ , zonotope order $o = 2$				
$D_0$ :	0.0084	1.1615	[1.0250, 1.2830]	0.0030
$D_2$ :	0.0156	1.0925	[1.0087, 1.2588]	0.0028
$D_4$ :	0.0600	1.0000	[1.0000, 1.0000]	0.0000
Dimension $n = 4$ , zonotope order $o = 6$				
$D_0$ :	0.0145	1.3001	[1.1930, 1.4498]	0.0018
$D_2$ :	0.0235	1.2857	[1.1782, 1.4827]	0.0018
$D_4$ :	0.0728	1.2607	[1.1699, 1.3860]	0.0018
Dimension $n = 6$ , zonotope order $o = 2$				
$D_0$ :	0.0225	1.2675	[1.0797, 1.4112]	0.0040
$D_2$ :	0.0795	1.1979	[1.0431, 1.4076]	0.0050
$D_4$ :	1.6178	1.1056	[1.0222, 1.3337]	0.0032
Dimension $n = 8$ , zonotope order $o = 2$				
$D_0$ :	0.9876	1.3638	[1.1706, 1.5302]	0.0054
$D_2$ :	0.5163	1.3150	[1.1653, 1.5132]	0.0049
$D_4$ :	39.189	1.2126	[1.0227, 1.4069]	0.0053

**Fig. 7.** Over-approximative halfspace representations.

#### 5.4.3. Intersection of halfspace representations

In order to improve the accuracy of the over-approximation to parallelotopes, zonotopes of order greater than one have been investigated previously in Section 5.4.2. An alternative approach, which is much more efficient, is based on parallelotopes:

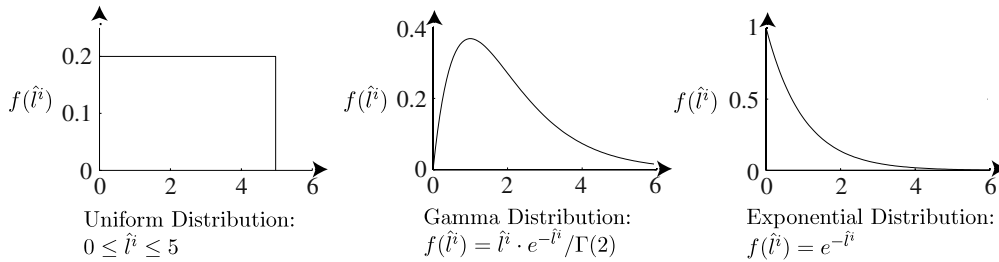
- (1) Compute the  $\zeta$  best over-approximations of zonotopes by parallelotopes.
- (2) Convert the generator representation of the parallelotopes to the halfspace representation as presented in Lemma 6.
- (3) Intersect the  $\zeta$  best parallelotopes in halfspace representation in order to obtain the over-approximating halfspace representation.

The method that intersects parallelotopes is denoted by  $E_\zeta$ , where  $\zeta$  is the number of intersected parallelotopes. For the over-approximation to parallelotopes, method C is applied. Using again 100 randomized zonotopes, the results for the over-approximation index  $\Theta$  and the computational time  $t^{comp}$  are listed in Table 3. The intersections of the halfspace representations have been performed using the MPT-Toolbox [22]. Compared to the results for zonotopes of order greater than one in Table 2, the results for the intersections of parallelotopes are computationally less costly and more accurate. Thus, method  $E_\zeta$  is the preferred method for the computation of halfspace representations. The difference in accuracy is also illustrated in Fig. 7 for a three-dimensional zonotope. The left halfspace representation is obtained using method  $E_1$  ( $\hat{=}$  C) and the right one is obtained using method  $E_4$ .

**Table 3**

Results for conversion to halfspace representation via parallelotope intersections.

Meth.	Mean of $t^{comp}$ (s):	Mean of $\Theta$ :	[min, max] of $\Theta$ :	Variance of $\Theta$ :
Dimension $n = 2$ , zonotope order $o = 6$				
$E_1$ :	0.0064	1.0918	[1.0352, 1.1380]	0.0005
$E_2$ :	0.0109	1.0548	[1.0088, 1.1368]	0.0003
$E_4$ :	0.0223	1.0191	[1.0025, 1.0561]	0.0002
Dimension $n = 4$ , zonotope order $o = 2$				
$E_1$ :	0.0085	1.1528	[1.0438, 1.2868]	0.0031
$E_2$ :	0.0168	1.0774	[1.0136, 1.1739]	0.0014
$E_4$ :	0.0382	1.0288	[1.0019, 1.0836]	0.0003
Dimension $n = 4$ , zonotope order $o = 6$				
$E_1$ :	0.0142	1.2948	[1.2294, 1.4025]	0.0014
$E_2$ :	0.0218	1.2106	[1.1191, 1.3932]	0.0019
$E_4$ :	0.0421	1.1383	[1.0808, 1.2892]	0.0010
Dimension $n = 6$ , zonotope order $o = 2$				
$E_1$ :	0.0234	1.2661	[1.1316, 1.4260]	0.0047
$E_2$ :	0.0363	1.1638	[1.0778, 1.2604]	0.0021
$E_4$ :	0.0739	1.0964	[1.0251, 1.1759]	0.0010

**Fig. 8.** Probability distributions of the generator lengths of randomized zonotopes.**Table 4**

Results for conversion to halfspace representation using different randomized zonotope categories.

Meth.	Distr.	Mean of $\Theta$ for dimension $n$ and order $o$			
		$n : 2, o : 6$	$n : 4, o : 2$	$n : 4, o : 6$	$n : 6, o : 2$
$E_1$ :	Uniform	1.0845	1.1625	1.2892	1.2678
$E_1$ :	Gamma	1.0872	1.1574	1.2816	1.2502
$E_1$ :	Exponential	1.0794	1.1217	1.2536	1.2060
$E_4$ :	Uniform	1.0178	1.0336	1.1355	1.1026
$E_4$ :	Gamma	1.0148	1.0355	1.1201	1.0955
$E_4$ :	Exponential	1.0106	1.0216	1.1013	1.0709

As method  $E_\zeta$  has been found to be the most efficient method to obtain a halfspace representation of a zonotope, the performance of the method is evaluated according to different categories of randomly generated zonotopes. The randomized zonotopes used so far had a uniform distribution of the length of the generators, see Section 5.4.1. If the length is e.g. exponentially distributed, there would be only a few dominant generators while the remaining generators have much smaller length. For this reason, the reduction performance is additionally evaluated for the distributions plotted in Fig. 8. The mean values of the over-approximation measure for the reduction of zonotopes generated by these distributions are shown in Table 4. It can be observed that distributions which thin out for high values (such as the exponential distribution) result in zonotopes that can be better reduced as those having only few characteristic generators. The performance of the halfspace conversion will also be evaluated with reachable sets of the numerical example at the end of this work. In the next section, the reverse problem of finding an over-approximative zonotope for a set of polytopes is studied.

## 6. Over-approximative generator representations of polytopes

Following the scheme in Fig. 5, the over-approximation of zonotopes by a halfspace representation has been addressed in the previous section. The intersection of the reachable sets with the guard sets is performed by a standard software package for polytopes (MPT-Toolbox [22]). Due to the intersection with the guard sets, the zonotopes in halfspace representation

become polytopes in halfspace representation. In case of jumps, the halfspace representation has to be changed according to the jump function, which has been limited to linear jumps, such that the mapping is exact. It remains to over-approximate the reachable sets by a single or several zonotopes, in order to continue the reachable set computations for the next continuous dynamics. The computation of the enclosing zonotope is carried out by computing the vertices of the polytopes, such that the problem is to tightly enclose points in  $\mathbb{R}^n$  by a zonotope. Much previous work solved the problem of finding tightly enclosing, oriented boxes in three-dimensional space, see e.g. [23]. For  $n$ -dimensions, enclosures of points have been investigated in [10].

The following subsections are structured as follows: the enclosure of polytopes by a single zonotope is discussed first in general. Next, some specific methods are presented, and finally the enclosure of the reachable set by several zonotopes is discussed.

### 6.1. General approach

The enclosure of polytopes by one or several zonotopes is carried out by the following proposition, which is analogous to the order reduction of zonotopes (Proposition 8), except that the  $\text{box}()$  operator is generalized for polytopes:

**Proposition 12** (Enclosure of Polytopes). *An over-approximating zonotope  $Z^{\text{encl}}$  of  $\mu$  polytopes  $P_l$  is obtained as:*

$$Z^{\text{encl}} = \Lambda \cdot \text{box}(\Lambda^{-1}P_1, \dots, \Lambda^{-1}P_\mu),$$

where  $\Lambda \in \mathbb{R}^{n \times n}$  is of full rank and  $P_l$  is the  $l$ th polytope. The  $\text{box}()$ -operator for several polytopes, returning an  $n$ -dimensional interval in generator representation, is computed as follows:

- (1) Compute the vertices  $v^j \in \mathbb{R}^n$  ( $V$ -representation) of all  $\mu$  polytopes  $P_l' = \Lambda^{-1}P_l$ .
- (2) Obtain the minimum and maximum values of each dimension  $i$ :

$$\underline{m}_i = \min(v_i^1, \dots, v_i^v), \quad \bar{m}_i = \max(v_i^1, \dots, v_i^v),$$

where  $v_i^j$  is the  $i$ th component of  $v^j$  and  $v$  is the number of the vertices of all polytopes.

- (3) Compute a generator representation of the  $n$ -dimensional interval  $[\underline{m}, \bar{m}]$ :

$$[\underline{m}, \bar{m}] = \left\{ x = c + \sum_{i=1}^n \beta^i \cdot g^{(i)}, -1 \leq \beta^i \leq 1 \right\},$$

where

$$c = 0.5(\underline{m} + \bar{m}), \quad g_i^{(j)} = \begin{cases} 0.5(\bar{m}_i - \underline{m}_i), & \text{if } i = j \\ 0, & \text{otherwise.} \end{cases}$$

The proof is omitted as it is analogous to Proposition 8. Similar to the order reduction of zonotopes, the challenge is to find a linear map  $\Lambda$ , which over-approximates the polytopes in a good or the best way. Note, that the column vectors of  $\Lambda$  determine the direction of the generators of the over-approximating zonotope. The choice of the  $\Lambda$  matrix, and hence the selection of generators for the over-approximating zonotope, is addressed next.

### 6.2. Generator selection for the enclosing zonotope

In order to obtain tight enclosures of the polytopes intersected with a guard set, the direction of the flow vector  $f(z, x, u)$  of the continuous dynamics

$$\dot{x} = f(z, x, u) = A_z x + u, \quad u \in U_z$$

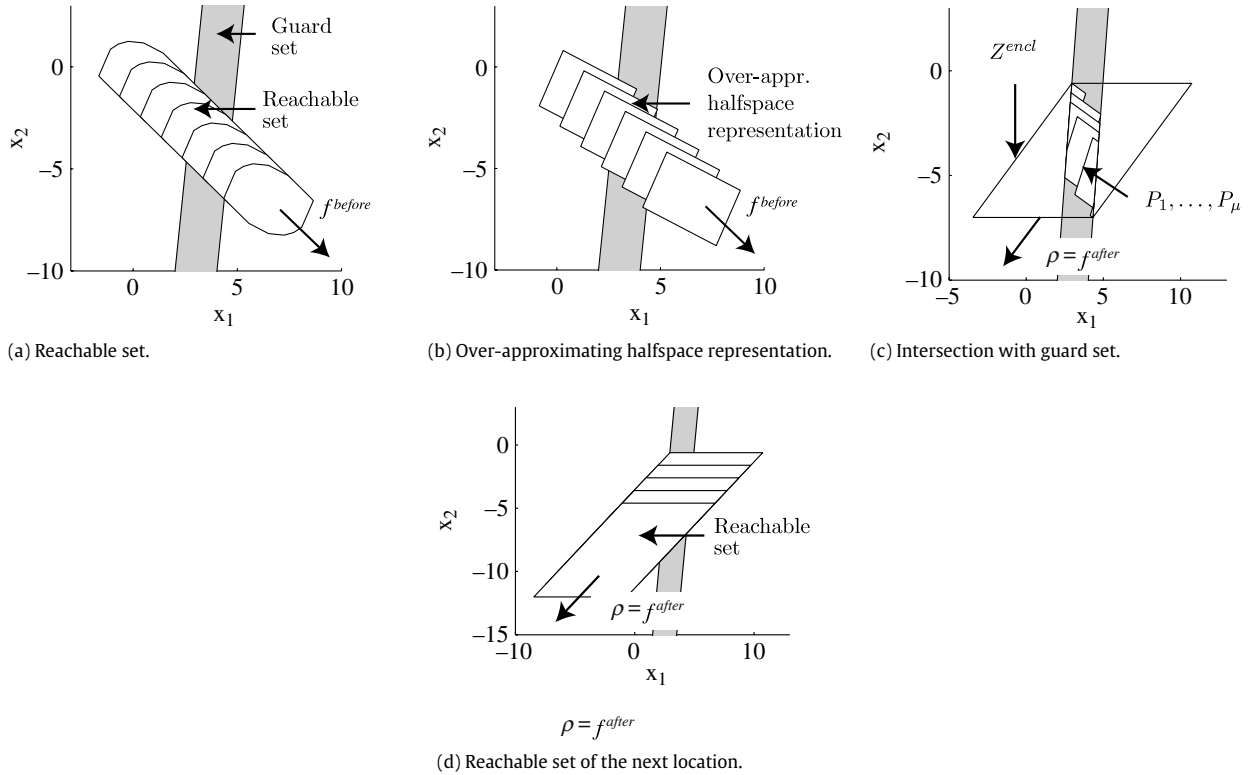
is taken into account, where  $z$  was the index referring to the location. There are two reasonable choices of flow vectors: the flow vector of the location which is active before the transition is taken, and the one after the transition is taken. First, the flow vector that is active after the transition is taken, is considered. The direction  $\rho$  of the flow vector is taken at the center of the enclosing boxes of the reachable set and the input:

$$\rho = \frac{f(z, x^*, u^*)}{\|f(z, x^*, u^*)\|_2}, \quad x^* = \text{center}(\text{box}(P_1, \dots, P_\mu)), \quad u^* = \text{center}(U_z),$$

where  $\text{box}()$  is defined as in Proposition 12 and  $\text{center}()$  returns the volumetric center of a set. By choosing  $\rho$  as one of the column vectors of  $\Lambda$ , the enclosing zonotope is oriented in the direction of the flow vector. The remaining column vectors are chosen as  $n - 1$  unit vectors  $e^j$  of the coordinate system, where  $e_i^j = 1$  if  $j = i$  and  $e_i^j = 0$  otherwise. The replaced unit vector  $e^j$  is the one, that is best aligned with  $\rho$ , such that:

$$\Lambda = [\dots, e^{j-1}, \rho, e^{j+1}, \dots], \quad |(e^j)^T \cdot \rho| \geq |(e^m)^T \cdot \rho|, \quad \forall m = 1 \dots n.$$

The procedure for the enclosure of the polytopes is illustrated for a simple two-dimensional example, where  $f^{\text{before}} = [1, -1]^T$  and  $f^{\text{after}} = [-1, -1]^T$  are the constant flow vectors before and after the transition without a jump. The reachable sets are illustrated in Fig. 9(a)–(d).



**Fig. 9.** Enclosure of the reachable set by a single zonotope.

Besides this method, there are alternative possibilities to obtain reasonable  $\Lambda$  matrices:

- Instead of replacing a unit vector  $e^i$  by the direction vector  $\rho$ , one can construct the  $\Lambda$  matrix as a  $n$ -dimensional rotation matrix ( $\det(\Lambda) = 1$ ) that maps the direction  $\rho$  to  $e^1$ .
- One can also apply principal component analysis (PCA) as presented in [10]. The  $\Lambda$  matrix then results from a singular value decomposition.

In the next subsection, it is outlined how these alternative approaches can be integrated.

### 6.3. Computing with several enclosing zonotopes

In order to combine the advantages of different approaches for the enclosure of points, one can compute several alternative enclosures  $Z^{encl,1}, \dots, Z^{encl,\omega}$ . From this follows, that starting from these enclosures, many reachable sets have to be computed in parallel within the active location. Note, that in this case, an unsafe set is only reached if the intersections of reachable sets (of common time intervals) intersect the unsafe set.

The slightly modified procedure for the computation of the intersection with guard sets is explained for the case where only two enclosing zonotopes  $Z^{encl,1}, Z^{encl,2}$  are used (what can be extended in a straightforward way to more enclosing zonotopes). One difference to the previous description is that the guard set is only reached, if it is intersected with both reachable sets  $R^1([t_{k-1}, t_k])$ ,  $R^2([t_{k-1}, t_k])$ . The polytopes obtained from the intersection with a guard set are denoted by  $P^1([t_{e1}, t_{e1+1}]), \dots, P^1([t_{l1-1}, t_{l1}])$  and  $P^2([t_{e2}, t_{e2+1}]), \dots, P^2([t_{l2-1}, t_{l2}])$ . Because of the parallel computation, the combined entry and exit time of the guard set are  $t_e = \max(t_{e1}, t_{e2})$  and  $t_l = \min(t_{l1}, t_{l2})$ . Further, the combined polytopes within a guard set are:  $P([t_{k-1}, t_k]) = P^1[t_{k-1}, t_k] \cap P^2[t_{k-1}, t_k]$  with  $t_{k-1} \geq t_e$ ,  $t_k \leq t_l$ . After obtaining the sets  $P([t_{k-1}, t_k])$ , the enclosure of the reachable sets is performed as shown in the previous subsection.

The tighter enclosure of the reachable set when using two enclosing zonotopes is shown for the same example that has been used in Fig. 9. The additional zonotope is obtained by the same method, but the flow vector of the previous location is additionally used. Both enclosing zonotopes as well as the intersected reachable sets  $R^1([t_{k-1}, t_k]) \cap R^2([t_{k-1}, t_k])$ , are illustrated in Fig. 10.

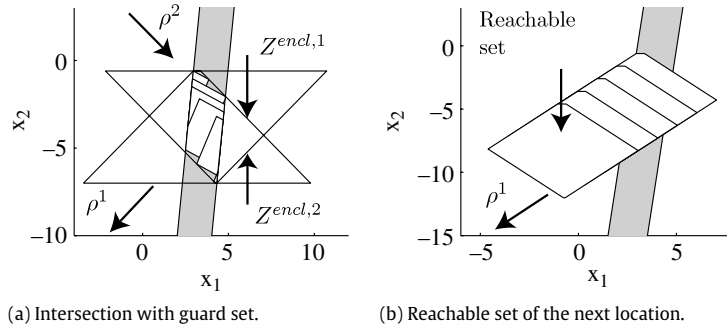


Fig. 10. Enclosure of the reachable set by a single zonotope.

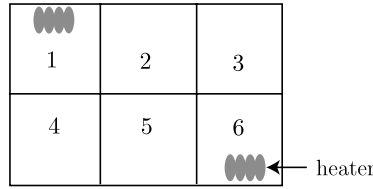


Fig. 11. Room layout.

## 7. Numerical example

The presented techniques are applied to a benchmark example proposed in [24]: It considers a room heating scenario, where in each room there is one or no heater. In contrast to the proposed benchmark example, it is not considered that heaters can be moved from one room into another. Extensions of the proposed example are that the input is modeled uncertain and that the switching is not deterministic. The example considered in this work consists of 6 rooms with heaters in the rooms 1 and 6, see Fig. 11. The heaters are switched on if the temperature drops below  $T^{low} + \Delta T$  and switched off if the temperature exceeds  $T^{high} - \Delta T$  with  $T^{low} = 22^\circ\text{C}$ ,  $T^{high} = 24^\circ\text{C}$  and  $\Delta T = [0, 0.05]^\circ\text{C}$ . The temperature dynamics in room  $i$  is:

$$\dot{x}_i = c \cdot h_i + b_i(u - x_i) + \sum_{j \neq i} a_{ij}(x_j - x_i)$$

with constants  $a_{ij}$ ,  $b_i$  and  $c$ . The rate of heat exchange  $a_{ij}$  between two adjacent rooms is 1. The transfer rate  $b_i$  from inside the building to the outside is 0.16 for rooms at corners and 0.08 for other rooms. The outside temperature  $u$  is in the interval of  $[0, 0.1]$  including modeling uncertainties, and the heating power is  $c = 15$  for both heaters. The variable  $h_i$  is 1 if a heater is switched on in room  $i$  and 0 otherwise. The reachable sets are computed for the time interval  $t \in [0, 1]$ .

The over-approximation of the zonotopes to a halfspace representation has been performed with method C with  $\kappa = 5$  for the preselection of generators by length. The enclosure of the polytopes within the guard sets has been performed by two zonotopes as explained in Section 6.3, where the flow vector of the location before and after the transition has been applied. The time step size has been set to  $r = 0.01$  and the computation time was 16.3 s on a single core desktop computer with 3.7 GHz for an implementation in Matlab.

In order to assess the halfspace conversion of this example, the zonotopes converted to halfspace representation have been recorded. As the uncertain input is small compared to the uncertainty arising from the switching of the system dynamics, the recorded zonotopes have only few dominant generators. This can be observed in the histogram in Fig. 12 showing the distribution of the generator length. From this follows that the halfspace conversion can be performed with high accuracy. The mean values of the relative over-approximation index ( $\Theta/\Theta^C$ ), where  $\Theta^C$  is the over-approximation index of method C, as well as the computational time for different halfspace conversion methods are presented in Table 5. The reason for the relative over-approximation index is that the absolute value cannot be computed – this requires the computation of the volume of the zonotopes of dimension 6 and order 10, which is intractable.

The reachable sets are shown as gray areas for selected projections in Fig. 13. Exemplary trajectories within the reachable set are plotted in black and have been computed for randomized initial states and inputs. In comparison with the trajectories, the reachable sets seem to have been computed for a longer time horizon. This is because the time point for a transition enabled by a guard  $g((z_i, z_j))$  is uncertain within  $[t_e^l, t_e^u]$  as discussed in Section 4, so that the uncertainty of computed time adds up to  $[t_f, t_f + \delta]$  and  $\delta$  is the total time uncertainty. A possible verification scenario would be to analyze whether a certain combination of room temperatures is enabled (or avoided) by the switching controller.



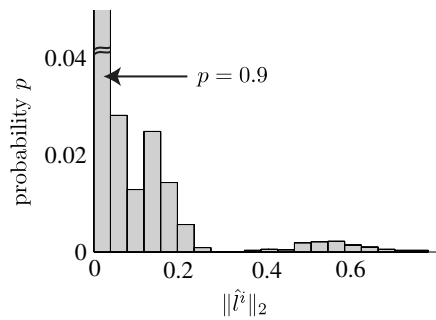


Fig. 12. Histogram of generator length  $\|\hat{l}^i\|_2$ .

Table 5

Results for the conversion to halfspace representation using zonotopes from the room heating example.

Mean of	Method B ( $\kappa = 3$ )	Method C ( $\kappa = 8, \tilde{\kappa} = 3$ )	Method D <sub>2</sub> ( $\kappa = 8, \tilde{\kappa} = 3$ )	Method E <sub>2</sub> ( $\kappa = 8, \tilde{\kappa} = 3$ )
$\Theta / \Theta^c$ :	1.0007	1.0000	1.0406	0.9203
$t^{comp}$ (s):	0.0442	0.0701	0.1549	0.0841

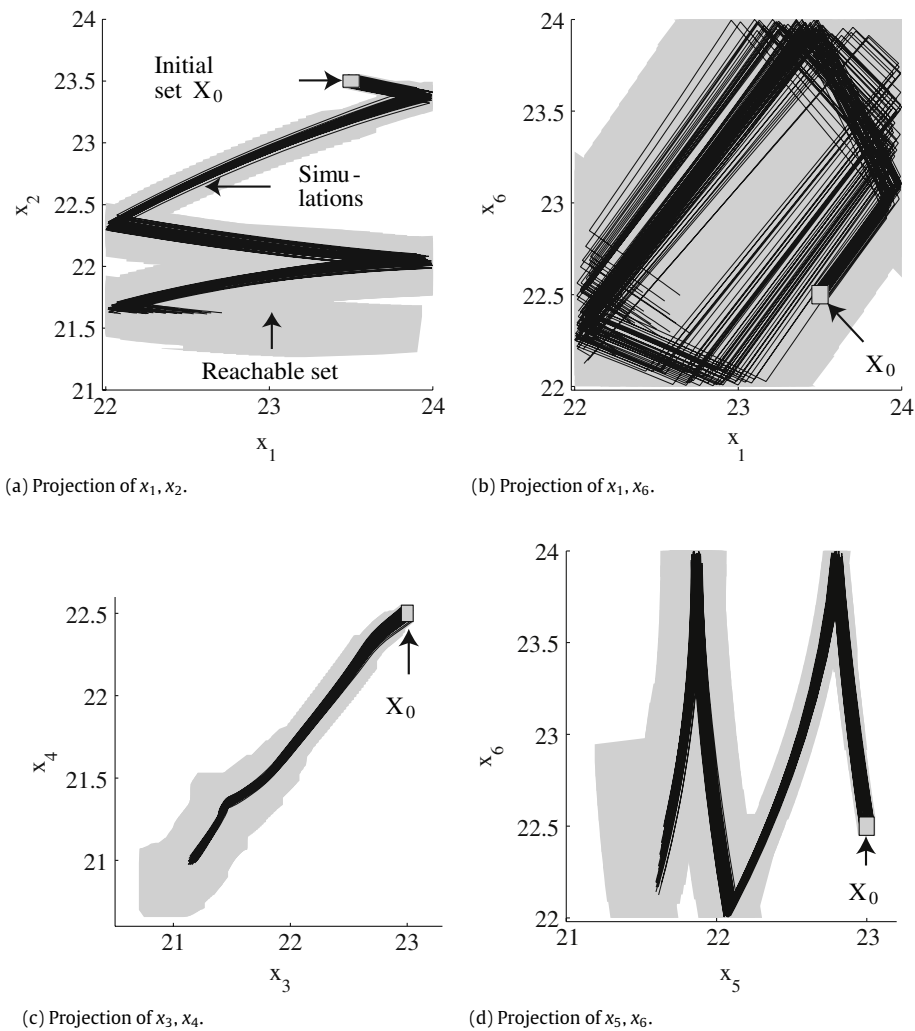


Fig. 13. Reachable sets of the room heating scenario.

## 8. Conclusion

The presented work has presented an approach which allows us to represent reachable sets of hybrid systems by zonotopes. The arising challenge when using zonotopes for hybrid systems is that reachable sets have to be intersected with guard sets and zonotopes are not closed under intersection. For this reason, methods have been developed allowing to over-approximate zonotopes by a halfspace representation, which then allows us to intersect the reachable sets with guard sets. In order to continue the computation with zonotopes, the inverse problem of finding an over-approximating zonotope for a halfspace representation has been addressed.

An advantage of the proposed techniques is that they scale well with the dimension  $n$ . For the transformation of zonotopes to halfspace representation and vice versa, one can tune the algorithms according to a tradeoff between accuracy and computational time. This allows using the approach for systems of order greater than 10, if high accuracy is not required. As a by-product, effective order reduction techniques for zonotopes have been presented which are useful for other applications, such as collision detectors using zonotopes.

## Acknowledgements

The authors gratefully acknowledge partial financial support by the German Research Foundation (DFG) within the Transregional Collaborative Research Centre 28 *Cognitive Automobiles*. They also would like to thank the reviewers for the valuable comments.

## References

- [1] E.M. Clarke, O. Grumberg, D.A. Peled, *Model Checking*, MIT Press, 2000.
- [2] R. Rubinstein, *Simulation and the Monte Carlo Method*, Wiley & Sons, 2007.
- [3] A.A. Julius, G.E. Fainekos, M. Anand, I. Lee, G.J. Pappas, Robust test generation and coverage for hybrid systems, in: *Hybrid Systems: Computation and Control*, in: LNCS, vol. 4416, Springer, 2007, pp. 329–342.
- [4] A. Donzé, O. Maler, Systematic simulation using sensitivity analysis, in: *Hybrid Systems: Computation and Control*, in: LNCS, vol. 4416, Springer, 2007, pp. 174–189.
- [5] J. Kapinski, B.H. Krogh, O. Maler, O. Stursberg, On systematic simulation of open continuous systems, in: *Hybrid Systems: Computation and Control*, in: LNCS, vol. 2623, Springer, 2003, pp. 283–297.
- [6] S. Prajna, A. Jadbabaie, Safety verification of hybrid systems using barrier certificates, in: *Hybrid Systems: Computation and Control*, in: LNCS, vol. 2993, Springer, 2004, pp. 477–492.
- [7] C. Tomlin, I. Mitchell, A. Bayen, M. Oishi, Computational techniques for the verification and control of hybrid systems, in: *Proceedings of the IEEE*, vol. 91, 2003, pp. 986–1001.
- [8] O. Botchkarev, S. Tripakis, Verification of hybrid systems with linear differential inclusions using ellipsoidal approximations, in: *Hybrid Systems – Computation and Control*, in: LNCS, vol. 1790, Springer, 2000, pp. 73–88.
- [9] A. Chutinan, B.H. Krogh, Computational techniques for hybrid system verification, in: *IEEE Transactions on Automatic Control*, vol. 48, 2003, pp. 64–75.
- [10] O. Stursberg, B.H. Krogh, Efficient representation and computation of reachable sets for hybrid systems, in: *Hybrid Systems: Computation and Control*, in: LNCS, vol. 2623, Springer, 2003, pp. 482–497.
- [11] A. Girard, Reachability of uncertain linear systems using zonotopes, in: *Hybrid Systems: Computation and Control*, in: LNCS, vol. 3414, Springer, 2005, pp. 291–305.
- [12] A. Girard, C. Le Guernic, Zonotope/hyperplane intersection for hybrid systems reachability analysis, in: *Proc. of Hybrid Systems: Computation and Control*, in: LNCS, vol. 4981, Springer, 2008, pp. 215–228.
- [13] M. Althoff, O. Stursberg, M. Buss, Verification of uncertain embedded systems by computing reachable sets based on zonotopes, in: *Proc. of the 17th IFAC World Congress*, 2008, pp. 5125–5130.
- [14] T. Alamo, J.M. Bravo, E.F. Camacho, Guaranteed state estimation by zonotopes, in: *Proc. of the 42nd IEEE Conference on Decision and Control*, 2003, pp. 5831–5836.
- [15] L.J. Guibas, A. Nguyen, L. Zhang, Zonotopes as bounding volumes, in: *Proc. of the Symposium on Discrete Algorithms*, 2005, pp. 803–812.
- [16] G.M. Ziegler, *Lectures on polytopes*, in: *Graduate Texts in Mathematics*, Springer, 1995.
- [17] M. Althoff, O. Stursberg, M. Buss, Reachability analysis of linear systems with uncertain parameters and inputs, in: *Proc. of the 46th IEEE Conference on Decision and Control*, 2007, pp. 726–732.
- [18] M. Althoff, O. Stursberg, 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.
- [19] P. Gritzmann, B. Sturmfels, Minkowski addition of polytopes: Computational complexity and applications to Gröbner bases, *SIAM Journal on Discrete Mathematics* 6 (1993) 246–269.
- [20] D. Mortari, The  $n$ -dimensional cross product and its application to the matrix eigenanalysis, in: *Proc. of the AIAA/AAS Astrodynamics Conference*, 1996.
- [21] M.E. Muller, A note on a method for generating points uniformly on  $n$ -dimensional spheres, *Communications of the ACM* 2 (1959) 19–20.
- [22] M. Kvasnica, P. Grieder, M. Baotić, *Multi-Parametric Toolbox (MPT)*, 2004.
- [23] G. Barequet, S. Har-Peled, Efficiently approximating the minimum-volume bounding box of a point set in three dimensions, *Journal of Algorithms* 38 (1999) 91–109.
- [24] A. Fehnker, F. Ivančić, Benchmarks for hybrid systems verification, in: *Hybrid Systems: Computation and Control*, in: LNCS, vol. 2993, Springer, 2004, pp. 326–341.