Brief paper

A comparison of zonotope order reduction techniques☆

Xuejiao Yang, Joseph K. Scott *

Department of Chemical and Biomolecular Engineering, Clemson University, Clemson, SC, USA

ARTICLE INFO

ABSTRACT

This brief paper provides a comparison of methods for enclosing a given zonotope within another of lower complexity, commonly called *order reduction*. These techniques are essential for maintaining efficiency in recursive computations with zonotopes and are widely used in set-based estimation, hybrid systems verification, and fault detection. We first review existing methods and provide a new theoretical analysis of the method recently introduced by Scott et al. (2016). We then compare methods in terms of computational cost and overestimation error, and investigate the effects of zonotope dimension, initial order, and reduced order on these metrics. These results provide valuable guidance for the design of robust estimation and control algorithms that more effectively balance accuracy with computational cost.

1. Introduction

Since the seminal work of Kühn (1998), zonotopes have been widely adopted as an accurate and efficient way to model bounded uncertainties and noises in a variety of control applications, including reachability analysis (Althoff, Stursberg, & Buss, 2008a; Girard, 2005; Kühn, 1998), state estimation (Bravo, Alamo, & Camacho, 2006; Combastel, 2003, 2005, 2015; Scott, Marseglia, Raimondo, & Braatz, 2016), hybrid systems verification (Althoff, Stursberg, & Buss, 2008b, 2010; Girard & Le Guernic, 2008), robust control (Ocampo-Martinez, Guerra, Puig, & Quevedo, 2007; Raimondo, Marseglia, Braatz, & Scott, 2013), and fault detection (Ingimundarson, Bravo, Puig, Alamo, & Guerra, 2009; Raimondo, Marseglia, Braatz, & Scott, 2016; Scott, Findeisen, Braatz, & Raimondo, 2014; Tabatabaeipour, Odgaard, Bak, & Stoustrup, 2012). Zonotopes are significantly more flexible than parallelotopes and ellipsoids, while requiring much less computational effort than general convex polytopes (Scott et al., 2016). However, many operations on zonotopes yield results with higher complexity than their arguments (Kühn, 1998), which is a serious limitation, particularly for recursive algorithms. To address this, *order reduction* methods bound a given zonotope within another of lower complexity. These methods are essential for many control algorithms, and can significantly impact their efficiency and performance. For example, inaccurate reduction can lead to overly conservative set-based estimators, and

hence to conservative control actions or ineffective fault detection (Althoff et al., 2010; Scott et al., 2016).

Order reduction was first addressed in Kühn (1998) in the context of reachability analysis. The first general purpose method was proposed in Combastel (2003), followed shortly by a similar method in Girard (2005). These methods (Methods 1 & 2, resp.) are both very efficient. However, while Method 1 has been overwhelmingly used in the literature (Bravo et al., 2006; Combastel, 2015; Ocampo-Martinez et al., 2007; Tabatabaeipour et al., 2012), there are no available studies comparing their accuracy. A more sophisticated approach was proposed in Althoff et al. (2008b) (Method 3) and shown to be significantly more accurate than Method 2, but only for a limited set of tests with low-dimensional zonotopes ($n \leq 4$). Moreover, Method 1 was not compared. Unfortunately, Method 3 requires a combinatorial search that is problematic in high-dimensions (see Section 3.3). To address this, a fourth method was recently proposed in Scott et al. (2016) (Method 4) that follows the main insights of Method 3 but eliminates the combinatorial search using an iterative matrix factorization. It was claimed in Scott et al. (2016) that Method 4 matches the accuracy of Method 3 at significantly lower cost. However, because Method 4 was not the focus of that article, it was described only in the appendix, with no theoretical justification and no comparisons.

This brief paper makes two main contributions. First, Method 4 is presented in detail and its validity is established. Second, a comprehensive comparison of Methods 1–4 is presented considering both computational cost and overestimation error for a large test set. The effects of problem dimension, initial zonotope order, and reduced zonotope order are also investigated. The results provide valuable guidance for designing set-based estimation and control algorithms that more effectively balance accuracy with computational cost.

2. Preliminaries

A zonotope is a convex polytope that can be represented as a Minkowski sum of line segments, or equivalently as the image of a unit hypercube under an affine mapping (Kühn, 1998). Formally, $Z \subset \mathbb{R}^n$ is a zonotope iff

$$\exists (\mathbf{G}, \mathbf{c}) \in \mathbb{R}^{n \times n_g} \times \mathbb{R}^n : Z = \{\mathbf{G}\boldsymbol{\xi} + \mathbf{c} : \|\boldsymbol{\xi}\|_\infty \le 1\}. \quad (1)$$

The vector $\mathbf{c}$ is the *center*, the n_g columns of $\mathbf{G}$ are the *generators*, and (1) is called the *generator-representation* (G-rep) of Z. We use the shorthand $Z = \{\mathbf{G}, \mathbf{c}\} \subset \mathbb{R}^n$ to denote zonotopes throughout. Increasing n_g makes zonotopes more flexible, but also more cumbersome. The complexity of a zonotope is described by its *order*, defined as $o \equiv n_g/n$ (Girard, 2005). A first-order zonotope is a parallelotope if $\mathbf{G}$ is full rank and an interval if $\mathbf{G}$ is diagonal.

For $Z, Y \subset \mathbb{R}^n$ and $\mathbf{R} \in \mathbb{R}^{m \times n}$, define the linear mapping and Minkowsi sum, respectively, as

$$\mathbf{R}Z \equiv \{\mathbf{R}\mathbf{z} : \mathbf{z} \in Z\}, \quad Z + Y \equiv \{\mathbf{z} + \mathbf{y} : \mathbf{z} \in Z, \ \mathbf{y} \in Y\}.$$

When $Z = \{\mathbf{G}_z, \mathbf{c}_z\}$ and $Y = \{\mathbf{G}_y, \mathbf{c}_y\}$ are zonotopes, $\mathbf{R}Z$ and $Z + Y$ can be computed exactly as (Kühn, 1998):

$$\mathbf{R}Z = \{\mathbf{R}\mathbf{G}_z, \mathbf{R}\mathbf{c}_z\}, \quad Z + Y = \{[\mathbf{G}_z \ \mathbf{G}_y], \mathbf{c}_z + \mathbf{c}_y\}. \quad (2)$$

Clearly, this can be done efficiently and reliably, even in high dimensions, which is not the case for general convex polytopes (Scott et al., 2016). However, $\mathbf{R}Z$ and $Z + Y$ can be higher order than Z and Y, and this holds for other important operations as well, such as the convex hull in Girard (2005). This is a major drawback, particularly when such operations are applied recursively (e.g., Minkowski sums in state estimation with additive uncertainty Bravo et al., 2006; Kühn, 1998).

Given $Z = \{\mathbf{G}, \mathbf{c}\} \subset \mathbb{R}^n$, *order reduction* addresses this issue by finding a lower-order zonotope Z_R that contains Z. Ideally, Z_R has minimal overestimation, which can be assessed using the following volume and Hausdorff error metrics, where $v(Z)$ is the volume of Z, $r(Z) \equiv \max_{\mathbf{z} \in Z} \|\mathbf{z} - \mathbf{c}\|_2$ is the radius of Z, and d_H is the Hausdorff distance:

$$\Theta_V \equiv \frac{v(Z_R)^{\frac{1}{n}} - v(Z)^{\frac{1}{n}}}{v(Z)^{\frac{1}{n}}}, \qquad \Theta_H \equiv \frac{d_H(Z_R, Z)}{r(Z)}. \quad (3)$$

Since $Z \subset Z_R$, the Hausdorff distance is given by

$$d_H(Z_R, Z) = \max_{\mathbf{y} \in Z_R} \min_{\mathbf{z} \in Z} \|\mathbf{y} - \mathbf{z}\|_2.$$

Thus, Θ_H is the maximum distance that a point in Z_R can be from Z, relative to the radius of Z, while Θ_V measures the volume added by reduction relative to the volume of Z.

Lemma 1. *The volume of $Z = \{\mathbf{G}, \mathbf{c}\} \subset \mathbb{R}^n$ is given by* (Bravo et al., 2006):

$$v(Z) = 2^n \sum |\det[\mathbf{g}_{s_1} \cdots \mathbf{g}_{s_n}]|,$$

where the sum runs over all combinations of n indices s_i from the set $\{1, \ldots, n_g\}$ and $\mathbf{g}_i$ is the ith column of $\mathbf{G}$.

Lemma 2. *Let $Z = \{\mathbf{G}, \mathbf{c}\}$ and let $Z_R = \{\mathbf{G}_R, \mathbf{c}_R\}$ be a superset of Z with $\mathbf{c}_R = \mathbf{c}$. Then,*

$$r(Z) = \max_{\|\boldsymbol{\lambda}\|_2 = 1} \|\boldsymbol{\lambda}^{\mathsf{T}}\mathbf{G}\|_1, \quad (4)$$

$$d_H(Z_R, Z) = \max_{\|\boldsymbol{\lambda}\|_2 = 1} |\|\boldsymbol{\lambda}^{\mathsf{T}}\mathbf{G}_R\|_1 - \|\boldsymbol{\lambda}^{\mathsf{T}}\mathbf{G}\|_1|. \quad (5)$$

Proof. Define the support function $h_Z(\boldsymbol{\lambda}) \equiv \max_{\mathbf{z} \in Z} \boldsymbol{\lambda}^{\mathsf{T}}\mathbf{z}$. It follows from a standard duality argument that $d_H(Z_R, Z) =$ $\max_{\|\boldsymbol{\lambda}\|_2 = 1} |h_{Z_R}(\boldsymbol{\lambda}) - h_Z(\boldsymbol{\lambda})|$ (see Lemma 2 in Salinetti and Wets (1979)). This is equivalent to (5) because, by (1),

$$h_Z(\boldsymbol{\lambda}) = \max_{\|\boldsymbol{\xi}\|_\infty \le 1} \boldsymbol{\lambda}^{\mathsf{T}}(\mathbf{G}\boldsymbol{\xi} + \mathbf{c}) = \|\boldsymbol{\lambda}^{\mathsf{T}}\mathbf{G}\|_1 + \boldsymbol{\lambda}^{\mathsf{T}}\mathbf{c}.$$

Moreover, (4) follows from (5) because $r(Z)$ is the Hausdorff distance between Z and the singleton $\{\mathbf{c}\}$. □

Lemma 3. *Let $Z = \{\mathbf{G}, \mathbf{c}\} \subset \mathbb{R}^n$, denote the elements of $\mathbf{G}$ by g_{ij}, and define $\mathbf{d} \in \mathbb{R}^n$ elementwise by $d_i \equiv \sum_j |g_{ij}|$. The interval hull of Z is given in G-rep by $\{\text{diag}(\mathbf{d}), \mathbf{c}\}$* (Combastel, 2003).

3. Order reduction methods

Let $Z = \{\mathbf{G}, \mathbf{c}\}$ be a zonotope with initial order $o_o = n_g/n$. To reduce Z to order $o < n_g/n$, existing methods all take the following four steps. First, the columns of $\mathbf{G}$ are reordered. It follows from (1) that this does not affect the set Z. Second, the reordered $\mathbf{G}$ matrix is partitioned as $[\mathbf{K} \ \mathbf{L}]$ with $\mathbf{K} \in \mathbb{R}^{n \times n(o-1)}$ and $\mathbf{L} \in \mathbb{R}^{n \times (n_g - n(o-1))}$. From (2), this corresponds to splitting Z into a sum of two zonotopes,

$$Z = K + L, \quad K \equiv \{\mathbf{K}, \mathbf{c}\}, \quad L \equiv \{\mathbf{L}, \mathbf{0}\}.$$

Third, L is overapproximated by a first order zonotope $L_R \equiv \{\mathbf{L}_R, \mathbf{0}\}$ with $\mathbf{L}_R \in \mathbb{R}^{n \times n}$. Finally, Z is overapproximated by

$$Z_R \equiv K + L_R = \{[\mathbf{K} \ \mathbf{L}_R], \mathbf{c}\}. \quad (6)$$

It is readily verified that this eliminates all but $n \times o$ generators, as desired. Methods 1–4 are now described in detail.

3.1. Method 1

Method 1 (Combastel, 2003) chooses L_R as the interval hull of L, which is easily computed as in Lemma 3. Clearly, it is desirable to choose L so that the overestimation introduced by taking its interval hull is minimized. Method 1 aims to achieve this by choosing $\mathbf{L}$ as the $n_g - n(o - 1)$ shortest generators in $\mathbf{G}$. This is implemented in Algorithm 1, where the subroutine INTERVALHULL($\mathbf{L}$) returns the generator matrix of the interval hull of L. The complexity of Algorithm 1 is dominated by line 2, with $\mathcal{O}(nn_g)$ for computing two-norms and $\mathcal{O}(n_g \log n_g)$ for sorting, for a total of $\mathcal{O}(n_g(n + \log n_g))$, or $\mathcal{O}(no_0(n + \log(no_0)))$.

Algorithm 1 Reduces $\{\mathbf{G}, \mathbf{c}\}$ to order o using Method 1

1: **procedure** REDUCEORDER1($\mathbf{G}, n, n_g, o$)
2:     Reorder the columns of $\mathbf{G}$ by decreasing two-norm
3:     $\mathbf{K} \leftarrow \mathbf{G}_{1:n,1:n(o-1)}$
4:     $\mathbf{L} \leftarrow \mathbf{G}_{1:n,n(o-1)+1:n_g}$
5:     $\mathbf{L}_R \leftarrow$ INTERVALHULL($\mathbf{L}$)
6:     **return** $\begin{bmatrix} \mathbf{K} & \mathbf{L}_R \end{bmatrix}$
7: **end procedure**

3.2. Method 2

Method 2 (Girard, 2005) also chooses L_R as the interval hull of L, but aims to minimize the error by making L interval-shaped. Specifically, $\mathbf{L}$ is chosen as the $n_g - n(o - 1)$ generators $\mathbf{g}_j$ that have the smallest values of the score

$$\gamma_j \equiv \|\mathbf{g}_j\|_1 - \|\mathbf{g}_j\|_\infty,$$

which measures how nearly axis-aligned $\mathbf{g}_j$ is and is zero when $\mathbf{g}_j$ is a scaled unit vector. This is implemented exactly as in Algorithm 1 by simply replacing line 2. The complexity is again $\mathcal{O}(n_g(n + \log n_g)) = \mathcal{O}(no_0(n + \log(no_0)))$.