Hyperplanes Based Zonotopic Contractor

Rahma Bengamra
LAAS-CNRS, Université de Toulouse, UPS, 7 avenue du Colonel Roche, 31400 Toulouse, France

Soheib Fergani
LAAS-CNRS, Université de Toulouse, UPS, 7 avenue du Colonel Roche, 31400 Toulouse, France

Carine Jauberthie
LAAS-CNRS, Université de Toulouse, UPS, 7 avenue du Colonel Roche, 31400 Toulouse, France

Abstract

This article introduces a novel method for constructing a "zonotopic contractor" based on hyperplane properties. While zonotopes, a special class of polytopes, offer computational advantages due to their symmetric matrix representation, they are not closed under intersection, often necessitating over-approximations that lead to conservatism in practical applications. The proposed contractor addresses this issue by providing a more efficient solution for approximating zonotope intersections, reducing conservatism. This method generates a new zonotope that closely covers the intersection of two zonotopes. The reliability and effectiveness of the proposed approach are demonstrated through simulations on a bicycle model, showing potential benefits in safety-critical applications like autonomous driving, where precise uncertainty management is crucial for decision-making and control.

2012 ACM Subject Classification Mathematics of computing → Continuous mathematics

Keywords and phrases Contractors, Zonotopes, Hyperplanes

Digital Object Identifier 10.4230/OASIcs.DX.2024.26

Category Short Paper

1 Introduction

With the increasing complexity of modern automated systems, precise state estimation has become essential for effective monitoring and control. Accurate state knowledge is crucial for ensuring the smooth operation of systems, such as robotics, autonomous vehicles, and aerospace platforms, where real-time decision-making is required. This precision is achieved by processing sensor data, which is often noisy or incomplete, through advanced mathematical tools known as filters.

Filters are designed to estimate key system variables like position and velocity, even in the presence of uncertainty. Depending on the nature of the data and assumptions about uncertainty, different types of filters are employed. For instance, probabilistic filters, such as the Kalman filter [6], use probability distributions to model uncertainty and are particularly effective in linear systems subjected to Gaussian noise. However, in many cases, statistical assumptions do not hold or are unavailable, necessitating alternative methods. In such scenarios, set-membership methods are often preferred, representing uncertainty through compact sets without relying on probabilistic models. These sets can be represented by various geometric shapes in the state space, each with distinct strengths and limitations:

- Ellipsoids [13] [3], which provide a compact and well-suited representation of sets but are not closed under Minkowski sum, limiting their use in some applications.
- Polytopes [14], which offer flexibility and accurate representations of uncertainty, though they are computationally intensive due to their complex vertex and facet representation.
- Intervals [12] [10], which simplify calculations by bounding each variable independently, but are highly conservative and may not capture the true uncertainty effectively.

- Zonotopes [7] [8] [1] [9], a special class of polytopes, strike a balance between simplicity and computational efficiency. They can be represented compactly using matrices, making operations faster and less conservative than intervals.

Despite their advantages, zonotopes face a significant limitation: they are not closed under intersection. This means that the intersection of two zonotopes does not necessarily yield another zonotope, complicating the representation of uncertainty in systems requiring intersection operations. As a result, overapproximation is often used, which introduces conservatism and reduces the precision of state estimation.

To overcome this challenge, we propose a novel "zonotopic contractor" that leverages hyperplane properties to approximate the intersection of two zonotopes while minimizing conservatism. The proposed contractor generates a new zonotope that closely covers the intersection, significantly improving the accuracy of set-based methods in systems where zonotope intersections are required. This method is designed to maintain the computational simplicity of zonotopes while addressing their limitations, offering a more efficient solution for state estimation in complex systems. We validate the performance of this zonotopic contractor through simulations using a bicycle model, demonstrating its effectiveness in reducing conservatism and enhancing reliability.

2 Preliminaries

▶ **Definition 1** (Zonotopes). *A zonotope is a type of geometric shape that is convex and symmetric. To create it, we start with a center point $c \in \mathbb{R}^n$ and then extend one or more vectors outward from this center in different directions. These vectors, which we call generating vectors and denote as $g_i \in \mathbb{R}^n : i \in 1, ..., d$, where n is the dimention and d represents the number of vectors, determine the shape of the zonotope. The zonotope's order $\rho = d/n$, represents a dimensionless measure of the representation size. i.e., This provides a relative measure of the complexity or dimension of the space covered by the zonotope. The higher this order, the larger and more complex the zonotope becomes.*

For each generator vector, we also include another vector pointing in the opposite direction. This pairing creates segments, i.e., forming a set of points lying within the interval I defined by:

$$I_i = c_i + [-1, 1]g_i = \{c_i + \alpha g_i : \alpha \in [-1, 1]\}, \quad \text{for } i = 1, \ldots, d. \tag{1}$$

After obtaining the intervals, we compute their Minkowski sum. Which is defined by :

$$I_1 + I_2 := \{x + y : x \in I_1, y \in I_2\} \tag{2}$$

So indeed, the zonotope is the Minkowski sum of d segments in $\mathbb{R}^n$.

Now that we have defined zonotopes, let's imagine a hypercube formed by the intervals I_i where i ranges from 1 to d. According to the definition we provided for a zonotope, we can construct a zonotope from these intervals by applying the Minkowski sum as :

$$Z = \sum_{i=1}^{d} I_i = \sum_{i=1}^{d} c_i + [-1, 1]g_i = \sum_{i=1}^{d} c_i + \sum_{i=1}^{d} [-1, 1]g_i = c + G * [-1, 1]^d \tag{3}$$

$$Z = \{x \in \mathbb{R}^n : x = c + G * \alpha^d : \alpha \in [-1, 1]\} \tag{4}$$

Therefore, the zonotope Z is defined by the new center c, which is the sum of the centers c_i of the intervals I_i and the generatrice matrix G such that $G = [g_1, ..., g_d]$.

Hence, another definition of a zonotope is that it is an affine transformation of the interval box of dimension $d \geq n$.

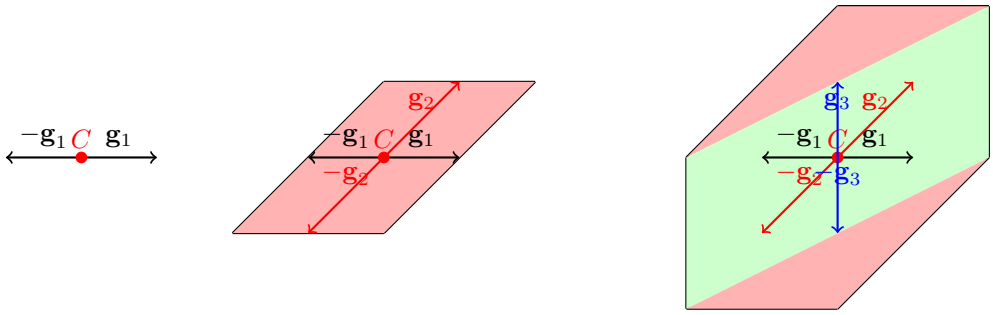

Figure 1 Illustration of the steps involved in constructing a zonotope from three vectors.

We have a matrix G composed of the following vectors: $g_1 = \begin{bmatrix}1\\0\end{bmatrix}$, $g_2 = \begin{bmatrix}1\\1\end{bmatrix}$, $g_3 = \begin{bmatrix}0\\1\end{bmatrix}$ *Each of these vectors represents a direction in the plane.*

The zonotope we are constructing is centered at the origin, meaning that $c = \begin{bmatrix}0\\0\end{bmatrix}$. *To construct this zonotope, we sequentially add each vector to the set of points already drawn. Let's first imagine drawing a segment using vector g_1. Then, we take this segment and, from each of its points, we add vector g_2 to create a new shape. Finally, we repeat this process with vector g_3. At each step, we progressively expand the shape by adding new segments, always starting from the center. The result is a zonotope, which represents the convex hull of the points generated by this addition process. The illustration in the figure 1 visually demonstrates how this construction evolves as we add generators to the zonotope. Each new vector contributes to extending the shape's boundaries in different directions, resulting in a more complex geometric figure. The final zonotope is a centrally symmetric, convex polytope whose sides are parallel to the generating vectors. The number of faces and the overall complexity of the zonotope increase with each added generator, thus demonstrating the flexibility and utility of zonotopes in representing multidimensional spaces in a compact and structured manner.*

Zonotopes have favorable properties as they can be represented compactly, and they are closed under the Minkowski sum and linear mapping, For clarity, we will denote the real interval $[-1, 1]$ as $\mathcal{I}$.

Composition of Zonotopes.

Let Z_1 and Z_2 be two zonotopes defined by their centers c_1 and c_2, as well as their generator matrices $G_1 = \{g_{1,1}, g_{1,2}, \ldots, g_{1,d_1}\}$ and $G_2 = \{g_{2,1}, g_{2,2}, \ldots, g_{2,d_2}\}$, respectively. Then, the Minkowski sum of these two zonotopes is itself a zonotope and can be expressed as:

$$c_1 + G_1\,\mathcal{I}^{d_1} + c_2 + G_2\,\mathcal{I}^{d_2} = c + [G_1\ G_2]\,\mathcal{I}^{d}, \tag{5}$$

where $c = c_1 + c_2$, $d = d_1 + d_2$ and $[G_1\ G_2]$ represents the concatenation of the generators matrix of the two zonotopes. This illustrates that a zonotope is closed under the Minkowski sum.

Linear Multiplication of Zonotopes.

Let $L \in \mathbb{R}^{n\times m}$ be a matrix and $G = \{g_1, g_2, \ldots, g_d\} \subseteq \mathbb{R}^m$. Then, the linear multiplication of L by $G\,\mathcal{I}^d$ is also a zonotope and is expressed as:

$$L * G\,\mathcal{I}^d = (L * G)\,\mathcal{I}^d, \tag{6}$$

Symmetry of Generator Vectors.

Let Z be a zonotope defined by its generators $\{g_1, g_2, \ldots, g_d\}$. The zonotope remains the same if a generator vector g_i is replaced by $-g_i$, that is:

$$Z = c + \sum_{i=1}^{m} \mathcal{I} g_i \text{ is equivalent to } Z = c + \sum_{i=1}^{m} (-\mathcal{I}) g_i.$$

The central symmetry of a zonotope means that if a generator g_i is replaced by $-g_i$, the zonotope remains the same because every point of the zonotope can still be expressed as a linear combination of the modified generators.

Invariance under Permutation of Generator Vectors.

Let Z be a zonotope defined by its generators $\{g_1, g_2, \ldots, g_d\}$. Permuting the columns of the generator matrix does not change the zonotope, that is:

$$Z = [g_1\ g_2\ \ldots\ g_d]\,\mathcal{I}^d \text{ is equivalent to } Z = [g_2\ g_1\ \ldots\ g_d]\,\mathcal{I}^d.$$

Zonotopes are defined by the set of their generators and are independent of the order in which these generators are listed. Permuting the columns of the generator matrix does not affect the zonotope because it does not change the set of possible linear combinations of the generators.

▶ **Definition 2** (Hyperplanes). *A hyperplane H is the set of points $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ that satisfy the equation:*

$$H = \{\mathbf{x} : \mathbf{a}^{\mathbf{T}} \cdot \mathbf{x} = \mathbf{b}\} \tag{7}$$

where:

- *$\mathbf{a} = (a_1, a_2, \ldots, a_n)$ is the normal vector, perpendicular to the hyperplane, and $\mathbf{a}^{\mathbf{T}}$ is its transpose.*
- *b is a scalar that determines the position of the hyperplane relative to the origin.*

In an n-dimensional space, a hyperplane is a subspace of dimension $n-1$. It generalizes the concepts of points, lines, and planes in higher dimensions.

▶ **Theorem 3** (Intersection of Convex Sets). *Let A and B be convex sets in a vector space. Then, their intersection $A \cap B$ is also a convex set.*

▶ **Theorem 4** (Hahn-Banach Separation Theorem). *Let C be a convex set in a finite-dimensional vector space. For any point x such that $x \notin C$, there exists a hyperplane that strictly separates x from C.*

3 Hyperplanes based zonotopic Contractor

The main advantage of contractors [5], [2], [11] is their ability to reduce the size of a box by narrowing the interval to retain only the desired set of data. The key idea is to refine the box while satisfying two important conditions:

1. The reduced box $[z]_0$ is always a subset of the original box $[z]$, meaning that no unnecessary data is included.
2. The intersection of the reduced box with the solution set S remains unchanged, ensuring that no potential solutions are lost.

In this work, we adopt a different approach aimed at reducing conservatism while preserving the accuracy of all potential solutions. Specifically, we leverage the use of zonotopes, which offer a more computationally efficient and less conservative alternative to intervals. By utilizing zonotopes, we are able to achieve a more precise representation of uncertainty, balancing computational manageability with improved solution integrity.

By definition, a zonotope is a convex set, and according to the theorem on the intersection of convex sets (Theorem 3) the intersection of two convex sets is also convex. However, this intersection does not necessarily result in a zonotope, which presents a challenge for efficient computation. Thus, a crucial step in our method is to approximate this convex intersection with a zonotope, simplifying subsequent calculations and manipulations.

To achieve this, hyperplanes are employed to enclose the intersection and construct a zonotopic approximation. The justification for using hyperplanes stems from the Hahn-Banach theorem (Theorem 4), which guarantees that two convex sets can always be separated by a hyperplane. This allows us to enclose any convex intersection with hyperplanes, creating a zonotopic envelope around the intersection. The process is carried out in three key steps:

1. **Selection of Non-Collinear Hyperplanes:** In this first step, the objective is to select two non-collinear hyperplanes, denoted as H_1 and H_2, which means that they are neither parallel nor aligned. Mathematically, this can be expressed by ensuring that the normal vectors a_1 and a_2 of H_1 and H_2 respectively are linearly independent, i.e., the rank of the matrix formed by a_1 and a_2 is equal to n:

$$\text{rank}([a_1, a_2]) = n \tag{8}$$

where $[a_1, a_2]$ is the concatenation of the normal vectors a_1 and a_2, and n is the dimension of the space in which the intersection of the zonotopes is defined.

These hyperplanes are specifically chosen to pass through facets of the intersection of two zonotopes, $Z_1 \cap Z_2$. Let F_i denote a facet of the convex hull of the intersection, which is a $(n-1)$-dimensional face. Each hyperplane H_i (for $i = 1, 2$) is defined such that it satisfies:

$$H_i : a_i^T x = b_i \quad \forall x \in F_i \tag{9}$$

2. **Construction of Parallel Hyperplanes:** Once the non-collinear hyperplanes H_i are defined, the next step involves constructing parallel hyperplanes to these. Each hyperplane is adjusted in space to be at a sufficient distance to encompass the entire convex intersection. This distance is determined by identifying the farthest vertex $\mathbf{v} \in V$ from the hyperplane H_i, where V is the set of vertices of the convex hull of the intersection. The distance from a vertex $\mathbf{v}$ to the hyperplane H_i is given by the formula:

$$d(\mathbf{v}, H_i) = \frac{|a_i^T \mathbf{v} - b_i|}{\|a_i\|} \tag{10}$$

The vertex $\mathbf{v}_{\max}$ corresponding to the maximum distance is selected:

$$\mathbf{v}_{\max} = \arg\max_{\mathbf{v} \in V} d(\mathbf{v}, H_i) \tag{11}$$

Once $\mathbf{v}_{\max}$ is identified, a parallel hyperplane H_i^{parallel} is constructed to pass through this vertex. The equation of the parallel hyperplane is defined as:

$$H_i^{\text{parallel}} : a_i^T \mathbf{x} = b_i^{\text{parallel}} \tag{12}$$

where $b_i^{\text{parallel}} = a_i^T \mathbf{v}_{\max}$

By selecting hyperplanes H_i that pass through all points x on the facet F_i, we minimize conservatism in the zonotope approximation of the intersection. Additionally, by taking the farthest vertex for the parallel hyperplane, we guarantee that the complete intersection is enclosed without loss.

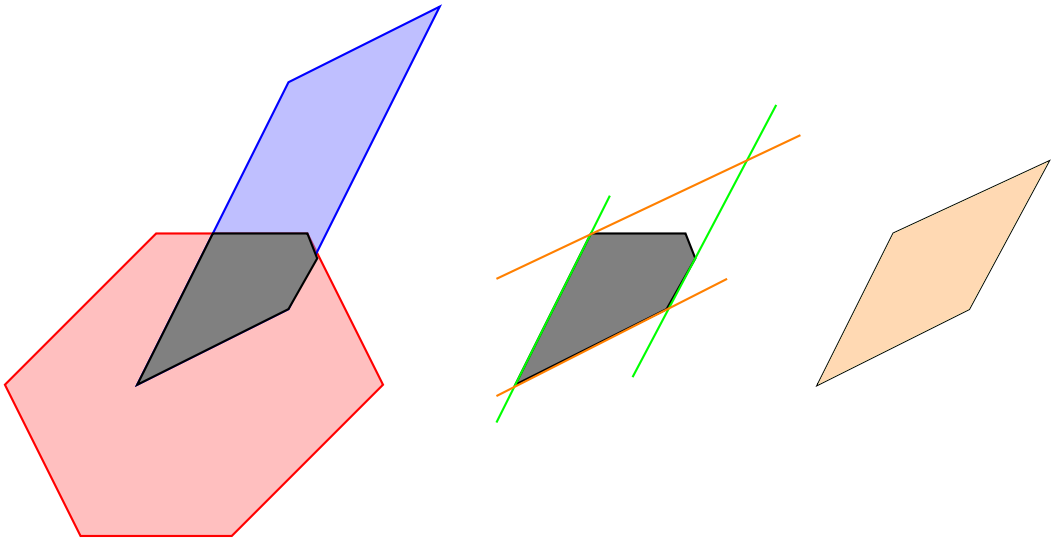

Figure 2 Illustration of the Zonotopic Contraction Method Using Hyperplanes.

This figure illustrates the key steps of the proposed zonotopic contraction method. The initial zonotopes are represented by the red and blue shapes, each generated by different sets of vectors. The intersection of these two zonotopes forms a complex convex region (in gray). To approximate this intersection while reducing conservatism, we use hyperplanes (depicted in green and orange) that enclose the intersection and define a new zonotopic boundary. The resulting zonotope, shown in orange, effectively captures the intersection of the original zonotopes, ensuring that the approximation remains both accurate and computationally manageable. This process is iteratively refined to minimize the volume of the final zonotope while preserving the integrity of all potential solutions within the intersection.

3. **Volume Comparison:** Once the hyperplanes are defined and placed, we proceed with an iterative volume comparison. The goal is to adjust the positions of the hyperplanes by making them pass through other edges of the intersection, if any remain, and to check all possibilities in order to find the optimal combination. This approach allows us to minimize the volume of the final zonotope while ensuring that the intersection is properly enclosed.

4 Model and simulation results

To test the proposed contractor, the commonly known "Linear Bicycle Model" [10] is used.

The considered continuous time model and the linearization assumptions are presented as follows:

1. **State Variables:**
   - $\beta(t)$: the sideslip angle, representing the difference between the direction of the vehicle's movement and its orientation.
   - $\psi(t)$: the yaw angle, corresponding to the vehicle's rotation around its vertical axis.

2. **Forces:**
   - $F_{tyf}(t)$: lateral forces on the front tires.
   - $F_{tyr}(t)$: lateral forces on the rear tires.
   - $F_{txf}(t)$: longitudinal forces on the front tires.
   - $\Delta F_{txr}(t)$: differential braking force on the rear tires, calculated from the braking torques T_{br}.
3. **Linearization Assumptions:**
   - The sideslip angles $|\beta|$ are small, less than 7 degrees.
   - The longitudinal slip ratio is small, less than 0.1.
   - The steering angles δ are small, allowing the approximation $\cos(\delta) \approx 1$.

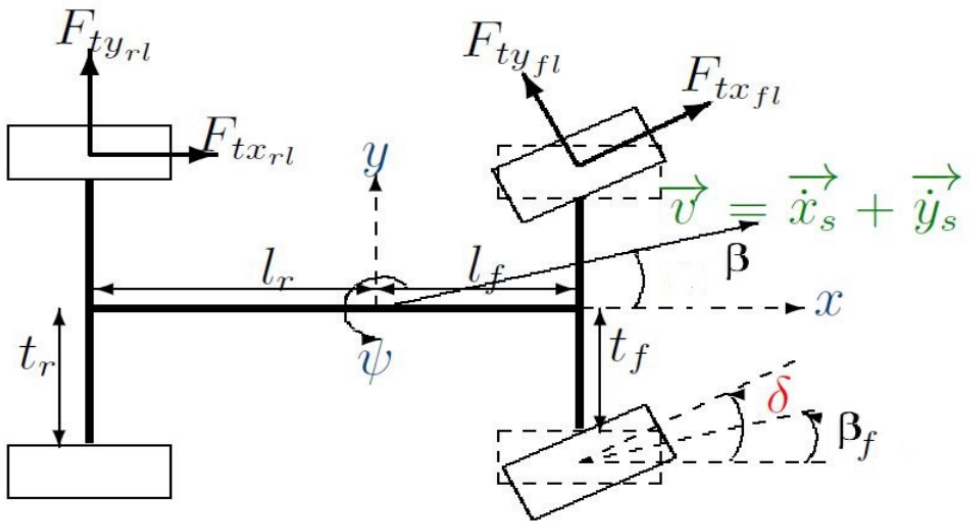

Figure 3 Representation of the bicycle model illustrating the car's lateral behavior [10].

The linearized model yields the following equations for lateral tire forces:

$$F_{tyf}(t) = C_f \beta_f(t) \tag{13}$$

$$F_{tyr}(t) = C_r \beta_r(t) \tag{14}$$

where $\beta_f(t)$ and $\beta_r(t)$ are the front and rear sideslip angles, given by:

$$\beta_f(t) = \delta(t) - \beta(t) - \frac{l_f \dot{\psi}(t)}{v}, \tag{15}$$

$$\beta_r(t) = \beta(t) + \frac{l_r \dot{\psi}(t)}{v}. \tag{16}$$

This leads to the following state-space representation:

$$\begin{bmatrix} \dot{\beta}(t) \\ \dot{\psi}(t) \end{bmatrix} = \begin{bmatrix} -\frac{C_f + C_r}{mv} & 1 + \mu - \frac{l_r C_r + l_f C_f}{mv^2} \\ -\frac{l_r C_r + l_f C_f}{I_z} & -\frac{l_f^2 C_f + l_r^2 C_r}{I_z v} \end{bmatrix} \begin{bmatrix} \beta(t) \\ \psi(t) \end{bmatrix} + \begin{bmatrix} \frac{C_f}{m} & 0 & 0 & 0 \\ \frac{l_f C_f}{I_z} & \frac{1}{I_z} & \frac{S_r R t_r}{2I_z} & -\frac{S_r R t_r}{2I_z} \end{bmatrix} \begin{bmatrix} \delta \\ M_{dz} \\ T_{brl} \\ T_{brr} \end{bmatrix} \tag{17}$$

The parameters used in the previous formulas are defined as follows:

- m: Vehicle mass, which is 1535 kg.
- I_z: Vehicle yaw inertia, equal to 2149 kg $\cdot$ m^2.
- C_f: Lateral stiffness of the front tires, given as 20000 N/degree.
- C_r: Lateral stiffness of the rear tires, also 20000 N/degree.
- S_r: Longitudinal stiffness of the rear tires, valued at 12720 N.
- l_f: Distance from the center of gravity (COG) to the front axle, which is 1.4 meters.
- l_r: Distance from the center of gravity to the rear axle, set at 1 meter.
- t_r: Length of the rear axle, specified as 1.4 meters.
- R: Tire radius, with a value of 0.3 meters.
- μ: Tire-to-road friction coefficient, ranging from 0.4 to 1.
- v: Vehicle speed, ranging from 50 to 130 km/h.
- δ: Steering angle.
- M_{dz}: Yaw moment disturbance.
- T_{brj}: Braking torques at the left and right rear tires (where $j = \{l, r\}$).

These parameters are crucial for modeling the vehicle's lateral dynamics. These values were obtained through an identification process on the Renault Mégane Coupé (see [4]).

Note that the bicycle model has been discretized to be used for the simulation. The filter undergoes the following steps:

1. Simulation Step :
   The equations for the actual system state and measured output are given by:

$$\begin{cases} x_{\text{real}}(k) = Ax_{\text{real}}(k-1) + Bu(k) \\ [y_{\text{measured}}]_z = Cx_{\text{real}}(k) + [v]_z \end{cases} \tag{18}$$

   where:
   - $x_{\text{real}}(k)$ is the time derivative of the actual state vector at time step k.
   - A and B are system matrices.
   - $u(k)$ represents the input vector at time step k.
   - C is the output matrix (identity matrix).
   - $[v]_z$ denotes the bounded noise affecting the measurements.

   Note the that the real state and the measurement are transformed into zonotopes to cope with the filter requirements.
2. Propagation Step:
   The equations for predicting and propagationg the next state and output are:

$$\begin{cases} [\hat{x}(k)]_z = A[\hat{x}(k-1)]_z + Bu(k) + [w]_z \\ [\hat{y}(k)]_z = C[\hat{x}(k)]_z + [v]_z \end{cases} \tag{19}$$

   where:
   - $[\hat{x}(k)]_z$ is the predicted zonotope state at time step k.
   - $[\hat{y}(k)]_z$ is the predicted zonotope measure at time k.
   - $[w]_z$ denotes the bounded noise affecting the system's dynamics.
3. Correction Step : The correction of the estimated state $\hat{x}(k)$ based on the measured output is performed using:

$$\begin{cases} intersect = [\hat{y}(k)]_z \cap [y_{\text{measured}}]_z \\ \hat{x}^r(k) = [intersect]_z \end{cases} \tag{20}$$

   where $[intersect]_z$ is an approximate zonotope representation that is close to the true intersection but simplifies the computation.

Here, $\hat{y}(k)$ is directly equivalent to $\hat{x}(k)$ due to the identity matrix C. Thus, the correction step adjusts the estimated state $\hat{x}^r(k)$ to better match the measured output y_{measured}, considering any discrepancies or noise in the measurements.

The system is represented by the following state-space matrices:

$$A = \begin{bmatrix} -\frac{C_f+C_r}{mv} & 1-\frac{l_rC_r+l_fC_f}{mv^2} \\ -\frac{l_rC_r+l_fC_f}{I_z} & -\frac{l_f^2C_f+l_r^2C_r}{I_zv} \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{C_f}{mv} & 0 & 0 & 0 \\ \frac{l_fC_f}{I_z} & \frac{1000}{I_z} & \frac{Rt_r}{2I_z} & \frac{-Rt_r}{2I_z} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The initial conditions are set as follows:

- The initial state vector x_0 is $[0, 0]$.
- The initial zonotope $[x_0]_z$ is initialized with a center of $[0, 0]$ and generators $[0, 0.5; 0.5, 0]$.

Input values u_k, which include steering angles, yaw moment disturbances, and braking torques, are loaded for simulation.

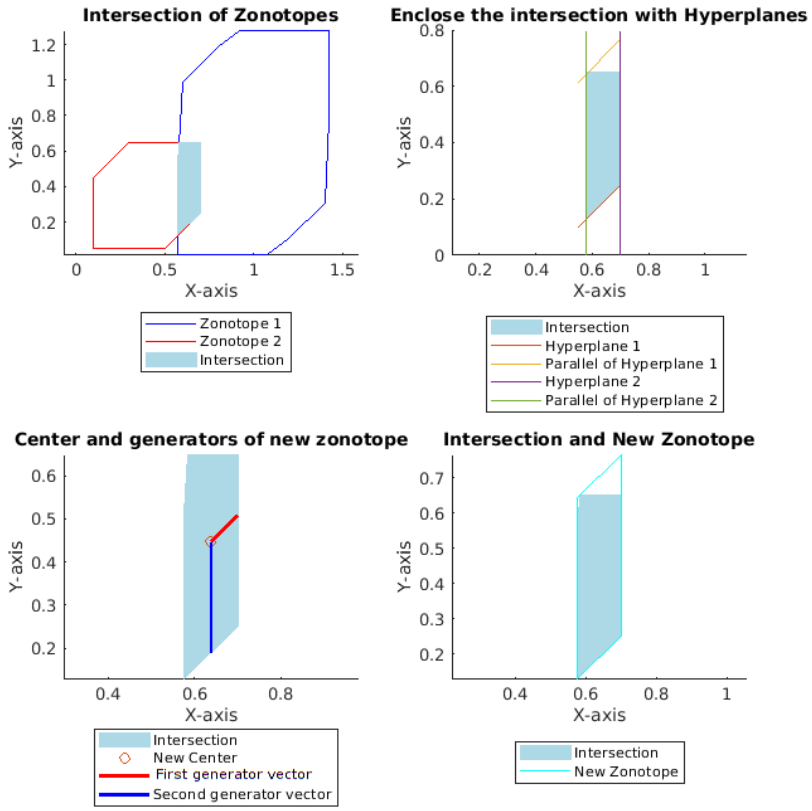

Figure 4 Example execution of the zonotope contractor method with $k = 30$ and an error rate of 0.1131.

The error rate is defined as the relative difference between the volume of the intersection of the predicted and actual zonotopes and the volume of the new zonotope. It is calculated as follows:

$$\text{error\_rate} = 1 - \frac{\text{volume(intersect)}}{\text{volume(new\_zonotope)}} \tag{21}$$

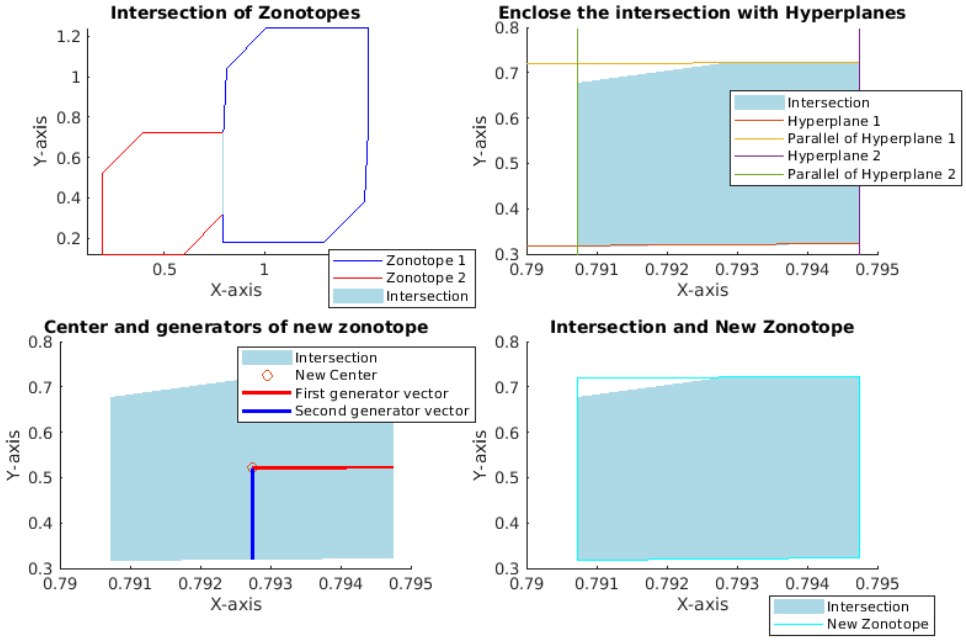

Figure 5 Example execution of the zonotope contractor method with $k = 600$ and an error rate of 0.03035.

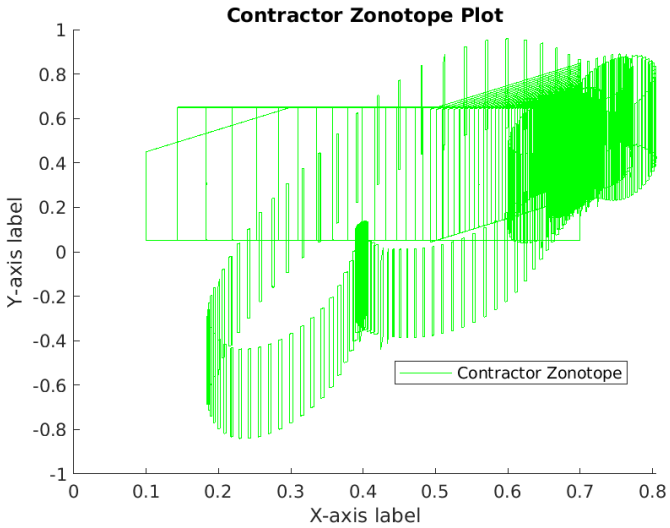

Figure 6 Contractor Zonotope over Iterations. This plot shows the evolution of the contractor zonotope as it becomes tighter with each iteration, illustrating how the method increasingly refines the estimated state.

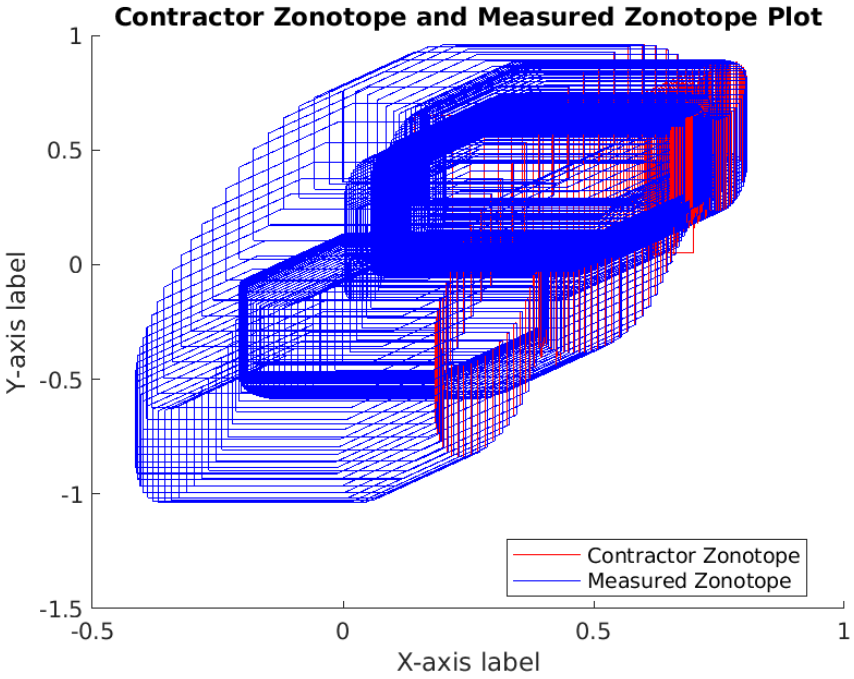

Figure 7 Comparison of the contractor zonotope estimates (in red) with the measured zonotopes (in blue) over $N = 864$ iterations. The contractor zonotope progressively tightens around the true state, demonstrating improved accuracy in state estimation as iterations advance.

The results demonstrate that the hyperplane based zonotopic contractor provides increasingly tighter estimates of the intersection as the iteration progresses from $k = 1$ to $k = N = 864$. This tightening is reflected in the improved accuracy of the state estimation, as the contractor becomes more refined with each iteration, effectively narrowing the uncertainty in the intersection with the measured zonotope.

Moreover, the error rate remains relatively low throughout the iterations. This low error rate indicates that the proposed contractor effectively approximates the true intersection, capturing the system's behavior with high fidelity. The tight bounding of the intersection suggests that it is a reliable method for state estimation in the presence of uncertainties and noise, making it a robust tool for systems modeled using zonotopes

5 Conclusion and future works

This work presents a novel method for zonotopic contraction using hyperplanes, with the goal of reducing conservatism while preserving the zonotope's structural integrity. The key advantage of this approach is its ability to maintain a zonotopic structure that closely approximates the convex intersection of the original zonotopes. By leveraging the properties of hyperplanes, we accurately approximated this intersection, ensuring that the resulting zonotope remains a manageable convex set. Through the use of non-collinear and parallel hyperplanes, we systematically refined the zonotope boundaries, minimizing volume while retaining the critical characteristics of the intersection. Although we cannot prove the optimality of this approach in all scenarios, we provide strong guarantees that it reduces conservatism and captures all potential solutions within the intersection. This method's

increased precision is especially relevant in safety-critical systems such as autonomous driving and robotics. For example, in autonomous vehicle navigation, the accurate capture of the intersection of state spaces represented by zonotopes leads to better trajectory planning and improved obstacle avoidance, thereby reducing collision risks and ensuring safer navigation in uncertain environments. Furthermore, in control and fault detection systems, this approach enhances state estimation accuracy and fault diagnosis. In fields like industrial robotics and aerospace, precise zonotopic contraction improves uncertainty management, optimizing system performance and reducing failure risks.

Another point concerns the computational complexity, particularly in Step 3 (Volume Comparison). Currently, the recursive volume comparison remains manageable with the use of two hyperplanes. However, as we extend the method to incorporate more hyperplanes in higher-dimensional spaces, we anticipate an increase in computational complexity. This potential increase has been taken into account, and future works will focus on optimizing the algorithm to mitigate this impact. Our current approach is designed to retain the inherent simplicity of zonotopes while exploring more efficient strategies for volume minimization. At this stage, the computational overhead remains low with the two-hyperplane scenario. Nevertheless, as we expand to higher dimensions and incorporate additional hyperplanes, we are actively pursuing optimization strategies for the recursive process. Our goal is to strike a balance between computational efficiency and accuracy, enabling the method to scale effectively with increasing dimensionality.

References

1 Matthias Althoff. On computing the minkowski difference of zonotopes. *arXiv preprint arXiv:1512.02794*, 2015. arXiv:1512.02794.

2 Gilles Chabert and Luc Jaulin. Contractor programming. *Artificial Intelligence*, 173(11):1079–1100, 2009. doi:10.1016/J.ARTINT.2009.03.002.

3 Felix L. Chernousko. Ellipsoidal state estimation for dynamical systems. *Nonlinear Analysis: Theory, Methods & Applications*, 63(5-7):872–879, 2005.

4 Soheib Fergani. *Robust multivariable control for vehicle dynamics*. PhD thesis, Université de Grenoble, 2014.

5 Luc Jaulin, Michel Kieffer, Olivier Didrit, Eric Walter, Luc Jaulin, Michel Kieffer, Olivier Didrit, and Éric Walter. *Interval analysis*. Springer, 2001.

6 Rudolf Emil Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME—Journal of Basic Engineering*, 82:35–45, 1960.

7 Wolfgang Kühn. Rigorously computed orbits of dynamical systems without the wrapping effect. *Computing*, 61(1):47–67, 1998. doi:10.1007/BF02684450.

8 Wolfgang Kühn. Zonotope dynamics in numerical quality control. In H.-C. Hege and K. Polthier, editors, *Mathematical Visualization*, pages 125–134. Springer, 1998. doi:10.1007/978-3-662-03567-2_10.

9 Jinsun Liu, Yifei Simon Shao, Lucas Lymburner, Hansen Qin, Vishrut Kaushik, Lena Trang, Ruiyang Wang, Vladimir Ivanovic, H. Eric Tseng, and Ram Vasudevan. Refine: Reachability-based trajectory design using robust feedback linearization and zonotopes. *IEEE Transactions on Robotics*, 40:2060–2080, 2024. doi:10.1109/TRO.2024.3366819.

10 Quoc Hung Lu. *Filtrage à incertitudes stochastiques et bornées: application au diagnostic actif en automobile*. PhD thesis, Université Paul Sabatier-Toulouse III, 2022.

11 Moussa Maiga, Nacim Ramdani, Louise Travé-Massuyès, and Christophe Combastel. A comprehensive method for reachability analysis of uncertain nonlinear hybrid systems. *IEEE Transactions on Automatic Control*, 61(9):2341–2356, 2015. doi:10.1109/TAC.2015.2491740.

12 Tuan Anh Tran. *Cadre unifié pour la modélisation des incertitudes statistiques et bornées: application à la détection et isolation de défauts dans les systèmes dynamiques incertains par estimation*. PhD thesis, Université Paul Sabatier-Toulouse III, 2017.

13 Bo Zhou, Jianda Han, and Guangjun Liu. A ud factorization-based nonlinear adaptive set-membership filter for ellipsoidal estimation. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, 18(16):1513–1531, 2008.

14 Günter M Ziegler. Graduate texts in mathematics, 1995.