Ellipsotopes: Combining Ellipsoids and Zonotopes for Reachability Analysis and Fault Detection

Shreyas Kousik\textsuperscript{1}, Adam Dai\textsuperscript{2} and Grace X. Gao\textsuperscript{1}

Abstract—Ellipsoids are a common representation for reachability analysis because they are closed under affine maps and allow conservative approximation of Minkowski sums; this enables one to incorporate uncertainty and linearization error in a dynamical system by expanding the size of the reachable set. Zonotopes, a type of symmetric, convex polytope, are similarly frequently used due to efficient numerical implementation of affine maps and exact Minkowski sums. Both of these representations also enable efficient, convex collision detection for fault detection or formal verification tasks, wherein one checks if the reachable set of a system collides (i.e., intersects) with an unsafe set. However, both representations often result in conservative representations for reachable sets of arbitrary systems, and neither is closed under intersection. Recently, constrained zonotopes and constrained polynomial zonotopes have been shown to overcome some of these conservatism challenges, and are closed under intersection. However, constrained zonotopes cannot represent shapes with smooth boundaries such as ellipsoids, and constrained polynomial zonotopes can require solving a non-convex program for collision checking (i.e., fault detection). This paper introduces ellipsotopes, a set representation that is closed under affine maps, Minkowski sums, and intersections. Ellipsotopes combine the advantages of ellipsoids and zonotopes, and enable convex collision checking at the expense of more conservative reachable sets than constrained polynomial zonotopes. The utility of this representation is demonstrated on several examples.

I. INTRODUCTION

In the controls, robotics, and navigation communities, it is often critical to place strict guarantees on the behavior of a dynamical system. Example applications of such guarantees include collision avoidance\textsuperscript{1}–\textsuperscript{4}, fault detection\textsuperscript{5},\textsuperscript{6}, and control invariance\textsuperscript{7},\textsuperscript{8}. A common strategy for enforcing such guarantees, especially for uncertain dynamical systems, is to compute the system’s reachable set of states, then guarantee that this set lies within certain bounds (e.g., for fault detection) or obeys non-intersection constraints (e.g., for collision avoidance).

Directly representing a continuum of possible system trajectories numerically is typically intractable, given that these trajectories are solutions to a nonlinear differential or difference equation. Instead, a variety of set representations have been introduced to enable approximating reachable sets. Two of the most common and well-studied representations are ellipsoids\textsuperscript{9},\textsuperscript{10} and zonotopes\textsuperscript{2},\textsuperscript{11},\textsuperscript{12}. In this work, an ellipsoid is best understood as an affine transformation of a unit 2-norm ball in an arbitrary-dimensional Euclidean space. A zonotope can similarly be understood as

\begin{equation}
\mathbb{E}(p,\mathbf{C}) = \{ \mathbf{x} \mid \mathbf{x} = \mathbf{C} \mathbf{z}, \mathbf{z} \in \mathbb{B}(p) \}
\end{equation}

where $\mathbb{E}(p,\mathbf{C})$ is an ellipsoid, $\mathbf{C}$ is an affine transformation, and $\mathbb{B}(p)$ is a unit $p$-norm ball.

Ellipsoids and Zonotopes

Both ellipsoids and zonotopes provide straightforward numerical implementations of operations that are commonly-used for reachability analysis, fault detection, and similar tasks. For example, both representations can be transformed readily via affine maps, thereby representing the flow of a (linearized) dynamical system. Furthermore, one can apply convex programming to efficiently detect when these sets intersect with e.g., obstacles for collision avoidance\textsuperscript{6},\textsuperscript{10},\textsuperscript{13}. However, choosing between the two representations comes with certain tradeoffs. For example, zonotopes are closed under Minkowski sums, which are used to incorporate uncertainty and linearization error, while ellipsoids are not. On the other hand, ellipsoids can exactly represent the confidence level sets of Gaussian distributions, while zonotopes cannot.

Note, we present a more detailed discussion of other set representations, both convex and non-convex, in Section II. Out of the convex representations, we consider ellipsoids and zonotopes the best-suited for reachability and fault detection tasks. For the non-convex representations, we typically lose the ability to perform efficient, convex collision-checking.

B. Contributions

Our main contribution is a novel set representation called the ellipsotope, which combines the advantages of both ellip-

---

1 Aeronautics and Astronautics, Stanford University, Stanford, CA.
2 Electrical Engineering, Stanford University, Stanford, CA.
Corresponding author: gracegao@stanford.edu

Fig. 1: Basic ellipsotopes with five generators and increasing $p$-norm ($p = 2, 4, \cdots, 10$), shown with lighter blue as the norm increases. The outermost shape is the $\infty$-norm zonotope, and the innermost shape is the 2-norm ellipsoid.
soids and zonotopes. For the purposes of reachability analysis and fault detection, we show that ellipsotopes are closed under linear maps, Minkowski sums, and intersections. We also introduce several order reduction strategies for managing ellipsoid complexity, which can grow during reachability analysis. We demonstrate the utility of these objects with a variety of numerical examples.

C. Paper Organization

Section II discusses a variety of set representations to clarify the context for ellipsotopes. Section III introduces notation and set representations relevant to developing ellipsotopes. Section IV defines ellipsotopes and discusses properties and closed operations. Section V covers methods for order reduction. Section VI covers numerical examples and applications of ellipsotopes. Section VII concludes the paper.

II. RELATED WORK

A variety of set representations exist for reachability analysis and fault detection. We now discuss these representations, and under which operations they are closed, meaning that an operation yields an instance of the same representation. We divide our discussion by convex and non-convex sets.

A. Convex Set Representations

Convex representations enable one to use convex programming to evaluate intersection and set membership. This enables one to certify that, e.g., a system’s reachable set lies within a safe region, because a convex program is guaranteed to converge. In particular, we discuss ellipsoids, convex polytopes, and support functions.

As mentioned before, ellipsoids are affine transformations of the 2-norm ball. This set representation is closed under affine transformations and hyperplane intersections. For operations such as Minkowski sum, intersection, Pontryagin (Minkowski) difference, and convex hull, efficient algorithms exist to generate inner- and outerapproximative ellipsoids. Most importantly, for tasks such as reachability analysis, confidence level sets of multivariate Gaussian distributions are ellipsoidal. Unfortunately, ellipsoidal representations of reachable sets can rapidly become conservative due to the overapproximation required for Minkowski sums. Furthermore, ellipsoids are not well-suited to representing polytopic sets such as occupancy grids, which are commonly used for tasks such as robot motion planning.

Convex polytopes can be thought of as the bounded intersection of a collection of affine halfspaces in arbitrary dimensions (H-representation); note, an unbounded intersection is called a polyhedron. Another common representation is as the convex hull of a set of vertices (V-representation). This broad category of objects is closed under Minkowski sum, intersection, Pontryagin difference, and convex hull. The H-representation is especially convenient for determining if a polytope contains a point and performing intersections. However, the remaining operations are not computationally efficient, especially in high dimensions or when a convex polytope is defined by a large number of halfspaces.

To avoid these challenges, zonotopes have become a popular representation that enable efficient Minkowski sums and set containment queries. A zonotope is a centrally-symmetric convex polytope constructed as a Minkowski sum of line segments. Zonotopes can be parameterized by a center and generator (see (2) in Section III), which we call a CG-representation; any point in the zonotope is the center plus a linear combination of the generators, each scaled by a coefficient in $[-1,1]$. Since zonotopes are not closed under intersection or Pontryagin difference, several authors have introduced AH-polytopes and constrained zonotopes. An AH-polytope is the affine transformation of an H-representation of a polytope (e.g., a zonotope is the affine transformation of a hypercube). A constrained zonotope is a zonotope with additional linear constraints on its coefficients. These representations are closed under affine transformation, Minkowski sum, intersection, and, for constrained zonotopes, Pontryagin difference and convex hull. All convex polytopes are constrained zonotopes, and set membership or intersection queries can be evaluated by solving a linear program. While AH-polytopes and constrained zonotopes overcome many of the challenges of zonotopes, they still cannot represent sets with smooth curved boundaries, such as ellipsoids.

Support functions enable one to represent arbitrary convex sets, allowing generalization beyond polytopes and ellipsoids. A support function is a convex function that maps a vector in Euclidean space to the maximum dot product between that vector and any element in a convex set, thereby providing an implicit set representation. Support functions of many convex sets, such as unit balls, ellipsoids, and zonotopes, have a simple analytical form, and the support function for polytopes can be expressed as the solution of a linear program. Furthermore, affine maps, Minkowski sums, and convex hulls have analytic formulations. Unfortunately, the intersection of sets represented by support functions can only be overapproximated and results in a non-convex representation, so using intersection for collision-checking and fault detection is neither straightforward nor conservative.

B. Non-Convex Set Representations

The reachable set of a dynamical system is not necessarily convex. Furthermore, robots and other autonomous systems frequently have non-convex bodies, and such systems are not necessarily subject to convex constraints for fault detection or collision avoidance. A variety of non-convex set representations exist that attempt to address these challenges. In particular, we discuss polynomial zonotopes, star sets, level sets, and Constructive Solid Geometry (CSG).

Polynomial zonotopes (PZs) are a generalization of zonotopes wherein the coefficients of a zonotope’s generators are instead allowed to be monomials. By leveraging the center/generator structure of zonotopes, these sets are closed under affine transformation and Minkowski sum. One
can add polynomial constraints on the coefficients to make constrained polynomial zonotopes (CPZs), which are closed under intersection and convex hull \( [24] \). PZs and CPZs provide much tighter (i.e., less conservative) approximations of reachable sets than zonotopes, at the expense of being non-convex. This means that collision checking requires solving a non-convex program, typically preventing solution guarantees. One alternative is to overapproximate a PZ or CPZ with a zonotope \( [25] \), resulting in a convex collision check at the expense of significant increase in conservatism.

Star sets are also a generalization of zonotopes and ellipsoids \( [29] - [28] \), which instead use a generic logical predicate constraint on their generator coefficients. These sets can be non-convex, and are closed under affine transformation, Minkowski sum, and intersection. However, similar to PZs and CPZs, checking for emptiness or collision can require solving a non-convex feasibility problem.

Departing from the center/generator construction used for zonotopes and similar objects, level sets are a popular representation for reachability analysis, because arbitrary sets can be represented as the 0-sublevel set of a function. Such a function can be approximated on a grid \( [29] - [30] \) or as a polynomial \( [31] \). Level sets can be used to conservatively compute reachable sets of robots and similar systems subject to uncertainty \( [1] - [4] , [32] - [33] \). In the special case of rigid-body robot motion planning with polynomial level sets, one can represent collision checking as a polynomial evaluation \( [1] \): however, in general, Minkowski sums, intersections, and convex hulls can be approximated using sums-of-squares programming. Level set methods typically do not require linear maps and Minkowski sums to perform reachability analysis; however, they instead require approximately solving either the Hamilton-Jacobi-Bellman or Liouville partial differential equations. Furthermore, level set representations still suffer from the curse of dimensionality, requiring decompositions or approximations for nonlinear dynamical systems with more than 5 dimensions \( [1] - [4] \).

The final set representation we discuss is Constructive Solid Geometry (CSG), which is used to model non-convex shapes in the computer graphics community \( [34] - [35] \). Similar to support functions, CSG leverages an implicit point membership classification function to express geometric primitives such as spheres, prisms, and cones. Non-convex bodies are represented as unions, intersections, and set differences or approximations for nonlinear dynamical systems \( [36] - [37] \). For these sets, computing Minkowski sums is challenging; furthermore, these representations are typically limited to 2D or 3D settings, and it is unclear how to reduce the growing complexity of a reachable set in a similar way to zonotope order reduction.

C. Summary

From this review of a wide variety of representations, we identify several advantages and challenges. The advantages of zonotopes and similar objects is their numerical simplicity for representing affine transformations, Minkowski sums, and collision/emptiness checking (via intersection operations). The challenges are to represent smooth or non-polytopic sets without incurring conservatism (as with ellipsoids) or non-convexity (as with polynomial zonotopes). Our proposed ellipsoid representation directly addresses this tradeoff by enabling efficient reachability and fault detection operations for both polytope-like and ellipsoid-like objects without introducing challenges from losing convexity.

III. Preliminaries

We now introduce notation and several set representations.

A. Notation

1) Points, Sets, and Set Operations: Scalars and vectors are lowercase and italic. Sets and matrices are uppercase italic. The real numbers are \( \mathbb{R} \), and the natural numbers are \( \mathbb{N} \). If \( n \in \mathbb{N} \), we denote \( \mathbb{N}_n = \{1, 2, \ldots, n\} \subset \mathbb{N} \). The \( p \)-norm unit ball in \( \mathbb{R}^n \) is \( B_{p,n} = \{ x \in \mathbb{R}^n \mid \| x \|_p \leq 1 \} \). An affine subspace (i.e., affine hyperplane) of \( \mathbb{R}^n \) parameterized by \( H \in \mathbb{R}^{n \times m} \) with \( m \in \mathbb{N} \) and \( f \in \mathbb{R}^n \), is \( \mathcal{P}(H, f) = \{ x \in \mathbb{R}^m \mid Hx = f \} \). A halfspace parameterized by \( h \in \mathbb{R}^n \) and \( s \in \mathbb{R} \), is \( \mathcal{H}(h, s) = \{ x \in \mathbb{R}^n \mid h^T x \leq s \} \).

Let \( A \) be a set such that \( A \subset \mathbb{R}^n \). Its power set is \( \text{pow}(A) \), its cardinality is \( |A| \), and its boundary is \( \text{bd}(A) \). Let \( B \subset \mathbb{R}^n \) as well. The Minkowski sum is \( A \oplus B = \{ a + b \mid a \in A, b \in B \} \). The convex hull of \( A \cup B \) is \( \text{CH}(A \cup B) = \{ \lambda a + (1 - \lambda) b \mid \lambda \in [0, 1], a \in A, b \in B \} \).

Consider a set of integers \( J = \{ j_1, j_2, \ldots, j_n \} \subset \mathbb{N} \) and \( m \in \mathbb{N} \); then \( J + m = \{ j_1 + m, \ldots, j_n + m \} \). Similarly, consider a set of sets of integers \( J = \{ J_1, J_2, \ldots, J_n \} \subset \text{pow}(\mathbb{N}) \). We denote \( \bigcup J_m \) to mean \( \{ j_1 + m, j_2 + m, \ldots, j_n + m \} \).

2) Vectors, Arrays, and Matrices: An \( n \times m \) matrix of ones is \( 1_{n \times m} \). Similarly, a matrix of zeros is \( 0_{n \times m} \). An \( n \times n \) identity matrix is \( I_n \). Let \( v \in \mathbb{R}^n, w \in \mathbb{R}^m \); we denote vector concatenation by \( (v, w) \in \mathbb{R}^{n+m} \). The diag(\( \cdot \)) operator places its arguments (block) diagonally on a matrix of zeros. The eig(\( \cdot \)) operator returns a column vector containing the eigenvalues of its input matrix. The det(\( \cdot \)) operator returns the determinant of a square matrix \( A \).

Let \( v \in \mathbb{R}^n \) and \( J \subset \mathbb{N} \). Then \( v(J) = v(J^J) \) is the vector of elements of \( v \) indexed by \( J \). Similarly, if \( A \in \mathbb{R}^{n \times m} \), \( J_1 \subset \mathbb{N} \), and \( J_2 \subset \mathbb{N} \), then \( A(J_1, J_2) \) is the \( |J_1| \times |J_2| \) sub-matrix of \( A \). We denote \( A(J, :) \) as the \( |J| \times m \) submatrix of \( A \) that is, the \( J \)-rows and all the columns. For a vector \( v \in \mathbb{R}^n \) and an integer matrix \( M \in \mathbb{N}^{n \times m} \), let \( v^M \in \mathbb{R}^m \) denote a vector for which \( v^M(j) = \prod_{i=1}^n (v(i))^M(i, j) \) with \( j = 1, \ldots, m \).

B. Set Representations

An ellipsoid is the set

\[
E(c, Q) = \left\{ x \in \mathbb{R}^n \mid (x - c)^T Q (x - c) \leq 1 \right\}.
\]

We call \( c \) its center and positive definite \( Q \) its shape matrix. Note that one may also see \( Q^{-1} \) as defining an ellipsoid \( [14] - [38] \).
A zonotope $Z(c, G) \subset \mathbb{R}^n$ is a convex, symmetrical polytope parameterized by a center $c \in \mathbb{R}^n$ and a generator matrix $G \in \mathbb{R}^{n \times m}$, given by

$$Z(c, G) = \{ c + G\beta \mid \|\beta\|_\infty \leq 1 \}.$$  

That is, a zonotope is a set of convex combinations of $c$ with the columns of the matrix $G$, which we call generators. We call $\beta$ the generator coefficients.

For context, we also provide the definitions for related set representations. From [1], a constrained zonotope is defined as follows. Let $A \in \mathbb{R}^{n \times m}$, $b \in \mathbb{R}^k$, $k \in \mathbb{N}$ is the number of linear constraints. We denote a constrained zonotope as

$$\mathcal{C}Z(c, A, b, m) = \{ c + G\beta \mid \|\beta\|_\infty \leq 1 \text{ and } A\beta = b \},$$  

where $c$ and $G$ are as in (2).

Finally, we introduce constrained polynomial zonotopes (CPZs) [24]. Given $c \in \mathbb{R}^n$, $G \in \mathbb{R}^{n \times m}$, $X \in \mathbb{N}^{m \times m}$, $A \in \mathbb{R}^{k \times m}$, $b \in \mathbb{R}^k$, and $D \in \mathbb{N}^{k \times n}$, a CPZ is the set

$$\mathcal{CPZ}(c, G, X, A, b, D) = \left\{ c + G\beta X \mid \|\beta\|_\infty \leq 1 \text{ and } A\beta D - b = 0 \right\}.$$  

As one would expect, polynomial zonotopes (PZs) are CPZs without constraints. Importantly, PZs and CPZs are not necessarily convex [23], [24].

IV. ELLIPSOTOPES

In this section, we define ellipsotopes, then discuss several useful properties. We then discuss the specific case of ellipsotopes defined using a 2-norm and conclude the section by relating ellipsotopes to other set representations.

A. Definition

To define ellipsotopes, we first introduce index sets.

**Definition 1.** Let $m \in \mathbb{N}$. Let $\mathcal{J} \subset \text{pow} (\mathbb{N}_m)$ be a partition of $\mathbb{N}_m$. We call $\mathcal{J}$ an index set. That is, $\mathcal{J}$ is a set of multi-indices such that $\mathbb{N}_m = \bigcup_{J \in \mathcal{J}} J$ and $J_1 \cap J_2 = \emptyset$ for any $J_1, J_2 \in \mathcal{J}$.

In other words, every integer from 1 to $m$ occurs in exactly one subset $J \in \mathcal{J}$. As an example, for $m = 3$, $\mathcal{J} = \{\{1, 2\}, \{3\}\}$ obeys the definition.

We now define ellipsotopes:

**Definition 2.** An ellipsotope is a set

$$\mathcal{E}_p(c, G, A, b, J) = \{ c + G\beta \mid \|\beta\|_p \leq 1 \text{ and } A\beta = b \},$$  

A basic ellipsotope, $\mathcal{E}_p(c, G)$, has no constraints or index set. A constrained ellipsotope, $\mathcal{E}_p(c, G, A)$, has no index set. An indexed ellipsotope, $\mathcal{E}_p(c, G, J)$, has no constraints.

One can go further and let the different subsets of $\beta$ be subject to different $p$-norms, but we have not yet needed this in practice. We call

$$\mathcal{B}_p(J) = \{ \beta \in \mathbb{R}^m \mid \|\beta(J)\|_p \leq 1 \text{ for all } J \in \mathcal{J} \}.$$
either add $m_1$ or halfspace $m_2$. We now present both of these

intersections are related to the Minkowski sum, see [13].

This property is illustrated in Figure 3. Note that, since $E_1 \cap E_2 = E_2 \cap E_1$, one can choose which center to keep in (11a) to minimize the number of zero generators (that is, one can either add $m_1$ or $m_2$ generators). For more details on how intersections are related to the Minkowski sum, see [13].

Often, for hybrid system reachability analysis, it is necessary to detect when a reachable set intersects a hyperplane or halfspace [12], [13]. We now present both of these

intersections for ellipsotopes. As before, we do not assume in either case that the intersection is nonempty.

Proposition 4 (Ellipsotope Intersection). Suppose that $E_1 = \mathcal{E}_2(c_1, G_1, A_1, b_1, \mathcal{J}_1) \subset \mathbb{R}^n$ with $m_1$ generators and $A_1 \in \mathbb{R}^{k_1 \times m_1}$, and $E_2 = \mathcal{E}_2(c_2, G_2, A_2, b_2, \mathcal{J}_2) \subset \mathbb{R}^n$ with $m_2$ generators and $A_2 \in \mathbb{R}^{k_2 \times m_2}$. Then $E_1 \cap E_2$ is an ellipsotope $E_\cap$ given by

$$E_\cap = \mathcal{E}_2(c_\cap, G_\cap, A_\cap, b_\cap, \mathcal{J}_\cap),$$

$$A_\cap = \begin{bmatrix} A_1 & 0_{k_1 \times m_2} \\ 0_{k_2 \times m_1} & A_2 \end{bmatrix}, b_\cap = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix},$$

$$\mathcal{J}_\cap = \mathcal{J}_1 \cup (\mathcal{J}_2 + m_1),$$

where $m_1$ is the number of generators of $E_1$.

Proof. This follows from [6] Proposition 1] by noticing (similar to the proof of Proposition 3] that $\mathcal{J}$ ensures that the $p$-norm constraints are applied separately to the coefficients of $E_\cap$, depending on whether they came from $E_1$ or from $E_2$. \hfill \Box

This property is illustrated in Figure 3. Note that, since $E_1 \cap E_2 = E_2 \cap E_1$, one can choose which center to keep in (11a) to minimize the number of zero generators (that is, one can either add $m_1$ or $m_2$ generators). For more details on how intersections are related to the Minkowski sum, see [13].

Often, for hybrid system reachability analysis, it is necessary to detect when a reachable set intersects a hyperplane or halfspace [12], [13]. We now present both of these

intersections for ellipsotopes. As before, we do not assume in either case that the intersection is nonempty.

Proposition 5 (Ellipsotope-Hyperplane Intersection). Let $E = \mathcal{E}_p(c, G, A, b, \mathcal{J}) \subset \mathbb{R}^n$ with $c \in \mathbb{R}^n$, $G \in \mathbb{R}^{n \times m}$, $A \in \mathbb{R}^{k \times m}$, $b \in \mathbb{R}^n$, and $\mathcal{J}$ a valid index set. Let $P = \mathcal{P}(H, f) \subset \mathbb{R}^n$ be an affine hyperplane. Then $E \cap P = E_\cap$ where

$$E_\cap = \mathcal{E}_p(c, G, A_p, b_p, \mathcal{J}),$$

$$A_p = \begin{bmatrix} A \\ H \cdot G \end{bmatrix}, b_p = \begin{bmatrix} b \\ f - Hc \end{bmatrix},$$

$$\mathcal{J} = \mathcal{J} \cup \{m+1\},$$

where $|G|$ denotes the element-wise absolute value of the generator matrix $G$.

Proof. Recall that $P = \{x \in \mathbb{R}^n \mid Hx = f\}$. Then, if $x \in E \cap P$, there exists $\beta \in \mathbb{R}^n$ feasible for $E$, for which

$$H(c + G\beta) = f \implies H \cdot G \beta = f - Hc,$$

which is the last block row of the linear constraint in $E_\cap$. \hfill \Box

To intersect an ellipsoid with a halfspace, we adapt [18] Theorem 1].

Proposition 6 (Ellipsotope-Halfspace Intersection). Let $E = \mathcal{E}_p(c, G, A, b, \mathcal{J}) \subset \mathbb{R}^n$ with $c \in \mathbb{R}^n$, $G \in \mathbb{R}^{n \times m}$, $A \in \mathbb{R}^{k \times m}$, $b \in \mathbb{R}^n$, and $\mathcal{J}$ a valid index set. Let $S = \mathcal{H}(h, s) \subset \mathbb{R}^n$, where $h \in \mathbb{R}^n$ and $s \in \mathbb{R}$. Then $E_\cap = E \cap S$ is given by

$$E_\cap = \mathcal{E}_p(c, G, A_\cap, b_\cap, \mathcal{J}_\cap),$$

$$A_\cap = \begin{bmatrix} A \\ 0_{k_1 \times 1} \end{bmatrix}, b_\cap = \begin{bmatrix} b \\ 0_{k_1 \times 1} \end{bmatrix},$$

$$\mathcal{J}_\cap = \mathcal{J} \cup \{m+1\},$$

where $|G|$ denotes the element-wise absolute value of the generator matrix $G$.

Proof. We prove this property constructively. Recall that $S = \{x \in \mathbb{R}^n \mid h^\top x \leq s\}$. Following the logic of Proposition 5 our strategy is to add a linear constraint to the coefficients $\beta$ of $E$ constraining the resulting set to lie within the halfspace; that is, we want $h^\top (c + G\beta) \leq s$. However, we need a slack variable to enforce this as an equality constraint: $h^\top (c + G\beta) + \gamma = s$, with $\gamma \geq 0$. We cannot add $\gamma$ directly as a coefficient to the ellipsoid, because it is unconstrained; instead, we want to bound $\gamma$ to lie within an interval, which
we can map to the interval $[-1, 1]$ containing a (scalar) ellipsotope coefficient. To do this, we first find an upper bound for $\gamma$ using the fact that $E$ is compact and lies fully within a zonotope: $E \subset Z(c, G)$. From [12 Sec. 5.1], we have

$$\gamma \leq s - h^T c + h^T |G| I_{m \times 1}$$  \hspace{1cm} (16)

Now, we want to pick $d$ such that, for any $\gamma$, $d(\beta, 1) = \gamma$ and $\|\beta\|_p \leq 1$, where $\beta$ is our additional coefficient. That is, we want to find an affine transform of the interval $[-1, 1]$ to the interval $[0, s - h^T c + h^T |G| I_{m \times 1}]$. Applying interval arithmetic, we can solve for $d$:

$$d([-1, 1] + 1) = [0, s - h^T c + h^T |G| I_{m \times 1}]$$  \hspace{1cm} (17)

We can then construct the necessary linear equality constraint on $\beta$ and $\beta_k$ as

$$h^T (c + G\beta) + d(\beta_k + 1) = s$$  \hspace{1cm} (19)

$$h^T G\beta + d\beta_k = s - h^T c - d$$  \hspace{1cm} (20)

Notice that $E_S$, as in (15), is the ellipsotope $E$ with one additional coefficient and the additional linear constraint in (20), with $j_3$ ensuring that $\beta_k \in [-1, 1]$, completing the proof. \hfill \square

To build intuition for what it means when $E \cap S$ is empty, consider the zonotope $Z = Z(c, G)$. Notice that, if $Z \cap S = \emptyset$, then $E \cap S = \emptyset$. In the case of the zonotope, we can interpret this to mean that the affine subspace $P(A_3, B_3) = \mathbb{R}^{m+1}$ does not intersect the $\infty$-norm unit ball in $\mathbb{R}^{m+1}$. Similarly for the ellipsotope case, the affine subspace does not intersect the ball product $B(x_j) = \mathbb{R}^{m+1}$.

**C. Emptiness and Point Containment**

Given a system’s state, it is often useful to check if it lies within a specific region of state space. Similarly, given a reachable set in state space, one may need to check if this set intersects with, e.g., an unsafe set. Assuming ellipsotope representation of the states and sets in question, we perform the desired checks as follows, by leveraging Proposition 4 wherein the intersection of ellipsotopes is again an ellipsotope.

**Proposition 7 (Emptiness and Point Containment).** Consider the ellipsotope $E = E_P(c, G, A, b, J) \subset \mathbb{R}^n$ with $m$ generators. Assume $P(A, b) \neq \emptyset$. Let $x \in \mathbb{R}^n$, and let

$$\text{cost}(\beta) = \max_{J \in \mathcal{J}} \| \beta (J) \|_p,$$  \hspace{1cm} (21)

where $\beta$ is the ellipsotope coefficient vector. Then

$$E \neq \emptyset \iff \min_{\beta \in \mathbb{R}^m} \{ \text{cost}(\beta) \mid A\beta = b \} \leq 1$$  \hspace{1cm} (22)

$$x \in E \iff \min_{\beta \in \mathbb{R}^m} \{ \text{cost}(\beta) \mid \begin{bmatrix} A & b \end{bmatrix} \beta = \begin{bmatrix} b & x - c \end{bmatrix} \} \leq 1,$$  \hspace{1cm} (23)

which are both convex programs.

**Proof.** We prove the claim for (22), as the claim for (23) then follows from Proposition 4 by checking the emptiness of $E \cap \mathbb{R}^n(x, [\bullet])$. Notice that, if $\beta \in \mathbb{R}^m$ is feasible for the ellipsotope definition constraints in (3), then $\text{cost}(\beta) \leq 1$ by construction. Therefore, (22) evaluates whether or not the set $P(A, b)$ intersects $B(x_j)$ (i.e., the set of feasible $\beta$ as in (6)). The constraint set is empty by assumption and convex by inspection. Since $\|\bullet\|_p$ is convex, and the max of convex functions is also convex, $\text{cost}(\beta)$ is convex. \hfill \square

We find in practice that, when an ellipsotope is nonempty, it takes on the order of $10^5$ $s$ to solve (22), but it takes two to four orders of magnitude longer for empty ellipsotopes. However, by instead searching for a feasible $\beta$ to the constraints $A\beta = b$ and $\beta \in B(x_j)$, we achieve much lower solve times in practice. We write the search for a feasible $\beta$ as follows:

**Corollary 8 (to Proposition 7).** Let $E = E_P(c, G, A, b, J) \subset \mathbb{R}^n$ with $m$ generators. Assume $P(A, b) \neq \emptyset$. Then

$$E \neq \emptyset \iff \min_{\beta \in \mathbb{R}^m} \{ \| A\beta - b \|_2^2 \mid \beta \in B(x_j) \} = 0.$$  \hspace{1cm} (24)

**Proof.** This formulation follows directly from the fact that, for any feasible $\beta$, we have $A\beta = b$ and $\beta \in B(x_j)$. \hfill \square

Notice that, in the case of a constrained zonotope, (24) becomes a bounded-value least squares problem.

**D. Properties of 2-Ellipsotopes**

We now briefly discuss the special case of 2-ellipsotopes (i.e., ellipsoids with $p = 2$). First, we confirm that basic 2-ellipsotopes are ellipsoids and vice-versa. Second, we notice that constrained 2-ellipsotopes are in fact basic 2-ellipsotopes. Later, in Sec. V, we leverage these properties to create an order reduction strategy for 2-ellipsotopes.

**Lemma 9 (Ellipsoid-Ellipsotope Equivalence).** (Claim 1) Let $E = E(c, Q) \subset \mathbb{R}^n$ be an ellipsoid as in (1). Then $E = E_2(c, (\sqrt{Q})^{-1}) \subset \mathbb{R}^n$. (Claim 2) Suppose $E = E_2(c, G) \subset \mathbb{R}^n$. Then there exists $Q \in \mathbb{R}^{m \times m}$, $Q > 0$, such that $E = E(c, Q)$. \hfill \square

**Proof.** (Claim 1) Note $(\sqrt{Q})^{-1} > 0$ exists because $Q > 0$. Suppose $x \in E$, so $(x - c)^T Q(x - c) \leq 1$. We want to find $G$ and $\beta$ such that $(x - c)^T G\beta \leq 1$ and $\|\beta\|_p \leq 1$. If we set $(G\beta)^T Q(G\beta) = \beta^T \beta$, then $G\beta = (\sqrt{Q})^{-1} \beta$.

(Claim 2) Suppose that $x \in E$, so there exists $\beta$ such that $G\beta = x - c$. It follows from Proposition 4 that $\beta = G^T (x - c)$, where $G^T$ is the Moore-Penrose pseudoinverse of $G$. Since $\|\beta\|_p^2 = \beta^T \beta$, we have $\|\beta\|_p^2 = (x - c)^T (G^T)^T G^T (x - c)$. Then, by picking $Q = (G^T)^T G^T$, the proof is complete. \hfill \square

While these claims are well-known in the literature (e.g., [6 (3)]), we write the proof to clarify Lemma 12 in Section V.

Next, we find a further equivalence between constrained and basic 2-ellipsotopes. To prove this, first, we confirm the well-known result that the (nonempty) intersection of an $n$-dimensional ellipsoid with an affine subspace is a lower-dimensional ellipsoid:

**Lemma 10.** Let $B = B_2(m)$ (the $m$-dimensional $2$-norm ball) and $H = P(A, b)$ (an affine hyperplane) where $A \in \mathbb{R}^{n \times m}$ is full row rank with $n < m$, and $b \in \mathbb{R}^n$. Suppose $B \cap H \neq \emptyset$
and $|B \cap H| > 1$. Then there exist $T \in \mathbb{R}^{m \times (m-n)}$ and $t \in \mathbb{R}^m$ such that $T \mathbf{B}_{2,m-n} + t = B \cap H \subset \mathbb{R}^m$.

**Proof.** We construct $t = A^T b \in H$. Notice that $t \in H$ because $n < m$. Then, $t \in B$ because it is a least-squares solution and $B \cap H \neq \emptyset$. To construct $T$, let $p \in \text{bd}(B \cap H)$, so $||p||_2 = 1$. Since $|B \cap H| > 1$, we have $||t||_2 < 1$. Let $c = |p - t||_2$, and note $c < 1$ by the triangle inequality. Also by triangle inequality, for any $q \in B \cap H$, if $||q - t||_2 \leq c$, then $||q||_2 \geq 1$. Let $K = \{e_1, \ldots, e_{m-n}\} \subset \mathbb{R}^m$ be an orthonormal basis for $\ker(A)$. Then $T = [ce_1, \ldots, ce_{m-n}]$. \hfill \square

**Lemma 11** (Basic and Constrained 2-Ellipsotope Equivalence). Let $E = \mathcal{E}_2(c; G, A, b)$ be a nonempty constrained ellipsoid over $A \in \mathbb{R}^{k \times m}$, $b \in \mathbb{R}^k$, and $k < m$. Then there exist $c', G'$ such that $E = \mathcal{E}_2(c', G')$.

**Proof.** This follows from Lemma 10. Since $E$ is nonempty, we can construct an affine map parameterized by $t$ and $T$ such that $T \mathbf{B}_{2,m-k} + t = B \subset \mathbb{R}^m$. Then, for any $\beta \in \mathbb{R}^{2,m-k}$, we have $c + G(t \beta + t) \in E$. Choose $c' = c + Gt$ and $G' = Gt$ to complete the proof. \hfill \square

Note that 2-ellipsotopes let us represent ellipsoidal Gaussian confidence level sets. We demonstrate this via a robot path verification example in Sec. VI-C.

**E. Relationships to Other Set Representations**

1) Ellipsoids, Zonotopes, and Similar Representations: Per Lemma 9, ellipsotopes generalize ellipsoids and, as a corollary, superellipsoids. We see from the Definition 2 specifically (5) that ellipsotopes generalize (constrained) zonotopes, by comparison to (2). In particular, if the index set $J = \{1, \{2\}, \ldots, \{m\}\}$ for an ellipsoid over $m$ generators, then the ellipsoid is also a (constrained) zonotope.

Another useful set representation is the **capsule**, often used to represent robot manipulator links for efficient collision detection [40], [41]. A capsule is the Minkowski sum of a line segment with a sphere, which we can represent as an ellipsoid over Lemma 9 and Proposition 3. Importantly, ellipsotopes allow us to generalize capsules to Minkowski sums of line segments with, e.g., confidence level sets ellipsoids of a Gaussian distribution.

2) Constrained Polynomial Zonotopes: We can also show that every ellipsoid is a CPZ in [2] by applying similar logic to the proof that every ellipsoid is a CPZ. Consider the basic case of $E = \mathcal{E}_p(c; G)$. Then,

$$ E = \left\{ c + G\beta \mid \beta \in \mathbb{R}^k, \beta_1 = 0 \right\}, $$

$$ E = \left\{ c + Gb \mid 0.5\beta_k + \beta_1 = 0.5, \text{ and } \beta \in \mathbb{R}^k \right\}, $$(26)

where $\beta \in \mathbb{R}^k$ acts as a slack variable. Then the ellipsotope is $E = \mathcal{CZ}_p(c; G, A, b, D)$ with $X = I_m$, $b = 0.5 \cdot I_{k \times 1}$, and $D = k \cdot I_{p \times 1}$. Adding linear constraints or an index set on the coefficients of $E$ necessitates only minor changes to $A$, $b$, and $D$ in the CPZ formulation.

**V. Order Reduction**

A commonly used operation in zonotope reachability analysis is order reduction, or the approximation of a zonotope by a new zonotope with fewer generators. This operation is necessary because reachability analysis commonly uses Minkowski sums, which increase the number of generators of a zonotope (or ellipsotopes, per Proposition 3).

A variety of order reduction techniques exist for zonotopes, most commonly achieved by enclosing a subset of a zonotope’s generators in a bounding box, the sides of which become new generators [11], [12]. This strategy can be improved or guided by a variety of heuristics [2] Ch. 2. In the case of polynomial zonotopes, which are not necessarily convex, one can apply a similar strategy of overapproximating a subset of generators with a zonotope or interval [23], [25]. For constrained zonotopes, the linear constraints necessitate alternative strategies [6], [18].

To proceed, we discuss order reduction for 2-ellipsotopes leveraging ellipsoid techniques. We then comment on more general strategies.

A. Order Reduction for 2-Ellipsotopes

For 2-ellipsotopes, we can leverage the properties of ellipsoids to perform order reduction.

1) Basic 2-Ellipsotopes: First, we note that a basic 2-ellipsotope in $\mathbb{R}^n$ never requires more than $n$ generators:

**Lemma 12** (Exact Order Reduction of Basic 2-Ellipsotopes). Let $E = \mathcal{E}_2(c; G) \subset \mathbb{R}^n$ with $G \in \mathbb{R}^{n \times n}$ full rank row and $m > n$. Then $E = \mathcal{E}_2(c; G^\dagger)$, where

$$ G^\dagger = \left( G^T (G^T G)^{−1} \right)^{−1}. $$

(27)

and $G^\dagger$ is the Moore-Penrose pseudoinverse of $G$.

**Proof.** This follows from Lemma 9 by converting $E$ to an ellipsoid, then back to an ellipsotope. Note, the matrix in the uppermost parentheses of (27) is invertible because it is the square root of a positive definite matrix. \hfill \square

Note that $G \in \mathbb{R}^{n \times n}$, so $E$ needs only $n$ generators.

2) General Strategy for 2-Ellipsotopes: Our general strategy is to treat 2-ellipsotopes as a Minkowski sum of ellipsoids. This is because order reduction is usually necessary after several Minkowski sum operations result in a large number of generators during, e.g., reachability analysis.

To explain our approach, we consider a simple case. Consider $E = \mathcal{E}_2(c, G, A, b, J) \subset \mathbb{R}^n$ with $m > n$ generators and with $k$ linear constraints. Suppose that we can write $E = E_1 \oplus E_2$ where $E_1 = \mathcal{E}_2(c_1, G_1, A_1, b_1)$ with $m_1$ generators and $E_2 = \mathcal{E}_2(c_2, G_2, A_2, b_2)$ with $m_2$ generators. Notice that $m = m_1 + m_2$. Our goal is to find $E$ for which $E = \mathcal{E}_2(c', G') \supset E$. First, by Lemma 11 we can find $T_1$ and $T_2$ such that $E_1 = \mathcal{E}_2(c_1 + G_1 T_1, G_1 T_1)$, and similarly for $E_2$. Then, per Lemma 9, we can find $O_1$ to represent $E_1$ as an ellipsoid, $E_1 = \mathcal{E}_2(c_1 + G_1 T_1, O_1)$, and similarly we can find $O_2$ for $E_2$.

Now we apply the method in [13] to create a minimum-volume outer ellipsoid (MVOE) $E_{\text{dE}} \supset E_1 \oplus E_2$. That is, we
can write $E_{\text{dc}} = \mathcal{E}(c_{\text{dc}}, Q_{\text{dc}}) \supset E_1 \oplus E_2$. By Lemma 9 we know that $E_{\text{dc}} \supset E$. By Lemma 12 we know that $E_{\text{dc}}$ needs no more than $n < m$ generators. Therefore, we can choose $E = E_{\text{dc}}$.

3) Choosing Which Ellipsoids to Overapproximate: The above example considered an ellipsotope created as the Minkowski sum of a pair of ellipsoids, so the order reduction strategy was to overapproximate this sum with a single ellipsoid. We now extend this idea to the case when an ellipsotope is a Minkowski sum of many ellipsoids.

First, we set up our assumptions. Consider again the ellipsotope $E = \mathcal{E}_2(E, G, A, b, J) \subset \mathbb{R}^n$ with $m$ generators. Assume that we can write $E$ as the Minkowski sum of several basic 2-ellipsotopes, which we call component ellipsotopes:

$$E = E_1 \oplus E_2 \oplus \cdots \oplus E_r,$$  

for some $r \in \mathbb{N}$. That is, each $E_i = \mathcal{E}_2(c_i, G_i)$. Notice that $E$ requires at most $r \times n$ generators.

Now, suppose that we want to find $\tilde{E}$ such that $\tilde{E} \supset E$ and $\tilde{E}$ has $m - n$ generators; in other words, we want to reduce the number of 2-ellipsotopes in (28) by one. To do so, we choose $i, j \in \mathbb{N}_r$ and construct $E_{\text{rdc}} = E_i \oplus E_j$ such that

$$E = \left( \bigoplus_{i \in \mathbb{N}_r \setminus \{i, j\}} E_i \right) \oplus E_{\text{rdc}}.$$  

The question is then how to choose $i$ and $j$. Our goal for choosing $i$ and $j$ is to minimize the conservatism introduced by overapproximating $E_i \oplus E_j$.

The most straightforward option is to choose the pair $(i, j)$ for which the MVOE has the smallest volume. We do this by applying the standard formula for the volume of an $n$-dimensional hyperellipsoid; if $E = \mathcal{E}(c, Q) \subset \mathbb{R}^n$, then volume($E$) = det($((\sqrt{Q})^{-1})_{i,j=1}^{n} \frac{\pi^{n/2}}{\Gamma(n/2)}$, where det$(\cdot)$ denotes the determinant and $\Gamma$ is the well-known gamma function. Since $G = (\sqrt{Q})^{-1}$ is the generator matrix given by Lemma and all component ellipsotopes share the same dimension $n$, we can choose those for which det$(G)$ is smallest.

However, we need an easier-to-compute heuristic when multiple ellipsoids have nearly-identical volumes or when there are many high-dimensional ellipsoids for which computing the MVOE and volume of every $(i, j)$ pair is intractable. We find in practice that, when the longest axes of $E_1$ and $E_2$ are nearly perpendicular, the resulting MVOE is a more conservative overapproximation. Therefore, we pick $i$ and $j$ to find the pair of longest ellipsoid axes that are closest to parallel:

$$(i, j) = \arg \max_{i,j \in \mathbb{N}_r} \left| v_i^\top v_j \right|$$  

where $v_i$ (resp. $v_j$) is the unit vector in the direction of the longest semi-axis of $E_i$ (resp. $E_j$). That is, $v_i$ is the eigenvector of $Q_i^{-1}$ corresponding to its largest eigenvalue.

4) Identifying Component Ellipsotopes: In Section IV, we found that intersections between ellipsotopes, hypersurfaces, and halfspaces all introduce linear constraints. Strategies exist to conservatively simplify these linear constraints for constrained zonotopes [6], [18]. For 2-ellipsotopes, we can instead use the index set and constraints to identify component ellipsotopes.

Notice that all intersections introduce a new block row to the ellipsotope constraints (see Propositions 4, 5, and 6), while placing any existing constraints either block-diagonally (in the case of ellipsotope-ellipsotope intersection) or with zero-padding (for halfspace intersection). Furthermore, the ellipsotope’s index set contains the indices of the columns corresponding to the constraints that existed before the intersection procedure. Therefore, given an arbitrary ellipsotope, if we identify indices in the index set that correspond to a block-diagonal arrangement of linear constraints, then we can extract the component ellipsotopes and simplify them with Lemma 11.

To illustrate this idea with an example, consider an ellipsotope $E = \mathcal{E}_2(E, G, A, b, J)$ with $m$ generators. Suppose that $A = \text{diag}(A_1, A_2) \in \mathbb{R}^{2 \times m}$, $A_1 \in \mathbb{R}^{1 \times m}$, and $A_2 \in \mathbb{R}^{1 \times m}$. Also suppose $J = \{N_m, N_{m_y} + m_1\}$. Then

$$E = \mathcal{E}_2(E, G, A_1 \oplus b(N_{m_1}), \oplus A_2 \oplus b(N_{m_2})).$$  

(31)

In other words, we have broken $E$ into two component ellipsotopes, which we can then reduce as above.

B. General Strategies for Order Reduction

We now briefly discuss order reduction when $p \neq 2$. In short, strategies from the literature for zonotopes and constrained zonotopes still apply to ellipsotopes. We leave strategies that leverage the $p$-norm structure to future work.

1) Leveraging Component Zonotopes: We noted above that order reduction for an arbitrary 2-ellipsotope follows from treating it as a Minkowski sum of component ellipsoids. For a basic $p$-ellipsotope, we can adopt a similar strategy by considering component zonotopes.

First, notice that, by making a single generator’s $p$-norm constraint independent from all other generators, we overapproximate an ellipsotope. We call this popping a generator:

**Lemma 13** (Generator Popping). Consider the indexed ellipsotope $E = \mathcal{E}_p(c, G, J)$. Consider an arbitrary $J \in \mathbb{N}$ and suppose $j \in J$. Define $J = (J \setminus \{j\})$ and $\beta = (J \cup \{j\}) \cup \{J \setminus \{j\}\}$. Then $E \subset E$ where $E = \mathcal{E}_p(c, G, J)$.

**Proof.** For any feasible $\beta$, $\|\beta(J)\|_p \leq \|\beta(J)\|_p + |\beta(J)|$ by the triangle inequality. □

Then, a strategy for order reduction is as follows. Suppose $E \subset \mathbb{R}^n$ has $m$ generators, and we seek to remove $n_{\text{rdc}}$ of them. First, we pop the $n_{\text{rdc}} + n$ smallest (in the 2-norm) generators. Let $G = [G_{\text{keep}}, G_{\text{rdc}}]$ where $G_{\text{rdc}}$ contains these $n_{\text{rdc}} + n$ generators; note we can reorder $G$ in this way without loss of generality. Let $Z_{\text{rdc}} = \mathcal{E}_p(0, G_{\text{rdc}}; \{\{1\}, \{2\}, \cdots, \{n_{\text{rdc}}\}\}$, which is a zonotope by construction. If we pop the $G_{\text{rdc}}$ generators, then $E = \mathcal{E}_p(c, G_{\text{keep}}; A_{\text{keep}}) \oplus Z_{\text{rdc}}$, where $A_{\text{keep}}$ is the original index set with the indices corresponding to $G_{\text{rdc}}$ removed, and then reorganized to match $G_{\text{keep}}$. Finally, we can apply zonotope order reduction (e.g., from [2], [11].
Let \( E \) be represented as an ellipsotope with \( n \) generators and \( k \) constraints. Let \( \Gamma \subseteq E \). Then
\[
E \subseteq \tilde{E} = \mathcal{E}_p(c+Gb - \Gamma A - AA, b - Ab, \mathcal{J}).
\] (32)

Proof. Let \( x \in E \), so there exists \( b \in \mathbb{R}^m \) such that \( x = c + Gb \) and \( A\beta = b \). It then follows that \( x = c + G\beta + \Gamma(b - \Lambda \beta) \) and \( A\beta = b + \Lambda(b - \Lambda \beta) \).

By choosing \( \Lambda \) as a matrix of zeros with a single one on the diagonal, one can cause a row of \( [A, b] \) to become zeros, thereby eliminating a constraint and producing an overapproximation. Note that [6] presents a further strategy for eliminating a constraint and a generator by choosing both \( \Gamma \) and \( \Lambda \). We leave adapting this strategy to future work.

VI. EXAMPLES

We now demonstrate properties and uses of ellipsotopes; in particular, we illustrate fault detection, assess the speed of the emptiness check, verify collision-avoidance for robot path planning under uncertainty, and show order reduction. All examples are run on a desktop computer with a 6-core 3.6 GHz processor and 32 GB of RAM.

A. Fault Detection

We implement the set-based fault detection example from [6] Section 6], in which a nominal model is given and a faulty model is propagated. A set-based estimator is propagated using the nominal dynamics and a set-inclusion check is performed at each timestep. The goal is to detect the fault, i.e. discrepancy between the nominal and faulty model, in the least number of timesteps. Using ellipsotopes, we are able to detect the fault in average of 23.54 timesteps over 500 simulation runs, compared to 27.718 when using constrained zonotopes as demonstrated in [6]. By maintaining tighter set representations, ellipsotopes lead to fewer missed detections and lower time to detect. In addition, ellipsotopes allow for fault detection with ellipsoid-like sets. Running the same example while using ellipsoids results in the fault being failed to be detected, due to the overapproximation of ellipsoid intersection.

B. Emptiness Checking

We now evaluate how long it takes to check if an ellipsotope is empty using Cor. [8]. We apply Cor. [8] because we find in practice that solving the feasibility problem (24) is orders of magnitude faster than solving (22) from Prop. [7]. This speed-up is because there is often a continuum of optimal solutions to (24), but only one optimal solution to (22).

Our evaluation method is as follows. First, we generate 10 random 2-D 2-ellipsotopes for each \( m = 1, 2, \cdots, 20 \) generators (each generator of length no more than 1/m) and \( k = 1 \) linear constraint. Then, we set \( b = 0_{k \times 1} \) or \( b = 2m \cdot 1_{k \times 1} \) (which ensures emptiness). Finally, we solve (24) using MATLAB’s fmincon solver with an initial guess of \( b_0 = \Lambda b \), and compute the solve time with MATLAB’s timeit tool.

The results, summarized in Fig. 4, show that it takes on the order of \( 10^{-5} \) s to confirm that an ellipsotope is nonempty, whereas it takes on the order of \( 10^{-2} \) s to identify that an ellipsotope is empty. This is because, for a nonempty ellipsotope, the initial guess of \( \Lambda b \) is often a feasible solution to (24), so the solver can terminate on the first iteration. In either case, this experiment demonstrates that ellipsotopes enable fast emptiness checking on the order of \( 10^{-2} \) s with a naive MATLAB implementation.

C. Robot Path Verification

We now present a planar path verification example in which ellipsotopes are used to represent the reachable set of the combined volume of a robot’s body and its uncertainty in state. This example illustrates the practicality of the ellipsotope Minkowski sum, intersection, and emptiness check.

Fig. 4: Timing results for solving the ellipsotope emptiness check (24) as a function of the number of generators for random 2-dimensional 2-ellipsotope. The top (resp. bottom) subplot shows the emptiness check times for nonempty (resp. empty) ellipsotopes. The box-and-whisker plots represent the 25-75% interquartile range (box), median (line through box), min/max of data (whiskers), and outliers (plus signs). Empty ellipsotopes typically take more time because they require multiple iterations to solve (24) instead of terminating early upon finding a single feasible solution.
properties. To demonstrate that ellipsotopes can provide tighter bounding reachable sets than zonotopes or ellipsoids, we also compute the reachable sets for the same trajectory using both zonotopes and ellipsoids using the CORA toolbox.

1) System Dynamics and Measurements: We consider a robot with a box-shaped rigid body with width \( w_{\text{rob}} \) and length \( l_{\text{rob}} \) and represent it with the 2-ellipsotope:

\[
E_{\text{rob}} = E_2 \left( \gamma \times 1, \frac{1}{2} \text{diag}(w_{\text{rob}}, l_{\text{rob}}) \right), \quad \{ \{1\}, \{2\} \},
\]

We model the system with discrete-time, nonlinear dynamics and measurements. In particular we consider a Dubins car model with state \( x(t) = [x_1(t), x_2(t), \theta(t)]^\top \), input \( u(t) = [v(t), \omega(t)]^\top \) and center-of-mass equations of motion:

\[
\begin{align*}
    x_1(t) &= x_1(t-1) + v(t-1) \cos(\theta(t-1)) \Delta t + w_1(t) \quad (34a) \\
    x_2(t) &= x_2(t-1) + v(t-1) \sin(\theta(t-1)) \Delta t + w_2(t) \quad (34b) \\
    \theta(t) &= \theta(t-1) + \omega(t-1) \Delta t + w_3(t), \quad (34c)
\end{align*}
\]

where \( p(t) = [x_1(t), x_2(t)]^\top \) is the robot’s center-of-mass position and \( \theta(t) \) is its heading at time \( t \in \mathbb{N} \). The process noise is \( w(t) \sim \mathcal{N}(0, Q) \) where \( Q \in \mathbb{R}^{3 \times 3} \) and \( Q > 0 \). The control inputs are longitudinal speed \( v(t) \) and yaw rate \( \omega(t) \). Time is discretized by \( \Delta t = 0.1 \) s.

The robot’s measurements consist of 4 ranges to beacons placed at fixed, known locations, as well as a heading measurement, all with additive Gaussian noise. Range measurements that are taken when \( x_1(t) < 30 \) have noise variance of 0.4 m, while measurements taken when \( x_1(t) \geq 30 \) (shown shaded in light red in Fig. 6) have a higher variance of 10.0 m.

2) Reachability under Position Uncertainty: The robot tracks a nominal trajectory \((\hat{x}(1), \ldots, \hat{x}(N))\) with a linear state estimator and controller, as in [3] and [43]. At time \( t \) the state estimator provides an uncertain estimate of the robot’s state parameterized by the mean and covariance of the Gaussian distribution \( \mathcal{N}(\mu(t), \Sigma(t)) \). We assume the position and heading covariance are decoupled, such that we can decompose \( \mu(t) \) and \( \Sigma(t) \) into position and heading components:

\[
\mu(t) = \begin{bmatrix} \mu_p(t) \\ \mu_\theta(t) \end{bmatrix}, \quad \Sigma(t) = \text{diag}(\Sigma_p(t), \Sigma_\theta(t)).
\]

Now consider the \( \alpha \)-probability confidence level set of the robot’s uncertain position, \( E_{\text{unc}} \), for which \( P(p(t) \in E_{\text{unc}}) \geq \alpha \). We can express \( E_{\text{unc}} \) as the ellipse:

\[
E_{\text{unc}} = \{ x + \hat{p}(t) \mid x^\top (\varepsilon \Sigma_p(t))^{-1} x \leq 1 \},
\]

\[
\varepsilon = -2\log(1 - \alpha).
\]

Using Lemma 9 we represent this ellipse with the 2-ellipsotope: \( E_{\text{unc}}(t) = E_2(\hat{p}(t), (\varepsilon \Sigma_p(t))^{1/2}) \). Given some initial state estimation covariance \( \Sigma_0 \), we propagate state uncertainty along the nominal trajectory according to Equations (17)-(21) and (33)), and obtain the associated \( \alpha \)-confidence ellipses that enclose the center-of-mass trajectory of the robot, under uncertainty due to noisy dynamics and measurements, with probability \( \alpha \).

3) Handling Robot Body and Heading Uncertainty: To account for the robot’s body, we cannot simply Minkowski sum the \( E_{\text{rob}} \) ellipsotope with the \( E_{\text{unc}} \) ellipsotope, because we must account for heading uncertainty. We do so by first taking the \( \alpha \)-confidence interval, \( (\hat{\theta} - \Delta \theta, \hat{\theta} + \Delta \theta) \), of the distribution \( \mathcal{N}(\hat{\theta}, \Sigma_\theta) \) of heading \( \theta \) estimates. Next, to overbound the area swept out by the robot’s body over this range of angles, we create an ellipsotope as the intersection of the circumscribing circle of the robot’s body with four halfplanes, shown in Figure 5 Then, for each timestep of the trajectory, we Minkowski sum this ellipsotope with the center of mass confidence ellipse from position uncertainty propagation to obtain a reachable set that accounts for the robot’s body plus position and heading uncertainty.

4) Collision Checking and Area Approximation: For each of the 187 timesteps of the nominal trajectory, we compute the intersection between the reachable set and each obstacle. We then solve the emptiness check in Cor. 8 to assess if the reachable set is in collision. To collision check the comparison ellipsoid and zonotope reachable sets, we overapproximate the obstacles as ellipsoids and zonotopes respectively, allowing us to use the CORA intersection and emptiness check implementations for comparison.

We compute the total area of each 2-D reachable set to assess conservatism. For ellipsotopes, we approximate area by sampling points from the boundary, constructing a polygon from the sampled points, then computing the area of the polygon. For zonotopes and ellipsoids we use the CORA built-in functions for computing area.

5) Results and Discussion: The ellipsotope reachable set is computed in 52.13 ms and collision checked in 8.14 s. Given that 187 timesteps is a 18.7 s long trajectory, this shows that we can validate uncertain trajectories with ellipsotope reachable sets faster than real time. For comparison, we also time the collision check of the zonotope and ellipsoid reachable sets with the obstacles. The zonotope reachable set is collision checked in 3.16 s and the ellipsoid reachable set in 0.25 s. Thus, ellipsotopes require slightly
more computation but provide a more accurate reachable set representation.

For the sake of comparison, the ellipsoid, zonotope, and ellipsotope reachable sets are shown together in Figure 6. The zonotope reachable set has an area of 2893.26 m², the ellipsoids 2399.94 m², and the ellipsotopes 2274.73 m². Thus, ellipsotopes provide a tighter reachable set than zonotopes or ellipsoids, as we would expect.

This example illustrates how ellipsotopes can tightly represent reachable sets of systems with uncertainty and geometric shape. Furthermore, we can use intersections and emptiness checking (Properties 4 and 7) to efficiently perform collision checking of this reachable set with obstacles also represented by ellipsotopes. Also note, this example is an improvement over [3], since we exactly represent the confidence bounds of the uncertain position and heading states as ellipsotopes, instead of overapproximating the bounds with zonotopes.

Fig. 6: Comparison of reachable sets represented by zonotopes, ellipsoids, and ellipsotopes, each shown with a different color. Although the ellipsoid and ellipsotope reachable sets, the ellipsotopes are able to more tightly bound the robot’s body, thus resulting in an overall tighter reachable set.

D. Order Reduction

We provide two brief examples of order reduction.

1) Method for 2-Ellipsotopes: We demonstrate the order-reduction heuristic in (30) on an example with three 2-ellipsotopes. First, we create $E_1 = E_2 (0_{2\times1}, G_2)$ with

$$ G_1 = \begin{bmatrix} 3 & 2 \\ -1 & 0 \end{bmatrix}. $$

Then, we create $E_2$ by rotating $G_1$ by $\pi/6$ radians and $E_3$ by rotating $G_3$ by $\pi/2$ radians, as shown in Fig. 7.

Next, we consider the possible $(i,j)$ pairs of ellipsotopes to overapproximate with an MVOE. The heuristic values for the different pairs are shown in Tab. I. We maximize the heuristic by picking $(i,j) = (1,2)$ (i.e., finding the MVOE of $E_1 \oplus E_2$).

We assess the heuristic as follows. First, for each possible $(i,j)$ combination, we compute the ratio between the area of the reduced ellipsotope and the area of the exact Minkowski sum $E = E_1 \oplus E_2 \oplus E_3$. We approximate each ellipsotope’s area using the method discussed above in the path planning example. We similarly approximate the Hausdorff distance between each reduced ellipsotope and the exact Minkowski sum ellipsotope. For both the area ratio and Hausdorff distance, our heuristic’s chosen pair (1,2) is the lowest, meaning our outer-approximation is the least conservative. The results are summarized in Tab. I and the Minkowski sums of the different $(i,j)$ pairs are shown in Fig. 7.

![Fig. 7: Example of order reduction heuristic (30) for an ellipsotope constructed as the Minkowski sum of three 2-ellipsotopes (left subplot). The right subplot shows the exact Minkowski sum as a dashed line, the reduced ellipsotope chosen by the heuristic in light blue, and the other two possible reduced ellipsotopes in red. The heuristic result is the closest overapproximation to the exact result by the metrics in Tab. I.](image)

![Fig. 7: Example of order reduction heuristic (30) for an ellipsotope constructed as the Minkowski sum of three 2-ellipsotopes (left subplot). The right subplot shows the exact Minkowski sum as a dashed line, the reduced ellipsotope chosen by the heuristic in light blue, and the other two possible reduced ellipsotopes in red. The heuristic result is the closest overapproximation to the exact result by the metrics in Tab. I.](image)

Table: Order reduction heuristic results; we see that the heuristic from (30) for reducing 2-ellipsotopes produces the tightest overapproximation by replacing a Minkowski sum of ellipsoids with a single ellipsoid. The $(i,j)$ pairs index which of the three ellipsotopes in the left subplot of Fig. 7 are being Minkowski summed.

| $(i,j)$ | Heuristic | Area Ratio | Hausdorff Dist. | MVOE Area |
|--------|-----------|------------|----------------|-----------|
| (1,2)  | 0.87      | 1.05       | 0.40           | 49.07     |
| (1,3)  | 0.00      | 1.05       | 0.42           | 87.96     |
| (2,3)  | 0.50      | 1.16       | 1.05           | 77.28     |

TABLE I: Order reduction heuristic results; we see that the heuristic from (30) for reducing 2-ellipsotopes produces the tightest overapproximation by replacing a Minkowski sum of ellipsoids with a single ellipsoid. The $(i,j)$ pairs index which of the three ellipsotopes in the left subplot of Fig. 7 are being Minkowski summed.

2) General Method: We apply the technique from [6, Proposition 5] to eliminate a single constraint. We create a random 2-D 2-ellipsotope with 8 generators and 2 constraints. We remove each constraint separately to produce two different overapproximations as shown in Figure 8.

VII. Conclusion

This work introduced ellipsotopes, a novel set representation created by generalizing the $\infty$-norm that defines zonotopes and constrained zonotopes. We illustrated that this set representation is closed under the operations critical to reachability analysis and fault detection: affine transformations, Minkowski sum, and intersection. Since ellipsotopes can grow in complexity similar to zonotopes when used for...
reachability analysis, we discussed several order reduction strategies. We also demonstrated the various properties of ellipsotopes via numerical examples, and illustrated their importance via a literature comparison to other set representations. For future work, we intend to formalize a stochastic variant of ellipsotopes and discover more applications of these objects to tasks in reachability, fault detection, and navigation. We also intend to explore applications of ellipsotopes in neural network verification.

REFERENCES

[1] S. Kousik, S. Vaskov, F. Bu, M. Johnson-Roberson, and R. Vasudevan, “Bridging the gap between safety and real-time performance in receding-horizon trajectory design for mobile robots,” The International Journal of Robotics Research, vol. 39, no. 12, pp. 1419–1449, 2020.

[2] M. Althoff, “Reachability analysis and its application to the safety assessment of autonomous cars,” Ph.D. dissertation, Technische Universität München, 2010.

[3] A. Shetty and G. X. Gao, “Predicting State Uncertainty Bounds Using Non-Linear Stochastic Reachability Analysis for Urban GNSS-Based UAS Navigation,” IEEE Transactions on Intelligent Transportation Systems, 2020.

[4] M. Chen, S. Herbert, H. Hu, Y. Pu, J. F. Fisc, S. Bansal, S. Han, and C. J. Tomlin, “Fastrack: a modular framework for real-time motion planning and guaranteed safe tracking,” IEEE Transactions on Automatic Control, 2021.

[5] S. Bhamidipati and G. X. Gao, “Integrity-Driven Landmark Attentions for GPS-Vision Navigation via Stochastic Reachability,” in Proceedings of the 33rd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2020), 2020, pp. 2311–2326.

[6] J. K. Scott, D. M. Raimondo, G. R. Marseglia, and R. D. Braatz, “Constrained zonotopes: A novel set representation for reachability analysis,” IEEE Transactions on Automatic Control, vol. 69, pp. 126–136, 2016.

[7] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, “Control barrier function based quadratic programs for safety critical systems,” IEEE Transactions on Automatic Control, vol. 62, no. 8, pp. 3861–3876, 2016.

[8] N. Smit-Anseeuw, C. D. Remy, and R. Vasudevan, “Walking with confidence: Safety regulation for full order biped models,” IEEE Robotics and Automation Letters, vol. 4, no. 4, pp. 4177–4184, 2019.

[9] A. B. Kurzhanski and P. Varaiya, “Ellipsoidal techniques for reachability analysis” in International Workshop on Hybrid Systems: Computation and Control, Springer, 2000, pp. 202–214.

[10] A. A. Kurzhansky and P. Varaiya, “Ellipsoidal toolbox (ETY)” in Proceedings of the 45th IEEE Conference on Decision and Control, IEEE, 2006, pp. 1498–1503.

[11] C. Combastel, “A state bounding observer for uncertain non-linear continuous-time systems based on zonotopes,” in Proceedings of the 44th IEEE Conference on Decision and Control, IEEE, 2005, pp. 7228–7234.
[37] I. M. Mitchell, M. Chen, and M. Oishi, “Ensuring safety of nonlinear sampled data systems through reachability (extended version),” Technical Report TR-2012-01, Department of Computer Science, University of California, Berkeley, Tech. Rep., 2012.

[38] V. Gaßmann and M. Althoff, “Scalable Zonotope-Ellipsoid Conversions using the Euclidean Zonotope Norm” in 2020 American Control Conference (ACC), IEEE, 2020, pp. 4715–4721.

[39] S. Kousik, P. Holmes, and R. Vasudevan, “Safe, aggressive quadrotor flight via reachability-based trajectory design” in Dynamic Systems and Control Conference, American Society of Mechanical Engineers, vol. 59162, 2019, V003T19A010.

[40] V. Macagon and B. Wünsche, “Efficient collision detection for skeletonally animated models in interactive environments” Proceedings of ICVRNZ’03, pp. 378–383, 2003.

[41] C. Liu and M. Tomizuka, “Algorithmic safety measures for intelligent industrial co-robots” in 2016 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2016, pp. 3095–3102.

[42] M. Althoff, “An introduction to cora 2015,” in Proc. of the Workshop on Applied Verification for Continuous and Hybrid Systems, 2015.

[43] A. Bry and N. Roy, “Rapidly-exploring random belief trees for motion planning under uncertainty,” in 2011 IEEE international conference on robotics and automation, IEEE, 2011, pp. 723–730.

[44] U. Malik, I. M. Jaimoukha, G. D. Halikias, and S. Gungah, “On the gap between the quadratic integer programming problem and its semidefinite relaxation” Mathematical Programming, vol. 107, no. 3, pp. 505–515, 2006.

APPENDIX

A. Ellipsotope Visualization

Ellipsotopes in general can be difficult to visualize due to the exponential number of faces as a function of the number of generators. Ellipsotopes are further challenging because we are now concerned with plotting an affine image of, in the most general sense, the intersection of a hyperplane with the convex hull of these points is a visually-acceptable boundary of the ellipsotope, our strategy for visualization is to generate points on the boundary of the feasible generator coefficients, then map them through the generator matrix. While many of these points may not lay on the boundary of the ellipsotope, the convex hull of these points is a visually-acceptable approximation of the ellipsotope in practice.

To proceed, we first need the following lemma.

Lemma 15. Suppose $C \in \mathbb{R}^n$ is a compact, convex set. Let $M : \mathbb{R}^n \to \mathbb{R}^m$ with $m < n$ be a surjective linear map. Suppose $x \in \text{bd}(MC)$. Then there exists $y \in \text{bd}(C)$ such that $x = My$.

Proof. The preimage $M^{-1}x$ is a linear subspace of $\mathbb{R}^n$ that intersects $C$, and therefore intersects $\text{bd}(C)$. \hfill \square

Now, to approximate the boundary of a high-dimensional ball, suppose that $Y \subset \mathbb{R}^n$ is a finite set of sampled points with $|Y| = n_{\text{plot}} \in \mathbb{N}$. To map $Y \to \text{bd}(B_{p,n})$, let $\text{bdproj}_{p}(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$ for which

$$\text{bdproj}_{p}(y) = \frac{y_{(j)}}{\|y_{(j)}\|_p}. \quad (39)$$

Suppose $E = \mathcal{E}_{p}(c,G)$. Let $X = c + G \cdot \text{bdproj}_{p}(Y)$. Then, applying Lemma 15, $\text{CH}(X)$ approximates $E$, where $\text{CH}(\cdot)$ returns the convex hull of a set.

2) Ball Product and Affine Subspace Ray Tracing: Now we generalize the previous approach to generate points on the boundary of the intersection of the ball product $B_x(J)$ and the affine subspace $\mathcal{P}(A,b)$. Let $E = \mathcal{E}_{p}(c,G,A,b,J)$. To plot this ellipsotope, our goal is to first pick $n_{\text{plot}} \in \mathbb{N}$ coefficients $\beta_i, i = 1, \ldots, n_{\text{plot}}$, such that

$$\beta_i \in \text{bd}(B_x(J) \cap \mathcal{P}(A,b)). \quad (40)$$

In other words, these coefficients obey the constraints

$$\forall J \in \mathcal{J}, \quad \|\beta_i(J)\|_p \leq 1, \quad (41a)$$

$$\exists K \in \mathcal{K} \text{ s.t. } \|\beta_i(K)\|_p = 1, \quad \text{and} \quad (41b)$$

$$A\beta = b. \quad (41c)$$

We can then approximate the ellipsotope as

$$E \approx \text{CH} \left( c + G\beta_{i=1}^{n_{\text{plot}}} \right). \quad (42)$$

We generate these points by tracing rays outwards in $\mathbb{R}^m$ from a point inside $B_x(J)$ until they contact the boundary of $B_x(J) \cap \mathcal{P}(A,b)$.

To generate a single $\beta$, first let $u_i \in \text{ker}(A)$ be a random vector in the nullspace of $A$. Let $\beta_0 \in B_x(J)$, which can be found by applying Corollary 8 note that $\beta_0 = A^t b$ is often such a point. Let $J \in \mathcal{J}$. We then solve

$$\|\alpha_i u_i(J) + \beta_0(J)\|_p = 1 \quad (43)$$

for $\alpha_i \in \mathbb{R}$ and set $\beta_i = \alpha_i u_i + \beta_0$ as a point that is guaranteed to obey (43). Let $\varphi(\alpha_i, u_i, \beta_0, J) = \|\alpha_i u_i(J) + \beta_0(J)\|_p^p - 1^p$. Notice that, for any $J \in \mathcal{J}$,

$$\varphi(\alpha_i, u_i, \beta_0, J) = \sum_{j \in J} (\alpha_i u_i(j)^p + (\beta_0(j))^p) - 1 \quad (44)$$

$$= \binom{p}{0} \alpha_i^p \sum_{j \in J} (u_i(j))^p (\beta_0(j))^0 + \binom{p}{1} \alpha_i^{p-1} \sum_{j \in J} (u_i(j))^{p-1} (\beta_0(j))^1 + \cdots + \binom{p}{p} \alpha_i^0 \sum_{j \in J} (u_i(j))^0 (\beta_0(j))^p - 1, \quad (45)$$

where we have applied the binomial theorem to expand the coefficients. Since $\varphi(\cdot,u_i,t,J)$ is a univariate polynomial in $\alpha_i$, we can solve for $\alpha_i$ efficiently. Critically, since $\beta_0 \in B_x(J)$, the direction $u_i$ points “outward” towards the boundary, so the smallest solution $\alpha_i$ is a point on the boundary; in other words, $\beta_i$ obeys (41).

3) Ray Tracing: We find in practice that plotting 2-D ellipsotopes with more than 5 generators with the above methods is computationally expensive, taking several seconds to generate a single plot. Furthermore, the above methods result in many unused points (that is, points on the boundary of the feasible set that are mapped to the interior of the ellipsotope, and therefore not used for plotting). To address...
this, we pose a convex program to identify points on the boundary of the ellipsotope directly in its workspace. In particular, we maximize the length of a ray extending from a point in the ellipsotope in an arbitrary direction while constraining it to lie within the ellipsotope.

We set up for ray tracing as follows. Let \( E = E_p(c,G,A,B,\mathcal{J}) \subset \mathbb{R}^n \) be an ellipsotope with \( m \) generators. Consider a ray

\[
\mathcal{R}(x,g) = \{ x + \lambda g \mid \lambda \geq 0 \} \subset \mathbb{R}^n,
\]

where \( g \in \mathbb{R}^n \) is arbitrary and \( x \in E \). We find \( x \) as any feasible point in the ellipsotope by applying the strategy above in Appendix A.2 to find a feasible coefficient \( \beta \), then setting \( x = c + G\beta \). Note, we cannot always set \( x = c \), because it is possible that \( c \notin E \), which occurs when \( 0 \notin \mathcal{P}(A,b) \).

Finally, to perform ray tracing, we solve

\[
\begin{align*}
\max_{\lambda \geq 0, \ \beta \in \mathbb{R}^m} & \lambda \\
\text{s.t.} & \|\beta(J)\|_p \leq 1 \ \forall \ J \in \mathcal{J}, \\
& A\beta = b, \text{ and} \\
& c + G\beta = x + \lambda g,
\end{align*}
\]

which is convex and always feasible if \( E \neq \emptyset \). By solving (47) for a variety of \( g \), we can sample the boundary of the ellipsotope. In practice, we sample \( g \) uniformly from the boundary of the 2-D or 3-D unit sphere and solve (47) once for each sample.

B. Minimum Volume Outer Ellipsoids

We use the following methods to compute minimum volume outer ellipsoids (MVOEs) for the Minkowski sum of ellipsoids and for zonotopes. These objects are useful for order reduction as discussed in Section V.

1) MVOE of Ellipsoid Minkowski Sum: We apply the method in [14, Thm. 1]. Consider the pair of ellipsoids \( E_1 = \mathcal{E}(c_1,Q_1) \) and \( E_2 = \mathcal{E}(c_2,Q_2) \) in \( \mathbb{R}^n \). Let \( \lambda = \text{eig}(Q_1Q_2^{-1}) \in \mathbb{R}^n \). Let \( \lambda_0 \in \mathbb{R} \) and consider the fixed-point iteration

\[
\zeta_{n+1} = \left( \frac{\sum_{i=1}^{n} \frac{1}{1+\zeta_0 \lambda(i)}}{\sum_{i=1}^{n} \frac{1}{1+\lambda(i)}} \right)^{\frac{1}{2}}.
\]

Define \( \zeta \) as the limit of (48) as \( n \to \infty \). Then

\[
E_1 \oplus E_2 \subseteq \mathcal{E}(c_1+c_2, Q_{\oplus}),
\]

where

\[
Q_{\oplus} = (1 + \frac{1}{\zeta}Q_1^{-1} + (1 + \zeta)Q_2^{-1})^{-1}.
\]

2) Overapproximating the MVOE of a Zonotope: We apply the method in [38, Thm. 1]. Let \( Z = Z(c,G) \subset \mathbb{R}^n \) be a zonotope with \( m \in \mathbb{N} \) generators. To overapproximate the MVOE, we first solve an SDP [38, Lem. 3]:

\[
\begin{align*}
r &= \min_{\lambda \geq 0, \ \lambda \in \mathbb{R}^m} & \frac{1}{m} \lambda^\top \lambda \\
\text{s.t.} & \text{diag}(\lambda) - G_0^\top G_0 \succeq 0,
\end{align*}
\]

where

\[
G_0 = E_0^{-\frac{1}{2}} \quad \text{and} \quad E_0 = mGG^\top.
\]

Then an outer approximation of the MVOE is given by

\[
Z \subset \mathcal{E}(c,rE_0).
\]

The MVOE approximation can be made tighter by applying [44, Lem. 3] in the case when \( n \approx m \).