Robust Constrained Control: The Polytopic Appraoch

Sadra Sadraddini sadra@mit.edu

In this lecture we overview a set of techniques to control (time-varying, possibly hybrid) linear systems with polytopic constraints and additive polytopic uncertainties. The central technique is characterizing the way polytopic uncertainties evolve through linear dynamics, and how one can ensure the constraints are satisfied by all the possible trajectories of the system - all points in the polytopes. We both consider i) finite-time trajectory synthesis, and ii) infinite-time invariance properties by focusing on polytopic robust control invariant sets for both time invariant systems and systems of periodic nature. We explain how one cast all computations based on linear/quadratic programming.

1 Introduction

Consider time-varying discrete-time systems of the following form:

$$x_{t+1} = f_t(x_t, u_t) + w_t \tag{1}$$

where $x_t \in \mathbb{R}^n$ is the state, $u_t \in \mathbb{R}^m$ is the control, $w_t \in \mathbb{R}^n$ is the additive external input (disturbance), and f_t is the state update rule at time $t, t \in \mathbb{N}$. The value of w_t is uncertain, but we assume that it is restricted to a given set $\mathbb{W}_t$. The system is fully observable. At time t, we have access to state $x_0, x_1, \cdots, x_t$ and the history of control inputs $u_0, u_1, \cdots, u_{t-1}$. We also know $f_t, t \in \mathbb{N}$. Therefore, one can compute $w_0, \cdots, w_{t-1}$ at time t. The problem, informally, is to design a control law that maps the history of states and controls to a control input such that the closed-loop trajectory satisfies certain constraints, no matter how disturbances hit the system. Let us formalize this problem.

A *trajectory* is defined as a timed sequence of tuples of visited states and controls:

$$\xi := (x_0, u_0), (x_1, u_1) \cdots .$$

Let Ξ denote the set of all trajectories. We are given $\Psi \subset \Xi$ as the set of *acceptable* trajectories. Consider the following examples as how one can characterize Ψ:

1. (Finite Time Reachability) a trajectory is acceptable if $x_t \in \mathbb{X}$, $\mathbb{X} \subset \mathbb{R}^n$, and $u_t \in \mathbb{U}$, $\mathbb{U} \subset \mathbb{R}^m$ for all $t \in \{0, 1, \cdots, T-1\}$ and $x_T \in \mathbb{X}_G$, where $\mathbb{X}_G \subset \mathbb{R}^n$ is the goal region.
2. (Invariance) a trajectory is acceptable if $x_t \in \mathbb{X}$, $\mathbb{X} \subset \mathbb{R}^n$, and $u_t \in \mathbb{U}$, $\mathbb{U} \subset \mathbb{R}^m$ for all $t \in \mathbb{N}$.
3. (Prescribed Regions) a trajectory is acceptable if $(x_t, u_t) \in \mathbb{H}_t$, where $\mathbb{H}_t \subset \mathbb{R}^{n+m}, t \in \{0, 1, \cdots, T\}$ are given regions in the joint state-control space.

Instead of searching for the schedule of control inputs in an open-loop manner, we want to search over control policies to incorporate feedback. A control policy Π is a set a of functions $\pi_t, t \in \mathbb{N}$, where $u_t = \pi_t(x_0, x_1, \cdots, x_t; u_0, \cdots, u_{t-1})$. The main question is to how to deal with uncertainties. There is often no way to confine the system to produce a single trajectory. Given an initial state $x_{\text{init.}} \in \mathbb{R}^n$ and a control policy Π, one can simulate the system with random allowable disturbances. The set of all allowable trajectories by the disturbances is denoted as follows:

$$\zeta(x_{\text{init.}}, \Pi) := \{\xi \in \Xi | \exists w_t \in \mathbb{W}_t, t \in \mathbb{N}, \text{ such that } x_{t+1} = f_t(x_t, u_t) + w_t, x_0 = x_{\text{init.}}\}. \tag{2}$$

Problem 1. Given system (1) and initial condition $x_{\text{init.}} \in \mathbb{R}^n$, find Π such that $\zeta(x_{\text{init.}}, \Pi) \subseteq \Psi$ - all the possible executions of the system are acceptable.

We have expressed Problem 1 in its most basic form. It is possible to ask for more. For example, an interesting problem is to find the (complete) set of initial conditions from which exists some Π that all trajectories are acceptable. Moreover, since Π is not often unique, we may be interested in finding the (sub)optimal one subject to a given cost function.

Linear Systems. In this lecture we focus on linear forms of f_t. Thus we have:

$$x_{t+1} = A_t x_t + B_t u_t + w_t. \tag{3}$$

We assume that $\mathbb{W}_t$ is a polytope. Also, we assume that Ψ is defined by linear constraints. As an extension, if there is a finite number of $(A_t, B_t, \mathbb{W}_t)$ tuples and each is valid in one polyhedral region of the joint state-control space, one can encode the dependence of (A_t, B_t, w_t) on (x_t, u_t) using mixed-integer linear constraints. We do not cover this topic in this lecture but we note that it is possible to combine mixed-integer encoding for hybrid planning presented in previous lectures with polytopic uncertainties here.

2 Deterministic Systems

First, let us focus on deterministic systems. Let the disturbance set be a singleton $\mathbb{W}_t = \{c_t\}$, $c_t \in \mathbb{R}^n, t \in \mathbb{N}$. Even though $\zeta(x_{\text{init.}}, \Pi)$ becomes a singleton, we still want to understand how a polytopic set is propagated through affine dynamics:

$$x_{t+1} = A_t x_t + B_t u_t + c_t. \tag{4}$$

Let $\mathbb{P}_t \subset \mathbb{R}^n$ be the set of states at time t. We parameterize $\mathbb{P}_t$ as an affine transformation of a given *base polytope* $\mathbb{Q} \subset \mathbb{R}^q$:

$$\mathbb{P}_t = \bar{x}_t + G_t \mathbb{Q}, \tag{5}$$

where $\bar{x}_t \subset \mathbb{R}^n$ and $G_t \in \mathbb{R}^{n \times n_q}$ - we interpret $G_t\mathbb{Q}$ as $\{G_t q | q \in \mathbb{Q}\}$. The $\mathbb{Q}$ and its dimensions n_q are chosen by the user. A popular choice is $\mathbb{Q} = [-1, 1]^{n_q}$ - a hyper-cube. Affine transformations of hyper-cubes are known as **zonotopes** and are very popular in set theoretic analysis in control theory.

Definition 1. A zonotope is defined as $\mathcal{Z}(\bar{x}, G) := \bar{x} + G[-1, 1]^{n_q}$, where n_q is the number of columns in G.

Increasing n_q makes the parameterization more flexible but increases the computational complexity. Note the richness and limitations of zonotopes in Figure. 2.

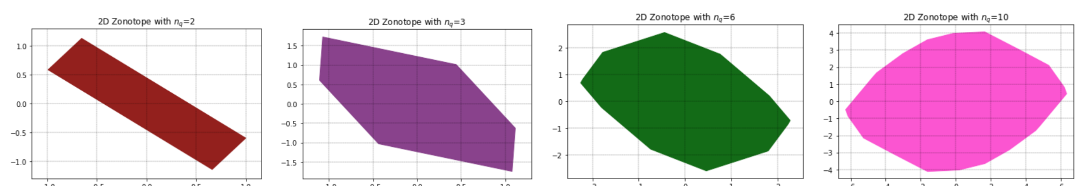

Figure 1: Some 2D zonotopes $\mathcal{Z}(0, G)$ with various values of n_q, the number of columns in G.

Now we parameterize the control policy. Consider the following control law:

$$x_t = \bar{x}_t + G_t q(x_t), \tag{6a}$$

$$u_t = \bar{u}_t + \theta_t q(x_t), \tag{6b}$$

where $q(x_t) \in \mathbb{Q}$ is an implicit variable depending on x_t, and $\bar{u}_t \in \mathbb{R}^m, \theta_t \in \mathbb{R}^{m \times q}$ are design parameters. Given x_t, one can solve a linear program (with any cost) to find $q(x_t)$ from (6a), and then plug it in (6b) to obtain the value for u_t. If matrix G_t has a left-inverse (which often it is **not** the case when $n_q > n$), then we simply obtain affine state-feedback law $u_t = \bar{u}_t + \theta_t G_t^\dagger (x_t - \bar{x}_t)$, where $G_t^\dagger G_t = I$.

The set of all possible control inputs at time t is

$$\mathbb{L}_t = \bar{u}_t + \theta_t \mathbb{Q}. \tag{7}$$

Plugging (6) in (4) yields the following:

$$\mathbb{P}_{t+1} = A_t \bar{x}_t + B_t \bar{u}_t + c_t + (A_t G_t + B_t \theta_t)\mathbb{Q} \tag{8}$$

or, equivalently:

$$\bar{x}_{t+1} = A_t \bar{x}_t + B_t \bar{u}_t + c_t, \tag{9a}$$

$$G_{t+1} = A_t G_t + B_t \theta_t. \tag{9b}$$

These are linear equations. We have successfully characterized the evolution of polytopic sets using linear constraints. What remains is the problem specification. For now, let assume we are given prescribed regions in the joint state-control space $\mathbb{H}_\tau \subset \mathbb{R}^{n+m}, \tau = 0, 1, \cdots, T-1$ and a goal region $\mathbb{X}_G \subset \mathbb{R}^n$, where T is the length of the trajectory we wish to synthesize. Now we can find all the parameters using the following optimization problem:

$$\{\bar{x}_\tau, \bar{u}_\tau, G_\tau, \theta_\tau\}_{\tau \in \{0,1,\cdots,T-1\}} = \begin{array}{ll} \arg\min & J \\ \text{subj. to} & (9), \mathbb{P}_\tau \times \mathbb{L}_\tau \subseteq \mathbb{H}_\tau, \tau \in \{0, 1, T-1\}, \mathbb{P}_T \subseteq \mathbb{X}_G, \end{array} \tag{10}$$

where J is a cost function. The subset constraints in (10) are polytope containment constraints, for which there exists linear encodings (see Appendix). Special care must be devoted to the design of J. It may consist of a performance criterion for $\bar{x}_t, \bar{u}_t$, but must also consist of a term that promotes large volumes for $\mathbb{P}_t$. Otherwise, zero values for G_t and θ_t would be obtained. A good heuristic is maximizing the singular values of G_t. When J is a linear cost (such as L_1 or L_∞ norm), then (10) becomes a linear program. For quadratic J, (10) becomes a quadratic program. Both forms are within the range of problems for which there exists very powerful software.

Remark 1 ([GKM06]). One may wonder why instead of the implicit control law in (6), we do not not parameterize the control law simply as an affine state feedback $u_t = \bar{u}_t + K_t x_t$, and search for $\bar{u}_t$ and K_t. As noted in Proposition 3 in [GKM06], the set of $\bar{u}_t, K_t$ that satisfy the constraints in (10) is non-convex. This fact is not surprising as one can check that propagating polytopes using such a law leads to multiplication of feedback gains.

3 Non-Deterministic Systems

Now we consider the case where we have additive disturbances:

$$x_{t+1} = A_t x_t + B_t u_t + w_t, w_t \in \mathbb{W}_t. \tag{11}$$

The propagation of polytopes is very similar to the deterministic case, with the the key difference that disturbances are added in a Minkowski sum sense. For the remainder of this note, we assume that $\mathbb{W}_t = \mathcal{Z}(\bar{w}_t, W_t)$, $W_t \in \mathbb{R}^{n \times n_w}, t \in \mathbb{N}$ - all disturbance sets are zonotopes. It is possible to consider non-zonotope additive disturbances, but we stick to zonotopes for the ease of notation here.

Similar to the deterministic case, consider the set of states at time t be represented by the following zonotope:

$$\mathbb{P}_t = \mathcal{Z}(\bar{x}_t, G_t). \tag{12}$$

Using the control law in (6), we characterize $\mathbb{P}_{t+1}$:

$$\mathbb{P}_{t+1} = \mathcal{Z}(A_t \bar{x}_t + B_t \bar{u}_t, A_t G_t + B_t \theta_t) \oplus \mathcal{Z}(\bar{w}_t, W_t). \tag{13}$$

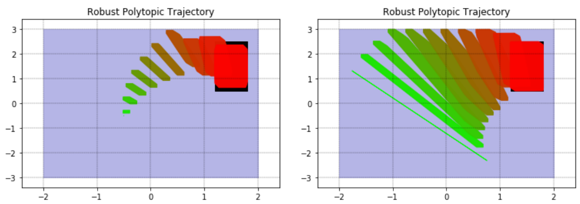

Figure 2: A robust trajectory synthesis for disturbed double integrator. The goal region is shown in black and the set of admissible states is shown in purple. Time progression of polytopes is colored from green to red. Two trajectories are shown. Note the polytopes respect the state constraints and the final zonotope is contained within the goal.

Using the properties of Minkowski sums for zonotopes (see Appendix), we can write (13) as:

$$\mathbb{P}_{t+1} = \mathcal{Z}(A_t\bar{x}_t + B_t\bar{u}_t + \bar{w}_t, (A_tG_t + B_t\theta_t, W_t)), \tag{14}$$

where $(.,.)$ denotes stacking matrices horizontally. The equivalent of (9) for (14) is the following:

$$\bar{x}_{t+1} = A_t\bar{x}_t + B_t\bar{u}_t + \bar{w}_t, \tag{15a}$$

$$G_{t+1} = (A_tG_t + B_t\theta_t, W_t). \tag{15b}$$

Note that the number of columns in G **increases with time**. This leads to more complex zonotopes. However, the encoding is still linear and we can cast the trajectory optimization problem similar to (10):

$$\{\bar{x}_\tau, \bar{u}_\tau, G_\tau, \theta_\tau\}_{\tau\in\{0,1,\cdots,T-1\}} = \begin{array}{ll} \arg\min & J, \\ \text{subj. to} & (15), \mathbb{P}_\tau \times \mathbb{L}_\tau \subseteq \mathbb{H}_\tau, \tau \in \{0, 1, T-1\}, \mathbb{P}_T \subseteq \mathbb{X}_G, \end{array} \tag{16}$$

where the cost function is often expressed over $\bar{x}_t, \bar{u}_t$ (characterizing the undisturbed trajectory). Unlike (10), we often obtain non-zero values for G_t and θ_t as disturbances are present and it is impossible to obtain a singleton trajectory.

Remark 2. In problems where proving worst-case guarantees for optimality is essential, a suitable framework instead of (16) is minimizing the maximum cost allowed by disturbances. This class of synthesis problems are known as *min-max* problems [BBM03] and are computationally very expensive, and practically very conservative.

Remark 3. In (16), we can introduce scalar variables α_t and replace W_t by $\alpha_t W_t$. Then we can maximize the summation of $\alpha_t, t = 0, 1, \cdots, T-1$, or use a similar cost criterion. This heuristic is meaningful: we search for a trajectory and a feedback policy that can tolerate a notion of the largest "magnitude" of additive disturbances and still satisfy the constraints.

4 Robust Controlled Invariance

Now we shift our focus to infinite-time properties. A core problem is designing a control policy that guarantees that a disturbed system remains in a designated set for all times, no matter how disturbances are acted. This problem is known as *robust controlled invariance* and has been extensively studied in the literature. A good starting reference is [Bla99].

4.1 RCI for Time Invariant Systems

Definition 2. Given a discrete-time dynamical system $x^+ = f(x, u, w), u \in U$, the set $\Omega \subset \mathbb{R}^n$ is a *Robust Controlled Invariant* (RCI) set if

$$\forall x \in \Omega, \exists u \in \mathbb{U}, \text{ such that } f(x, u, w) \in \Omega, \forall w \in \mathbb{W}. \tag{17}$$

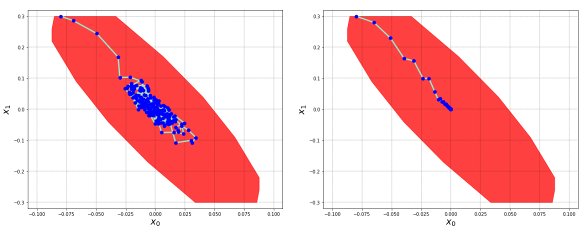

Figure 3: A Robust Control Invariant (RCI) set for a disturbed linear dynamics. Left: disturbed system under the set-invariance control law. Right: the state of undisturbed system under the same control law reaches the origin in finite time.

Once an RCI set is available, extracting the controller is straightforward - we can basically implement (6). We are interested in finding a RCI set Ω such that $\Omega \subset \mathbb{X}$. If this problem is solved, we have obtained a controller that guarantees the state remains within $\mathbb{X}$ for all times using control inputs from $\mathbb{U}$, while assuming disturbances take values from $\mathbb{W}$.

Computation of an RCI set and the associated control law is a non-trivial problem, and often very hard. Even for linear systems, the standard technique required to compute the maximal RCI set (which exists and is unique) suffers from severe computational complexity - the interested reader is encouraged to check the thesis on this subject [Ker00]. Here we present one of the many techniques to efficiently compute RCI sets for time-invariant linear systems. The method is closely related to the one in [RKMK07].

Let the RCI set be described by the zonotope $\Omega = \mathcal{Z}(\bar{x}, G)$. We use the control law in the form of (6). The set of states that are one-step-reachable from Ω is:

$$\Omega^+ = \mathcal{Z}\big(A\bar{x} + B\bar{u} + \bar{w}, (AG + B\theta, W)\big) \tag{18}$$

The central idea is imposing $\Omega^+ = \Omega$. However, the number of columns in the generator matrix of Ω^+ is more than those of G (because of additive disturbances). A solution is to get over this issue is imposing the following constraint:

$$A\bar{x} + B\bar{u} + \bar{w} = \bar{x}, \tag{19a}$$

$$(AG + B\theta, W) = (0, G), \tag{19b}$$

where the number of columns of the zero matrix is equal to the number of columns in W. The constraints in (19) are linear and if a feasible $G, \theta, \bar{x}, \bar{u}$ exists, a RCI set is obtained. The user has only need to pick the number of columns in G (n_q). The standard way is to start with $n_q = n$ and implement $n_q \leftarrow n_q + 1$ until feasibility or a termination criterion is reached.

There are interesting properties of RCI sets obtained from (19). For example, one can show that all undisturbed trajectories reach $\bar{x}$ in finite time, and a polytopic Lyapunov function exists. Moreover, it can be shown that if $(AG + B\theta, W) = (0, G)$ is replaced by $(AG + B\theta, W) = (\alpha E, G)$, where $\mathcal{Z}(0, E) \subseteq \mathcal{Z}(0, W)$ and $\alpha \in [0, 1)$, then $\mathcal{Z}\big(\bar{x}, (1-\alpha)^{-1} G\big)$ is an RCI set. The proof is left to the interested reader as an exercise.

4.2 RCI Sets for Periodic Systems

An extended version of (19) can be obtained for systems of periodic nature, which are encountered a lot in robot locomotion. Consider the system in (3) to be T-periodic:

$$A_{t+T} = A_t, B_{t+T} = B_t, \mathbb{W}_{t+T} = \mathbb{W}_t. \tag{20}$$

Now we write (15) and instead of (19), we impose the following constraint:

$$\bar{x}_T = \bar{x}_0, \tag{21a}$$

$$G_T = (0, G_0). \tag{21b}$$

With the linear encoding in (21), we obtain a periodic polytopic trajectory. All trajectories starting from $\mathbb{P}_0$ are back to $\mathbb{P}_0$ in T-steps.

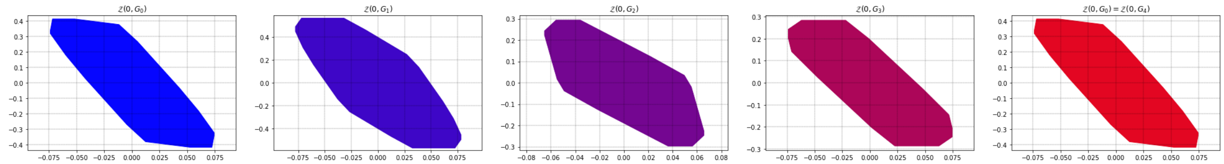

Figure 4: A Robust Control Invariant (RCI) set for a 4-periodic disturbed linear system. Note that the first and final zonotopes are the same.

References

[Alt15] Matthias Althoff. On computing the minkowski difference of zonotopes. *arXiv preprint arXiv:1512.02794*, 2015.

[BBM03] Alberto Bemporad, Francesco Borrelli, and Manfred Morari. Min-max control of constrained uncertain discrete-time linear systems. *IEEE Transactions on automatic control*, 48(9):1600–1606, 2003.

[Bla99] F. Blanchini. Set invariance in control–a survey. *Automatica*, 35(11):1747–1767, 1999.

[GKM06] Paul J Goulart, Eric C Kerrigan, and Jan M Maciejowski. Optimization over state feedback policies for robust control with constraints. *Automatica*, 42(4):523–533, 2006.

[Ker00] Eric C Kerrigan. *Robust Constraint Satisfaction: Invariant Sets and Predictive Control*. PhD thesis, University of Cambridge, Department of Engineering, 2000.

[RB10] Saša V Rakovic and Miroslav Baric. Parameterized robust control invariant sets for linear systems: Theoretical advances and computational remarks. *IEEE Transactions on Automatic Control*, 55(7):1599–1614, 2010.

[RKMK07] SV Raković, Eric C Kerrigan, David Q Mayne, and Konstantinos I Kouramas. Optimized robust control invariance for linear discrete-time systems: Theoretical foundations. *Automatica*, 43(5):831–841, 2007.

[ST19] Sadra Sadraddini and Russ Tedrake. Linear encodings for polytope containment problems. *arXiv preprint arXiv:1903.05214*, 2019.

[Zie12] Günter M Ziegler. *Lectures on polytopes*, volume 152. Springer Science & Business Media, 2012.

5 appendix

5.1 Minkowski Sums

Given two sets $\mathbb{S}_1, \mathbb{S}_2 \subset \mathbb{R}^n$, their Minkowski sum is denoted by

$$\mathbb{S}_1 \oplus \mathbb{S}_2 = \{s_1 + s_2 | s_1 \in \mathbb{S}_1, s_2 \in \mathbb{S}_2\}.$$

Given $s \in \mathbb{R}^n$, $s + \mathbb{S}$ is interpreted as $\{s\} \oplus \mathbb{S}$. Given sets $\mathbb{S}_i \subset \mathbb{R}^n, i = 1, \cdots, N$, their convex hull is defined as:

$$\text{Convexhull}(\{\mathbb{S}_i\}_{i=1,\cdots,N}) = \bigoplus_{i=1}^{N} \lambda_i \mathbb{S}_i,$$

where $\lambda_i \geq 0, i = 1, \cdots, N$, and $\sum_{i=1}^{N} \lambda_i = 1$.

5.2 Polytopes

Notation Given matrices A_1, A_2 of appropriate dimensions, we use $[A_1, A_2]$, (A_1, A_2), and $\text{blk}(A_1, A_2)$ to denote the matrices obtained by stacking A_1 and A_2 vertically, horizontally, and block-diagonally, respectively.

Definition 3. [Zie12] An H-*polyhedron* $\mathbb{P} \subset \mathbb{R}^n$ is a set that is represented as the intersection of a finite number of closed half-spaces in the form $\mathbb{P} = \{x \in \mathbb{R}^n | Hx \leq h\}$, where $H \in \mathbb{R}^{n_H \times n}, h \in \mathbb{R}^{n_H}$ define the hyperplanes.

Definition 4. A bounded H-polyhedron is called a *H-polytope*.

A polytope is a set that can be represented as a H-polytope. Polytopes are closed under affine transformations, Minkowski sums, convex hulls, and intersections. Another popular representation for polytopes is by its vertices.

Definition 5. A *V-polytope* is defined as the convex hull of its vertices: $\mathbb{P} = \text{Convexhull}(\{v_1, v_2, \cdots, v_N\}$.

The conversion between V-polytopes and H-polytope is always possible, but it is often computationally prohibitive. Often, the number of vertices N grows exponentially with the number of hyperplanes n_H in high dimensions. For example, the unit box in $\mathbb{R}^n$ has $n_H = 2n$ hyperplanes (two faces on each axis), but $N = 2^n$ vertices.

Definition 6. [ST19] An *AH-polytope* $\mathbb{X} \subset \mathbb{R}^n$ is a polytope that is given as an affine transformation of an H-polytope $\mathbb{P} \subset \mathbb{R}^m$, $\mathbb{X} = \bar{x} + G\mathbb{P}$, where $G \in \mathbb{R}^{n \times m}, \bar{x} \in \mathbb{R}^n$.

Converting an AH-polytope to its equivalent H-polytope may have an exponential complexity. The special case in which conversion is simple is when G has a left inverse, in which case we have:

$$\{\bar{x} + Gx | Hx \leq h\} = \{y \in \mathbb{R}^n | HG^\dagger y \leq h + HG^\dagger \bar{x}\}, G^\dagger G = I. \tag{22}$$

Given two H-polytopes $\mathbb{P}_i = \{x \in \mathbb{R}^n | H_i x \leq h_i\}, i = 1, 2$, their intersection can be easily represented as the following H-polytope:

$$\mathbb{P}_1 \cap \mathbb{P}_2 = \{x \in \mathbb{R}^n | [H_1, H_2]x \leq [h_1, h_2]\}.$$

However, the H-polytope form of $\mathbb{P}_1 \oplus \mathbb{P}_2$ is not easy to obtain. Unlike H-polytopes, AH-polytopes are suitable for affine transformations and Minkowski sums, while the case of intersections is less trivial but still possible [ST19]. Let $\mathbb{X}_i = \bar{x}_i + G_i \mathbb{P}_i, \mathbb{P}_i = \{z \in \mathbb{R}^{n_i} | H_i z \leq h_i\}, i = 1, 2, G_i \in \mathbb{R}^{n \times n_i}, \bar{x}_i \in \mathbb{R}^n$, be two AH-polytopes.

- (Affine transformation) Given $f \in \mathbb{R}^q, F \in \mathbb{R}^{q \times n}$, we have:

$$F(\bar{x} + G\mathbb{P}) + f = (F\bar{x} + g) + FG\mathbb{P}. \tag{23}$$

- (Minkowski Sum) We have the following relation:

$$(\bar{x}_1 + G_1\mathbb{P}_1) \oplus (\bar{x}_2 + G_2\mathbb{P}_2) = \bar{x}_1 + \bar{x}_2 + (G_1, G_2)\mathbb{P}_{\oplus}, \tag{24}$$

  where

$$\mathbb{P}_{\oplus} = \{z \in \mathbb{R}^{n_1+n_2} |\, \mathrm{blk}(H_1, H_2)z \leq [h_1, h_2]\}.$$

Zonotopes are a special case of AH-polytopes. An appealing feature of zonotopes is its operational convenience with Minkowski sums:

$$\mathcal{Z}(\bar{x}_1, G_1) \oplus \mathcal{Z}(\bar{x}_2, G_2) = \mathcal{Z}\left(\bar{x}_1 + \bar{x}_2, (G_1, G_2)\right). \tag{25}$$

Finding the H-polytope version of a zonotope requires facet enumeration, which its worst-case complexity is exponential in n and n_x [Alt15], the number of rows and columns of the generator, respectively.

5.3 Polytope Containment Encoding

Theorem 1. Let $\mathbb{X} = \{x \in \mathbb{R}^m | H_x x \leq h_x\}, \mathbb{Y} = \{y \in \mathbb{R}^n | H_y y \leq h_y\} \subset \mathbb{R}^n$, $H_x \in \mathbb{R}^{q_x \times n}, H_y \in \mathbb{R}^{q_y \times n}$ be polytopes. Then $\bar{x} + G\mathbb{X} \subseteq \mathbb{Y}$, $G \in \mathbb{R}^{n \times m}, \bar{x} \in \mathbb{R}^n$, if and only if

$$\exists \Lambda \in \mathbb{R}_{\geq 0}^{q_y \times q_x} \text{ such that } \Lambda H_x = H_y G, \Lambda h_x \leq h_y - H_y \bar{x}. \tag{26}$$

Proof. The conditions in (26) is equivalent to $\bar{x} + G\mathbb{X}$ being contained within each closed half-space of the hyperplanes in $\mathbb{Y}$. This condition is verified by checking q_y inequalities:

$$\max_{x \in \mathbb{X}} H_{y,i}(\bar{x} + Gx) \quad \leq \quad h_{y,i}, i = 1, \cdots, q_y, \tag{27}$$

where $H_{y,i}$ is the i'th row of H_y (the same notation applies to h_y). By writing the dual of the left hand side in (27), we arrive at

$$H_{y,i}\bar{x} + \min_{u_i \in \mathbb{R}_{\geq 0}^{q_x}, u_i' H_x = H_{y,i} G} u_i' h_x \quad \leq \quad h_{y,i}, i = 1, \cdots, q_y, \tag{28}$$

which is equivalent to $\exists u_i \in \mathbb{R}_{\geq 0}^{q_x}, u_i' H_x = H_{y,i} G$, such that $u_i' h_x \leq h_{y,i}$. Let $\Lambda = [u_1', u_2', \cdots, u_{q_y}']$, and (26) immediately follows. □