

Chapter 2

Non-convex Region Description by Hyperplane Arrangements

This chapter presents some prerequisites and basic notions which will be instrumental in the rest of the manuscript. Namely, details about hyperplane arrangements (the scaffolding over which the feasible region characterization is defined) and various standard set theoretic elements (families of sets like polyhedra, zonotopes as well as basic set operations) are presented.

Without being exhaustive, for describing this “set of tools” the presentation follows the ideas and notations in [1–3]. Furthermore, these notions can be readily found and reviewed by the interested readers in complementary materials referenced in this chapter.

2.1 Hyperplane Arrangements

Let us consider a finite collection of hyperplanes $\mathbb{H} = \{\mathcal{H}_i\}_{i \in \mathbb{I}}$ from \mathbb{R}^n :

$$\mathcal{H}_i = \{x \in \mathbb{R}^n : h_i x = k_i\}, \quad i \in \mathbb{I}, \quad (2.1)$$

with $\mathbb{I} \triangleq \{1 \dots N\}$ and $(h_i, k_i) \in \mathbb{R}^{1 \times n} \times \mathbb{R}$.

Each of these hyperplanes partitions the space into two disjoint¹ regions (which halve the space and hence are called “half-spaces”):

$$\mathcal{R}_i^+ = \{x \in \mathbb{R}^n : h_i x \leq k_i\}, \quad (2.2a)$$

$$\mathcal{R}_i^- = \{x \in \mathbb{R}^n : -h_i x \leq -k_i\}. \quad (2.2b)$$

¹The relative interiors of these regions do not intersect, but their closures have as common boundary the affine subspace \mathcal{H}_i in (2.1).

Considering the above basic concepts the combinatorial notion of hyperplane arrangement is defined in the following.

Definition 2.1 (*Hyperplane arrangements*—[4]) The collection of hyperplanes \mathbb{H} will partition the space into a union of disjoint cells $\mathcal{A}(\sigma)$ characterized by a sign tuple $\sigma \in \{-, +\}^N$ defined as follows:

$$\mathcal{A}(\sigma) = \bigcap_{i \in \mathbb{I}} \mathcal{R}_i^{\sigma(i)}. \quad (2.3)$$

The collection of all feasible sign tuples describes a hyperplane arrangement of feasible cells covering the entire space:

$$\mathcal{A}(\mathbb{H}) = \bigcup_{\sigma \in \Sigma} \mathcal{A}(\sigma), \quad (2.4)$$

where

$$\Sigma = \{\sigma \in \{-, +\}^N : \mathcal{A}(\sigma) \neq \emptyset\}, \quad (2.5)$$

denotes the collection of sign tuples resulting into non-empty intersections of regions (2.2a)–(2.2b). ♦

Remark 2.1 Arrangement (2.4) is said to be in *general position* if for any ‘ i ’ a perturbation $\epsilon_i > 0$ (e.g., $\mathcal{H}_i \rightarrow \mathcal{H}'_i = \{x : h_i x = k_i + \epsilon_i\}$) will not change the distribution and number of regions. ♦

Remark 2.2 Any sub-arrangement $\mathcal{A}(\mathbb{H}')$ of $\mathcal{A}(\mathbb{H})$ (i.e., $\mathbb{H}' \subseteq \mathbb{H}$) is said to be *central* if $\bigcap_{\mathcal{H} \in \mathbb{H}'} \mathcal{H} \neq \emptyset$. ♦

Illustrative example for hyperplane arrangements in \mathbb{R}^2

Here and thereafter, we make use of the numerical data provided in Appendix A such that the present and the forthcoming illustrative examples are based on reproducible constructions.

Figure 2.1a depicts a hyperplane as described by (2.1), $\mathcal{H} = \{x : [0.07 \ 0.99] x = 4.98\}$ and its two half-spaces, $\mathcal{R}^- = \{x : [0.07 \ 0.99] x \leq 4.98\}$ and $\mathcal{R}^+ = \{x : [0.07 \ 0.99] x \geq 4.98\}$ as in (2.2a)–(2.2b).

Figure 2.1b illustrates 4 hyperplanes with the numerical data provided in Appendix A.1. These generate a hyperplane arrangement as in (2.3) with 10 cells, each defined by a unique sign tuple (the feasible sign combinations are: $(- + - -)$, $(- - - -)$, $(+ - - -)$, $(+ - + -)$, $(+ - - +)$, $(+ - + +)$, $(+ + + +)$, $(- + + +)$, $(- + + -)$, $(+ + + -)$). Also, the figure delineates the cell $\mathcal{A}(\sigma) = \bigcup_{\sigma(i)} \mathcal{R}_i^{\sigma(i)} =$

$\mathcal{R}_1^+ \cap \mathcal{R}_2^+ \cap \mathcal{R}_3^+ \cap \mathcal{R}_4^-$, corresponding to the sign tuple $\sigma = (+ + + -)$. Note that all the other sign tuples (the remaining $2^4 - 10 = 6$ sign combinations) are infeasible. For example, $\sigma = (- + - +)$ which corresponds to $\mathcal{A}(\sigma) = \bigcup_{\sigma(i)} \mathcal{R}_i^{\sigma(i)} = \mathcal{R}_1^- \cap$

$\mathcal{R}_2^+ \cap \mathcal{R}_3^- \cap \mathcal{R}_4^+$ is an empty set (due to the fact that $(\mathcal{R}_1^- \cap \mathcal{R}_2^+ \cap \mathcal{R}_3^-) \cap \mathcal{R}_4^+ = \emptyset$).

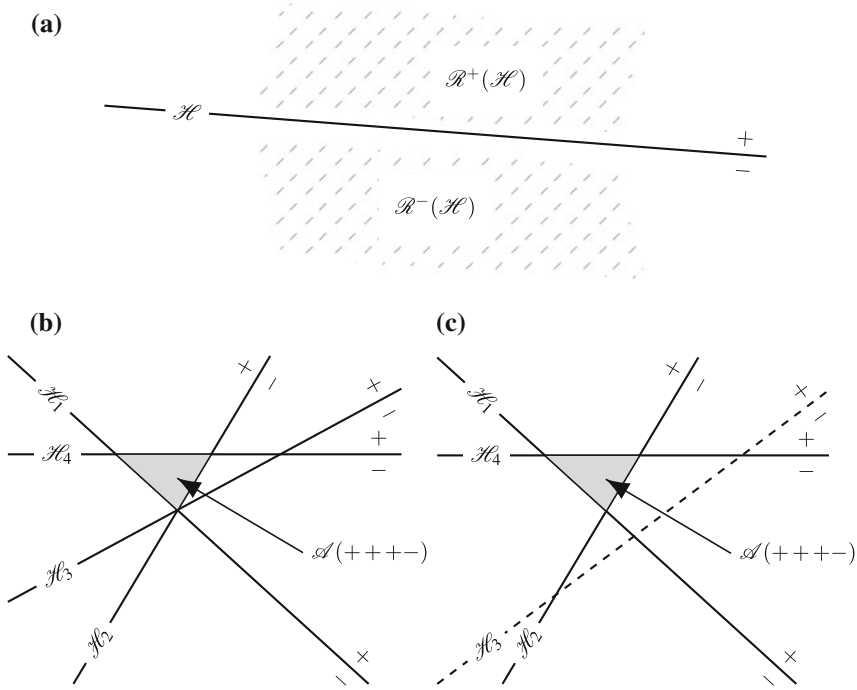


Fig. 2.1 Hyperplane arrangements. **a** One hyperplane and its associated half-spaces. **b** Hyperplane arrangement. **c** Perturbed hyperplane arrangement

Furthermore, note that the hyperplane arrangement depicted in Fig. 2.1b is not in a general position since three of the hyperplanes intersect $\{\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3\}$. These three hyperplanes also define a central sub-arrangement but no other 3 hyperplanes define one (e.g., $\{\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_4\}$), as explained in Remark 2.2. Note that any combination of the hyperplanes is a central sub-arrangement ($\{\mathcal{H}_1, \mathcal{H}_2\}$, $\{\mathcal{H}_1, \mathcal{H}_3\}$, $\{\mathcal{H}_1, \mathcal{H}_4\}$, $\{\mathcal{H}_2, \mathcal{H}_4\}$, $\{\mathcal{H}_3, \mathcal{H}_4\}$).

Figure 2.1c delineates the arrangement perturbed by translating the 3rd hyperplane ($\mathcal{H}_3 \rightarrow \{[0.6 - 0.8]x = 1\}$). The corresponding numerical data is to be found in Appendix A.1. We observe that no three hyperplanes intersect in a common point and that the number of cells has increased to 11. This was expected since in \mathbb{R}^2 three randomly taken hyperplanes should not intersect in a common point (see Remark 2.1).

2.1.1 Region Counting

Each of the regions (2.4) can be either bounded or unbounded. Hence, denote² the total number of regions as $r(\mathcal{A})$ and the number of bounded regions as $b(\mathcal{A})$.

To proceed further, the auxiliary notion of a *characteristic polynomial* is required.

Definition 2.2 (Whitney's formula [5]) Let there be an arrangement \mathcal{A} in \mathbb{R}^n and \mathcal{A}' the shorthand notation for any of its sub-arrangements (as defined in Remark 2.2), then:

$$\mathcal{E}(\mathcal{A}, t) = \sum_{\mathcal{A}' \subseteq \mathcal{A}} (-1)^{\text{card}(\mathcal{A}')} t^{n - \text{rank}(\mathcal{A}')}, \quad (2.6)$$

where \mathcal{A}' are central sub-arrangements of \mathcal{A} , $\text{card}(\mathcal{A}')$ denotes the number of hyperplanes of \mathcal{A}' and $\text{rank}(\mathcal{A}')$ denotes the dimension of the sub-space spanned by the hyperplanes of \mathcal{A}' . By definition, $\text{rank}(\emptyset) = 0$. ♦

The number of regions (total and bounded) can now be given as a function of the characteristic polynomial.

Theorem 2.1 (Zaslavsky's Theorem [6]) For a given arrangement \mathcal{A} , the number of regions is

$$r(\mathcal{A}) = |\mathcal{E}(\mathcal{A}, -1)|, \quad b(\mathcal{A}) = |\mathcal{E}(\mathcal{A}, 1)|. \quad (2.7)$$

□

Remark 2.3 The above notions provide powerful tools for counting the regions (which is of interest as it reflects on the number of binary variables required in the mixed-integer constructions of the next chapter). Note that all the computations involve simple operations and avoid explicitly computing the regions (hence one can proceed with combinatorial notions and avoid a costly geometrical analysis). Furthermore, whenever the arrangement is in general position (see Remark 2.1) these numbers are known. For example,

$$r(\mathcal{A}) = \sum_{i=0}^n \binom{N}{i}, \quad (2.8)$$

as per Buck's formula [7]. ♦

Illustrative example for region counting

Let us revisit first the example in Fig. 2.1b where the hyperplanes are not in general position (as defined in Remark 2.1) and hence the bound given by Buck's formula is not reached. We proceed to enumerate all the central sub-arrangements of the arrangement: $\{\emptyset\}$, $\{\mathcal{H}_1\}$, $\{\mathcal{H}_2\}$, $\{\mathcal{H}_3\}$, $\{\mathcal{H}_4\}$, $\{\mathcal{H}_1, \mathcal{H}_2\}$, $\{\mathcal{H}_1, \mathcal{H}_3\}$, $\{\mathcal{H}_1, \mathcal{H}_4\}$, $\{\mathcal{H}_2, \mathcal{H}_3\}$,

²Whenever the notation is not ambiguous, we use the shorthand \mathcal{A} for the arrangement $\mathcal{A}(\mathbb{H})$.

$\{\mathcal{H}_2, \mathcal{H}_4\}$, $\{\mathcal{H}_3, \mathcal{H}_4\}$ and $\{\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3\}$. To each of these central sub-arrangements corresponds a term in Whitney's formula. For example $\mathcal{A}' = \{\mathcal{H}_1, \mathcal{H}_4\}$ has $\text{card}(\mathcal{A}') = 2$ elements and $\text{rank}(\mathcal{A}') = 2$ (since the two composing hyperplanes span \mathbb{R}^2) thus leading to term $(-1)^2 t^{2-2} = 1$. Applying the same reasoning for the rest of the sub-arrangements enumerated before we obtain the characteristic polynomial (2.6):

$$\begin{aligned}\mathcal{E}(\mathcal{A}, t) &= 1 \cdot (-1)^3 t^{2-2} + 6 \cdot (-1)^2 t^{2-2} + 4 \cdot (-1)^1 t^{2-1} + 1 \cdot (-1)^1 t^{2-0} \\ &= 5 - 4t + t^2,\end{aligned}$$

which permits to apply Zaslavsky's Theorem and obtain the total number of regions $r(\mathcal{A}) = |\mathcal{E}(\mathcal{A}, -1)| = 10$ and the number of bounded regions $b(\mathcal{A}) = |\mathcal{E}(\mathcal{A}, 1)| = 2$.

The same constructions can be applied to the arrangement from Fig. 2.1c. The only difference is that there are no longer three intersecting hyperplanes and hence the term $(-1)^3 t^{2-2}$ no longer appears in the characteristic polynomial:

$$\mathcal{E}(\mathcal{A}, t) = 6 \cdot (-1)^2 t^{2-2} + 4 \cdot (-1)^1 t^{2-1} + 1 \cdot (-1)^1 t^{2-0} = 6 - 4t + t^2,$$

In turn, by again applying the Zaslavsky's Theorem, the total number of regions $r(\mathcal{A}) = |\mathcal{E}(\mathcal{A}, -1)| = 11$ and the number of bounded regions $b(\mathcal{A}) = |\mathcal{E}(\mathcal{A}, 1)| = 3$ are obtained. For the total number of regions, we may as well apply Buck's formula, i.e., $r(\mathcal{A}) = \sum_{i=0}^2 \binom{4}{i} = \binom{4}{0} + \binom{4}{1} + \binom{4}{2} = 11$.

The bounded and total number of regions are illustrated in Fig. 2.2 which uses the numerical data provided in Appendix A.1.

2.1.2 Parametrized Hyperplane Arrangements

In many practical control applications the hyperplane arrangement modifies from one instant to the next in relationship with a dynamical system evolution. Perhaps the most illustrative example is to consider a moving obstacle: the hyperplanes which characterize it will translate from one instant to the next. Re-computing the arrangement is not trivial and may be impractical for small discretization steps. The solution is to analyze the problem in a lifted space (variables and parameters), obtain the corresponding arrangement and to 'cut' it at runtime for the current parameter in order to go back to the original space.

Henceforth, let us consider a collection of parametrized hyperplanes $\mathbb{H}(p) = \{\mathcal{H}_i(p)\}$ from \mathbb{R}^n where each of them has a linear dependence in the right-hand side with respect to a variable $p \in \mathbb{R}^{n_p}$:

$$\mathcal{H}_i(p) = \{x \in \mathbb{R}^n : h_i x = k_i - h_i^p p\}. \quad (2.9)$$

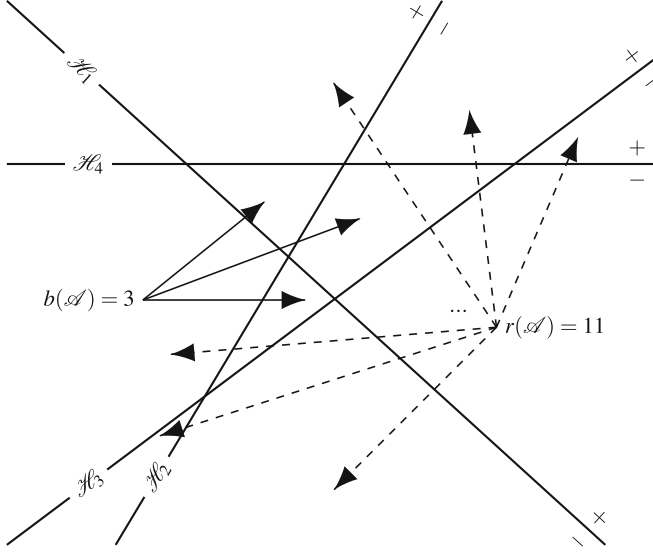


Fig. 2.2 Hyperplane arrangement region counting

A new hyperplane collection $\mathbb{H}' = \{\mathcal{H}'_i\}$ is considered:

$$\mathcal{H}'_i = \left\{ \begin{bmatrix} h_i & h_i^p \end{bmatrix} \cdot \begin{bmatrix} x \\ p \end{bmatrix} = k_i \right\}, \quad (2.10)$$

and the arrangement $\mathcal{A}(\mathbb{H}')$ is defined with feasible cells $\mathcal{A}(\sigma')$ where $\sigma' \in \Sigma'$.

To retrieve the cells composing the arrangement in \mathbb{R}^n for a certain parameter p^* we simply intersect $\mathcal{A}(\sigma')$ with the subspace $\{p = p^*\}$:

$$\mathbb{H}(p^*) = \bigcup_{\sigma' \in \Sigma'} \{x : A(\sigma') \cap \{p = p^*\}\}. \quad (2.11)$$

At this point we might stop if not for the fact that many of the cells $A(\sigma') \cap \{p = p^*\}$ will be empty for certain values of p . To reduce the runtime computations we can therefore consider an additional analysis step. That is, we compute the domain of existence of each cell $A(\sigma')$ in the parameter space (i.e., project on \mathbb{R}^{n_p}):

$$\text{dom}(\mathcal{A}(\sigma')) = \left\{ p : \exists x \text{ s.t. } \begin{bmatrix} x \\ p \end{bmatrix} \in \mathcal{A}(\sigma') \right\}. \quad (2.12)$$

This provides a partitioning of \mathbb{R}^{n_p} into overlapping domains which can be further written (after suitable manipulations) as a union of disjoint domains D'_k . To each domain corresponds a list of sign tuples Σ'_k such that $\forall \sigma' \in \Sigma'_k, \exists p \in D'_k$ s.t. $\mathcal{A}(\sigma') \cap \{p = p^*\} \neq \emptyset$. Lastly, at runtime, we identify index 'k' such that

$p \in D_k$ and retrieve Σ'_k from which we generate the current arrangement, as in (2.11) but with Σ' reduced to Σ'_k .

Remark 2.4 These notions can be related to the parametrized polyhedra and associated validity domains (see Sect. 2.2 and [8]). ♦

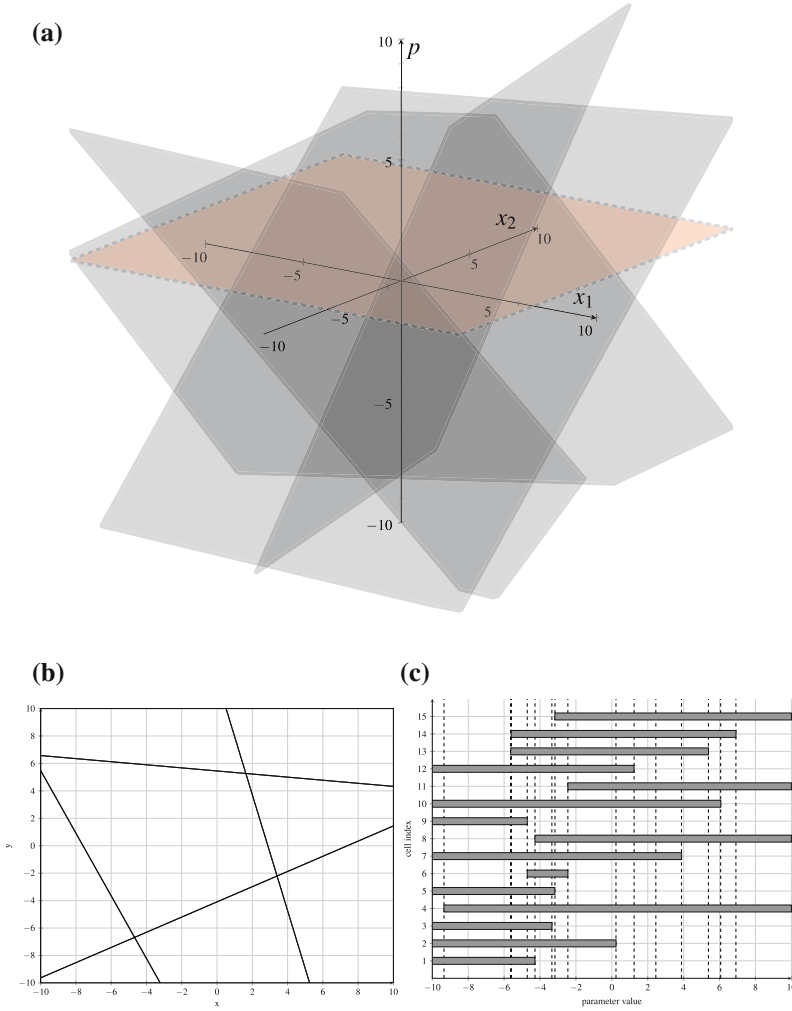


Fig. 2.3 Parametrized hyperplane arrangements. **a** Hyperplane arrangement lifted in \mathbb{R}^3 . **b** Domain. **c** Region

Illustrative example for hyperplane parametrization

Let us consider the hyperplane arrangement given in Appendix A.2 with 4 hyperplanes in \mathbb{R}^2 parametrized after a scalar $p \in \mathbb{R}$. We apply (2.10) and obtain an arrangement with 4 hyperplanes in \mathbb{R}^3 which has (according to (2.8)) 15 cells.

In Fig. 2.3a we depict the resulting arrangement with the original space represented along dimensions x_1 and x_2 and the parameter space represented along dimension x_3 . It can be seen that ‘cutting’ for a certain value of p will result in different arrangements in \mathbb{R}^2 , see Fig. 2.3b which is obtained for $p = 1.5$.

Lastly, in Fig. 2.3c we illustrate the domains of existence for the cells of the lifted arrangement. Since the parameter is a scalar, these domains are intervals along the real axis (represented as solid segments). As it can be seen, each of the 15 cells has its own domain which in some cases overlaps with others. Considering all the overlaps we compute the disjoint intervals D'_k (separated by vertical dashed lines). For example, for $p = -1.5 \in [-2.45 \ 2.45]$ we have 10 cells whose cutting will result in non-empty cells in \mathbb{R}^2 .

2.2 Polyhedral and Zonotopic Sets

There exists a wealth of families which describe convex (or non-convex) sets with varying degrees of accuracy. An important limiting factor is the numerical reliability of their representation. That is, a particular family may be able to represent a great number of shapes but due to computationally expensive manipulations will be useless in practice. Usually there exists an inverse relation between flexibility of a family and the numerical cost of the representation.

2.2.1 Polyhedral Sets

Polyhedra³ provide a useful geometrical representation for the linear constraints that appear in diverse fields such as linear control and optimization. In a convex setting, they provide a good compromise between complexity and flexibility. Due to their linear and convex nature, the basic set operations are relatively easy to implement [8]. Principally, this is related to their dual (half-spaces/vertices) representation [9] which allows to choose which formulation is best suited for a particular task. Note that the transformation from one representation to another may be time-consuming with various well-known algorithms: Fourier-Motzkin elimination—[10], Double Description method—[11], Equality Set Projection—[12]. With respect to their flexibility it is worthwhile to note that any convex body can be approximated arbitrarily well by a polytope [13].

³In here we will use the notions of *polyhedron* and *polytope*. The first represents the element of the polyhedral class under discussion whereas the latter denotes a bounded polyhedron.

The set operations implemented over the polyhedral family represent a main topic in the domain which lies at the intersection of convex geometry, mathematical programming and computer science [14]. To mention just a few, the algorithms used to implement the Minkowski addition (the sum over the space of sets), the Pontryagin difference (the non-dual counterpart of the sum over the space of sets—see formal definitions below), the translation between vertex and half-space representations are sensitive to the space dimension and the complexity of the chosen set representation (see [15, 16]).

We start by recalling some theoretical concepts (from Chap. 1 of [4]). Firstly, we provide the notion of \mathcal{H} -polyhedron which denotes an intersection of closed half-spaces:

Definition 2.3 A set $P \in \mathbb{R}^n$ is a \mathcal{H} -polyhedron if it can be implicitly presented in the form

$$P = \mathcal{P}(F, \theta) = \{x \in \mathbb{R}^n : Fx \leq \theta\}, \quad (2.13)$$

for some $F \in \mathbb{R}^{m \times n}$, $\theta \in \mathbb{R}^m$.

The cone of a finite collection of vectors is defined by Definition 2.4 and the convex hull of a finite set of points by Definition 2.5:

Definition 2.4 For a finite collection of vectors $Y = \{y_1 \dots y_d\} \subseteq \mathbb{R}^n$, the cone of Y is defined as

$$\text{Cone}(Y) \triangleq \{t_1 y_1 + \dots t_d y_d : t_i \in \mathbb{R}_+\} = \{Yt, t \in \mathbb{R}_+^d\}.$$

Definition 2.5 For a finite collection of points $V = \{v_1 \dots v_d\} \subseteq \mathbb{R}^n$, the convex hull of V is defined as

$$\text{Conv}(V) \triangleq \{\alpha_1 v_1 + \dots \alpha_d v_d : \alpha_i \in \mathbb{R}_+, \sum_i \alpha_i = 1\} = \{V\alpha, \alpha \in \mathbb{R}_+^d, \mathbf{1}^T \alpha = 1\}.$$

In the context of set theory, the generalizations of addition (the *Minkowski sum*), difference (the *Pontryagin difference*) and distance (the *Hausdorff distance*) are provided below:

Definition 2.6 The Minkowski sum of two sets $P, Q \subseteq \mathbb{R}^n$ is defined to be

$$P \oplus Q = \{x + y : x \in P, y \in Q\},$$

and the Pontryagin difference is defined as

$$P \ominus Q = \{x \in P : x + y \in P, \forall y \in Q\}.$$

Definition 2.7 Given two convex sets P, Q , the Hausdorff distance is defined as

$$d_H(P, Q) = \max \{\bar{d}_H(P, Q), \bar{d}_H(Q, P)\},$$

where $\bar{d}_H(P, Q) = \max_{x \in P} \min_{y \in Q} d(x, y)$, and $d(x, y)$ is a distance measured in a given norm in the \mathbb{R}^n space.

Illustrative example for polyhedral set operations

In Fig. 2.4 polyhedral construction and some important set operations are illustrated. Figure 2.4a shows a bounded polyhedron (i.e., a polytope) which is characterized as half-space intersections or equivalently as convex sum of extremal vertices. Figure 2.4b delineates an unbounded polyhedron (a cone) which is again characterized by half-space intersections and cone representation (positive sum of rays). Lastly, in Fig. 2.4c the Minkowski addition and Pontryagin difference operations are illustrated graphically.

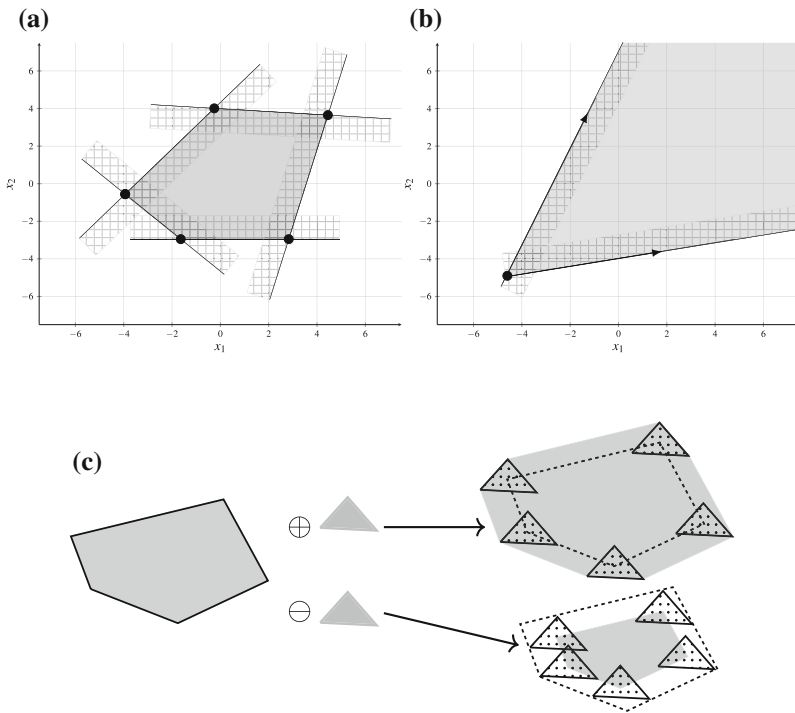


Fig. 2.4 Some primitives and operations for polytopic sets. **a** Convex hull of a polytope. **b** Cone. **c** Minkowski sum and Pontryagin difference representations

2.2.2 Zonotopic Sets

Zonotopes represent a particular class of polytopes (i.e., projections of a hypercube from a higher dimension) and can be described in “generator form” as

$$\mathcal{Z}(c, G) = \left\{ x \in \mathbb{R}^n : x = c + \sum_{i=1}^m \lambda_i g_i, |\lambda_i| \leq 1 \right\}, \quad (2.14)$$

with $i = 1 \dots m$ and where $c \in \mathbb{R}^n$ represents the center and $G = [g_1 \dots g_m] \in \mathbb{R}^{n \times m}$ the matrix of generators. A zonotope defined as in (2.14) has some interesting properties [17]:

- is closed under linear transformations:

$$L\mathcal{Z}(c, G) = \mathcal{Z}(Lc, LG); \quad (2.15)$$

- is closed under the Minkowski sum:

$$\mathcal{Z}(c_1, G_1) \oplus \mathcal{Z}(c_2, G_2) = \mathcal{Z}(c_1 + c_2, [G_1 \ G_2]). \quad (2.16)$$

Due to its particular structure, the number of vertices and facets for a zonotope (2.14) is significantly reduced in comparison with a randomly generated polyhedron.^{4, 5}

In realistic situations, often the constraints are given in polytopic form but posses an encapsulated symmetry which allows a description in terms of zonotopic sets. Even when this is not the case, zonotopic approximations may be constructed. Since the generator representation (2.14) is more compact than either the half-space and vertex representations associated to polytopes, it becomes obvious why for numerical and theoretical reasons the zonotopes will be used whenever possible in the set constructions of this manuscript. However, we stress that this practical preference remains a subjective choice and the set-theoretic results appearing in the next developments hold for any class of sets (when convexity is mandatory, this requirement will be specified accordingly).

For polytopic sets, [20] proposes tight approximations in fixed directions and [21] discusses an iterative algorithm. A more general case is represented by convex bodies defined by nonlinear inequalities. Common characterizations of such sets include the

⁴From [18] we recall the following bounds on the number of facets f_i of order ‘ i ’ for a given zonope \mathcal{Z} (bounds which are reached whenever the zonotope’s generators are in general position):

$$f_0(\mathcal{Z}) \leq 2 \sum_{i=0}^{n-1} \binom{m-1}{i}, \quad f_{n-1}(\mathcal{Z}) \leq 2 \binom{m}{n-1}.$$

⁵A zonotope is topologically equivalent with an associated hyperplane arrangement (see Definition 2.1) offering thus efficient descriptions of the faces (e.g., by using reverse search algorithms as in [19]).

unit ball of the weighted p -norm (usually some weighted Euclidean norm defining an ellipsoid). In [22, 23] it is proven that any such Euclidean ball can be approximated arbitrarily close, in the sense of the Hausdorff distance, by a zonotope given by a uniform distribution on the surface of the (hyper)sphere.

2.3 Non-convex Region Description

In this manuscript we aim to characterize the complement of a given region which may or may not be convex. To this end let us consider a collection of (possibly overlapping) polyhedral sets in \mathbb{R}^n that may be related to obstacles to better fix the ideas (usually encountered in \mathbb{R}^2 or \mathbb{R}^3 for motion planning problems):

$$\mathbb{S} = \bigcup_{l=1}^{N_o} S_l. \quad (2.17)$$

Describing the region of interest (2.17) as a union of polyhedral sets is not random: a polyhedral set is in fact a finite intersection of regions of type (2.2a)–(2.2b). We can then consider $\mathbb{H} = \{\mathcal{H}_i\}_{i \in \mathbb{I}}$ as the collection of all hyperplanes appearing in the description of polytopes S_l from (2.17) and construct the hyperplane arrangement (2.4).

Then, there exists a sign tuple σ_l which characterizes⁶ S_l , i.e.,

$$S_l = \mathcal{A}(\sigma_l) = \bigcap_{i \in \mathbb{I}} \mathcal{R}_i^{\sigma_l(i)} = \mathcal{P} \left(\begin{bmatrix} \dots \\ \sigma_l(i)h_i \\ \dots \end{bmatrix}, \begin{bmatrix} \dots \\ \sigma_l(i)k_i \\ \dots \end{bmatrix} \right). \quad (2.18)$$

Further we can partition the collection of feasible tuples (2.5) into⁷:

- (i) the collection of forbidden tuples

$$\Sigma^\bullet = \{\sigma \in \Sigma : \text{interior}(\mathcal{A}(\sigma) \cap \mathbb{S}) \neq \emptyset\}, \quad (2.19)$$

- (ii) the collection of admissible tuples

$$\Sigma^\circ = \{\sigma \in \Sigma : \text{interior}(\mathcal{A}(\sigma) \cap \mathbb{S}) = \emptyset\} = \Sigma \setminus \Sigma^\bullet. \quad (2.20)$$

Remark 2.5 Note that we have chosen as classification criteria $\text{interior}(\mathcal{A}(\sigma) \cap \mathbb{S}) \neq \emptyset$ rather than $\mathcal{A}(\sigma) \subseteq \mathbb{S}$. The question is mote as long as the hypeplanes

⁶We assume without loss of generality that each set S_l is characterized by a unique tuple.

⁷The intersection is meant to discern the cells which overlap with the obstacles. The common frontier is irrelevant in this context and is formally discarded by taking the interior of the intersection into consideration.

generating the arrangement are taken from the half-space representation of the obstacles (in this case the inclusion and the intersection operations are equivalent). Still, we preferred the more robust interior $(\mathcal{A}(\sigma) \cap \mathbb{S}) \neq \emptyset$ which assigns a sign tuple as forbidden if there is any intersection between the current cell and the obstacle union. \blacklozenge

With notation (2.19) and (2.20) we may describe regions \mathbb{S} and $\bar{\mathbb{S}}$ in terms of forbidden and admissible tuples:

$$\mathbb{S} = \bigcup_{\sigma \in \Sigma^\bullet} \mathcal{A}(\sigma), \quad (2.21a)$$

$$\bar{\mathbb{S}} = \bigcup_{\sigma \in \Sigma^\circ} \mathcal{A}(\sigma). \quad (2.21b)$$

Note that $\Sigma^\bullet \cap \Sigma^\circ = \emptyset$ and $\Sigma^\bullet \cup \Sigma^\circ = \Sigma \subset \{-, +\}^N$.

Remark 2.6 At this point, we may ask what was first, the arrangement or the obstacles? The answer is that it depends on the problem at hand. Do we start with predefined obstacles (polyhedral sets) and find the hyperplane arrangement associated to them? Or, do we start with a predefined hyperplane arrangement (e.g., by grid-ing the space and assigning to the resulting cells admissible/forbidden values? Each approach has its merits as on one hand we may have more precise bounds but difficult formulation and on the other hand we have over-approximations but under reduced/fixed complexity. \blacklozenge

Illustrative example of non-convex regions

For the purpose of illustration let us consider a simple example as depicted in Fig. 2.5. A union of two obstacles, $\mathbb{S} = S_1 \cup S_2$ in \mathbb{R}^2 is considered. The forbidden regions are defined by 5 and respectively 3 hyperplanes (see Appendix A.4 for the numerical data). Furthermore, these partition the space into 37 cells from which 3 describe the obstacles and the rest characterize the feasible space $\bar{\mathbb{S}} = \mathbb{R}^2 \setminus \mathbb{S}$. More precisely, $\Sigma^\bullet = \{\sigma_1, \sigma_2, \sigma_3\}$ is identified such that $S_1 = \mathcal{A}(\sigma_1)$ and $S_2 = \mathcal{A}(\sigma_2) \cup \mathcal{A}(\sigma_3)$ for $\sigma_1 = (+++ - + + -)$, $\sigma_2 = (+ - + + + + +)$ and $\sigma_3 = (- - + + + + +)$. Note that the obstacle S_2 is described by more than one cell (this usually happens if a hyperplane from another obstacle cuts the obstacle under consideration, as for example the hyperplane \mathcal{H}_1 in Fig. 2.5). As it can be observed in the example, this is not an issue, it simply means that Σ^\bullet , the collection of forbidden tuples, has more tuples than there are obstacles.

2.3.1 Cell Merging

Recall that any of the cells from (2.21b) is described by a unique sign tuple and is disjoint with respect to the others (up to its boundary, see footnote 1). For our

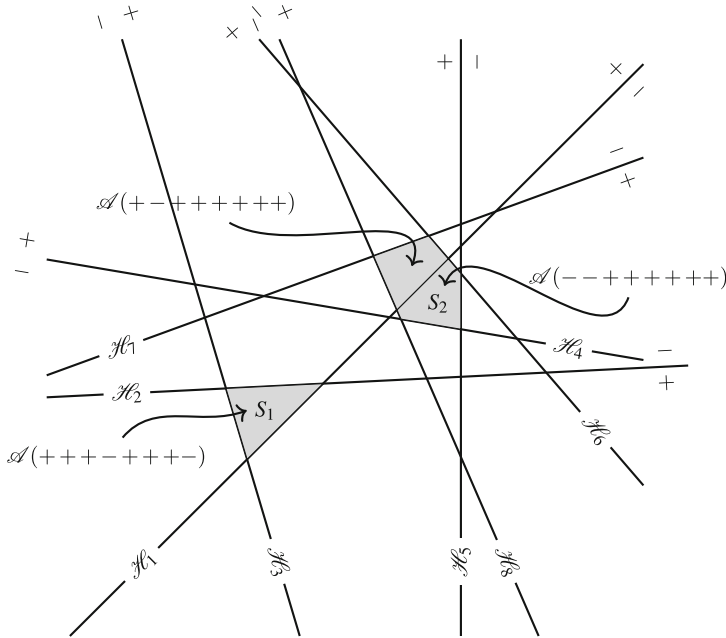


Fig. 2.5 Collection of obstacles and their associated hyperplane arrangement

purposes, we may be⁸ satisfied with any collection of regions not necessarily disjoint which covers the feasible space. In this context, the question arises whether it is possible to merge the existing cells of (2.21b) into a reduced number of regions which still describe \bar{S} . Note that this will help later on when it will lead to a compact mixed-integer representation.

In general, these *merged cells* have to be formed as unions of cells from (2.21b) such that the result is a convex set. Existing merging algorithms are usually computationally expensive because in general a union of convex sets is non-convex. In the present study the problem can be simplified by observing two properties of the cells from (2.21b) (and in general, from (2.4)):

- the sign tuples σ describe an adjacency graph since any two cells whose sign tuples differ at only one position are neighbors (i.e., they share a hyperplane as boundary),
- the union of any two adjacent cells is a polyhedron.

Using these properties a merged cell can be formally characterized.

Definition 2.8 The union of all cells $\mathcal{A}(\sigma)$ whose sign tuples retain the same values over a subset of indices ($\mathbf{i} \subset \mathbb{I}$) and span all possible combinations for the rest of

⁸In the forthcoming chapter constructions which accept both formulations will be discussed in detail.

indices $(\mathbb{I} \setminus \mathbf{i})$ is a “merged” cell. This cell is characterized by the sign tuple $\sigma^* \in \{-, *, +\}^N$ where $\sigma^*(i) \in \{-, +\}, \forall i \in \mathbf{i}$ and $\sigma^*(i) = *, \forall i \in \mathbb{I} \setminus \mathbf{i}$ such that:

$$\mathcal{A}(\sigma^*) = \bigcap_{\sigma^*(i) \neq *, i \in \mathbb{I}} \mathcal{R}_i^{\sigma^*(i)} = \bigcup_{\substack{\sigma(i) = \sigma^*(i), \forall i \in \mathbf{i} \\ \sigma(j) \in \{-, +\}, \forall j \notin \mathbf{i}}} \mathcal{A}(\sigma). \quad (2.22)$$

◆

Remark 2.7 Using Definition 2.8, the number of cells appearing in a merged cell is expected to be $2^{\text{card}(\mathbb{I} \setminus \mathbf{i})}$ because “spanning all possible combinations” would lead to such a number. In practice the number of disjoint cells will be smaller since many of these sign combinations result in empty cells. ◆

With the previously given properties, merging algorithms can be employed (see, for example, [24], which adapt a “branch and bound” strategy). This is still cumbersome, and henceforth the problem is tackled in the Boolean algebra framework. The merging problem of regions from (2.21b) is functionally identical to the minimization of a Boolean function given in the “sum-of-products” form. A cell describing the (in)feasible cell from (2.3) corresponds to a “1” (“0”) value in the truth-table at the position determined by its associated sign tuple, whereas infeasible sign tuples correspond to “don’t care” values. It is then straightforward to apply minimization algorithms (Karnaugh maps, the Quine-McCluskey algorithm or the Espresso heuristic logic minimizer for example) in order to obtain Boolean minterms who describe the merged cells of (2.21b). Note that a similar merging procedure was proposed in [25] in order to deal with polyhedral piecewise affine systems.

The next theorem provides constructive means for generating merged cells (similar to the results in [26]).

Theorem 2.2 Consider the forbidden and admissible regions (2.21a), (2.21b) characterized respectively by the sign tuples (2.19) and (2.20). Then, the feasible region (2.21b) is compactly described as a union of merged cells (2.22):

$$\bar{\mathbb{S}} = \bigcup_{\sigma^*} \mathcal{A}(\sigma^*), \quad (2.23)$$

where $\sigma^* \in \{-, *, +\}^N$ are given by the “sum-of-products” representation of the Boolean function $f : \{-, +\}^N \rightarrow \{0, *, 1\}$ verifying:

$$f(\sigma) = 0, \forall \sigma \in \Sigma^\bullet, \quad (2.24a)$$

$$f(\sigma) = 1, \forall \sigma \in \Sigma^\circ, \quad (2.24b)$$

$$f(\sigma) = *, \forall \sigma \in \{-, +\}^N \setminus (\Sigma^\bullet \cup \Sigma^\circ). \quad (2.24c)$$

□

Proof Let us consider function (2.24) and its truth-table: we have “0” for sign tuples characterizing cells inside (2.21a), “1” for sign tuples characterizing cells inside

(2.21b) and “don’t care” values for all the sign tuples which correspond to empty cells (infeasible combinations of regions (2.2a)–(2.2b)). All that remains is to group the combinations which are “1” or “don’t care” in the truth-table and express the function (2.24) in the canonical sum-of-products form. Each term of the product describes a region of form (2.22) thus reaching (2.23) and concluding the proof. ■

Furthermore, Theorem 2.2 can be adapted so that we can actually avoid calculating the collection of feasible sign tuples (2.5). Rather, by using the set of cells which describe (2.21a) (usually containing many fewer cells than the total number, see Theorem 2.1) and Boolean algebra notions we provide a compact representation of (2.21b) without explicitly making the decomposition (2.4).

Corollary 2.1 *Consider the forbidden region (2.21a) characterized by the sign tuples (2.19). Then, the feasible region (2.21b) is compactly described as a union of merged cells of form (2.22):*

$$\bar{\mathbb{S}} = \bigcup_{\sigma^*} \mathcal{A}(\sigma^*), \quad (2.25)$$

where $\sigma^* \in \{-, *, +\}^N$ are the sums from the “sum-of-products” representation of the Boolean function $f : \{-, +\}^N \rightarrow \{0, 1\}$ verifying

$$f(\sigma) = 0, \forall \sigma \in \Sigma^\bullet, \quad (2.26a)$$

$$f(\sigma) = 1, \forall \sigma \in \{-, +\}^N \setminus \Sigma^\bullet. \quad (2.26b)$$

□

Proof Let us consider function (2.26) and its truth-table: we have “0” for sign combinations characterizing cells inside (2.21a) and “1” in all the other cases (regardless if the sign tuples describe admissible cells or infeasible combinations of regions). All that remains is to group the combinations which are “1” and express the function in the canonical sum-of-products form. Each term of the product describes a region of form (2.22) thus reaching (2.25). ■

Several remarks are in order:

Remark 2.8 In the construction of the truth table of (2.26) as explained in the proof of Corollary 2.1 lies the major difference with respect to Theorem 2.2. By not computing the cells composing the feasible region, it is assumed implicitly (and conservatively) that all the remaining combinations of signs correspond to non-empty admissible cells (thus “1” in the table). This will result in a larger number of merged cells. More than that, one can even obtain merged cells which are infeasible (because the sign combination describes an empty region). Such a cell is denoted as feasible (and discarded otherwise) at a later post-processing stage, in contrast with the approach in Theorem 2.2 where the validation is made in a pre-processing stage. ♦

Remark 2.9 For a compact representation it is suitable to write (2.24) or (2.26) in a minimal sum-of-products form. To this end, minimization algorithms (e.g., Karnaugh

Each of the Boolean products appearing here denotes a merged cell. Taking them in order, we have: $\mathcal{A}(* - * * * * -)$, $\mathcal{A}(- * * - * * * *)$, $\mathcal{A}(* * * * * - *)$, $\mathcal{A}(* * - * * * *)$, $\mathcal{A}(* * * * * - *)$ and $\mathcal{A}(* * * * - * * *)$. The union of these merged cells describes the feasible region (2.21b) as in (2.25).

For illustration, Fig. 2.7 delineates the merged cell $\mathcal{A}(* - * * * * -) = \mathcal{H}_2^- \cap \mathcal{H}_8^-$ and the hyperplane arrangement which spanned them.

Note that, in addition to reducing the number of regions in (2.25) comparative with (2.21b)—from 24 to 6, we have also reduced the number of hyperplanes appearing in the region's half-space representation (see Remark 2.10). These differences become more significant with a larger number of hyperplanes and of obstacles.

The same reasoning, but under the conservative view expressed in Corollary 2.1, can be applied to the problem. In the truth-table depicted in Fig. 2.8 we assign '0' whenever the cell corresponds to an obstacle and '1' otherwise. The result, once the associated Boolean function (2.26) is put into the canonical sum-of-products form:

$$f(\sigma) = \bar{\sigma}(1)\bar{\sigma}(8) + \bar{\sigma}(2)\bar{\sigma}(8) + \bar{\sigma}(4)\sigma(8) + \sigma(2)\sigma(4) \\ + \bar{\sigma}(7) + \bar{\sigma}(6) + \bar{\sigma}(5) + \bar{\sigma}(3),$$

are the merged cells $\mathcal{A}(- * * * * * *)$, $\mathcal{A}(* - * + * * * *)$, $\mathcal{A}(* * * * * -)$, $\mathcal{A}(* * - + * * * *)$, $\mathcal{A}(* * * * * - *)$, $\mathcal{A}(* * * * - * *)$, $\mathcal{A}(* * * * - * *)$ and $\mathcal{A}(- - + * * * * *)$. As before, the union of these merged cells describes the fea-

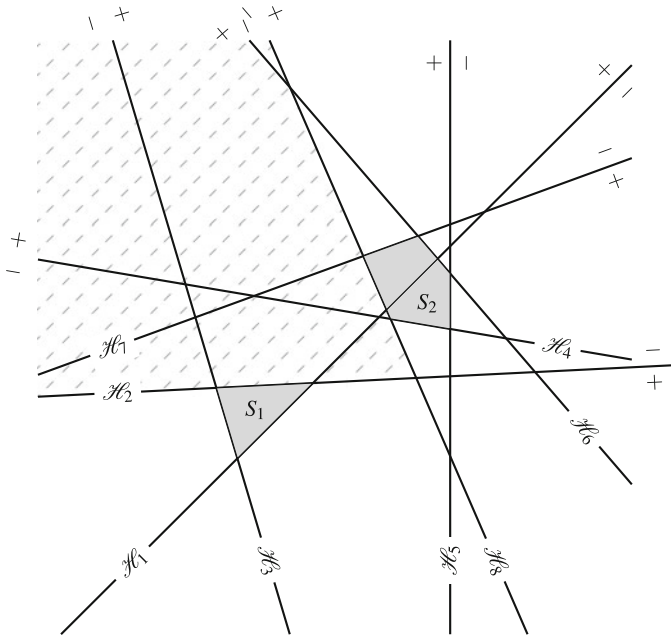


Fig. 2.7 Hyperplane arrangement with merged cells computed as in Theorem 2.2

$\sigma_1, \sigma_2, \sigma_3, \sigma_4$	-	-	+	+	-	-	+	+	-	-	+	+	-	-	+	+	-
$\sigma_5, \sigma_6, \sigma_7, \sigma_8$	-	-	-	-	+	+	-	-	+	+	-	-	+	+	-	-	+
----	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
----+	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
---++	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
---+-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
-+---	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
-++--	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
-+++-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
+-+-+	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
+-+-+	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
++---	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
++-+-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
++-++	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
+++-+	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
++--	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
++-+-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
+++-+	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
++++	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1
+++++	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

Fig. 2.8 Karnaugh diagram for obtaining the reduced cell representation with Corollary 2.1

sible region (2.21b) as in (2.25). In this particular case we observe that all the sign tuples correspond to non-empty regions (which, in general might not be the case—Remark 2.8). The drawback is that, even for this relatively small example the implicit representation is more conservative than the explicit one (8 merged cells instead of 6).

2.3.2 Numerical Considerations

The two main issues discussed in this chapter are the computation of the hyperplane arrangement (2.3) and the merging procedures which lead to compact representation (2.22). The first issue is an enumeration problem. The second issue, at least in the approach followed here, reduces to the minimization of a Boolean function. Both these problems have worst-case resolution times of exponential complexity. In practice there are algorithms which provide (sub)optimal solutions in a reasonable time. These algorithms require a broad mathematical background and fall beyond the scope of this book. Rather, we will consider several representative examples (with the numerical data provided in Appendices A.3–A.9) and apply existing algorithms to observe the computation times and various properties of interest.

The computation times are of course dependent on the hardware and software platforms and therefore the numbers obtained should not be seen as “best” values, rather they should serve to underline the link between computational difficulty and various design parameters. The numbers illustrated in the following tables have been obtained after extensive simulation in Matlab 2014a, using MPT3 toolbox [28] and the Espresso heuristic logic minimizer [29, 30]. MPT contains routines which permit the computation of hyperplane arrangements (used internally for simplifying

Table 2.1 Computation times and number of feasible cells for hyperplane arrangements in \mathbb{R}^2

(N, d)	(3, 2)	(8, 2)	(10, 2)	(15, 2)	(20, 2)	(25, 2)	(30, 2)
Card Σ	7	27	46	98	179	252	365
Card Σ cf. (2.8)	7	56	90	210	380	600	870
Time [s]	0.7314	1.2375	1.9720	3.8619	7.7412	12.7355	17.8422

piecewise affine representations [25, 31]) and the Espresso minimizer produces sub-optimal simplifications of Boolean functions which are order of magnitude faster than optimal algorithms [32].

Table 2.1 illustrates the times required for the computation of a hyperplane arrangement, starting from its collection of hyperplanes. The first row enumerates various collection of hyperplanes characterized through the pair (N, d) —number of hyperplanes and space dimension. The second and third rows provide the number of cells obtained with the MPT3 routines and as the upper bound given in (2.8). Note that the computed number of cells is usually smaller. This is due to the fact that the arrangement might not be in general position and because the MPT3 routine used counts only the cells inside a finite box (a reasonable assumption for realistic control problems—we took here a bound $[-20, 20]$ along each axis). Finally, the last row gives the computation times. As it can be seen, the values obtained are reasonable, at least for the \mathbb{R}^2 case which can cope with robotic applications for example.

In the following, the proposed merging procedures (see Theorem 2.2 and Corollary 2.1) are analyzed and solved via the Espresso minimizer. The data sent to the minimizer is divided into three disjoint categories: combinations of signs to which corresponds ‘1’—an admissible cell; combinations to which corresponds ‘0’—a forbidden cell and combinations to which corresponds ‘*’—empty cells. Furthermore, the input can be composed from any combination of these categories. Two of these cases hold interest to us:

- (i) both the ‘0’ and ‘1’ cases are given explicitly; this corresponds to Theorem 2.2,
- (ii) only the ‘0’ cases are given explicitly; this corresponds to Corollary 2.1.

Table 2.2 gives the arrangements corresponding to the examples in Table 2.1 and the number of forbidden tuples in the first two rows. The next four rows are paired two by two and provide the number of merged cells and the time to compute them in case (i) and (ii) respectively (card $\Sigma^{\bullet,1}$ is the number of forbidden tuples, card $\Sigma^*_{(i)/(ii)}$ is the number of merged cells and time_{(i)/(ii)} are the computation times).

Several observations can be made. First, the explicit approach is clearly better in terms of number of merged cells: in each case the number obtained through the implicit method is larger (sometimes much larger). Interestingly the computation times favor the explicit approaches in most all instances. Here we have to note that the explicit approach requires the computation of the arrangement as a pre-processing step. Including this additional time as well the implicit approach becomes clearly faster than the explicit one. A silver lining is the observation that the arrangement

Table 2.2 Computation times and number of merged cells for hyperplane arrangements in \mathbb{R}^2

(N, d)	(3, 2)	(8, 2)	(10, 2)	(15, 2)	(20, 2)	(25, 2)	(30, 2)
Card Σ^\bullet . ¹	1	3	4	9	9	31	22
Card $\Sigma_{(i)}^*$	3	6	8	11	10	19	16
Time _(i) [s]	0.1700	0.1061	0.1141	0.1401	0.1665	0.2632	0.3612
Card $\Sigma_{(ii)}^*$	3	8	11	16	23	45	43
Time _(ii) [s]	0.1561	0.1022	0.1113	0.1423	0.1766	0.3759	0.4665

Table 2.3 Computation times and number of merged cells for hyperplane arrangements in \mathbb{R}^2

(N, d)	(3, 2)	(8, 2)	(10, 2)	(15, 2)	(20, 2)	(25, 2)	(30, 2)
Card Σ^\bullet . ²	2	5	7	23	14	40	37
Card $\Sigma_{(i)}^*$	2	7	7	10	11	20	21
Time _(i) [s]	0.1670	0.1071	0.1062	0.1554	0.1787	0.2580	0.4630
Card $\Sigma_{(ii)}^*$	2	8	13	24	27	51	55
Time _(ii) [s]	0.1687	0.1232	0.1312	0.1616	0.1944	0.6450	1.0896

needs to be computed only once. After this operation, the obstacle classification reduces to a separation between admissible and forbidden tuples.

While the number of hyperplanes and the space dimension have a clear influence on the computation time not the same can be said about the forbidden tuples. Even keeping constant the number of hyperplanes, it is intuitive that changing the distribution and number of forbidden cells will modify the number and shape of the merged cells. To illustrate the point, in Table 2.3 we perform the same operations as in Table 2.2 but for a different collection of forbidden tuples, Σ^\bullet .², these and the initial forbidden tuples are available in Appendix A.

As expected, the number of merged cells is larger (although there is no formal relation between this number and the number of forbidden tuples). The computation times are in almost all cases larger (again, due to the increased complexity of the problem).

To have a better grasp of the computation times for the merging procedures under the various methods and initial data, Fig. 2.9 plots the relevant results provided in Tables 2.2 and 2.3.

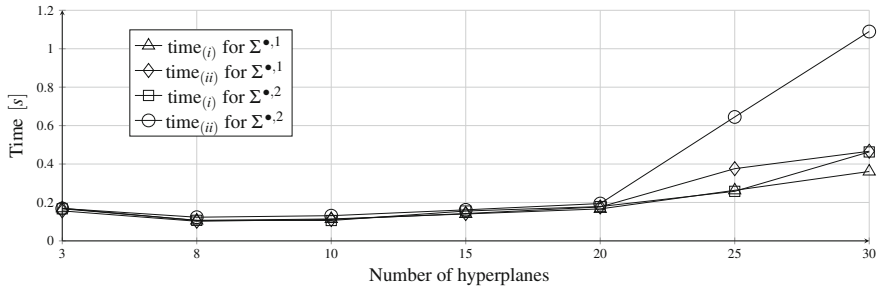


Fig. 2.9 Illustration of the merging procedure computation times provided in Tables 2.2 and 2.3

2.4 Notes and Comments

The scope of this chapter was to present basic combinatorial and set-theoretic notions necessary to characterize the feasible space (e.g., the complement of a union of obstacles). Assuming polyhedral characterization of the forbidden regions we can take the hyperplanes defining their borders and use them to create an appropriate hyperplane arrangement. From our viewpoint, the main advantage is represented by the fact that this arrangement characterizes each cell in terms on which side (either negative or positive) of the hyperplanes it lies. Hence, we no longer need to provide a geometrical description of the cell but rather of the tuple of signs which uniquely defines it. Furthermore, to characterize each of these cells as forbidden (i.e., the part of the space defining the obstacles) or feasible (i.e., the remaining part) it suffices to gather and separate the sign tuples into two disjoint collections. Moreover, since we are equally satisfied with overlapping descriptions of the feasible space we also present the notion of merged cells (hence reducing the complexity of the feasible space description).

Nevertheless, while here we presented some of the basics, extensive additional information can be gathered from a multitude of sources. Classical monographs about polyhedral or zonotopic notions can be found in [4, 9, 33] whereas, their application in control are detailed in [34, 35], amongst others. Next, detailed mathematical descriptions and various applications of hyperplane arrangements notions are provided by Zaslavsky, Orlik, Geyer, De Concini, Procesi and others in [1, 6, 24, 36–38].

References

1. Orlik, P.: Hyperplane arrangements. In: Floudas, C., Pardalos, P. (eds.) *Encyclopedia of Optimization*, pp. 1545–1547. Springer, US (2009)
2. Stanley, R.: An introduction to hyperplane arrangements. In: *Lecture notes*, IAS/Park City Mathematics Institute. Citeseer (2004)
3. Blanchini, F.: Set invariance in control—a survey. *Automatica* **35**(11), 1747–1767 (1999)

4. Ziegler, G.: Lectures on Polytopes, vol. 152. Springer (1995)
5. Birkhoff, G.: Abstract linear dependence and lattices. *Am. J. Math.* 800–804 (1935)
6. Zaslavsky, T.: Facing up to arrangements: face-count formulas for partitions of space by hyperplanes. *Am. Math. Soc.* (1975)
7. Buck, R.: Partition of space. *Am. Math. Monthly* 541–544 (1943)
8. Loechner, V.: Polylib: a library for manipulating parameterized polyhedra (1999)
9. Motzkin, T., Raiffa, H., Thompson, G., Thrall, R.: The double description method. *Contrib. Theory Games* **2**, 51 (1959)
10. Dantzig, G.: Fourier-Motzkin elimination and its dual. Technical Report DTIC Document (1972)
11. Fukuda, K.: CDD/CDD+ Reference Manual. Institute for operations Research ETH-Zentrum, Zurich (1999)
12. Jones, C.N., Kerrigan, E.C., Maciejowski, J.M.: A new algorithm for the projection of polytopes in halfspace representation. Citeseer (2004)
13. Bronstein, E.: Approximation of convex sets by polytopes. *J. Math. Sci.* **153**(6), 727–762 (2008)
14. Wilde, D.: A library for doing polyhedral operations. *Int. J. Parallel, Emergent Distrib. Syst.* **15**(3), 137–166 (2000)
15. Gritzmann, P., Klee, V.: On the complexity of some basic problems in computational convexity: I. containment problems. *Discrete Math.* **136**(1–3), 129–174 (1994)
16. Gritzmann, P., Klee, V.: On the complexity of some basic problems in computational convexity: II. volume and mixed volumes. *NATO ASI Ser. C Math. Phys. Sci. Adv. Study Inst.* **440**, 373–466 (1994)
17. Fukuda, K.: From the zonotope construction to the Minkowski addition of convex polytopes. *J. Symbolic Comput.* **38**(4), 1261–1272 (2004)
18. Fukuda, K.: Polytope examples. <ftp://ftp.ifor.math.ethz.ch/pub/fukuda/reports/polyfaq041121.pdf>
19. Avis, D., Fukuda, K.: Reverse search for enumeration. *Discrete Appl. Math.* **65**(1), 21–46 (1996)
20. Alamo, T., Bravo, J., Camacho, E.: Guaranteed state estimation by zonotopes. *Automatica* **41**(6), 1035–1043 (2005)
21. Dang, T.: Approximate reachability computation for polynomial systems. *Hybrid Syst. Comput. Control* 138–152 (2006)
22. Bourgain, J., Lindenstrauss, J.: Distribution of points on spheres and approximation by zonotopes. *Isr. J. Math.* **64**(1), 25–31 (1988)
23. Linhart, J.: Approximation of a ball by zonotopes using uniform distribution on the sphere. *Archiv der Mathematik* **53**(1), 82–86 (1989)
24. Geyer, T., Torrisi, F., Morari, M.: Optimal complexity reduction of piecewise affine models based on hyperplane arrangements. In: *Proceedings of the 23th American Control Conference*, vol. 2, pp. 1190–1195. Boston (2004)
25. Geyer, T., Torrisi, F., Morari, M.: Optimal complexity reduction of polyhedral piecewise affine systems. *Automatica* **44**(7), 1728–1740 (2008)
26. Prodan, I., Stoican, F., Olaru, S., Niculescu, S.I.: Enhancements on the hyperplanes arrangements in mixed-integer techniques. *J. Optim. Theory Appl.* **154**(2), 549–572 (2012)
27. Brayton, R., Hachtel, G., McMullen, C., Sangiovanni-Vincentelli, A.: Logic minimization algorithms for VLSI synthesis, vol. 2. Springer (1984)
28. Herceg, M., Kvasnica, M., Jones, C., Morari, M.: Multi-parametric toolbox 3.0. In: *Proceedings of the European Control Conference*, pp. 502–510. Zürich (2013). <http://control.ee.ethz.ch/~mpt>
29. McGeer, P., Sanghavi, J., Brayton, R., Sangiovanni-Vincentelli, A.: Espresso-signature: a new exact minimizer for logic functions. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **1**(4), 432–440 (1993)
30. Hlavíčka, J., Fišer, P.: Boom: a heuristic boolean minimizer. In: *Proceedings of the 2001 IEEE/ACM International Conference On Computer-aided Design*, pp. 439–442. IEEE Press (2001)

31. Geyer, T., Torrisi, F., Morari, M.: Efficient mode enumeration of compositional hybrid systems. *Int. J. Control* **83**(2), 313–329 (2010)
32. Kunz, W., Stoffel, D.: Logic optimization. In: *Reasoning in Boolean Networks*, pp. 101–161. Springer (1997)
33. Schneider, R.: *Convex Bodies: the Brunn-Minkowski Theory*. Cambridge University Press (1993)
34. Althoff, M., Stursberg, O., Buss, M.: Computing reachable sets of hybrid systems using a combination of zonotopes and polytopes. *Nonlinear Anal. Hybrid Syst.* **4**(2), 233–249 (2010)
35. Blanchini, F., Miani, S.: *Set-Theoretic Methods In Control*. Birkhauser (2007)
36. Edelsbrunner, H., Seidel, R., Sharir, M.: On the zone theorem for hyperplane arrangements. *New Results New Trends Comput. Sci.* 108–123 (1991)
37. Orlik, P., Terao, H.: *Arrangements of Hyperplanes*, vol. 300. Springer (1992)
38. De Concini, C., Procesi, C.: *Topics in Hyperplane Arrangements, Polytopes and Box-Splines*. Springer (2010)

Mixed-Integer Representations in Control Design

Mathematical Foundations and Applications

Prodan, I.; Stoican, F.; Olaru, S.; Niculescu, S.-I.

2016, XII, 107 p. 30 illus. in color., Softcover

ISBN: 978-3-319-26993-1