2013 European Control Conference (ECC)
July 17-19, 2013, Zürich, Switzerland.

Hyperplane Arrangements in Mixed-Integer Programming Techniques. Collision avoidance application with Zonotopic Sets.

Florin Stoican[†,*], Ionela Prodan[‡], Sorin Olaru[‡];

***Abstract*— The current paper addresses the problem of minimizing the computational complexity of optimization problems with non-convex and possibly non-connected feasible region of polyhedral type. Using hyperplane arrangements and Mixed-Integer Programming we provide an efficient description of the feasible region in the solution space. Moreover, we exploit the geometric properties of the hyperplane arrangements and adapt this description in order to provide an efficient solution of the mixed-integer optimization problem. Furthermore, a zonotopic representation of the sets appearing in the problem is considered. The advantages of this representation are highlighted and exploited through proof of concepts illustrations as well as simulation results.**

I. Introduction

It is often the case that optimization problems arising in control theory have to be solved over a non-convex feasible region. This issue arises naturally in many control engineering problems like persisting exciting control [1], robust fault detection [2] or the control of multi-agent systems [3] where collision and obstacle avoidance conditions appear and result in non-convex constraints. It is important to emphasize that, these constraints are not just an artifact of the problem, but rather an intrinsic property which cannot be avoided. They are used explicitly in the Mixed-Integer (MI) formulations [4], [5], and implicitly in the Potential Field formulations [6], but regardless of the approach, they are always present.

Solving optimization problems over non-convex regions is not a new issue in the literature and, it is well known that MI formulations provide one of the best ways of dealing with this type of problems [7], at least from the point of view of the feasible domain representation. The advantage is that the problem is reformulated into a mixed continuous+binary form which is manageable for reasonable dimensions and permits the use of efficient algorithms. However, the computational complexity is highly dependent on the size of the binary part and limits its usefulness to small-size problems. There are works that try to reduce the number of binary variables used in the problem formulation. For example, in [8] and the references therein, a logarithmic formulation is discussed. This and other techniques help to reduce the computational burden but, ultimately, the complexity of the problem is directly related with the difficulty of representing the feasible region. In [9], a variant of the logarithmic formulation is revisited and geometric insights into the nature of the feasible region help to provide an efficient MI description. The idea was to use the notion of hyperplane arrangements [10] in order to describe in a formal way the *non-connected and non-convex* feasible region.

In the present paper we revisit the results in [9] and we go further in improving the novel (to the best of the authors' knowledge) geometrical interpretations. We concentrate on compact ways of describing the feasible region as a way of reducing the number of binary variables. In particular, we aim to avoid decomposing the hyperplane arrangement into cells (as was done in [9]). The solution we propose here is to never calculate the collection of cells describing the feasible region, which turns to be a computationally demanding pre-processing. Rather, by using the set of cells which describe the interdicted regions (usually containing many fewer cells than the ones of the feasible region) and *boolean algebra* notions we provide a compact representation. In fact, and this is a central point of the paper, we make the MI formulation dependent on the complexity of the feasible region rather than on the complexity of the infeasible region.

To better illustrate these results we consider a multi-agent optimization problem with zonotopic constraints. This allows us to show both a typical genesis for a non-convex feasible region and the importance of choosing a particular family of sets.

Notation: The closure of a set S, $cl(S)$, is the intersection of all closed sets containing S. The collection of all possible combinations of N binary variables will be noted $\{0,1\}^N := \{(b_1,\dots,b_N) :\ b_i \in \{0,1\}, \forall i = 1,\dots,N\}$. The same definition holds for the set of sign tuples $\{-,+\}^N$. For a scalar $x \in \mathbb{R}$, we denote by $\lceil x \rceil$ the upper integer part. $\#S$ denotes the number of elements of set S.

II. Preliminaries

Let us consider a collection of hyperplanes from $\mathbb{R}^n$

$$\mathcal{H}_i = \{x \in \mathbb{R}^n :\ h_i x = k_i\},\ i \in \mathcal{I} \tag{1}$$

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

† Norwegian University of Science and Technology (NTNU) - Department of Engineering Cybernetics, Trondheim, Norway;
* The author is currently with the Automatic Control and Systems Engineering Department, UPB, Romania florin.stoican@acse.pub.ro;
‡ SUPELEC Systems Sciences (E3S) - Automatic Control Department, Gif sur Yvette, France {ionela.prodan,sorin.olaru}@supelec.fr

Each of these hyperplanes partitions the space into two disjunct[1] regions (which halve the space and hence are called "half-spaces"):

$$\mathcal{R}_i^+ = \{x \in \mathbb{R}^n : \quad h_i x \leq \quad k_i\}, \tag{2a}$$

$$\mathcal{R}_i^- = \{x \in \mathbb{R}^n : -h_i x \leq -k_i\}. \tag{2b}$$

In the following, we define a polytope, $P \subset \mathbb{R}^n$, as a bounded intersection of half-spaces[2]:

$$P = \bigcap_{i \in \mathcal{I}} \mathcal{R}_i^+, \tag{3}$$

and its complement $\mathcal{C}_X(P) \triangleq cl(X \setminus P)$ over $X \subseteq \mathbb{R}^n$ as the union[3] of regions that cover all space except P:

$$\mathcal{C}_X(P) = \mathcal{C}_X\left(\bigcap_{i \in \mathcal{I}} \mathcal{R}_i^+\right) = \bigcup_{i \in \mathcal{I}} \mathcal{C}_X\left(\mathcal{R}_i^+\right) = \bigcup_{i \in \mathcal{I}} \mathcal{R}_i^-. \tag{4}$$

We use the reduced notation $\mathcal{C}(P)$ whenever X is presumed known or is considered to be the entire space $\mathbb{R}^n$.

As explained in the introduction, it is often the case that the feasible region of an optimization problem is non-convex and is actually the complement of some convex set. With the present notation, this feasible region is (4) and in order to reach a tractable formulation, we have to use mixed integer techniques. By adding the binary variables $(\alpha_1 \ldots \alpha_N) \in \{0,1\}^N$ we obtain the polytopic set in the extended space of *state + auxiliary binary variables*

$$-h_i x \leq -k_i + M\alpha_i, \; i \in \mathcal{I}, \tag{5a}$$

$$\sum_{i \in \mathcal{I}} \alpha_i \leq N - 1, \tag{5b}$$

with M a constant chosen appropriately (that is, significantly larger than the rest of the variables and playing the role of a relaxation constant – hence, the "big-M" name for this type of MI formulation).

Remark 1. Constraints (5a)–(5b) can be projected into the original feasible region (4) by suitable choices of the binary variables. E.g., a region $\mathcal{R}_i^-$ can be obtained from (5a) by letting the associated binary variable to be "0":

$$(\alpha_1 \ldots \alpha_N) = (1 \ldots 1 \underbrace{0}_{i} 1 \ldots, 1). \tag{6}$$

Note that the converse is false since no choice of binary variables can lead to the description of a region $\mathcal{R}_i^+$. If a binary variable is "1", the corresponding inequality degenerates such that it covers any point $x \in \mathbb{R}^n$ (this represents the limit case for $M \to \infty$). The condition (5b) is thus required to ensure that at least one binary value is "0" and, consequently, that at least one inequality is verified. ♦

[1]The relative interiors of these regions do not intersect, but their closures have as a common boundary the affine subspace $\mathcal{H}_i$.

[2]The "+" superscript was chosen for the homogeneity of notation, equivalently one could have chosen any combination of signs from (2a)–(2b) in order to describe the polytope P.

[3]The union and intersection operators interchange under complementation.

A. The complement of a union of convex sets

The same reasoning can be applied to the more general case when the interdicted region is defined as a union of polytopes:

$$\mathbb{P} = \bigcup_l P_l, \tag{7}$$

with $P_l = \bigcap_{i_l \in \mathcal{I}_l} R^+(\mathcal{H}_{i_l})$ and $\mathcal{I}$ redefined for convenience as $\mathcal{I} = \bigcup_l \mathcal{I}_l$.

Its complement, $\mathcal{C}_X(\mathbb{P}) = cl(X \setminus \mathbb{P})$, is defined by

$$\mathcal{C}_X(\mathbb{P}) = \mathcal{C}_X\left(\bigcup_l P_l\right) = \bigcap_l \mathcal{C}_X(P_l) = \bigcap_l \left(\bigcup_{i_l \in \mathcal{I}_l} R_{i_l}^-\right) \tag{8}$$

and can be described by a MI formalism as in (5a)–(5b) by replacing the index i with i_l for all $l \in \{1, 2 \ldots\}$:

$$-h_{i_l} x \leq -k_{i_l} + M\alpha_{i_l}, \; \forall i_l \in \mathcal{I}_l, \tag{9a}$$

$$\sum_{i_l \in \mathcal{I}_l} \alpha_{i_l} \leq \#\mathcal{I}_l - 1. \tag{9b}$$

B. Logarithmic formulation

As it can be seen in the representation (5a)–(5b), a binary variable is associated to each region of form (2b) in the description of the feasible region (4). Obviously, this may increase the number of binary variables significantly. A commonly used solution is the "logarithmic formulation" of the binary part.

Let us recall and adapt Proposition 3.1 from [9].

Proposition 1. *For each region $\mathcal{R}_i^-$ from (4) we associate a unique combination of binary variables $\lambda^i \in \{0,1\}^{\lceil log_2 N \rceil}$. Then, we can construct the affine functions $\alpha_i : \{0,1\}^{\lceil log_2 N \rceil} \to \{0\} \cup [1, \infty)$:*

$$\alpha_i(\lambda) = \sum_{k=0}^{\lceil log_2 N \rceil} \left(\lambda_k^i + (1 - 2\lambda_k^i) \cdot \lambda_k\right). \tag{10}$$

Index 'k' denotes the k^{th} variable and λ_k^i its value for the tuple λ^i, associated to region $\mathcal{R}_i^-$.

Proof. See the proof of Proposition 3.1 of [9]. ■

Illustrative example

For the purpose of illustration let us consider two simple examples. First we take a triangle in $\mathbb{R}^2$ represented as in (3) ($\{h_i x \leq k_i\}$ with $i = 1, 2, 3$) and depict it in Fig. 1. Then the MI formulation can be written as in (5a)–(5b):

$$-h_1 x \leq -k_1 + M\alpha_1, \tag{11a}$$

$$-h_2 x \leq -k_2 + M\alpha_2, \tag{11b}$$

$$-h_3 x \leq -k_3 + M\alpha_3, \tag{11c}$$

$$\alpha_1 + \alpha_2 + \alpha_3 \leq 2. \tag{11d}$$

As it can be seen, we need 3 binary variables, one for each constraint, and an additional constraint – (11d) to force at least one of the regions (2b) to be active at any time. An alternative construction is the one provided in Proposition 1 with only $\lceil log_2 3 \rceil = 2$ binary variables:

$$\begin{aligned} &- h_1 x \leq -k_1 + M(\quad \lambda_1 + \lambda_2), && \text{(12a)}\\ &- h_2 x \leq -k_2 + M(1 - \lambda_1 + \lambda_2), && \text{(12b)}\\ &- h_3 x \leq -k_3 + M(1 + \lambda_1 - \lambda_2), && \text{(12c)}\\ &2 - \lambda_1 - \lambda_2 > 0. && \text{(12d)} \end{aligned}$$

Again, a constraint is needed in order to make the problem well-posed. Constraint (12d) makes $(\lambda_1, \lambda_2) = (1, 1)$ infeasible and thus, assures that at least one of the constraints (12a)–(12c) is active.

In Fig. 1 we highlight region R_2^- and show to which combination of binary variables it corresponds in either formulation (11) or (12). For the first case, R_2^- is obtained by taking $(\alpha_1, \alpha_2, \alpha_3) = (1, 1, 0)$ whereas for the second case we have $(\lambda_1, \lambda_2) = (1, 0)$.

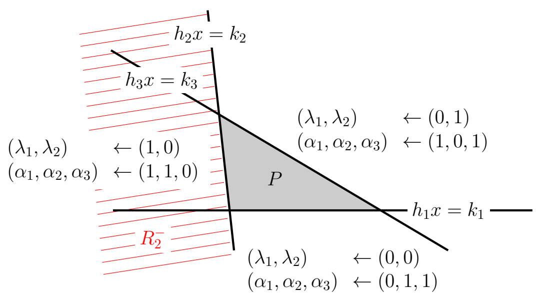

Fig. 1: Illustration of MI formulation for (4) in both classical and logarithmic formulation.

A more complex example is depicted in Fig. 2. The interdicted region (7) is composed from 2 polytopes (the first defined by $\{h_i x \leq k_i\}$ for $i = 1, 2, 3$ and the latter by $\{h_i x \leq k_i\}$ for $i = 4, 5, 6, 7$).

The feasible region (8) is described as in the MI formulation (9a)–(9b) and we know that for various combinations of binary variables we can cover any part of the feasible region. For example, by taking $(\alpha_1, \alpha_2, \alpha_3) = (0, 1, 1)$ and $(\alpha_4, \alpha_5, \alpha_6, \alpha_7) = (1, 0, 1, 1)$ we are able to describe $R_1^- \cap R_5^-$, as depicted in Fig. 2.

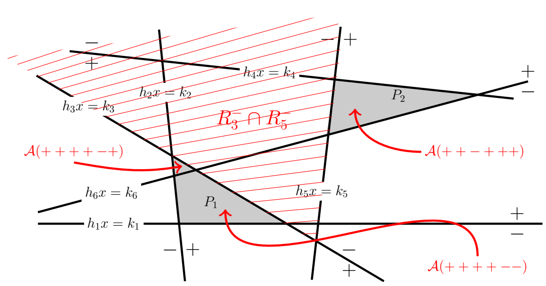

Fig. 2: Illustration of MI formulation for (8) and hyperplane arrangement description.

III. Hyperplane arrangements

Although the constructions given in Section II are adequate, we still wish to improve upon the formulation in order to reduce the number of binary variables and thus, the computational burden. The issue is that[4] (9a)–(9b) describe the feasible set as "whatever is out of the interdicted region" which means that the number of binary variables depends on the complexity of the interdicted region rather than on the complexity of the feasible region.

Here we aim to give an explicit description of (8) as a union of polyhedral sets. To this end, let us denote $\mathbb{H} = \{\mathcal{H}_i\}_{i \in \mathcal{I}}$ as the collection of all hyperplanes appearing in the description of polytopes P_l from (7) and introduce the following combinatorial notion.

Definition 1 (Hyperplane arrangements – [11])**.** The collection of hyperplanes $\mathbb{H}$ will partition the space into a union of disjoint cells $\mathcal{A}(\sigma)$ characterized by a sign tuple $\sigma \in \{-, +\}^N$ and defined as follows:

$$\mathcal{A}(\sigma) = \bigcap_{i \in \mathcal{I}} \mathcal{R}_i^{\sigma(i)}. \tag{13}$$

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

$$\mathcal{A}(\mathbb{H}) = \bigcup_{l = 1 \ldots \gamma(N)} \mathcal{A}(\sigma_l), \tag{14}$$

where $\sigma_l \in \{-, +\}^N$ denotes sign tuples resulting into non-empty intersections of regions (2a)–(2b). ♦

Remark 2. The number of feasible cells, $\gamma(N)$, (in relation with the space dimension – d and the number of hyperplanes – N) is bounded by Buck's formula [12]:

$$\gamma(N) \leq \sum_{i=0}^{d} \binom{N}{i}, \tag{15}$$

with equality satisfied if the hyperplanes are in general position[5] and $X = \mathbb{R}^n$. ♦

Incidentally, this representation permits an equivalent (with respect to (9a)–(9b)) MI construction.

Proposition 2. *Consider the interdicted region (7) and its associated hyperplane arrangement (14). We denote the subset of sign tuples*

$$\Sigma_{\mathbb{P}} = \{\sigma_l : \mathcal{A}(\sigma_l) \subseteq \mathbb{P}\}, \tag{16}$$

as the tuples defining cells from (14) which are inside $\mathbb{P}$.

Then, the feasible region (8) is given by the following

[4]We take the case (9a)–(9b) rather than (5a)–(5b) in order to keep a general formulation.

[5]We call a hyperplane arrangement to be "in general position" whenever any small change in the position of the composing hyperplanes does not change the number of cells.

MI formulation:

$$h_i x \leq \quad k_i + M(1-\alpha_i), \tag{17a}$$
$$-h_i x \leq -k_i + M\alpha_i, \tag{17b}$$
$$\sum_{\sigma_l(i)='+'} (1-\alpha_i) + \sum_{\sigma_l(i)='-'} \alpha_i > 0,\ \forall \sigma_l \in \Sigma_{\mathbb{P}}. \tag{17c}$$

Proof. Constraints (17a)–(17b) assure that either region $\mathcal{R}_i^+$ or $\mathcal{R}_i^-$ is active, depending on the value taken by the associated binary variable α_i. Considering only (17a)–(17b) means that the entire space will be covered, including the interdicted region (7). To avoid this, we add conditions (17c) which, by making infeasible the binary codes associated with interdicted cells, assure that only the cells describing (8) are feasible. ■

Remark 3. In general, MI representation (9a)–(9b) should be more compact than (17a)–(17c) in the number of necessary binary variables (especially if for the former we use the logarithmic formulation). Nonetheless, the latter's construction is interesting since the number of binary variables remains fix, no matter how many cells are labeled as interdicted. Note that the number of interdiction constraints (17c) can be further reduced if by a single constraint more than one point is made infeasible (see Proposition 3.2 of [9]). ♦

The arrangement (14) can be used to obtain a compact description of (8) by merging neighboring disjoint cells. This is efficient due to their characteristics, i.e., the union of any two neighboring cells is convex. In fact, we can formally define a merged cell as follows.

Definition 2. The union of all cells $\mathcal{A}(\sigma)$ whose sign tuples retain the same values over a subset of indices $(i \in \mathcal{I})$ and span all possible combinations for the rest of indices $(j \in \mathcal{I})$ is a "merged" cell and is defined as:

$$\mathcal{A}(\sigma^*) = \bigcup_{\sigma} \mathcal{A}(\sigma) = \bigcap_{\sigma^*(i) \neq '*',\, i \in \mathcal{I}} \mathcal{R}_i^{\sigma^*(i)}, \tag{18}$$

where $\sigma^* \in \{-,*,+\}^N$ denotes the sign tuple associated with the merged cell. That is, we define σ^* as $\sigma^*(i) = \sigma(i)$ and as $\sigma^*(j) = '*'$ with indices i, j described as above.♦

The main issue for the approach presented in [9] is that in order to obtain the merged cells (18) we had to calculate first the cells (13). Efficient algorithms exist (e.g., a reverse search was presented in [13]) but still the time increases exponentially in number of hyperplanes "N" and space dimension "d".

The idea we propose here is to never actually calculate the collection of feasible sign tuples. Rather, by using the set of cells which describe (7) (usually containing many fewer cells than the total number (15)) and boolean algebra notions we provide a compact representation of (8) without explicitly making the decomposition (14).

Theorem 1. *Consider the interdicted region (7) characterized by the sign tuples (16). Then, the feasible region (8) is compactly described as a union of merged cells (18):*

$$\mathcal{C}_X(\mathbb{P}) = \bigcup_{j \in \mathcal{J}} \mathcal{A}(\sigma_j^*), \tag{19}$$

where $\sigma_j^* \in \{-,*,+\}^N$ are the sums from the "product-of-sums" representation of the boolean function $f : \{-,+\}^N \to \{0,1\}$ verifying

$$f(\sigma_l) = 0,\ \forall \sigma_l \in \Sigma_{\mathbb{P}}. \tag{20}$$

$\mathcal{J}$ denotes the set of all the indices of the merged cells.

Proof. Let us consider function (20) and its truth-table: we have "0" for binary combinations describing a cell inside (7) and "1" in all the other cases. All that remains is to group the combinations which are "1" and express the function in a product-of-sums form. Each term of the product describes a region of form (18) which is either i) an empty set (infeasible combination of regions) or ii) a non-empty set (a feasible combination of regions). ■

Remark 4. In the construction of the truth table of (20) as explained in the proof of Theorem 1 lies the major difference with respect to [9]. By not computing the cells composing the feasible region, we assume implicitly (and conservatively) that all the remaining combinations of binary variables are feasible (thus "1" in the table). This means that we obtain merged cells which may be infeasible. Such a cell is denoted as feasible (and discarded otherwise) at a later post-processing stage, in contrast with the approach in [9] where the validation is made in a pre-processing stage. ♦

Having obtained the merged cells of (19) it is straightforward to associate[6] to each of them a binary variable as in (5a)–(5b):

$$-h_{j_l} x \leq -k_{j_l} + M\alpha_j,\ \forall j_l \text{ s.t. } \sigma_j^*(j_l) \neq '*', \tag{21a}$$
$$\sum_j \alpha_j \leq \#\mathcal{J} - 1. \tag{21b}$$

This leads to a more compact representation (i.e, less binary variables) than in (9a)–(9b) with obvious benefits for the computation times. The number of binary variables can be further reduced by employing the logarithmic formulation of Subsection II-B.

Remark 5. For a compact representation we wish to write (20) in a minimal product-of-sums form. To this end, we have to apply minimization algorithms (e.g., Karnaugh maps, the Quine-McCluskey algorithm or the Espresso heuristic logic minimizer). We emphasize on the latter technique which although heuristic gives near-optimal results and is orders of magnitude faster than other methods [14]. We note that a similar approach was proposed in [15] in order to deal with polyhedral piecewise affine systems. ♦

[6] The binary variable will make active not only a constraint but rather all the constraints defining a merged cell.

Illustrative example

Let us recall the previous illustrative example. We have shown in Fig. 2 the sign tuples associated with each of the hyperplanes. By convention we are choosing '+' to describe the half-spaces defining the polytopes of the union.

We note that the interdicted region (i.e., polytopes P_1 and P_2) is described by 3 cells associated with sign tuples $(+ + + + --)$, $(+ + + + -+)$ and $(+ + - + ++)$. This means that in formulation (17), the constraints that will make the cells which are inside $P_1 \cup P_2$ un-selectable are:

$$4 - \alpha_1 - \alpha_2 - \alpha_3 - \alpha_4 + \alpha_5 + \alpha_6 > 0, \tag{22a}$$
$$5 - \alpha_1 - \alpha_2 - \alpha_3 - \alpha_4 + \alpha_5 - \alpha_6 > 0, \tag{22b}$$
$$5 - \alpha_1 - \alpha_2 + \alpha_3 - \alpha_4 - \alpha_5 - \alpha_6 > 0. \tag{22c}$$

In order to obtain the compact description given in Theorem 1 we construct the boolean function (20) and obtain the maxterms defining it.

After applying the minimization (using, e.g., the Espresso heuristic minimizer) we obtain 6 merged cells defined by the sign tuples: $(- * * * **)$, $(* - * * **)$, $(* * * - **)$, $(* * - * -*)$, $(* * - * * -)$ and $(+ + + * + *)$. The first five denote non-empty regions (with the fourth being $\mathcal{R}_3^- \cap \mathcal{R}_5^-$ highlighted in Fig. 2) and the last one has no physical meaning, the combination of half-spaces which it describes being an empty set.

In the end we remain with 5 feasible merged cells which means that in logarithmic formulation, only[7] $\lceil log_2 5 \rceil = 3$ are needed and the MI formulation is as follows:

$$-h_1 x \leq -k_1 + M(\quad \lambda_1 + \lambda_2 + \lambda_3), \tag{23a}$$
$$-h_2 x \leq -k_2 + M(1 - \lambda_1 + \lambda_2 + \lambda_3), \tag{23b}$$
$$-h_4 x \leq -k_4 + M(1 + \lambda_1 - \lambda_2 + \lambda_3), \tag{23c}$$
$$\begin{matrix} -h_3 x \leq -k_3 \\ -h_5 x \leq -k_5 \end{matrix} + M(1 + \lambda_1 + \lambda_2 - \lambda_3), \tag{23d}$$
$$\begin{matrix} -h_5 x \leq -k_5 \\ -h_6 x \leq -k_6 \end{matrix} + M(2 - \lambda_1 - \lambda_2 + \lambda_3), \tag{23e}$$
$$\begin{matrix} 2 - \lambda_1 + \lambda_2 - \lambda_3 \\ 2 + \lambda_1 - \lambda_2 - \lambda_3 > 0 \\ 3 - \lambda_1 - \lambda_2 - \lambda_3 \end{matrix} \tag{23f}$$

IV. Collision avoidance with zonotopic sets

Multi-agent problems are one of the topics which provide non-convex formulations naturally: obstacle and collision avoidance requirements are modeled by non-convex constraints. In general, the complexity of the resulting problem is affected by the sets chosen to describe obstacles and/or safety regions around agents. In what follows we will consider zonotopic sets due to their interesting characteristics. They represent a particular class of polytopes (are the projection of a hypercube from a higher dimension) and can be described in "generator form" as

$$Z(c, G) = \left\{ x \in \mathbb{R}^n : \; x = c + \sum_{i=1}^{m} \lambda_i g_i \,, |\lambda_i| \leq 1 \right\}, \tag{24}$$

with $i = 1, \dots, m$ and where $c \in \mathbb{R}^n$ represents the center and $G = \begin{bmatrix} g_1 & \dots & g_m \end{bmatrix} \in \mathbb{R}^{n \times m}$ the matrix of generators. A zonotope defined as in (24) has some interesting properties [16]:

- is closed under linear transformations:
$$LZ(c, G) = Z(Lc, LG); \tag{25}$$
- is closed under the Minkowski sum:
$$Z(c_1, G_1) \oplus Z(c_2, G_2) = Z(c_1 + c_2, \begin{bmatrix} G_1 & G_2 \end{bmatrix}). \tag{26}$$

A zonotope cannot in general approximate arbitrary well all convex shapes (as a polytope does) but can nonetheless provide over-approximations of these shapes. We consider this approach for a collection of safety regions around agents and obstacles. We define these sets as $Z(x_i, G_i)$ and $Z(o_l, G_l^o)$ respectively. That is, around each agent $i \in \{1 \dots N_a\}$ with position x_i there is a zonotopic safety region with generators G_i and each obstacle $l \in \{1 \dots N_o\}$ is described by a fix center o_l and the generators G_l^o.

If the generators' directions are fixed, an optimization problem can scale them such that they over-approximate efficiently a given shape [17]. Using the same construction here we assume a common "seed", the generators G, and Δ_i and Δ_l^o the scaling factors (diagonal matrices with positive elements) for agent i and obstacle l respectively. Then, the collision and obstacle avoidance conditions can be written as follows:

$$Z(x_i, G\Delta_i) \cap Z(o_l, G\Delta_l^o) = \emptyset, \; \forall i, l, \tag{27a}$$
$$Z(x_i, G\Delta_i) \cap Z(x_j, G\Delta_j) = \emptyset, \; \forall i, j \text{ and } i \neq j. \tag{27b}$$

Further, using the zonotope properties (25)–(26) we reach the equivalent formulation:

$$x_i \notin Z(o_l, G(\Delta_i + \Delta_l^o)), \; \forall i, l, \tag{28a}$$
$$x_i - x_j \notin Z(0, G(\Delta_i + \Delta_j)), \; \forall i, j \text{ and } i \neq j. \tag{28b}$$

Conditions (28a)–(28b) have to be rewritten into a half-space representation in order to be used as in Section III. As per [17], each combination of $n-1$ generators uniquely defines a pair of half-spaces. For a zonotope defined as in (24), these half-space are given as $h_i x \leq k_i^+$ and $h_i x \leq k_i^-$ with

$$h_i = \left(G^{\mathbf{i}}\right)^{\perp} / || \left(G^{\mathbf{i}}\right)^{\perp} ||_2, \tag{29a}$$
$$k_i^{\pm} = \pm h_i^T \cdot c + \sum_{i \in \mathbf{i}} |h_i^T g_i|, \tag{29b}$$

where $\mathbf{i}$ represents a subset of $n-1$ indices from $\{1 \dots m\}$ and $G^{\mathbf{i}}$ represents the associated subset of generators from G. With the notation from (29a), h_i represents the

[7] This is less than the $\lceil log_2 3 \rceil + \lceil log_2 3 \rceil = 4$ binary variables needed for construction (9a)–(9b) in logarithmic formulation or the 6 needed in construction (17).

normed vector perpendicular on the subspace generated by the $n-1$ generators from $G^{\mathbf{i}}$.

In here we can see the advantage of a zonotopic representation with respect to the politopic approach. By keeping a common "seed" of generators we have that each the half-spaces that they generate share common normal vectors no matter which are the scaling factors. Thus, for any of the zonotopes from (28a)–(28b) we have half-spaces parallel with each other which is important for the hyperplane arrangement construction since it reduces the number of cells and thus simplifies the formulation.

Illustrative example

We consider a simple example with a single agent and 2 obstacles defined by the LTI dynamics:

$$x^+ = \begin{bmatrix} 1 & -0.3 \\ 0.13 & 0.5 \end{bmatrix} x + \begin{bmatrix} 0.14 \\ 0.53 \end{bmatrix} u$$

and with constraints $-0.5 \leq u \leq 0.5$.

We describe these obstacles by over-approximating them with zonotopic sets as in Section IV. We take 3 generators $G = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$ and find the scalings which best approximate the obstacles: $\Delta_1 = diag(\begin{bmatrix} 1 & 0.25 & 0.5 \end{bmatrix})$ and $\Delta_2 = diag(\begin{bmatrix} 0.25 & 0.5 & 1 \end{bmatrix})$.

Further, we enumerate the half-spaces as in (29a)–(29b) and describe the feasible region as in Section III. Applying an MPC optimization problem ($Q = I_2$, $R = 1$ and $N = 5$) over these non-convex constraints we obtain trajectories which avoid the obstacles and converge towards the origin. These results are depicted in Fig. 3. We show the zonotpic sets over-approximating the obstacles (together with their generators) and trajectories which avoid them.

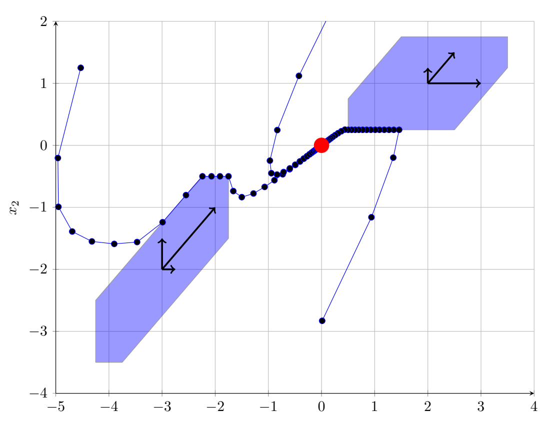

Fig. 3: Illustration of obstacle avoidance with zonotopic constraints.

V. Conclusions

The current paper addresses the problem of minimizing the computational complexity of optimization problems under a non-convex and possibly non-connected feasible region. Using the hyperplane arrangements notion we provide a compact description of the feasible region and thus obtain a compact MI formulation with fewer binary variables. We emphasize the advantage of constructing the MI formulation by considering an explicit description of the feasible region rather than looking at it as the complement of the infeasible one. Lastly, we apply these results over an obstacle avoidance problem where the obstacles are described by zonotopic sets.

Acknowledgements

The research of Ionela Prodan is financially supported by the EADS Corporate Foundation (091-AO09-1006).

References

[1] G. Marafioti, F. Stoican, R. R. Bitmead, and M. Hovd, "Persistently exciting model predictive control for siso systems," in *4th IFAC Nonlinear Model Predictive Control Conference International Federation of Automatic Control*, Noordwijkerhout, Netherlands, August 23-27 2012, pp. 448–453.

[2] F. Stoican, S. Olaru, M. M. Seron, and J. A. De Doná, "Reference governor design for tracking problems with fault detection guarantees," *Journal of Process Control*, vol. 22, no. 5, pp. 829–836, 2012.

[3] D. Grundel, R. Murphey, and P. Pardalos, *Cooperative systems, Control and optimization*. Springer, 2007, vol. 588.

[4] I. Prodan, S. Olaru, C. Stoica, and S.-I. Niculescu, "On the tight formation for multi-agent dynamical systems," in *KES – Agents and Multi-agent Systems – Technologies and Applications*. Springer, 2012.

[5] I. Prodan, G. Bitsoris, S. Olaru, C. Stoica, and S. Niculescu, "On the limit behavior for multi-agent dynamical systems," in *The IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles*, Porto, Portugal, 10-12 April 2012.

[6] I. Prodan, S. Olaru, S. Stoica, and S.-I. Niculescu, "Predictive Control for Trajectory Tracking and Decentralized Navigation of Multi-Agent Formations," *International Journal of Applied Mathematics and Computer Science*, vol. 23, no. 1, pp. 91–102, 2013.

[7] M. Jünger, T. Liebling, D. Naddef, G. Nemhauser, and W. Pulleyblank, *50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art*. Springer, 2009.

[8] J. Vielma and G. Nemhauser, "Modeling disjunctive constraints with a logarithmic number of binary variables and constraints," *Mathematical Programming*, vol. 128, no. 1, pp. 49–72, 2011.

[9] I. Prodan, F. Stoican, S. Olaru, and S. Niculescu, "Enhancements on the Hyperplanes Arrangements in Mixed-Integer Techniques," *Journal of Optimization Theory and Applications*, vol. 154, no. 2, pp. 549–572, 2012.

[10] P. Orlik, "Hyperplane arrangements," in *Encyclopedia of Optimization*, C. A. Floudas and P. M. Pardalos, Eds. Springer US, 2009, pp. 1545–1547.

[11] G. M. Ziegler, *Lectures on polytopes*. Springer, 1995, vol. 152.

[12] R. Buck, "Partition of space," *American Mathematical Monthly*, pp. 541–544, 1943.

[13] D. Avis and K. Fukuda, "Reverse search for enumeration," *Discrete Applied Mathematics*, vol. 65, no. 1, pp. 21–46, 1996.

[14] R. Brayton, G. Hachtel, C. McMullen, and A. Sangiovanni-Vincentelli, *Logic minimization algorithms for VLSI synthesis*. Springer, 1984, vol. 2.

[15] T. Geyer, F. Torrisi, and M. Morari, "Optimal complexity reduction of polyhedral piecewise affine systems," *Automatica*, vol. 44, no. 7, pp. 1728–1740, 2008.

[16] K. Fukuda, "From the zonotope construction to the Minkowski addition of convex polytopes," *Journal of Symbolic Computation*, vol. 38, no. 4, pp. 1261–1272, 2004.

[17] M. Althoff, O. Stursberg, and M. Buss, "Computing reachable sets of hybrid systems using a combination of zonotopes and polytopes," *Nonlinear Analysis: Hybrid Systems*, vol. 4, no. 2, pp. 233–249, 2010.