2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Enhancements on the hyperplane arrangements in mixed integer techniques

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

***Abstract*— The current paper addresses the problem of optimizing a cost function over a non-convex and possibly non-connected feasible region. A classical approach for solving this type of optimization problem is based on Mixed integer technique. The exponential complexity as a function of the number of binary variables used in the problem formulation highlights the importance of reducing them. Previous work which minimize the number of binary variables is revisited and enhanced. Practical limitations of the procedure are discussed and a typical control application, the control of Multi-Agent Systems is exemplified.**

I. Introduction

Collision avoidance plays an important role in the context of managing multiple agents. In the same time is known to be a difficult problem, since certain constraints are non-convex. For example, the evolution of a dynamical system in an environment presenting obstacles can be modeled in terms of a non-convex feasible region. More precisely, it is possible to set up an optimization problem to optimize the agent state trajectory in order to avoid a convex region, representing an obstacle (static constraints) or another agent (dynamic constraints - leading in fact to a parametrization of the set of constraints with respect to the current state).

A popular framework for the treatment of a such optimization problem is represented by *Mixed-Integer-Programming* (MIP), described in [1]. This method has proved to be very useful due to the ability to include non-convex constraints and discrete decisions in the optimization problem. Research on these types of problems, using MIP techniques has focused on optimization of agent trajectories [2], multi-vehicle target assignment and intercept problems [3], or on coordinating the efficient interaction of multiple agents in scenarios with many sequential tasks and tight timing constraints [4]. In [5], the authors used a combination of MIP and Model Predictive Control (MPC) to stabilize general hybrid systems around equilibrium points.

However, despite its modeling capabilities and the availability of good solvers, MIP has serious numerical drawbacks. As stated in [6], mixed-integer techniques are in the NP-hard computation class. Consequently, these methods may not be fast enough for real-time control of systems with large problem formulations.

† SUPELEC Systems Sciences (E3S) - Automatic Control Department, Gif sur Yvette, France {florin.stoican,ionela.prodan,sorin.olaru}@supelec.fr
‡ DISCO Team – INRIA, France

There has been a number of attempts in the literature to reduce the computational requirements of MIP formulations in order to make them attractive for real-time applications. In [7] an iterative method for including the obstacles in the best path generation is provided. Other works, like [8], consider a predefined path constrained by a sequence of convex sets. In all of these papers the original decision problems are reformulated in a simplified MIP form.

The negative influence of the increased number of binary variables in the problem formulation highlights the importance of reducing them. In [9], we introduce a novel *linear* constraints expression for reducing the number of binary variables necessary in describing the exterior of convex sets.

In the present paper, we revise these preliminary results and introduce enhancements in the description of non-connected convex sets (or their complement). We list some of the noteworthy aspects of our approach representing also the main contributions of this paper:

- a convex representation in the extended space of state plus binary variables using a hyperplane arrangement;
- reduced complexity of the problem upon merging techniques;
- a notable property of optimal association between regions and their binary representation leading to the minimization of the number of constraints.

The method presented here can be used in several fields of application. We choose to exemplify here with the control of an agent operating in a dynamic environment with obstacles. The agent is required to maneuver successfully in a hostile environment. The obstacles are designed as convex polyhedral regions. In this context the reduction technique is embedded within an MPC path planning for multiple agents.

The rest of the paper is organized as follows. In Section II the preliminaries are presented, the main idea being detailed in Section III. Discussions based on the control of multiple agents operating in a hostile environment are presented in Section IV, while the conclusions are drawn in Section V.

Notation: The following notation will be used throughout the paper. The closure of a set S, $cl(S)$ is the intersection of all closed sets containing S. The collection of all possible N combinations of binary variables will be noted $\{0,1\}^N = \{(b_1,\ldots,b_N) : b_i \in \{0,1\}, \forall i = 1,\ldots,N\}$. The ceiling value of $x \in \mathbb{R}$ denoted as $\lceil x \rceil$ is the smallest integer greater than x. We denote $|I|$ as the cardinal of set I. $lp(n,d)$ ($qp(n,d)$) denotes the complexity of solving a linear (quadratic) program with n constraints and d variables.

II. Preliminaries

The main objective of this paper is to provide a technique for solving an optimization problem over a non-convex and possibly non-connected region of the state space. We recall here methods from [9] which use hyperplane arrangements and mixed integer techniques in order to provide an equivalent (in an extended space augmented with auxiliary binary variables) optimization problem over a convex region.

A. Polyhedral notions

In the following we define a bounded polyhedral set, $P \subset \mathbb{R}^n$ through its implicit half-space description:

$$P = \{x \in \mathbb{R}^n : \; h_i x \leq k_i, \quad i = 1 \dots N\} \tag{1}$$

with $(h_i, k_i) \in \mathbb{R}^{1\times n} \times \mathbb{R}$ and its complement, as:

$$\mathcal{C}_X(P) \triangleq cl(X \setminus P) \tag{2}$$

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

By definition, every affine subspace which defines P

$$\mathcal{H}_i = \{x : \; h_i x = k_i\} \tag{3}$$

will partition the space into two disjoint regions:

$$\mathcal{R}^+(\mathcal{H}_i) \;=\; \{x : \quad h_i x \leq \quad k_i\} \tag{4}$$
$$\mathcal{R}^-(\mathcal{H}_i) \;=\; \{x : \; -h_i x \leq -k_i\} \tag{5}$$

with $i = 1 \dots N$. $\mathcal{R}_i^+$ and $\mathcal{R}_i^-$ denote simplified notation for region (4) and (5), respectively, associated to the i^{th} inequality of (1).

B. Non-convex and non-connected region

Without restricting the problem let us consider our non-convex and non-compact region as the complement of an union of convex (bounded polyhedral) sets $\mathbb{P} = \bigcup\limits_l P_l$:

$$\mathcal{C}_X(\mathbb{P}) = cl(X \setminus \mathbb{P}) \tag{6}$$

with $P_l = \bigcap\limits_{k_l=1}^{\mathcal{K}_l} R^+(\mathcal{H}_{k_l})$ and $N \triangleq \sum\limits_l \mathcal{K}_l$.

This type of regions arises naturally in the context of obstacle/collision avoidance when there is more than a single object to be taken into account.

In order to deal with the complement of a non-convex region in the context of mixed-integer techniques several additional theoretical tools need to be introduced.

Definition 1 (Hyperplane arrangements – [10])**.** A collection of hyperplanes $\mathbb{H} = \{\mathcal{H}_i\}_{i=1:N}$ will partition the space in an union of disjoint cells defined as follows:

$$\mathcal{A}(\mathbb{H}) = \bigcup_{l=1,\dots,\gamma(N)} \left(\underbrace{\bigcap_{i=1}^{N} \mathcal{R}^{\sigma_l(i)}(\mathcal{H}_i)}_{A_l} \right) \tag{7}$$

where $\sigma_l \in \{-,+\}^N$ denotes feasible combinations of regions (4)–(5) obtained for the hyperplanes in $\mathbb{H}$. ♦

Several computational aspects are of interest. 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 ([11]):

$$\gamma(N) \leq \sum_{i=0}^{d} \binom{N}{i} \tag{8}$$

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

We note that there exists a subset $\{B_l\}_{l=1,\dots,\gamma^b(N)}$ of *feasible* cells from (7) (with $\gamma^b(N) \leq \gamma(N)$) which describes region (6):

$$\mathcal{C}_X(\mathbb{P}) = \bigcup_l B_l, \tag{9}$$

such that, for any l there exists i such that $B_l = A_i$ and

$$B_l \in \left\{A_i \in \mathcal{A}(\mathbb{H}) : A_i \bigcap \mathcal{P} = \emptyset\right\}. \tag{10}$$

Mixed integer programming (MIP) allows to express the union (9) as a polyhedra in an extended space $X \times \{0,1\}^N$ of *state + auxiliary binary variables* as follows:

$$\left.\begin{array}{c} \vdots \\ \sigma_l(1)h_1 x \leq \sigma_l(1)k_1 + M\alpha_l \\ \vdots \\ \sigma_l(N)h_N x \leq \sigma_l(N)k_N + M\alpha_l \\ \vdots \end{array}\right\} \quad B_l \tag{11}$$

with M a positive scalar chosen appropriately (that is, significantly bigger than the rest of the variables in the right hand side of the inequalities) and $(\alpha_1,\dots,\alpha_N) \in \{0,1\}^N$ the auxiliary binary variables.

It is straightforward to note that any of the regions B_l can be obtained from (11) with an adequate choice of binary variables:

$$\alpha^l = (1,\dots,1,\underbrace{0}_{l},1,\dots,1). \tag{12}$$

The number of binary variables negatively influences the computation time, in the worst case, an exponential bound is reached. In [9] it was noted that a set

$$(\lambda_1,\dots,\lambda_{N_0}) \in \{0,1\}^{N_0} \tag{13}$$

with $N_0 = \lceil log_2(N) \rceil$ permits a reduced representation where any variable α_l is written as a linear combination in the space of variables $(\lambda_1,\dots,\lambda_{N_0}) \in \{0,1\}^{N_0}$:

$$\alpha_i = \alpha_i\left(\lambda_1 \dots \lambda_{N_0}\right). \tag{14}$$

As a prerequisite for explicitly defining relation (14) we associate a tuple

$$\lambda^i \triangleq \left(\lambda_1^i \dots \lambda_{N_0}^i\right) \tag{15}$$

to each region B_i. We mention that this association is not unique, and various possibilities can be considered; in the following, unless otherwise specified, the tuples will be appointed in lexicographical order. We can now recall the next result:

Proposition 1. *A mapping* $\alpha_i(\lambda) : \{0,1\}^{N_0} \to \mathbb{R}$ *which verifies that* $\alpha_i(\lambda^i) = 0$ *and* $\alpha_i(\lambda^j) \geq 1$ *for any* $j \neq i$ *is given by:*

$$\alpha_i(\lambda) = \sum_{k=1}^{N_0} t_k^i, \quad \text{where } t_k^i = \begin{cases} \lambda_k, & \text{if } \lambda_k^i = 0 \\ 1-\lambda_k, & \text{if } \lambda_k^i = 1 \end{cases} . \tag{16}$$

where λ_k *denotes the* k^{th} *variable and* λ_k^i *its value for the tuple associated to region* A_i.

Proof: See the proof of Proposition 1 in [9]. ■

Note that Proposition 1 gives a linear mapping $\alpha_i(\lambda)$ which will have value 0 only for the tuple associated to region B_i and values ≥ 1 for any other tuple. This proves that (14) can be used to extract region B_i as in (12).

Note that if a tuple is unallocated (i.e., has no associated region), the substitution of its binary values in the extended polyhedra (11) will result in a degenerate projection onto X which for all intents and purposes encompasses the entire space. To counteract this undesired behavior we need to add constraints which will force the unallocated tuples to be infeasible. This reasoning leads the following corollary:

Corollary 1. *Let there be a tuple* $\lambda^i \in \{0,1\}^{N_0}$. *The point it describes is made infeasible with respect to the constraint:*

$$-\sum_{k=1}^{N_0} t_k^i \leq -\epsilon \tag{17}$$

with t_k^i *defined as in Proposition 1 and* $\epsilon \in (0,1)$ *a scalar (for simplicity, in the rest of the paper we will use* 0.5*).*

Proof: The left side of the inequality (17) will vanish only at tuple λ^i and for the rest of the tuples in the discrete set $\{0,1\}^{N_0}$ will give values greater of equal to 1. Thus, the only point made infeasible by inequality (17) is λ^i. ■

Using Corollary 1 it follows that by adding inequalities of form (17) for each unallocated tuple in (11) we obtain a complete representation of (6).

C. Exemplification

Consider the following example depicted in Fig. 1 where the complement of the union of two triangles ($\mathbb{P} = P_1 \cap P_2$) represents the feasible region. We take $\mathbb{H} = \{\mathcal{H}_i\}_{i=1:4}$ a collection of $N = 4$ hyperplanes (given as in (3)) which define P_1, P_2 as follows:

$$P_1 = R_1^+ \cap R_2^+ \cap R_3^+$$
$$P_2 = R_1^- \cap R_2^- \cap R_4^+.$$

We observe that the bound given in (8) is reached, that is, we have 11 cells (obtained as in the arrangement (7)). From them, a total of 9, which we denote here as $B_1, \ldots, B_9$, describe the non-convex region (6). To each of them we associate a unique tuple from $\{0,1\}^{N_0}$ as seen in Fig. 1 with $N_0 = \lceil log_2 9 \rceil = 4$.

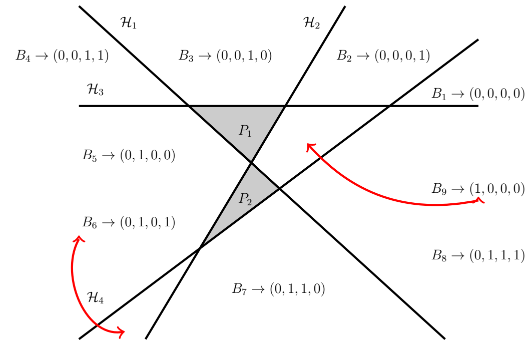


FIG. 1: Exemplification of hyperplane arrangement

As per Proposition 1 and (11), we are now able to write the following set of inequalities:

$$\begin{aligned}
&\left.\begin{aligned} -h_3x &\leq -k_3 \\ h_4x &\leq k_4 \end{aligned} + M\,(\lambda_1+\lambda_2+\lambda_3+\lambda_4)\right\} && B_1\\
&\left.\begin{aligned} -h_2x &\leq -k_2 \\ -h_3x &\leq -k_3 + M\,(1+\lambda_1+\lambda_2+\lambda_3-\lambda_4) \\ h_4x &\leq k_4 \end{aligned}\right\} && B_2\\
&\left.\begin{aligned} h_1x &\leq k_1 \\ h_2x &\leq k_2 + M\,(1+\lambda_1+\lambda_2-\lambda_3+\lambda_4) \\ -h_3x &\leq -k_3 \end{aligned}\right\} && B_3\\
&\left.\begin{aligned} -h_1x &\leq -k_1 \\ -h_3x &\leq -k_3 \end{aligned} + M\,(2+\lambda_1+\lambda_2-\lambda_3-\lambda_4)\right\} && B_4\\
&\left.\begin{aligned} -h_1x &\leq -k_1 \\ -h_2x &\leq -k_2 \\ h_3x &\leq k_3 \\ h_4x &\leq k_4 \end{aligned} + M\,(1+\lambda_1-\lambda_2+\lambda_3+\lambda_4)\right\} && B_5\\
&\left.\begin{aligned} h_2x &\leq k_2 \\ -h_4x &\leq -k_4 \end{aligned} + M\,(2+\lambda_1-\lambda_2+\lambda_3-\lambda_4)\right\} && B_6\\
&\left.\begin{aligned} h_1x &\leq k_1 \\ h_2x &\leq k_2 + M\,(2+\lambda_1-\lambda_2-\lambda_3+\lambda_4) \\ -h_4x &\leq -k_4 \end{aligned}\right\} && B_7\\
&\left.\begin{aligned} h_1x &\leq k_1 \\ h_3x &\leq k_3 + M\,(3+\lambda_1-\lambda_2-\lambda_3-\lambda_4) \\ -h_4x &\leq -k_4 \end{aligned}\right\} && B_8\\
&\left.\begin{aligned} h_1x &\leq k_1 \\ h_2x &\leq k_2 \\ h_3x &\leq k_3 \\ h_4x &\leq k_4 \end{aligned} + M\,(1+\lambda_1-\lambda_2+\lambda_3+\lambda_4)\right\} && B_9.
\end{aligned} \tag{18}$$

Note that in the above set we simplified the description by cutting the redundant hyperplanes in a cell representation(e.g., for cell A_1, 2 hyperplanes suffice for a complete description).

Since only 9 tuples, from a total number of 16 are associated to cells, we need to add constraints to the problem such that remaining 7 unallocated tuples will never be feasible:

$$\begin{aligned}
&-(2-\lambda_1-\lambda_2+\lambda_3+\lambda_4) \leq -0.5\\
&-(3-\lambda_1-\lambda_2-\lambda_3+\lambda_4) \leq -0.5\\
&-(3-\lambda_1-\lambda_2+\lambda_3-\lambda_4) \leq -0.5\\
&-(4-\lambda_1-\lambda_2-\lambda_3-\lambda_4) \leq -0.5\\
&-(2-\lambda_1+\lambda_2-\lambda_3+\lambda_4) \leq -0.5\\
&-(3-\lambda_1+\lambda_2-\lambda_3-\lambda_4) \leq -0.5\\
&-(2-\lambda_1+\lambda_2+\lambda_3-\lambda_4) \leq -0.5.
\end{aligned} \tag{19}$$

III. Main idea

As seen in [9], palliatives for reducing the computational load exist but ultimately, the computation time is in the worst case scenario is exponentially dependent on the number of binary variables which in turn depends on the number of cells of the hyperplane arrangements (see (8)). We conclude then, that the problem becomes prohibitive for a relatively small number of polyhedra in $\mathbb{P}$ and that any reduction in the number of cells is worthwhile and should be pursued.

This can be accomplished in two complementary ways. Firstly, we note that bound (8) is reached for a given number of hyperplanes only if they are in general position. As such, particular classes of polyhedra may somewhat reduce the actual number of cells in arrangement (7) and consequently, the number of auxiliary binary variables.

The other direction, which we chose to pursue in the rest of the paper is the merging of adjacent cells into possibly overlapping regions which describe the complement of our initial union of polyhedra. Using, the cells of (7) and some well known notions of set theory we will describe a reduced representation of (6), both in number of cells and of interdicting constraints (similar to the ones discussed in Corollary 1).

A. Cell merging

Recall that any of the cells of (9) is described by an unique sign tuple ($B_l \leftrightarrow \sigma_l$). As such, we obtain that the cells are disjunct and cover the entire feasible space. For our purposes we are satisfied with any collection of regions not necessarily disjoint which covers the feasible space. In this context we may ask if it is not possible to merge the existing cells of (9) into a reduced number of regions which will still cover region (6). Note that by reducing the number of regions, the number of necessary auxiliary variables my also decrease substantially.

We can formally represent the problem by requiring the existence of a collection of regions,

$$\mathcal{C}_X(\mathbb{P}) = \bigcup_{k=1,\ldots,\gamma^c(N)} C_k \tag{20}$$

which verifies the next conditions:

- the new polyhedra are formed as unions of the old ones (i.e., for any k there exists a set I_k which selects indices from $1,\ldots,\gamma^c(N)$ such that $C_k = \bigcup_{i\in I_k} B_i$)
- the union is minimal, that is, the number $\gamma^c(N)$ of regions is minimal

Existing merging algorithms are usually computationally expensive but here we can simplify the problem by noting two properties of the cells in (9):

- the sign tuples σ_l describe an adjacency graph since any two cells whose sign tuples differ at only one position are neighbors
- the union of any two adjacent cells is a polyhedra

Remark 1. Note that a region C_k is described by at most $N-d$ hyperplanes where d denotes the number of indices in the sign tuples which flip the sign. It makes sense then to, not only reduce the number of regions, but also to maximize the number of disjoint cells that go into the description of a region from (20). ♦

In order to construct (20) we may use merging algorithms (see for example [12] which adapts a "branch and bound" algorithm to merge cells of a hyperplane arrangement) or we can pose the problem in the boolean algebra framework [13].

B. Constraint reduction

Note that,

$$N_{int} \triangleq 2^{\lceil log_2\gamma^c(N)\rceil} - log_2\gamma^c(N), \tag{21}$$

the number of unallocated tuples, may have significant values. If we associate to each tuple an inequality intended to discard the combination from the set of feasible points as in Corollary 1, we negatively influence the speed of the associated optimization algorithm. This can be alleviated by noting (as previously mentioned) that the association between feasible cells in (7) and tuples is arbitrary. One could then chose favorable associations which will permit more than one tuple to be removed through a single inequality. To this end, we present the following proposition.

Proposition 2. *Let there be a collection of tuples* $\{\lambda^i\}_{i\in 1,\ldots,2^d} \in \{0,1\}^{N_0}$ *which completely spans a d-facet of hypercube* $\{0,1\}^{N_0}$*. Let* $\mathcal{I}$ *be the set of the* N_0-d *indices which retain a constant value over all the tuples* $\{\lambda^i\}_{i\in 1,\ldots,2^d}$ *composing the facet. Then there exists the constraint*

$$-\sum_{k\in\mathcal{I}} t_k^* \leq -\epsilon, \tag{22}$$

which renders the tuples of the given facet (and only these ones) infeasible.

Variables t_k^* *and* ϵ *are taken as in Corollary 1 with* t_k^* *associated to* λ_k^**, the common value of variable* λ_k *over the set of tuples* $\{\lambda^i\}_{i\in 1,\ldots,2^d}$*.*

Proof: Geometrically, the tuples are extreme points on the hypercube $\{0,1\}^{N_0}$ and the inequalities we are dealing with are half-spaces which separate the points of the hypercube. If a set of tuples completely spans a d-facet it is always possible to isolate a half-space that separates the points of the d-facet from the rest of the hypercube. ■

By a suitable association between feasible cells and tuples we may label as unallocated the extreme points which compose entire facets on the hypercube $\{0,1\}^{N_0}$ which permits to apply Proposition 2 in order to obtain constraints (22).

Remark 2. By writing N_{int} as a sum of consecutive powers of 2 ($N_{int} = \sum_{i=0}^{\lceil log_2 N_{int}\rceil -1} b_i 2^i$), an upper bound N_{hyp} for the number of inequalities (22) can be computed:

$$N_{hyp} = \sum_{i=0}^{\lceil log_2 N_{int}\rceil -1} b_i \leq \lceil log_2 \gamma^c(N)\rceil - 1 \qquad (23)$$

where $b_i \in \{0,1\}$. ♦

C. Exemplification

By applying the merging algorithm of Subsection III-A we obtain that the feasible region (6) is expressed by an union as in (11) and we depict the result in Fig. 2.

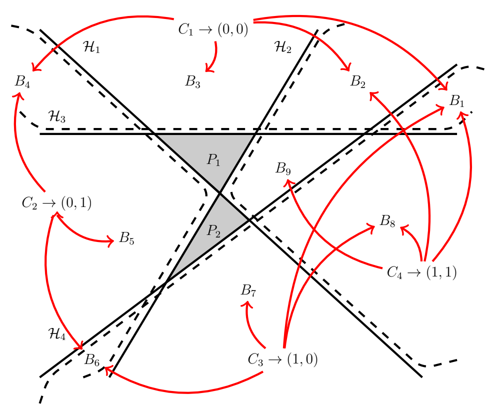


FIG. 2: Exemplification of hyperplane arrangement with merged regions

As it can be seen, we obtain 4 overlapping regions: $C_1 = B_1 \cup B_2 \cup B_3 \cup B_4$, $C_2 = B_4 \cup B_5 \cup B_6$, $C_3 = B_6 \cup B_7 \cup B_8 \cup B_1$ and $C_1 = B_8 \cup B_9 \cup B_1 \cup B_2$. Consequently, we note that $N_0 = 2$ auxiliary binary variables suffice in coding the regions. As per Proposition 1 and (11), we are now able to write the following set of inequalities (we attach to each of the regions a tuple in lexicographical order):

$$\begin{array}{ll}
-h_3 x \leq -k_3 + M\,(\qquad \lambda_1 + \lambda_2)\} & C_1 \\
\left.\begin{array}{l} -h_1 x \leq -k_1 \\ -h_4 x \leq -k_4 \end{array}\right. + M\,(1 + \lambda_1 - \lambda_2)\Big\} & C_2 \\
-h_4 x \leq -k_4 + M\,(1 - \lambda_1 + \lambda_2)\} & C_3 \\
\left.\begin{array}{l} h_1 x \leq k_1 \\ -h_2 x \leq -k_2 \end{array}\right. + M\,(2 - \lambda_1 - \lambda_2)\Big\} & C_4
\end{array} \qquad (24)$$

In the reduced representation (24) there are no unallocated tuples, since the number of tuples coincides with the number of regions. For the sake of the presentation we take the construction from (18) where there remain 7 unallocated tuples and consider Proposition 2. In the original formulation (19) we required 7 constraints to discard these tuples. We apply now the results of Subsection III-B and observe the following improvements. For the 7 unallocated tuples, we observe that 4 of them, $(1,1,0,0)$, $(1,1,0,1)$, $(1,1,1,0)$ and $(1,1,1,1)$ form a 2-facet of the hypercube $\{0,1\}^4$. Tuples $(1,0,1,0)$ and $(1,0,1,1)$ form an edge and $(1,0,0,1)$ is on a vertex. We can now apply Proposition 2 and obtain the following constraints

$$\begin{aligned}
&-(2 - \lambda_1 - \lambda_2 \qquad\qquad) \leq -0.5 \\
&-(2 - \lambda_1 + \lambda_2 - \lambda_3 \qquad) \leq -0.5 \\
&-(2 - \lambda_1 + \lambda_2 + \lambda_3 - \lambda_4) \leq -0.5.
\end{aligned} \qquad (25)$$

Note that we were able to diminish the number of inequalities from 7 in (19) to only 3 in (25): the first 4 constraints of (19) are replaced by the 1^{st} constraint of (25). The same holds for the next 2 that correspond to the 2^{nd} and for the last that is identical with the 3^{rd}.

IV. COLLISION AVOIDANCE EXAMPLE

A number of commonly found situations in the control related to Multi-Agent Systems imply a cost function has to be minimized, while in the same time, the agent avoids collision with obstacles and other agents.

In this illustrative example we will describe an agent that has to navigate its way around a group of fixed obstacles. We consider the dynamics of the agent described by a LTI system as follows:

$$\xi_{k+1} = A\xi_k + Bu_k. \qquad (26)$$

The agent model is used in a predictive control context which permits the use of non-convex state constraints for obstacle avoidance behavior.

An optimal control action u^* is obtained from the control sequence $\mathbf{u} \triangleq \{u_{k|k}, u_{k+1|k}, \cdots, u_{k+N-1|k}\}$ as a result of the optimization problem:

$$u^* = \arg\min_{\mathbf{u}} (\xi^T_{k+N_h|k} P \xi_{k+N_h|k} + \sum_{l=1}^{N_h-1} \xi^T_{k+l|k} Q \xi_{k+l|k} + \\ + \sum_{l=0}^{N_h-1} u^T_{k+l|k} R u_{k+l|k})$$

$$\text{subject to: } \begin{cases} \xi_{k+l|k} = A\xi_{k+l-1|k} + Bu_{k+l-1|k} \\ \xi_{k+l|k} \in \mathcal{C}(\mathbb{P}), \quad l = 1, \ldots, N_h \end{cases} \qquad (27)$$

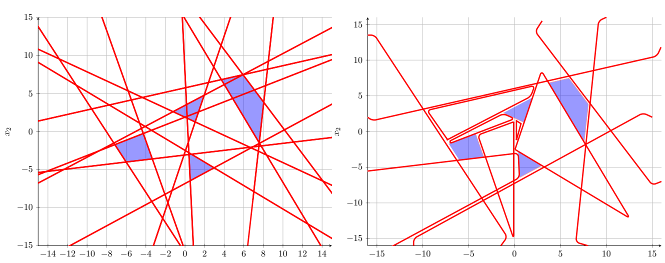
Here $Q \geq 0$, $R > 0$ are the weighting matrices, $P \geq 0$ defines the terminal cost, N_h the prediction horizon and $\mathbb{P}$ is an union of polytopes describing the obstacles.

As a practical application we consider a linear system (vehicle, pedestrian or agent in general form) whose dynamics are described by:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{\mu}{m} & 0 \\ 0 & 0 & 0 & -\frac{\mu}{m} \end{bmatrix}, \; B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{m} & 0 \\ 0 & \frac{1}{m} \end{bmatrix} \quad (28)$$

where $\xi = [x\ y\ v_x\ v_y]^T$, $u = [u_x\ u_y]^T$ are the state and the input of the system. With the components of the state being (x, y), the position, and (v_x, v_y) the velocities of the agent, m is the mass of the agent and μ its damping factor.

We consider the position component of the agent state to be constrained by 4 obstacles as shown in Fig. 3 (in blue). Considering the 14 hyperplanes which describe the polyhedra associated with the obstacles we have $\gamma(14) = 106$ regions obtained as in (7). Additionally we observe that 10 of the cells will describe the interdicted regions and the rest, $\gamma^b(14) = 96$ will describe the feasible region, as shown in (9). Further, we apply the notions from Subsection III-A (in this particular situation, the problem is "small" enough to be solved using a Karnaugh map) to obtain a reduced representation for the feasible region as in (20). We observe that the number of cells is substantially reduced, from $\gamma^b(14) = 96$ to $\gamma^c(14) = 11$ which warrants in turn a reduction of the auxiliary binary variables from 7 to 4 that, for a worst case scenario, equals to an eightfold speed up. In Fig. 3 (a) we depict the cells of (9) and the obstacles while in Fig. 3 (b) we show the covering (20) of merged cells.



(A) Partitioning with disjoint cells obtained as in (9)
(B) Partitioning with merged cells (20)

FIG. 3: Cell partitionings of the feasible region.

We apply the predictive control strategy for horizon $N = 3$ and cost matrices $Q = 10^5 \cdot I_4$, $R = I_2$ and $P = 10^5 \cdot I_4$ and obtain the trajectory depicted in Fig. 4.

V. Conclusions

In this paper we revisit a technique which transforms a non-convex and possibly non-connected region into a polyhedra in an augmented space (state and auxiliary binary variables) through the use of hyperplane arrangements. With respect to previous results we minimized the number of cells describing the feasible region through merging methods and discussed an improvement of the optimization problem such that the number of additional constraints is minimized. These numerical improvements were presented and tested in an obstacle avoidance control problem.

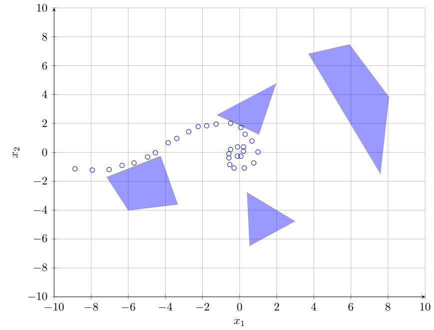


FIG. 4: Simulations of agent trajectories.

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

References

[1] A. Osiadacz, "Integer and combinatorial optimization, George L. Nemhauser and Laurence A. Wolsey, Wiley-Interscience Series in Discrete Mathematics and Optimization, New York, 1988, ISBN 0-471-82819-X, 763pp," *International Journal of Adaptive Control and Signal Processing*, vol. 4, no. 4, pp. 333–334, 1990.

[2] A. Richards and J. How, "Model predictive control of vehicle maneuvers with guaranteed completion time and robust feasibility," in *Proceedings of the 24th American Control Conference*, vol. 5, Portland, Oregon, USA, 8-10 June 2005, pp. 4034–4040.

[3] M. Earl and R. D'Andrea, "Modeling and control of a multi-agent system using mixed integer linear programming," in *Proceedings of the 40th IEEE Conference on Decision and Control*, vol. 1, Orlando, Florida, USA, 4-7 December 2001, pp. 107–111.

[4] C. Schumacher, "UAV task assignment with timing constraints via mixed-integer linear programming," Air Force research lab Wright-Patterson AFB of air vehicles directorate, Tech. Rep., 2004.

[5] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, pp. 407–428, 1999.

[6] M. Garey and D. Johnson, *Computers and intractability. A guide to the theory of NP-completeness. A Series of Books in the Mathematical Sciences.* WH Freeman and Company, San Francisco, Calif, 1979.

[7] M. G. Earl and R. D'Andrea, "Iterative MILP methods for vehicle-control problems," *IEEE Transactions on Robotics*, vol. 21, no. 6, p. 11581167, 2005.

[8] M. P. Vitus, V. Pradeep, G. Hoffmann, S. L. Waslander, and C. J. Tomlin, "Tunnel-milp: Path planning with sequential convex polytopes," in *AIAA Guidance, Navigation, and Control Conference*, Honolulu, Hawaii, USA, 18-21 August 2008.

[9] F. Stoican, I. Prodan, and S. Olaru, "On the hyperplanes arrangements in mixed-integer techniques," 2011, accepted to the 30th American Control Conference.

[10] G. M. Ziegler, *Lectures on polytopes.* Springer, 1995.

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

[12] T. Geyer, F. Torrisi, and M. Morari, "Optimal complexity reduction of piecewise affine models based on hyperplane arrangements," in *Proceedings of the 23th American Control Conference*, vol. 2, Boston, Massachusetts, USA, 30 June - 2 July 2004, pp. 1190–1195.

[13] ——, "Optimal complexity reduction of polyhedral piecewise affine systems," *Automatica*, vol. 44, no. 7, pp. 1728–1740, 2008.