Mixed Integer Representations in Control Design. Applications for the Control of Multi-agent Dynamical Systems

Ionela PRODAN, Florin STOICAN

Grenoble INP - LCIS (Laboratoire de Conception et d'Intégration des Systémes EA 3747), Valence 26000, France,
ionela.prodan@lcis.grenoble-inp.fr, https://sites.google.com/site/iprodanionela/.

University Politehnica of Bucharest, Department of Automatic Control and Systems Engineering, Bucharest, Romania,

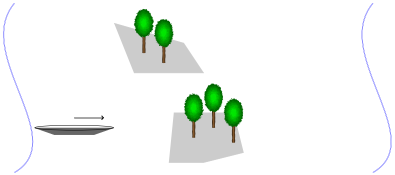
florin.stoican@acse.pub.ro, http://florinstoican.com/os/academic.

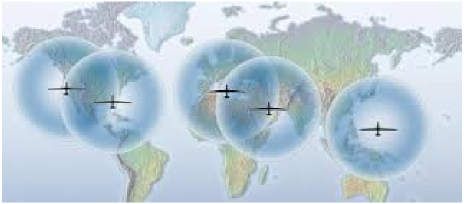
June 11, 2015

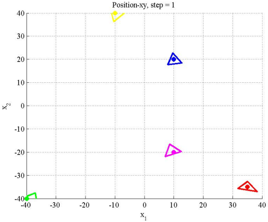
Journée commune aux GT CPNL et SDH, Paris, France

"To be or not to be ?" *(Hamlet, Shakespeare, 1601)*

"analysis and control of dynamical systems with conflicting objectives"



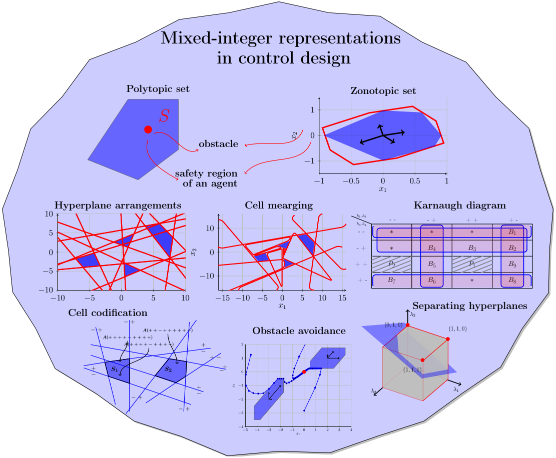




"To be or not to be ?" *(Hamlet, Shakespeare, 1601)*

"analysis and control of dynamical systems with conflicting objectives"

Mixed-integer optimization problems for which part or all of the arguments are required to be integers. NP-hard in general, but can also solve many large problems in practice



Brief history

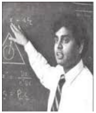
"50 Years of Integer Programming 1958-2008 : From the Early Years to the State-of-the-art", Jünger et al. [2009]

Mathematicians have started first to analyze problems with integer variables (early 1820's).
Fourier [1826], Minkowski [1896], Dantzig [1951], Fulkerson [1954], Hoffman and Kruskal [1956], Gomory et al. [1958], Edmonds [1965], Garey and Johnson [1979], Khachiyan [1979], Karmarkar [1984]

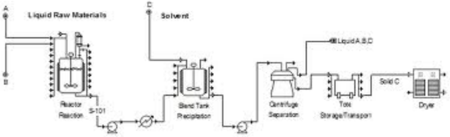






In the 90's MIP/MILP becomes a widely explored approach for chemical process scheduling problems.
Pritsker et al. [1969], Sahinidis and Grossmann [1991]



The availability of computing power increased the interest in optimization problems which can be formulated through the use of MI techniques (early 2000).
Earl and D'Andrea [2001], Richards et al. [2002], Schumacher et al. [2003]

Outline

Outline

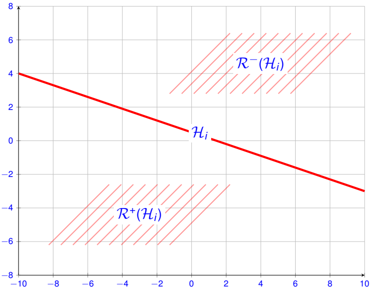
Half-spaces and polytopic sets

Let there be a collection of hyperplanes

$$\mathcal{H}_i = \left\{x : h_i x = k_i,\ (h_i, k_i) \in \mathbb{R}^{1\times n} \times \mathbb{R}\right\}$$

which partition the space in regions

$$\mathcal{R}^+(\mathcal{H}_i) = \left\{x : \quad h_i x \leq \quad k_i\right\}$$
$$\mathcal{R}^-(\mathcal{H}_i) = \left\{x : -h_i x \leq -k_i\right\}$$



Half-spaces and polytopic sets

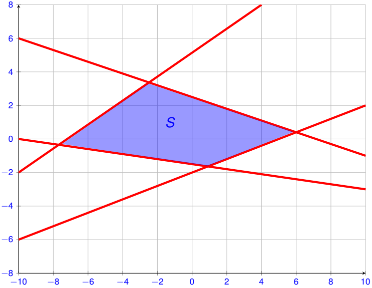
Let there be a collection of hyperplanes

$$\mathcal{H}_i = \left\{x : h_i x = k_i,\ (h_i, k_i) \in \mathbb{R}^{1\times n} \times \mathbb{R}\right\}$$

which partition the space in regions

$$\mathcal{R}^{+}(\mathcal{H}_i) = \left\{x : \quad h_i x \leq \quad k_i\right\}$$
$$\mathcal{R}^{-}(\mathcal{H}_i) = \left\{x : -h_i x \leq -k_i\right\}$$

describing a bounded polyhedral set



$$S = \bigcap_i \mathcal{R}^{+}(\mathcal{H}_i), \quad i = 1 : N$$

Half-spaces and polytopic sets

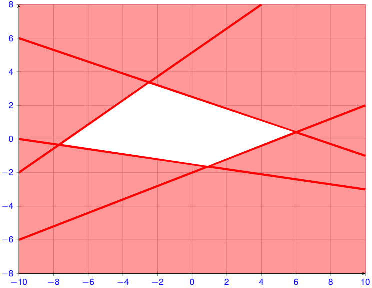
Let there be a collection of hyperplanes

$$\mathcal{H}_i = \left\{x : h_i x = k_i, (h_i, k_i) \in \mathbb{R}^{1\times n} \times \mathbb{R}\right\}$$

which partition the space in regions

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

describing a bounded polyhedral set



$$S = \bigcap_i \mathcal{R}^+(\mathcal{H}_i), \quad i = 1 : N$$

The complement of P is defined as

$$\mathcal{C}(S) \triangleq cl(\mathbb{R}^n \setminus P) = \bigcup_i \mathcal{R}^-(\mathcal{H}_i), \quad i = 1 : N$$

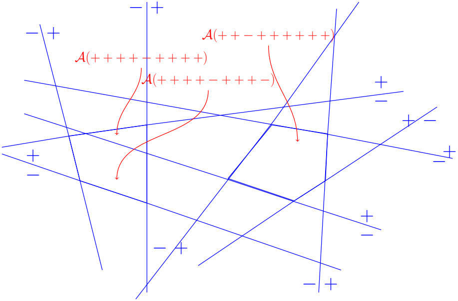
Hyperplane arrangement decomposition

The collection of hyperplanes $\mathbb{H}$ partitions $\mathbb{R}^n$ into a union of disjoint cells $\mathcal{A}(\sigma)$ characterized by the sign tuple $\sigma \in \{-,+\}^N$:

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

The feasible sign tuples describe a hyperplane arrangement of cells covering the entire space :

$$\mathcal{A}(\mathbb{H}) = \bigcup_{\sigma_l \in \sum_N} \mathcal{A}(\sigma_l).$$



where $\sum_N \subset \{-,+\}^N$ denotes the collection of combinations of regions $\mathcal{R}_i^+$, $\mathcal{R}_i^-$ resulting into non-empty cells (Buck's formula gives $\gamma(N) \leq \sum_{i=0}^{n} \binom{N}{i}$ as an upper bound Buck [1943]).

Description of the feasible region – I

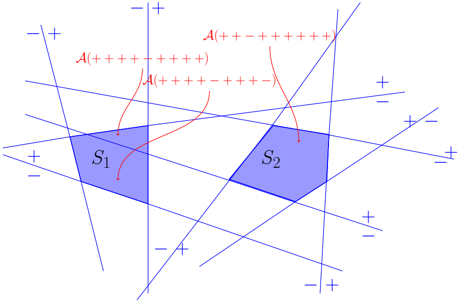
Describe the feasible region ($\mathbb{R}^n \setminus \mathbb{S}$) by dividing the existing cells ($\sigma \in \sum_N$) into :

- forbidden (they describe $\mathbb{S}$) :

$$\sigma \in \bar{\sum}_N$$

- feasible (they describe $\mathbb{R}^n \setminus \mathbb{S}$) :

$$\sigma \in \sum_N \setminus \bar{\sum}_N$$



Then the feasible region can be defined as the union of feasible cells :

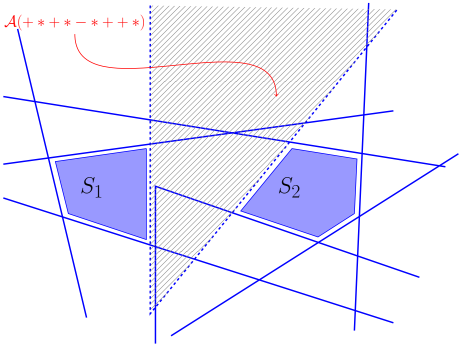
$$\mathbb{R}^n \setminus \mathbb{S} = \bigcup_{\sigma_l \in \sum_N \setminus \bar{\sum}_N} \mathcal{A}(\sigma_l).$$

Description of the feasible region – II

Feasible cells can be concatenated into "merged" cells (through merging procedures, e.g., Karnaugh maps, Espresso minimizer Geyer et al. [2008], Prodan et al. [2012a])

$$\mathcal{A}(\sigma^*) = \bigcup_{\sigma,\begin{cases}\sigma(i)=\sigma^*(i), & \sigma^*(i)\neq' *' \\ \sigma(i)\in\{-,+\}, & \sigma^*(i)=' *'\end{cases}} \mathcal{A}(\sigma) = \bigcap_{\sigma^*(i)\neq' *',\, i\in\mathcal{I}} \mathcal{R}_i^{\sigma^*(i)}$$

where $\sigma^* \in \{-,*,+\}^N$ denotes the sign tuple associated with the merged cell.



Then the feasible region can be defined as the union of feasible merged cells :

$$\mathbb{R}^n \setminus \mathbb{S} = \bigcup_{\sigma^*} \mathcal{A}(\sigma^*).$$

Outline

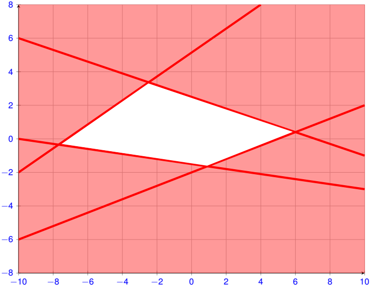
Classical MIP representation

Define an extended linear representation of $\mathcal{C}(P)$

$$-h_i x \leq -k_i + M\alpha_i, \quad i = 1 : N$$

$$\sum_{i=1}^{i=N} \alpha_i \leq N - 1$$

where $(\alpha_1, \ldots, \alpha_N) \in \{0,1\}^N$



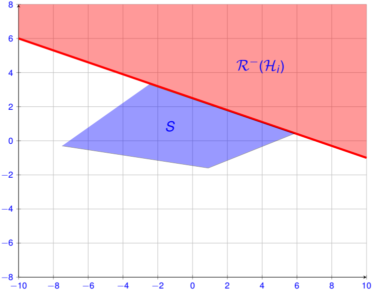
Classical MIP representation

Define an extended linear representation of $\mathcal{C}(P)$

$$-h_i x \leq -k_i + M\alpha_i, \quad i = 1 : N$$

$$\sum_{i=1}^{i=N} \alpha_i \leq N - 1$$

where $(\alpha_1, \ldots, \alpha_N) \in \{0, 1\}^N$



Any of the regions $\mathcal{R}^-(\mathcal{H}_i)$ of $\mathcal{C}(P)$ can be obtained by a suitable choice of binary variables

$$\mathcal{R}^-(\mathcal{H}_i) \longleftrightarrow (\alpha_1, \ldots, \alpha_N)^i \triangleq (1, \ldots, 1, \underbrace{0}_{i}, 1, \ldots, 1)$$

Logarithmic representation

For each region $\mathcal{R}^-(\mathcal{H}_i)$ a unique combination of binary variables $\lambda^i \in \{0,1\}^{\lceil log_2 N \rceil}$ is associated. Then, the affine functions $\alpha_i : \{0,1\}^{\lceil log_2 N \rceil} \to \{0\} \cup [1,\infty)$ are constructed :

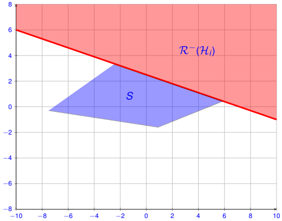
$$\alpha_i(\lambda) = \sum_{k=0}^{\lceil log_2 N \rceil} \left(\lambda_k^i + (1 - 2\lambda_k^i) \cdot \lambda_k\right).$$

λ_k denotes the kth component of λ and λ_k^i its value for the tuple associated to region $\mathcal{R}^-(\mathcal{H}_i)$:

$$\alpha_i(\lambda) = \begin{cases} 0, & \text{only if } \lambda = \lambda^i \\ \geq 1, & \text{for any } \lambda \neq \lambda^i \end{cases}$$

which leads to the compact formulation

$$-h_i x \leq -k_i + M\alpha_i(\lambda), \quad i = 1 : N,$$
$$0 \leq \beta_l(\lambda).$$



Interdicted tuples

In the mixed-integer representation we interdict tuples which describe the obstacle :

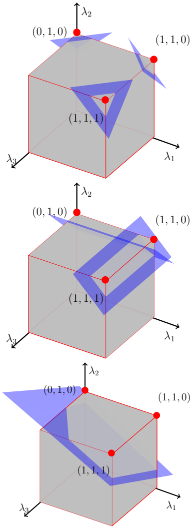
- in the classical formulation we force that at least one constraint is active :

$$\sum_{i=1}^{i=N} \alpha_i \leq N-1$$

- in the logarithmic formulation
  - multiple constraints to interdict tuples Prodan et al. [2012a]

$$0 < \beta_l(\lambda)$$

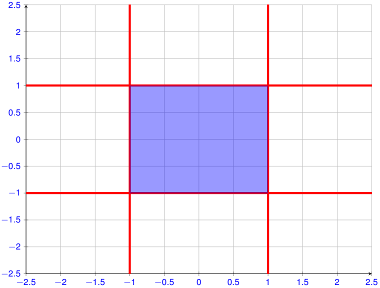
  - if the allocated tuples are ordered a single constraint suffices Afonso and Galvão [2013]



Illustrative example

Consider a polytope $P \subset \mathbb{R}^2$ given by

$$\begin{bmatrix} -1 & 0 \\ 1 & 0 \\ 0 & -1 \\ 0 & 1 \end{bmatrix} x \leq \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

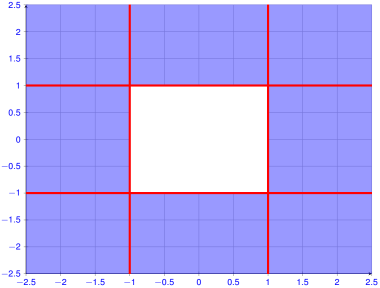


Illustrative example

and its complement $\mathcal{C}(P)$ by

$$\begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} x \leq \begin{bmatrix} -1 + M\alpha_1 \\ -1 + M\alpha_2 \\ -1 + M\alpha_3 \\ -1 + M\alpha_4 \end{bmatrix}$$

in the classical MI formulation.

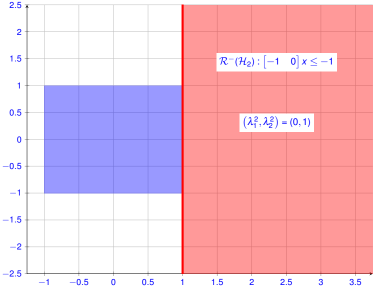


Illustrative example

and its complement $\mathcal{C}(P)$ by

$$\begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} x \leq \begin{bmatrix} -1 + M(\lambda_1 + \lambda_2) \\ -1 + M(1 - \lambda_1 + \lambda_2) \\ -1 + M(1 + \lambda_1 - \lambda_2) \\ -1 + M(2 - \lambda_1 - \lambda_2) \end{bmatrix}$$

in the reduced MI formulation.



In the reduced representation only $N_0 = \lceil \log_2 4 \rceil = 2$ binary variables are needed.

For region $\mathcal{R}^{-}(\mathcal{H}_2)$ associate tuple $(\lambda_1^2, \lambda_2^2) = (0, 1)$ which leads to the mapping

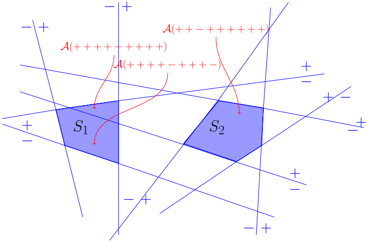
$$\alpha_2 = 1 + \lambda_1 - \lambda_2$$

MIP for hyperplane arrangements – I

Reminder :

- hyperplane arrangement :

$$\mathcal{A}(\mathbb{H}) = \bigcup_{\sigma_l \in \Sigma_N} \mathcal{A}(\sigma_l).$$



- interdicted and feasible tuples :

$$\bar{\sum}_N = \{\sigma : \mathcal{A}(\sigma) \subseteq \mathbf{S}\}$$
$$\sum_N \setminus \bar{\sum}_N = \{\sigma : \mathcal{A}(\sigma) \cap \mathbf{S} = \emptyset\}$$

There are several possible formulations of the feasible region :

- by making at least a constraint from each obstacle active
- as union of feasible (merged) cells
- explicitly forbid the cells describing obstacles

MIP for hyperplane arrangements – II

- by making at least a constraint from each obstacle active :

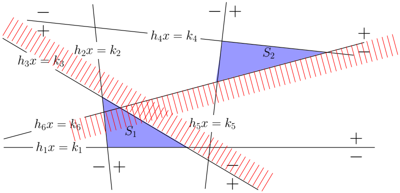
$$\mathcal{C}(\mathbb{S}) = \bigcap_{l} \mathcal{C}(S_l)$$

$$\ldots$$
$$-h_{i_l} x \leq -k_{i_l} + M\alpha_{i_l}, \ \forall i_l \in \mathcal{I}_l$$
$$\sum_{i_l \in \mathcal{I}_l} \alpha_{i_l} \leq \#\mathcal{I}_l - 1$$
$$\ldots$$

- as union of feasible (merged) cells :

$$\mathcal{C}(\mathbb{S}) = \bigcup_{j \in \mathcal{J}} \mathcal{A}(\sigma_j^*)$$

- explicitly forbid the cells describing obstacles ($\sigma_l \in \bar{\Sigma}_N$)



Characteristics :

- the number of binary variables depends on the complexity of the obstacles
- efficient in the logarithmic formulation

MIP for hyperplane arrangements – II

- by making at least a constraint from each obstacle active :

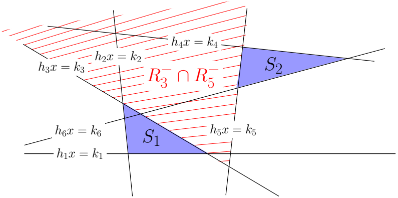
$$\mathcal{C}(\mathbb{S}) = \bigcap_{l} \mathcal{C}(S_l)$$

- as union of feasible (merged) cells :

$$\mathcal{C}(\mathbb{S}) = \bigcup_{j\in\mathcal{J}} \mathcal{A}(\sigma_j^*)$$

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

- explicitly forbid the cells describing obstacles ($\sigma_l \in \bar{\Sigma}_N$)



Characteristics :

- the number of binary variables depends on the complexity of $\mathbb{R}^n \setminus \mathbb{S}$
- efficient when using merged cells and logarithmic formulation
- difficult to compute the feasible cells and merge them

MIP for hyperplane arrangements – II

- by making at least a constraint from each obstacle active :

$$\mathcal{C}(\mathbb{S}) = \bigcap_{l} \mathcal{C}(S_l)$$

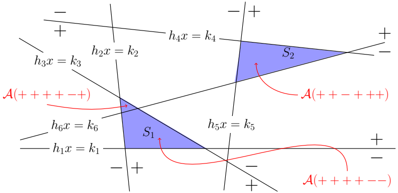
- as union of feasible (merged) cells :

$$\mathcal{C}(\mathbb{S}) = \bigcup_{j\in\mathcal{J}} \mathcal{A}(\sigma_j^*)$$

- explicitly forbid the cells describing obstacles ($\sigma_l \in \sum_{\mathbb{S}}$)

$$\begin{aligned} h_i x &\leq \quad k_i + M(1-\alpha_i),\ i \in \mathcal{I} \\ -h_i x &\leq -k_i + M\alpha_i \end{aligned}$$

$$\sum_{\sigma_l(i)='+'} (1-\alpha_i) + \sum_{\sigma_l(i)='-'} \alpha_i > 0,\ \forall \sigma_l \in \sum_{\mathbb{P}}$$



Characteristics :

- the number of binary variables does not depend on the number of forbidden cells

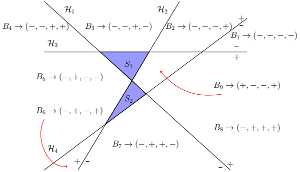
Numerical aspects

For a hyperplane arrangement we associate a truth table :

- '0' for forbidden tuples
- '1' for tuples describing part of the feasible domain
- '$*$' for tuples which result in empty cells

$P_1 = R_1^+ \cap R_2^+ \cap R_3^+$

$P_2 = R_1^- \cap R_2^- \cap R_4^+$



$\Rightarrow$

h_1, h_2 / h_3, h_4	- -	- +	+ +	+ -
- -	$*$	$*$	$*$	B_1
- +	$*$	B_4	B_3	B_2
+ +	P_2	B_5	P_1	B_9
+ -	B_7	B_6	$*$	B_8

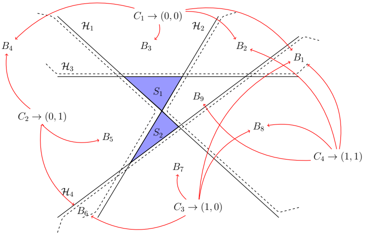
The resulting Boolean function (written as sum-of-products) describes the feasible cells of the hyperplane arrangements :

- the canonic form leads to merged cells
- the greatest time is spent finding the feasible tuples $\Rightarrow$ don't look for them
  - merge in the truth table all the cells which are not explicitly forbidden
  - discard the combinations which have no geometrical meaning

Numerical aspects

For a hyperplane arrangement we associate a truth table :

- '0' for forbidden tuples
- '1' for tuples describing part of the feasible domain
- '$*$' for tuples which result in empty cells



$\Rightarrow$

h_3,h_4 \ h_1,h_2	- -	- +	+ +	+ -
- -	$*$	$*$	$*$	B_1
- +	$*$	B_4	B_3	B_2
+ +	P_2	B_5	P_1	B_9
+ -	B_7	B_6	$*$	B_8

The resulting Boolean function (written as sum-of-products) describes the feasible cells of the hyperplane arrangements :

- the canonic form leads to merged cells
- the greatest time is spent finding the feasible tuples $\Rightarrow$ don't look for them
  - merge in the truth table all the cells which are not explicitly forbidden
  - discard the combinations which have no geometrical meaning

Caveat emptor

We use hyperplane arrangements in order to reduce the complexity of the non-convex representation.

Nonetheless, this pre-processing can be difficult itself :

- computation of the hyperplane arrangement increases exponentially with space dimension and number of hyperplanes Avis and Fukuda [1996]
- computing merged cells is relatively easy (the union of two cells which differ through a single bit is always convex) Geyer et al. [2008]
- we can use sub-optimal strategies
  - heuristic Boolean minimizer (the Espresso solver)
  - write conservatively the truth-table Stoican et al. [2013]

Outline

Applications for the control of multi-agent dynamical systems

Aircraft formation

Blake and Multhopp [1998]
Richards and How [2002]

Astronomical observations

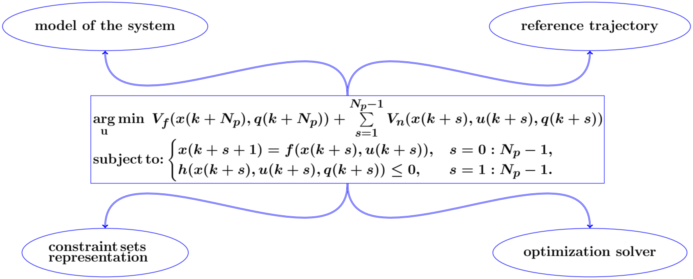
Massion et al. [2008]
Mora and Solar [2010]

Mobile Offshore Base

Sousa et al. [2000]
Girard et al. [2005]

Model Predictive Control (MPC)

Propoi [1963], Richalet et al. [1978], Cutler and Ramaker [1980]



$$\arg\min_{\mathbf{u}} \; V_f(x(k+N_p), q(k+N_p)) + \sum_{s=1}^{N_p-1} V_n(x(k+s), u(k+s), q(k+s))$$

$$\text{subject to: } \begin{cases} x(k+s+1) = f(x(k+s), u(k+s)), & s = 0 : N_p - 1, \\ h(x(k+s), u(k+s), q(k+s)) \leq 0, & s = 1 : N_p - 1. \end{cases}$$

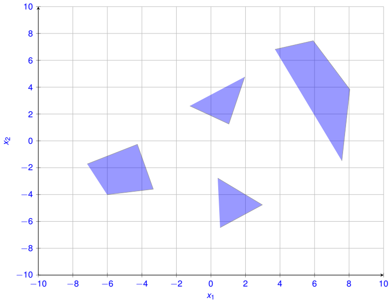
Obstacle avoidance problems

(Prodan et al., Springer'11)

Consider 4 obstacles and a single agent defined by a LTI dynamics.

- 14 hyperplanes ;
- 106 regions obtained with hyperplane arrangements ;
- 10 cells describing the forbidden regions ;
- 96 cells describing the feasible region ;

- $N_0 = 4$ the number of the binary variables ;
- apply an MPC optimization problem ($N_p = 3$, $Q = 10^5 \cdot I_4$, $R = I_2$, $P = 10^5 \cdot I_4$).



Conclusion : 72% complexity reduction of binary variables.

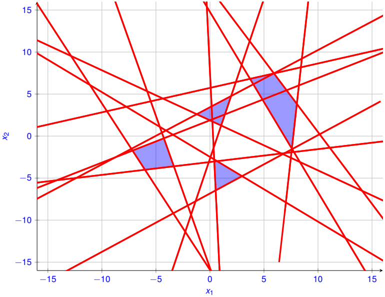
Obstacle avoidance problems

(Prodan et al., Springer'11)

Consider 4 obstacles and a single agent defined by a LTI dynamics.

- 14 hyperplanes ;
- 106 regions obtained with hyperplane arrangements ;
- 10 cells describing the forbidden regions ;
- 96 cells describing the feasible region ;

- $N_0 = 4$ the number of the binary variables ;
- apply an MPC optimization problem ($N_p = 3$, $Q = 10^5 \cdot I_4$, $R = I_2$, $P = 10^5 \cdot I_4$).



Conclusion : 72% complexity reduction of binary variables.

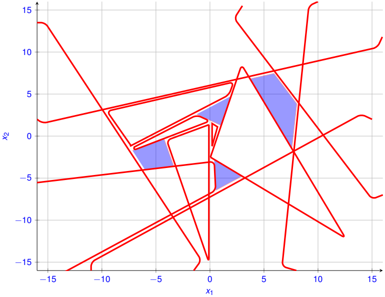
Obstacle avoidance problems

(Prodan et al., Springer'11)

Consider 4 obstacles and a single agent defined by a LTI dynamics.

- 14 hyperplanes ;
- 106 regions obtained with hyperplane arrangements ;
- 10 cells describing the forbidden regions ;
- 96 cells describing the feasible region ;

- $N_0 = 4$ the number of the binary variables ;
- apply an MPC optimization problem ($N_p = 3$, $Q = 10^5 \cdot I_4$, $R = I_2$, $P = 10^5 \cdot I_4$).



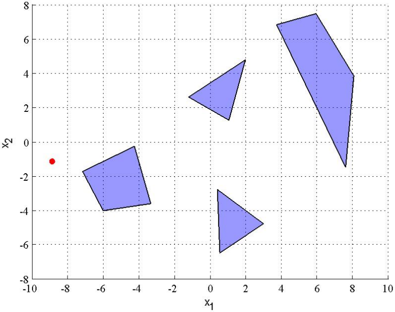
Conclusion : 72% complexity reduction of binary variables.

Obstacle avoidance problems

(Prodan et al., Springer'11)

Consider 4 obstacles and a single agent defined by a LTI dynamics.

- 14 hyperplanes ;
- 106 regions obtained with hyperplane arrangements ;
- 10 cells describing the forbidden regions ;
- 96 cells describing the feasible region ;
- $N_0 = 4$ the number of the binary variables ;
- apply an MPC optimization problem ($N_p = 3$, $Q = 10^5 \cdot I_4$, $R = I_2$, $P = 10^5 \cdot I_4$).



Conclusion : 72% complexity reduction of binary variables.

The corner cutting problem

Stoican, Grotli, Prodan, 2015

One challenging and not extensively studied issue in obstacle avoidance is the corner cutting problem.

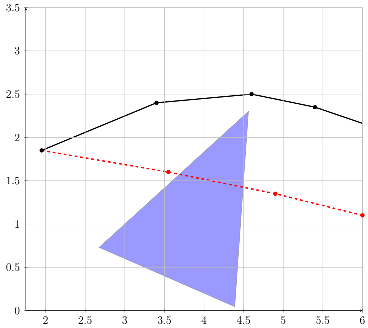
Avoidance constraints are usually imposed at the sampling time without regards to the intra-sample behavior of the dynamics.

Current results Richards and Turnbull [2015], Maia and Galvão [2009], Deits and Tedrake [2015] are

- conservative in their description
- do not treat efficiently the case of multiple obstacles

Main ideas :

- the future position should not lie in the shadow of the obstacle(s)
- consider exact and approximate descriptions



The corner cutting problem

Stoican, Grotli, Prodan, 2015

One challenging and not extensively studied issue in obstacle avoidance is the corner cutting problem.

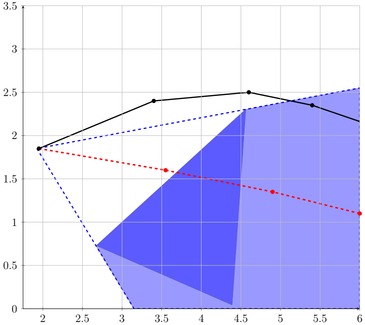
Avoidance constraints are usually imposed at the sampling time without regards to the intra-sample behavior of the dynamics.

Current results Richards and Turnbull [2015], Maia and Galvão [2009], Deits and Tedrake [2015] are

- conservative in their description
- do not treat efficiently the case of multiple obstacles

Main ideas :

- the future position should not lie in the shadow of the obstacle(s)
- consider exact and approximate descriptions

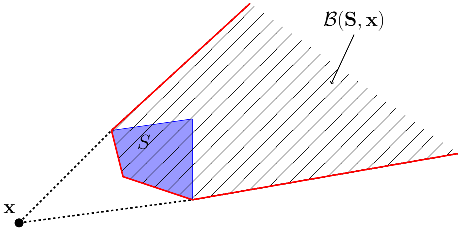


Shadow region description

We can define the "shadow" region $\mathcal{B}(S,x)$ as the collection of all the points from $\mathbb{R}^n$ which are "in the shadow" from the point of view of x :

$$\mathcal{B}(S,x) = \{y : [x,y] \cap S \neq \emptyset\}$$

- S is the obstacle
- x is the sensor/agent



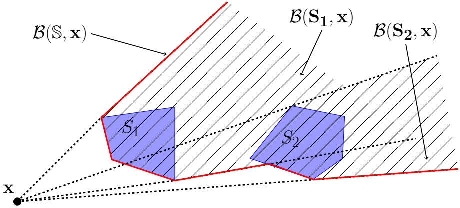
If the segment $[x,y]$ intersects S it means that point y is "hidden" by obstacle S and therefore is not "visible" from the point of view of x.

Shadow region description

We can define the "shadow" region $\mathcal{B}(\mathbb{S}, x)$ as the collection of all the points from $\mathbb{R}^n$ which are "in the shadow" from the point of view of x :

$$\mathcal{B}(\mathbb{S}, x) \quad = \{y \in \mathbb{R}^n : [x, y] \cap \mathbb{S} \neq \emptyset\} = \bigcup_{l=1}^{N_o} \mathcal{B}(S_l, x)$$

- $\mathbb{S} \triangleq \bigcup_{l=1}^{N_o} S_l$ is the collection of obstacles
- x is the sensor/agent



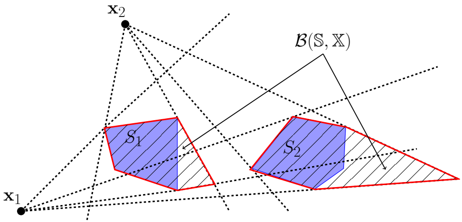
If the segment $[x, y]$ intersects S it means that point y is "hidden" by obstacle $S \in \mathbb{S}$ and therefore is not "visible" from the point of view of x.

Shadow region description

We can define the "shadow" region $\mathcal{B}(\mathbb{S},\mathbb{X})$ as the collection of all the points from $\mathbb{R}^n$ which are "in the shadow" from the point of view of $\mathbb{X}$:

$$\begin{aligned}\mathcal{B}(\mathbb{S},\mathbb{X}) &= \bigcap_{k=1}^{N_a} \mathcal{B}(\mathbb{S},x_k) = \bigcap_{k=1}^{N_a}\left[\left(\bigcup_{l=1}^{N_o} \mathcal{B}(S_l,x_k)\right)\right] \\ &= \bigcap_{k=1}^{N_a}\left(\bigcup_{l=1}^{N_o} \mathcal{B}(S_l,x_k)\right)\end{aligned}$$

- $\mathbb{S} \triangleq \bigcup_{l=1}^{N_o} S_l$ is the collection of obstacles
- $\mathbb{X} \triangleq \{x_1,\ldots,x_{N_a}\}$ is the collection of sensors/agents

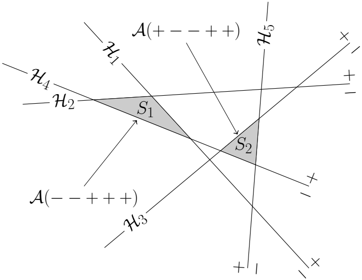


If the segment $[x,y]$ intersects S it means that point y is "hidden" by obstacle $S \in \mathbb{S}$ and therefore is not "visible" from the point of view of $x \in \mathbb{X}$.

Shadow area construction

Let there be $S = \mathcal{A}(\sigma^\bullet)$ where $\sigma^\bullet \in \sum^\bullet$, then we define the auxiliary set

$$\mathcal{E}(\sigma^\bullet, x) = \mathcal{A}(\sigma^\bullet) \cap \left(\bigcup_{x \notin \mathcal{H}_i^{\sigma^\bullet(i)}} \mathcal{H}_i \right) \cap \left(\bigcup_{x \in \mathcal{H}_i^{\sigma^\bullet(i)}} \mathcal{H}_i \right)$$



which denotes the tangent points of S from the viewpoint of x.

For any $x \in \mathcal{A}(\sigma^\circ)$ we have that $\mathcal{E}(\sigma^\bullet, x)$ remains fixed :

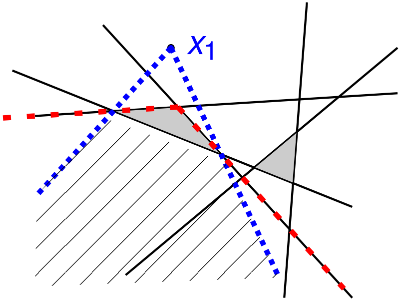
$$\mathcal{E}(\sigma^\bullet, \sigma^\circ) = \mathcal{A}(\sigma^\bullet) \cap \left(\bigcup_{\sigma^\circ(i) \neq \sigma^\bullet(i)} \mathcal{H}_i \right) \cap \left(\bigcup_{\sigma^\circ(i) = \sigma^\bullet(i)} \mathcal{H}_i \right)$$

$\mathcal{E}(\sigma^\bullet, x)$ is parametrized after $\sigma^\circ \in \sum^\circ \Rightarrow$ it remains constant with respect to σ° !

Shadow area construction

Let there be $S = \mathcal{A}(\sigma^\bullet)$ where $\sigma^\bullet \in \sum^\bullet$, then we define the auxiliary set

$$\mathcal{E}(\sigma^\bullet, x) = \mathcal{A}(\sigma^\bullet) \cap \left(\bigcup_{x \notin \mathcal{H}_i^{\sigma^\bullet(i)}} \mathcal{H}_i \right) \cap \left(\bigcup_{x \in \mathcal{H}_i^{\sigma^\bullet(i)}} \mathcal{H}_i \right)$$



which denotes the tangent points of S from the viewpoint of x.

For any $x \in \mathcal{A}(\sigma^\circ)$ we have that $\mathcal{E}(\sigma^\bullet, x)$ remains fixed :

$$\mathcal{E}(\sigma^\bullet, \sigma^\circ) = \mathcal{A}(\sigma^\bullet) \cap \left(\bigcup_{\sigma^\circ(i) \neq \sigma^\bullet(i)} \mathcal{H}_i \right) \cap \left(\bigcup_{\sigma^\circ(i) = \sigma^\bullet(i)} \mathcal{H}_i \right)$$

$\mathcal{E}(\sigma^\bullet, x)$ is parametrized after $\sigma^\circ \in \sum^\circ \Rightarrow$ it remains constant with respect to σ° !

Illustrative example

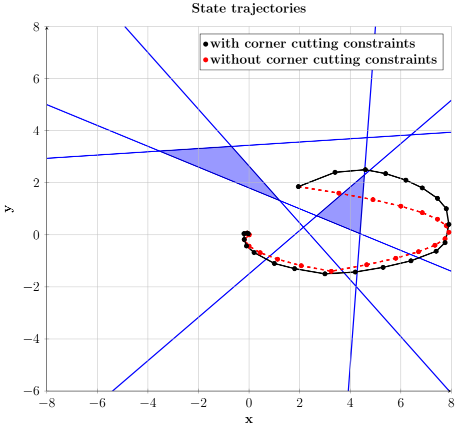
For dynamics $x_{k+1} = Ax_k + Bu_k$ we impose that $x_{k+1} \notin \bigcup_{\sigma^\bullet \in \Sigma^\bullet} \mathcal{B}(\sigma^\bullet, \sigma_k)$

$$u^* = \arg \min_{u_{k+i}, \sigma_{k+i}} \sum_{i=0}^{N_p-1} \|x_{k+i+1}\|_Q + \|u_k\|_R$$

$$\text{s.t.} x_{k+i+1} = Ax_{k+i} + Bu_{k+i}$$

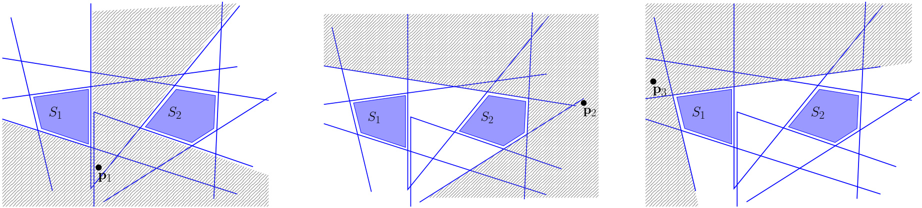
$$x_{k+i} \in \mathcal{X},\ u_{k+i} \in \mathcal{U}$$

$$x_{k+i+1} \notin \bigcup_{\sigma^\bullet \in \Sigma^\bullet} \mathcal{B}(\sigma^\bullet, \sigma_{k+i}),\ \forall i$$



A word about the coverage problem

In this talk the shadow region has been used to characterize the (in)feasible future position of an agent.
Alternatively we can use the the region to characterize the area *under shadow*



and use it for

- static approach : multiple agents shuffle their positions until they minimize (cancel) the unobserved space
- dynamic approach : successive points are taken such that an agent passing through them minimizes (cancels) the unobserved space – the agent dynamic has to be considered
- a combination of the previous approaches

Centralized/Distributed/Decentralized MPC for formation control

Two-stage procedure :

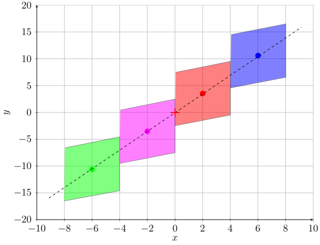
- Solve the task assignment problem (re-evaluated at each time step)
- Solve the mixed-integer optimization problem :

$$\mathbf{u}_{\mathcal{I}}^{*} = \arg \min_{\mathbf{u}_{\mathcal{I}}(k),\ldots,\mathbf{u}_{\mathcal{I}}(k+N_p-1)} V_n(\mathbf{x}_{\mathcal{I}}(k), \mathbf{u}_{\mathcal{I}}(k), \ldots, \mathbf{u}_{\mathcal{I}}(k+N_p-1))$$

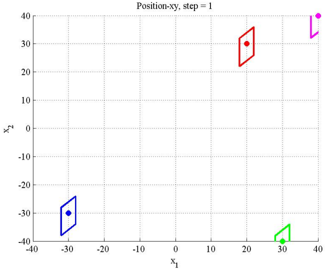
$$\text{subject to : } \begin{cases} \mathbf{x}_{\mathcal{I}}(k+s+1) = \mathbf{A}_{\mathcal{I}}\mathbf{x}_{\mathcal{I}}(k+s) + \mathbf{B}_{\mathcal{I}}\mathbf{u}_{\mathcal{I}}(k+s), & s = 0 : N_p - 1, \\ \mathbf{x}_{\mathcal{I}}(k+s) \in \mathcal{C}(\mathbb{S}), & s = 1 : N_p, \end{cases}$$

where the cost function is minimized in the target positions.

4 homogeneous agents
minimal configuration - "off-line"



agents motion - "on-line"



Centralized/Distributed/Decentralized MPC for formation control

Two-stage procedure :

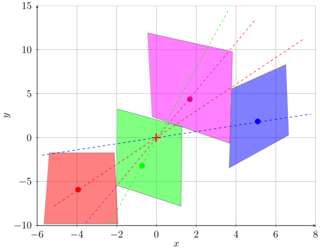
- Solve the task assignment problem (re-evaluated at each time step)
- Solve the mixed-integer optimization problem :

$$\mathbf{u}_{\mathcal{I}}^* = \arg \min_{\mathbf{u}_{\mathcal{I}}(k),\dots,\mathbf{u}_{\mathcal{I}}(k+N_p-1)} V_n(\mathbf{x}_{\mathcal{I}}(k), \mathbf{u}_{\mathcal{I}}(k), \dots, \mathbf{u}_{\mathcal{I}}(k+N_p-1))$$

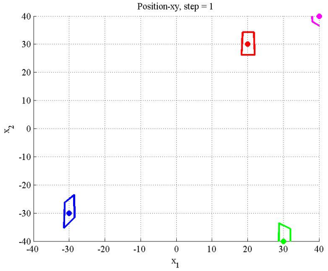
$$\text{subject to :} \quad \begin{cases} \mathbf{x}_{\mathcal{I}}(k+s+1) = \mathbf{A}_{\mathcal{I}}\mathbf{x}_{\mathcal{I}}(k+s) + \mathbf{B}_{\mathcal{I}}\mathbf{u}_{\mathcal{I}}(k+s), & s = 0 : N_p - 1, \\ \mathbf{x}_{\mathcal{I}}(k+s) \in \mathcal{C}(\mathbb{S}), & s = 1 : N_p, \end{cases}$$

where the cost function is minimized in the target positions.

4 heterogeneous agents
minimal configuration - "off-line"



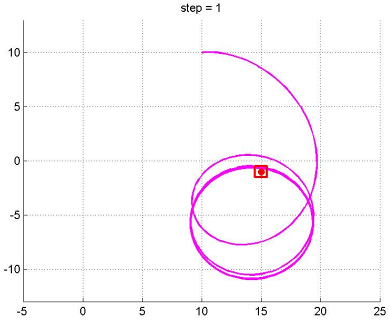
agents motion - "on-line"



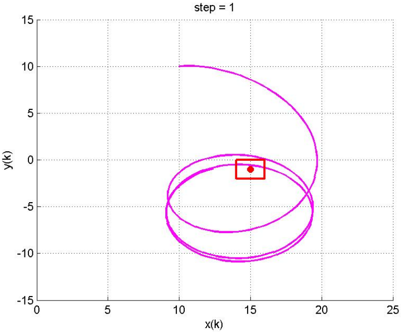
MIP-based solution for trajectory tracking of multi-agent formation

(Prodan et al., IFAC World Congress'11, Springer'13)

Centralized predictive control with non-convex state constraints



Decentralized predictive control with non-convex state constraints



FDI adjusted reference governor

Stoican and Olaru, Wiley, 2013

Fix z ($z \in S_z$) and let x_{ref} be the decision variable :

$$D_{x_{ref}} \triangleq \left\{ x_{ref} : \left(\{ -C_i x_{ref} \} \oplus N_i^F \right) \cap \left(C_i S_z \oplus N_i \right) = \emptyset,\ i = 1 \dots N \right\}.$$

Reference governor (Stoican et al. [2010]) :

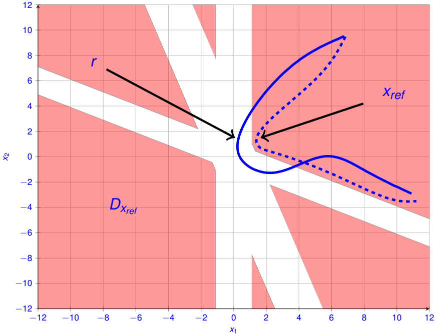
$$u^*_{ref[0,\tau-1]} = \underset{u_{ref[0,\tau-1]}}{\arg\min} \sum_{i=0}^{\tau-1} \left(||r_{[i]} - x_{ref[i]}||_{Q_r} + ||u_{ref[i]}||_{R_r} \right)$$

subject to :

$$x^+_{ref[i]} = A x_{ref[i]} + B u_{ref[i]}$$
$$x^+_{ref[i]} \in D_{x_{ref}}$$

Characteristics :

- fix gain
- flexible reference



Dynamic models of the microgrid components

(Prodan and Zio, Int. Journal of Electrical Power and Energy Systems, 2014)

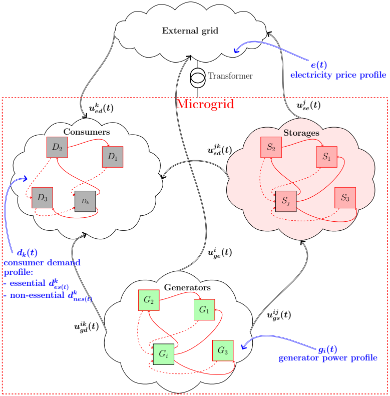
Consider the dynamic model of the electrical storage units S_j :

$$x_j(t+1) = (1-\sigma_j)x_j(t) + \sum_{M_{gs}(i,j)\neq 0} u_{gs}^{ij}(t) - \sum_{M_{sd}(i,j)\neq 0} u_{sd}^{jk}(t) - \sum_{M_{se}(j,k)\neq 0} u_{se}^{j}(t) + w_j(t),$$

with the *mixed-integer conditions* :

$$\begin{cases} 0 \leq u_{gs}^{ij}(t) \leq M\alpha_j(t), & \forall i \text{ with } M_{gs}(i,j)\neq 0, \\ 0 \leq u_{sd}^{jk}(t) \leq M(1-\alpha_j(t)), & \forall k \text{ with } M_{sd}(j,k)\neq 0, \\ 0 \leq u_{se}^{j}(t) \leq M(1-\alpha_j(t)), & \text{if } \exists j \text{ with } M_{se}(j)\neq 0, \end{cases}$$

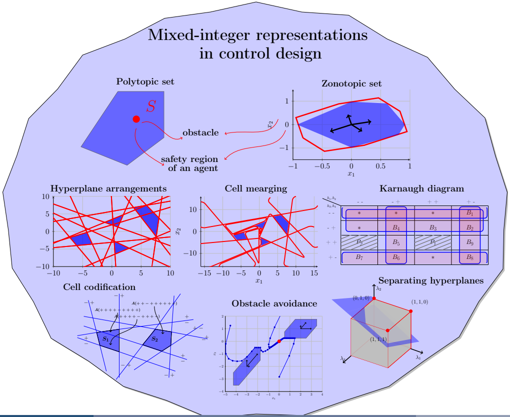
- $x_j(t) \in \mathbb{R}$ represents the amount of energy stored in S_j at time step t ;
- $\alpha_j(t) \in \{0,1\}$ are the auxiliary binary variables which govern the mode switching ;
- $\sigma_j \in \mathbb{R}^+$ hourly self-discharge decay ;
- M_{ab} adjacency matrix characterizing the links between components.



Outline

Conclusions

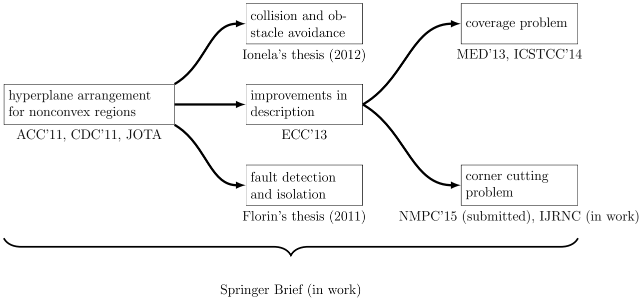
- Develop and bring to light new insights in the use of mixed-integer (MI) formulations for efficiently describing non-convex and non-connected regions appearing in a wide range of applications in control theory.
- Once the overall problem is brought to an improved formulation specialized solvers like CPLEX, Gurobi or SCIP are employed.
- Mixed-integer formulations provide one of the best ways of dealing with optimization problems with conflicting objectives.



Research interactions

Work done together with

- Sorin Olaru, Silviu Niculescu (L2S, CentraleSupélec)
- Morten Hovd (NTNU), Esten Grotli (SINTEF)
- Mircea Strutu, Dan Popescu (UPB)
- Enrico Zio (Chair EDF, CentraleSupélec)



References I

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

J.B.J. Fourier. Solution d'une question particulière du calcul des inégalités. *Nouveau Bulletin des Sciences par la Société Philomatique de Paris*, pages 317–319, 1826.

H. Minkowski. Geometrie der zahlen (erste lieferung). 1896.

G.B. Dantzig. Maximization of a linear function of variables subject to linear inequalities, in : Activity analysis of production and allocation. *(T.C. Koopmans, ed.), Wiley N.Y.*, pages 339–347, 1951.

Alan J Hoffman and Joseph B Kruskal. Integral boundary points of convex polyhedra. pages 223–246, 1956.

Ralph E Gomory et al. Outline of an algorithm for integer solutions to linear programs. *Bulletin of the American Mathematical Society*, 64(5) : 275–278, 1958.

J. Edmonds. Paths, trees, and flowers. *Canadian Journal of Mathematics 17*, pages 449–467, 1965.

M.R. Garey and D.S. 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.

L.G. Khachiyan. A polynomial algorithm in linear programming. *Soviet Mathematics Doklady 20*, pages 191–194, 1979.

N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica 4*, pages 373–395, 1984.

A Alan B Pritsker, Lawrence J Waiters, and Philip M Wolfe. Multiproject scheduling with limited resources : A zero-one programming approach. *Management science*, 16(1) :93–108, 1969.

NV Sahinidis and Ignacio E Grossmann. Reformulation of multiperiod milp models for planning and scheduling of chemical processes. *Computers & Chemical Engineering*, 15(4) :255–272, 1991.

M.G. 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*, volume 1, pages 107–111, Orlando, Florida, USA, 4-7 December 2001.

A. Richards, J. Bellingham, M. Tillerson, and J. How. Coordination and control of multiple uavs. *In AIAA Guidance, Navigation, and Control Conference*, Monterey, CA, 2002.

C. Schumacher, P. Chandler, and M. Pachter. UAV task assignment with timing constraints. *In AIAA Guidance, Navigation, and Control Conference*, Austin, Texas, 2003.

R.C. Buck. Partition of space. *American Mathematical Monthly*, 50(9) :541–544, 1943.

References II

T. Geyer, F.D. Torrisi, and M. Morari. Optimal complexity reduction of polyhedral piecewise affine systems. *Automatica*, 44(7) :1728–1740, 2008. ISSN 0005-1098.

I. Prodan, F. Stoican, S. Olaru, and Niculescu S.I. Enhancements on the Hyperplanes Arrangements in Mixed-Integer Techniques. *Journal of Optimization Theory and Applications*, 154(2) :549–572, 2012a. doi : 10.1007/s10957-012-0022-9.

Rubens JM Afonso and Roberto KH Galvão. Comments on “enhancements on the hyperplanes arrangements in mixed-integer programming techniques”. *Journal of Optimization Theory and Applications*, pages 1–8, 2013.

D. Avis and K. Fukuda. Reverse search for enumeration. *Discrete Applied Mathematics*, 65(1) :21–46, 1996.

F. Stoican, I. Prodan, and S. Olaru. Hyperplane arrangements in mixed-integer programming techniques. collision avoidance application with zonotopic sets. In *In Proceedings of the IEEE European Control Conference*, pages 3155–3160, 2013.

W. Blake and D. Multhopp. Design, performance and modeling considerations for close formation flight. *American Institute of Aeronautics and Astronautics Journal*, 1998.

A. Richards and J.P. How. Aircraft trajectory planning with collision avoidance using mixed integer linear programming. In *Proceedings of the 21th American Control Conference*, pages 1936–1941, Anchorage, Alaska, USA, 8-10 May 2002.

I.P. Massion, T. Keviczky, and M. Verhaegen. New approaches to distributed control of satellite formation flying. In *Proceedings 3rd International Symposium on Formation Flying, Missions and Technologies*, 2008.

M. Mora and M. Solar. A survey on the dynamic scheduling problem in astronomical observations. *Artificial Intelligence in Theory and Practice III*, pages 111–120, 2010.

J. Sousa, A.R. Girard, J.K. Hedrick, and F. Kretz. Real-time hybrid control of mobile offshore base scaled models. In *Proceedings of the 19th American Control Conference*, volume 1, pages 682–686, Chicago,Illinois, USA, 28-30 June 2000.

A.R. Girard, J.B. De Sousa, and J.K. Hedrick. A selection of recent advances in networked multivehicle systems. *Proceedings of the Institution of Mechanical Engineers, Part I : Journal of Systems and Control Engineering*, 219(1) :1–14, 2005.

A.I. Propoi. Use of linear programming methods for synthesizing sampled-data automatic systems. *Automation and Remote Control*, 24(7) : 837–844, 1963.

J. Richalet, A. Rault, J.L. Testud, and J. Papon. Model predictive heuristic control : Applications to industrial processes. *Automatica*, 14(5) : 413–428, 1978.

References III

C.R. Cutler and B.L. Ramaker. Dynamic matrix control-a computer control algorithm. In *Proceedings of the joint Automatic Control Conference*, volume 1, pages Wp5–B. American Automatic Control Council Piscataway, NJ, 1980.

A. Richards and O. Turnbull. Inter-sample avoidance in trajectory optimizers using mixed-integer linear programming. *Int. J. Robust Nonlinear Control*, 25 :521–526, 2015. doi : 10.1002/rnc.3101.

M. H. Maia and R. K. H. Galvão. On the use of mixed-integer linear programming for predictive control with avoidance constraints. *Int. J. Robust Nonlinear Control*, 19 :822–828, 2009. doi : 10.1002/rnc.1341.

R. Deits and R. Tedrake. Efficient mixed-integer planning for UAVs in cluttered environments. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, 2015.

F. Stoican, S. Olaru, M.M. Seron, and J.A. De Doná. Reference governor for tracking with fault detection capabilities. In *Proceedings of the 2010 Conference on Control and Fault Tolerant Systems*, pages 546–551, Nice, France, 6-8 October 2010.

I. Prodan and E. Zio. A model predictive control for reliable microgrid energy management. *International Journal of Electrical Power and Energy Systems*, 61(1) :399–409, 2014. doi : 10.1016/j.ijepes.2014.03.017.

I. Prodan, S. Olaru, R. Bencatel, J. Sousa, C. Stoica, and S.I. Niculescu. Receding horizon flight control for trajectory tracking of autonomous aerial vehicles. *Control Engineering Practice*, 21(10) :1334–1349, 2013a. doi : 10.1016/j.conengprac.2013.05.010.

I. Prodan, S. Olaru, C. Stoica, and S.I. Niculescu. Predictive control for trajectory tracking and decentralized navigation of multi-agent formations. *International Journal of Applied Mathematics and Computer Scienc*, 23(1) :91–102, 2013b. doi : 10.2478/amcs-2013-0008.

I. Prodan, S. Olaru, S. Stoica, and Niculescu S.I. Path following with collision avoidance and velocity constraints for multi-agent group formations. *Annals of the University of Craiova, Series : Automation, Computers, Electronics and Mechatronics*, 7(34) :33–38, 2010a. doi : ISSN:1841-0626.

I. Prodan, S. Olaru, C. Stoica, and S-I. Niculescu. On the tight formation for multi-agent dynamical systems. In *Agents and Multi-agent Systems Technologies and Applications*, volume LNAI 7372, pages 554–565. Springer, 2012b.

I. Prodan, F. Stoican, S. Olaru, C. Stoica, and S-I. Niculescu. Mixed-integer programming techniques in distributed mpc problems. In *Distributed MPC made easy*, number 69, pages 273–288. Springer, 2012c.

I. Prodan, R. Bencatel, S. Olaru, J. Sousa, C. Stoica, and S.I. Niculescu. Predictive control for autonomous aerial vehicles trajectory tracking. In *In Proceedings of the IFAC Nonlinear Model Predictive Control Conference*, pages 508–513, Noordwijkerhout, The Netherlands, 23-27 August 2012d.

References IV

F. Stoican, I. Prodan, and S. Olaru. On the hyperplanes arrangements in mixed-integer techniques. In *Proceedings of the 30th American Control Conference*, pages 1898–1903, San Francisco, California, USA, 29 June-1 July 2011a.

F. Stoican, I. Prodan, and S. Olaru. Enhancements on the hyperplane arrangements in mixed integer techniques. In *In Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference*, pages 3986–3991, Orlando, Florida, USA, 12-15 December 2011b.

I. Prodan, S. Olaru, C. Stoica, and S.-I. Niculescu. Predictive control for tight group formation of multi-agent systems. In *In Proceedings of the 18th IFAC World Congress*, pages 138–143, Milan, Italy, 28 August - 2 September 2011a.

I. Prodan, S. Olaru, C. Stoica, and S.I. Niculescu. On the limit behavior of multi-agent systems. In *The 8th IEEE International Conference on Informatics in Control, Automation and Robotics*, pages 344–349, Noordwijkerhout, The Netherlands, 28-31 July 2011b.

I. Prodan, G. Bitsoris, S. Olaru, C. Stoica, and S.I. 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 2012e.

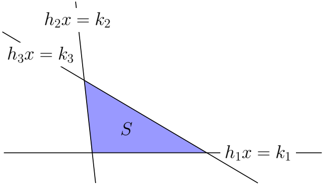
I. Prodan, S. Olaru, C. Stoica, and S.I. Niculescu. Predictive control for trajectory tracking and decentralized navigation of multi-agent formations. In *The 4th IEEE International Conference on Agents and Artificial Intelligence*, pages 206–214, Vilamoura, Portugal, 6-8 February 2012f.

I. Prodan, S. Olaru, C. Stoica, and S.I. Niculescu. Collision avoidance and path following for multi-agent dynamical systems. In *IEEE International Conference on Control Automation and Systems*, pages 1930–1935, Seoul, Korea, 27-30 October 2010b.

Illustrative example

Consider a triangle from $\mathbb{R}^2$ given by

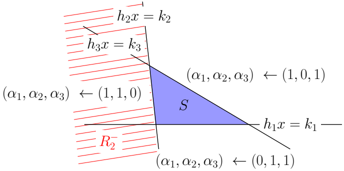
$$\begin{aligned} h_1 x &\leq k_1 \\ h_2 x &\leq k_2 \\ h_3 x &\leq k_3 \end{aligned}$$



Illustrative example

and its complement

$$\begin{aligned} -h_1 x &\leq -k_1 + M\alpha_1 \\ -h_2 x &\leq -k_2 + M\alpha_2 \\ -h_3 x &\leq -k_3 + M\alpha_3 \end{aligned}$$

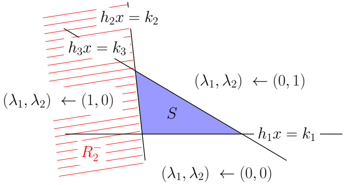


in the classical MI formulation.

Illustrative example

and its complement

$$\begin{aligned} -h_1 x &\leq -k_1 + M(\quad\ \lambda_1 + \lambda_2) \\ -h_2 x &\leq -k_2 + M(1 - \lambda_1 + \lambda_2) \\ -h_3 x &\leq -k_3 + M(1 + \lambda_1 - \lambda_2) \end{aligned}$$



in the reduced MI formulation.

In the reduced representation only $N_0 = \lceil \log_2 3 \rceil = 2$ binary variables are needed.

For region $\mathcal{R}^-(\mathcal{H}_2)$ associate tuple $(\lambda_1^2, \lambda_2^2) = (1, 0)$ which leads to the mapping

$$\alpha_2(\lambda) = 1 - \lambda_1 + \lambda_2$$