HAL
open science

An hybrid system approach to nonlinear optimal control problems

Jean-Guillaume Dumas, Aude Rondepierre

To cite this version:

Jean-Guillaume Dumas, Aude Rondepierre. An hybrid system approach to nonlinear optimal control problems. [Research Report] Université Grenoble Alpes (UGA). 2008. hal-00004191v4

HAL Id: hal-00004191

https://hal.science/hal-00004191v4

Submitted on 7 Jan 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

hal-00004191, version 4 - 7 Jan 2008

An hybrid system approach to nonlinear optimal control problems

Jean-Guillaume DUMAS* Aude RONDEPIERRE†

Abstract

We consider a nonlinear ordinary differential equation and want to control its behavior so that it reaches a target by minimizing a cost function. Our approach is to use hybrid systems to solve this problem: the complex dynamic is replaced by piecewise affine approximations which allow an analytical resolution. The sequence of affine models then forms a sequence of states of a hybrid automaton. Given a sequence of states, we introduce an hybrid approximation of the nonlinear controllable domain and propose a new algorithm computing a controllable, piecewise convex approximation. The same way the nonlinear optimal control problem is replaced by an hybrid piecewise affine one. Stating a hybrid maximum principle suitable to our hybrid model, we deduce the global structure of the hybrid optimal control steering the system to the target.

1 Introduction

Aerospace engineering, automatics and other industries provide a lot of optimization problems, which can be described by optimal control formulations: change of satellites orbits, flight planning, motion coordination (see e.g. [23] or [36] for more applications in aerospace industry). In general those optimal control problems are fully nonlinear ; since the years 1950-1970, the theory of optimal control has been extensively developed and has provided us with powerful results like dynamic programming [9] or the maximum principle [37]. A large amount of theory has been developed and resolutions are mainly numerical.

We here consider a dynamical system whose state is described by the solution of the following ordinary differential equation (ODE):

$$\begin{cases} \dot{X}(t) = f(X(t), u(t)) \\ X(0) = X_0 \end{cases} \tag{1}$$

*Université de Grenoble, laboratoire Jean Kuntzmann, Mathématiques Appliquées et Informatique, UMR CNRS 5224, Tour IRMA, BP 53, 38041 Grenoble, France. Email: Jean-Guillaume.Dumas@imag.fr

†Université Paul Sabatier, Institut de Mathématiques de Toulouse, 118 route de Narbonne, 31062 Toulouse, France. Email: rondep@mip.ups-tlse.fr

We present a hybrid (discrete-continuous) algorithm controlling the system (1) from an initial state X_0 at time $t = 0$ to a final state $X_f = 0$ at an unspecified time t_f. To reach this state, we allow the admissible control functions u to take values in a convex and compact polyhedral set $\mathbb{U}_m$ of $\mathbb{R}^m$, in order to minimize the following cost-function:

$$J(X, u) = \int_0^{t_f} l(X(t), u(t))dt \tag{2}$$

There exists two main approaches for the resolution of nonlinear optimal control problems (see e.g. [44] for a review of different solving methods).

A first class of solving methods is based on the Bellman dynamic principle and the characterization of the value function $J(X, u)$ in terms of viscosity solutions of a nonlinear Hamilton-Jacobi-Bellman (HJB) equation [19, 8, 7, 15, 10]. The related numerical methods, called direct methods, often uses time and/or space discretizations and are really efficient in small dimension. Unfortunately, their cost in memory is in general too expensive in high dimension.

The second class of methods is based on the Pontryagin Maximum Principle [37, 25]. This principle provides a pseudo-Hamiltonian formulation of optimal control problems and some necessary optimality conditions. In this context, the foremost numerical methods are the indirect shooting methods. These methods are very efficient in any dimension but require an *a priori* knowledge of the optimal trajectory structure.

Today's challenge is to reach a compromise between direct and indirect methods in order to enable the handling of high dimensional problems with a satisfactory precision and without any pre-simulation to locate optimal trajectories.

In this paper, we propose a different approach: the use of hybrid computing [20]. The idea is to approach nonlinear systems like (1) by simplified models which we can analytically study. Basically, an analytical approach must allow to improve approximations as is done by [27, 21]: the level of details allows to reach a compromise between quantitative quality of the approximation and the computational time. The latter has been done e.g. for biological systems, where simplifications (in relation to real data and in regard of model simulations) are possible, see [21]. We here choose to use a hybrid system modeling i.e. to approximate the original system (1) by a continuous and piecewise affine one. Contrary to [29], we want to focus on *both* the discrete and continuous aspects of the hybrid systems. We thus first compute on the fly a polyhedral mesh of phase space *and* the control space and approach the global system by piecewise affine dynamics in each mode: in each cell of this mesh, the system is piecewise affine ($\dot{X}(t) = AX(t) + Bu(t) + c$) and can be locally solved, mostly with analytical tools. We then find inside the cells the affine feedback controls à la [28] which enable a global controllability. Finally, to optimize the choice of the solutions, we derive from [38, Theorem 2.3] a hybrid maximum principle where the transitions are not constrained.

The paper is organized as follows: §2 presents algorithms for the hybrid approximation of nonlinear system. Next, in section 3, the controllability of the

system is studied and a new efficient algorithm computing a piecewise convex approximation of the controllable set is developed. The last section §4 is devoted to a heuristic approaching the resolution of optimal control problems: we first validate our hybrid approach for this resolution and establish a suitable hybrid maximum principle. We then show that the analysis of the optimality conditions gives us the general structure of optimal trajectories.

2 Hybrid Approximation of Nonlinear Systems

In this section, we model nonlinear control systems by way of piecewise affine systems. Our main objective here is to replace the nonlinear system (1) by another one, complex enough to reproduce the intrinsic properties of the initial system, but also simple enough to be analytically studied.

Let us consider the general nonlinear control system $\dot{X}(t) = f(X(t), u(t))$, where the admissible controls are measurable. Moreover we assume that the bounded functions u take values in a compact polytope $\mathbb{U}_m$ of $\mathbb{R}^m$. Classical numerical methods like Euler or Runge-Kutta usually propose time discretizations of nonlinear systems and then obtain discretizations of solution trajectories. On the contrary, we here use hybrid computation introduced in [20] and whose main idea is to compute an approximation of the nonlinear field f by means of a *state discretization*. Extended to control systems, the principle is as follows: for a given mesh of the control and state space $\mathbb{R}^n \times \mathbb{U}_m$, we compute a piecewise affine approximation of the nonlinear vector field f. This partition, associated to the so computed piecewise affine dynamic, defines a hybrid system which we can thus analyze. We thus first describe the construction of a piecewise affine approximation of the nonlinear system. We then propose a complete study of the interpolation error and the convergence of the approximation. Then we develop algorithms building an implicit mesh of the state and control space $\mathbb{R}^n \times \mathbb{U}_m$. This will enable us to define our hybrid model for nonlinear control systems.

2.1 Piecewise affine approximation of nonlinear systems

In this paragraph we want to approximate the nonlinear system (1) by a system:

$$\dot{X}(t) = f_h(X(t), u(t)) \tag{3}$$

where f_h is a piecewise affine approximation of the nonlinear vector field f.

2.1.1 Computation of the piecewise affine approximation

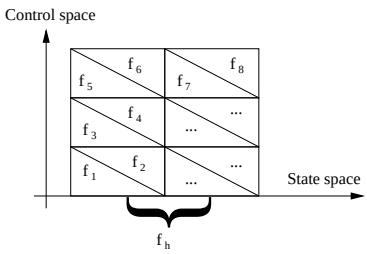
There are many possible linearizations by part ; especially in the context of nonlinear systems without terms of control, [16] and [27] propose a linearization technique based on the interpolation of the nonlinear vector field at the vertices of a given simplicial mesh of the state space. This technique is extended in this paragraph to control dynamics like (1) and therefore enables the computation

of a piecewise affine approximation with respect to the state X *and* the control u.

Let $\Delta = (\Delta_i)_{i\in I}$ be a given simplicial mesh of $\mathbb{R}^n \times \mathbb{U}_m$. Any affine function in dimension $n+m$ is uniquely defined by its values at $(n+m+1)$ affine independent points. Thus, as done e.g. in [26], in each cell Δ_i, we can compute an affine approximation f_h of f by interpolation at the vertices of the cell. Moreover, only one supplementary function evaluation can be sufficient to compute an approximation when the system is evolving to an adjacent cell.

Then, we consider a simplicial cell Δ_i of Δ, defined as the convex hull of its $(n+m+1)$ vertices: $\sigma_1, \ldots, \sigma_{n+m+1}$. Let f_i be the affine approximation of f computed by interpolation at the vertices of Δ_i. We have that: $f_i(X,u) = A_iX + B_iu + c_i$. The approximation f_i is computed by interpolation of f at the vertices of the cell Δ_i, so that: $\forall j,\ f(\sigma_j) = f_i(\sigma_j)$. Hence, this proves that $f(\sigma_j) = [\ A_i \mid B_i\ ]\sigma_j + c_i$ which in turn induces that $\forall j = 1, \ldots, n+m+1,\ f(\sigma_j) - f(\sigma_1) = [\ A_i \mid B_i\ ](\sigma_j - \sigma_1)$. We thus define M_i to be the $(n+m)\times(n+m)$ matrix, whose columns are the vectors of vertices: $\{\sigma_j - \sigma_1; j = 2, \ldots, n+m+1\}$ and F_i is the $(n+m)\times n$ matrix, whose columns are the vectors $\{f(\sigma_j) - f(\sigma_1); j = 2, \ldots, n+m+1\}$ so that $F_i = [\ A_i \mid B_i\ ]M_i$.

Consequently, by linear independence of the vertices of the simplex Δ_i, the square matrix M_i is non singular, so that we obtain the piecewise affine approximation of figure 1 by[1]:



$$\begin{cases} [\ A_i \mid B_i\ ] &= F_i\ M_i^{-1} \\ c_i &= f(\sigma_1) - [\ A_i \mid B_i\ ]\sigma_1 \end{cases} \tag{4}$$

$$f_h(X,u) = A_iX + B_iu + c_i, \quad \text{if } (X,u) \in \Delta_i$$

Figure 1: Definition of the piecewise affine approximation

2.1.2 Interpolation error and convergence of the Hybridization scheme

For autonomous nonlinear systems: $\dot{x}(t) = g(x(t))$, [27, §11.1] presents a study of the approximation error inside a compact set D under some regularity as-

[1] the c_i can be solved as well with any column as $\forall j = 1, \ldots, n+m+1,\ f(\sigma_j) - [\ A_i \mid B_i\ ]\sigma_j = c_i$.

sumptions on the field g. In the context of nonlinear control systems $\dot{X}(t) = f(X(t), u(t))$, the control is in general only measurable so that the required regularity for f is lost. Regularity assumptions on the initial nonlinear field f are usually chosen to fulfill the conditions of the Cauchy-Lipschitz theorem applied to control systems, see e.g. [41, 44, 40]. From now on we therefore assume that f is continuous, Lipschitz continuous over any compact set in X, uniformly in u. In other words, we have the condition (A_1):

$$(A_1) \quad \forall \Omega \Subset \mathbb{R}^n,\ \exists L_\Omega > 0,\ \forall (X_1, X_2) \in \Omega^2,\ \forall u \in \mathbb{U}_m,$$
$$\|f(X_2, u) - f(X_1, u)\| \leq L_\Omega \|X_2 - X_1\|.$$

Then for any given control u, we can prove the existence and uniqueness of solution trajectories of the nonlinear system (1): we first establish the regularity properties of the piecewise affine approximation f_h, required by the Cauchy-Lipschitz theorem. This then enables us to get an accurate bound on the error produced by our linearization scheme. We end this study by convergence results on our scheme.

Let us introduce the size h of the simplicial mesh $\Delta = (\Delta_i)_{i \in I}$ defined by: $h = \sup_{i \in I}\ h_i$ where $h_i = \sup_{x,y \in \Delta_i}\ \|x - y\|$ and $\|.\|$ denotes the ∞-norm on $\mathbb{R}^{n+m}$. We first need the continuity of the vector field f_h which is an application of [27, proposition 11.1.3] to a state-control space:

Proposition 1 *f_h is continuous over $\mathbb{R}^n \times \mathbb{U}_m$ and locally Lipschitz in X, i.e. for all compact subset Ω in $\mathbb{R}^n$, the restriction of f_h to $\Omega \times \mathbb{U}_m$ is Lipschitz continuous in X, uniformly in u.*

Then, from this continuities of f and f_h, we deduce the following bound on the magnitude of f_h:

Lemma 1 *f and f_h are locally bounded in X, uniformly in u on $\mathbb{R}^n \times \mathbb{U}_m$, i.e.:*

$$\forall R > 0, \exists C_R > 0, \forall (X, u) \in B(0, R) \times \mathbb{U}_m, \|f_h(X, u)\| \leq C_R$$

Moreover, if f is bounded on $\mathbb{R}^n \times \mathbb{U}_m$, then f_h is also bounded on $\mathbb{R}^n \times \mathbb{U}_m$ (with the same bound).

Consequently for any given control u, the conditions of the Cauchy-Lipschitz theorem are satisfied and we therefore obtain the existence and uniqueness of solutions to the Cauchy problem: $\dot{X}(t) = f_h(X(t), u(t)),\ X(0) = X_0$. Now under some non restrictive assumptions on the nonlinear vector field f, our interpolation error can be evaluated with respect to the size h of the mesh Δ of $\mathbb{R}^n \times \mathbb{U}_m$:

Proposition 2 (Interpolation error) *i. If f is L-Lipschitz on $\mathbb{R}^n \times \mathbb{U}_m$, then:*

$$\sup_{(X,u) \in \mathbb{R}^n \times \mathbb{U}_m} \|f(X, u) - f_h(X, u)\| \leq \frac{4L(n+m)}{n+m+1} h = \varepsilon(h)$$

ii. If f is continuously differentiable on $\mathbb{R}^n \times \mathbb{U}_m$, then for any compact set $\Omega \subset \mathbb{R}^n$, then there exists $L_\Omega > 0$ such that:

$$\sup_{(X,u)\in\Omega\times\mathbb{U}_m} \|f(X,u) - f_h(X,u)\| \leq \frac{4L_\Omega(n+m)}{n+m+1}h = \varepsilon_\Omega(h)$$

where: $L_\Omega = \max_{q'\in I/\Delta_{q'}\cap(\Omega\times\mathbb{U}_m)\neq\emptyset} \left(\sup_{(X,u)\in\Delta_{q'}} \|Df\| \right)$.

The principle of the proof of proposition 2 is first to evaluate the interpolation error inside one cell $\Delta_{q'}$ of the mesh Δ. If the so computed error does not depend on the given cell $\Delta_{q'}$, then we deduce the global interpolation error ; otherwise the error is evaluated on any compact set $\Omega \times \mathbb{U}_m$ of the state and control space. See [40, proposition 2.3.1] for more details.

Now, let $X_0 \in \mathbb{R}^n$ be a given initial point and let the state $X(.)$ (resp. $X_h(.)$) be the corresponding solution of the initial nonlinear (resp. piecewise affine) system: $\dot{X} = f(X,u)$ (resp. $\dot{X}_h = f_h(X_h,u)$). Now by introducing the attainable sets $\mathcal{A}(X_0,t)$ and $\mathcal{A}_h(X_0,t)$ i.e. the sets of points that can be reached from X_0 by the related system in time $s \leq t$, we obtain the following convergence results [40, proposition 2.3.2] of our piecewise affine approximation:

Proposition 3 (C^1 Convergence)
i. If f is Lipschitz continuous in X, uniformly in u, then for all $t > 0$ such that $X(t)$ and $X_h(t)$ are well defined:

$$\begin{aligned} \|X(t) - X_h(t)\| &\leq \frac{\varepsilon(h)}{L}(e^{Lt} - 1) \\ \|\dot{X}(t) - \dot{X}_h(t)\| &\leq \varepsilon(h)e^{Lt}. \end{aligned}$$

ii. Let f be Lipschitz continuous over any compact set Ω in X, uniformly in u (hypothesis A1) and assume that the attainable sets $\mathcal{A}(X_0,s)$, $s > 0$ are compact. Then for $t > 0$,

$$\begin{aligned} \|X(t) - X_h(t)\| &\leq \frac{\varepsilon_\Omega(h)}{L_\Omega}(e^{L_\Omega t} - 1) \\ \|\dot{X}(t) - \dot{X}_h(t)\| &\leq \varepsilon_\Omega(h)e^{L_\Omega t}, \end{aligned}$$

where Ω is defined by: $\Omega = \mathcal{A}(X_0,t) \cup \mathcal{A}_h(X_0,t)$.

We can thus see that the key point is the definition of the mesh Δ and its size h. We now show that meshing the whole space is not mandatory. We rather compute the mesh on the fly.

2.2 Implicit simplicial mesh

Previously we have seen how to build a piecewise affine approximation of the nonlinear dynamic f for a given simplicial mesh of the state and control domain. In order to perform our hybrid approximation, we need now a method to build a simplicial mesh of $\mathbb{R}^n \times \mathbb{U}_m$. As $\mathbb{R}^n \times \mathbb{U}_m$ is not bounded, it is algorithmically inconceivable to mesh the whole space $\mathbb{R}^n \times \mathbb{U}_m$. Our approach is then to implicitly define a mesh of our space so that the simplicial subdivision is made on the fly.

Let $h > 0$ be the discretization step. There exists several ways to mesh a given domain. To each mesh corresponds only one piecewise affine approximation (3) by interpolation. Among all the possible meshes of the space $\mathbb{R}^n \times \mathbb{U}_m$, we choose to select those for which $(0,0)$ is a fixed point of the system (3). Indeed, if the target is $X_f = 0$, we want to stay there once we have reached it. Therefore, $u(t_f)$ also has to be zero. A simple way to thus enforce $f_h(0,0) = (0,0)$ is to request the following property:

Property 1 *If $0 \in \Delta_i$ then 0 is a vertex of Δ_i.*

Lemma 2 *Let us assume that $f(0,0) = 0$. Let Δ be a mesh of $\mathbb{R}^n \times \mathbb{U}_m$ that satisfies the property 1. Then $(0,0)$ is also a fixed point of the piecewise affine system (3) built over Δ.*

Proof. At the vertices of the mesh, $f(\sigma) = f_h(\sigma)$ by the interpolating constraints and $(0,0)$ is a fixed point of the initial system by definition. Therefore, $f_h(0,0) = f(0,0) = 0$. □

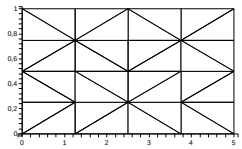
Moreover we assume that the mesh Δ also satisfies the property:

Property 2 *Let $D = (D_q)_{q \in \mathcal{Q}}$ be the projection of Δ over the state space $\mathbb{R}^n$.*

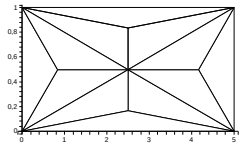
1. *D is a simplicial mesh of $\mathbb{R}^n$.*
2. *Let $(i,j) \in I^2$. We state: $D_i = p^{\perp}_{\mathbb{R}^n}(\Delta_i)$ and $D_j = p^{\perp}_{\mathbb{R}^n}(\Delta_j)$. Then we have: $D_i = D_j$ or $\mathring{D}_i \cap \mathring{D}_j = \emptyset$.*

The property 2 is a geometrical constraint on the position of the cells Δ_i in $\mathbb{R}^n \times \mathbb{U}_m$. Indeed, we now assume that, to each cell D_i corresponds a "column" of cells Δ_j whose projection over $\mathbb{R}^n$ is exactly D_i (see for example, figure 2a). We note $\mathcal{K}(q) = \{i \in I \ ; \ p^{\perp}_{\mathbb{R}^n}(\Delta_i) = D_q\}$ the set of all the indices of cells Δ_j belonging to the same column q. Then each domain $D_q \times \mathbb{U}_m$ can be decomposed into exactly *card* $\mathcal{K}(q)$ simplices in this way:

Lemma 3 $D_q \times \mathbb{U}_m = \bigcup_{q' \in \mathcal{K}(q)} \Delta_{q'}$.



(a)



(b)

Figure 2: Examples of meshes, for $n = m = 1$, with (a) satisfying the properties 1 and 2 and where (b) satisfies property 1 and not property 2.

The latter property will allow us in section 4 to have the same output constraints from a state cell D_q for each Δ_i in the column defined by $\mathcal{K}(q)$.

We are now able to define an algorithm computing an implicit mesh of the space $\mathbb{R}^n \times \mathbb{U}_m$. To satisfy property 2, we start by computing an implicit mesh $D = (D_q)_{q\in\mathcal{Q}}$ of the state space. Next, we compute an explicit triangulation of the control domain $\mathbb{U}_m$ (small and bounded). We lastly deduce an implicit simplicial mesh $(\Delta_i)_{i\in I}$ of $\mathbb{R}^n \times \mathbb{U}_m$.

2.2.1 Simplicial mesh of $\mathbb{R}^n$

The state space $\mathbb{R}^n$ is implicitly cut into n-dimensional cubes. Each cube is then meshed into $n!$ simplices and the resulting partition is a mesh of $\mathbb{R}^n$ (see [24],[27, chapter 11] for more details). Here, a n-cube C is defined by one point $a = (a_i)_{i=1\ldots n} \in \mathbb{R}^n$ and the length[2] h of its edges: $C = [a_1, a_1+h]\times\cdots\times[a_n, a_n+h]$. This will also be denoted by $C = a + [0, h]^n$. In the same way we define a mesh of $\mathbb{R}^n$ into n-cubes:

Definition 1 (Mesh of $\mathbb{R}^n$ into n-cubes) *Let $a = (a_i)_{i=1\ldots n}$ be a given point in $\mathbb{R}^n$ and $h > 0$. Then $(a + kh + [0, h]^n)_{k=(k_1,\ldots,k_n)\in\mathbb{Z}^n}$ is a mesh of $\mathbb{R}^n$ into n-cubes.*

Now, let us consider a n-cube $C = [a_1, a_1 + h] \times \cdots \times [a_n, a_n + h]$. We then introduce $\mathcal{S}_n$ the set of permutations of $\{1, \ldots, n\}$. For all $\varphi \in \mathcal{S}_n$, $D_{a,\varphi} = \{(x_1, \ldots, x_n) \in \mathbb{R}^n; 0 \le x_{\varphi(1)} - a_{\varphi(1)} \le \cdots \le x_{\varphi(n)} - a_{\varphi(n)} \le h\}$ is a simplex in $\mathbb{R}^n$, whose vertices are defined by:

$$\begin{cases} \forall i = 1, \ldots, p, & x_{\varphi(i)} = a_{\varphi(i)} \\ \forall i = p+1, \ldots, n, & x_{\varphi(i)} = a_{\varphi(i)} + h \end{cases}, p = 0, \ldots, n \tag{5}$$

[2]The Euclidean norm is used to compute lengths or sizes in the mesh.

Note that property 1 is satisfied if $a = 0$. Now each n-cube can be meshed into $n!$ simplices [24]:

Proposition 4 *$(D_{b,\varphi})_{\varphi \in S_n}$ is a mesh of size $\sqrt{n}h$ of the n-cube $C = b + [0, h]^n$.*

We then check that the respective meshes of two adjacent n-cubes coincide at their intersection so that we deduce a global mesh of $\mathbb{R}^n$ as in [40, proposition 3.1.2]:

Proposition 5 *$(D_{k,\varphi})_{k \in \mathbb{Z}^n, \varphi \in \mathcal{S}_n}$ is a simplicial mesh of size $\sqrt{n}h$ of the state space $\mathbb{R}^n$.*

2.2.2 Triangulation of $\mathbb{U}_m$

The control domain $\mathbb{U}_m$ is a bounded convex polyhedron, defined as the convex hull of its vertices. The study of simplicial subdivisions of such polytopes has been extensively developed in recent years and provides us with some efficient tools to compute them (e.g. Delaunay triangulation of the *Qhull* [6] software[3]). To build a regular mesh of size h of $\mathbb{U}_m$, we use an algorithm combining a Sierpinski-like discretization with the Delaunay triangulation.

Remark 1 *The triangulation of the control domain is computed once and for all at the beginning of the algorithm: indeed, in every control problem, there is in general no initial condition on the control. Therefore in each state cell, we need to compute all the possible related control cells and the triangulation can not be computed on the fly as for the state. This is possible for the control polytope as it is bounded and usually much smaller than the state space. Also this same explicit triangulation will be reused in every column cell of our hybrid automaton.*

2.2.3 Simplicial mesh of $\mathbb{R}^n \times \mathbb{U}_m$

Let $(D_q)_{q \in \mathcal{Q}}$ and $(U_j)_{j \in I}$ be the respective simplicial subdivisions of $\mathbb{R}^n$ and $\mathbb{U}_m$. The triangulation $(\Delta_i)_i$ of the state and control space $\mathbb{R}^n \times \mathbb{U}_m$ is built without new vertices via the Delaunay triangulation. This last criterion guarantees the property 2 to be satisfied and gives the following size of the resulting mesh:

Lemma 4 *Let D_q and U_j be two polytopes respectively in $\mathbb{R}^n$ and $\mathbb{R}^m$. Then we have:*

$$diam(D_q \times U_j) = \sqrt{diam(D_q)^2 + diam(U_j)^2}$$

Combining this lemma with proposition 5 gives that our mesh Δ is of size $\sqrt{n+1}h$.

In this section, we have described the main steps of the construction of an implicit simplicial mesh. Next, we will see that this mesh and the corresponding linear approximations, actually define a hybrid approximation of the initial system (1).

[3] tool for computation of convex hulls, Delaunay triangulation, Voronoi diagrams in 2d, 3d or higher dimension.

2.3 Hybrid Automaton

2.3.1 Definition of the hybrid model

The state of a hybrid system is described by two variables: the first one, denoted by $q(t)$, is discrete and takes values in a countable set $\mathcal{Q}$. The second one, denoted by $X(t)$, is continuous and takes values in the space $\mathbb{R}^n$. In each discrete mode $q \in \mathcal{Q}$, the variable X evolves continuously according to a dynamic f_q specific to the mode q. Therefore the hybrid dynamic and consequently the solving methods, depend on the choice of the discrete set $\mathcal{Q}$. Since the control functions are generally measurable and not continuous (at best piecewise continuous), trajectories $(X(.), u(.))$ in the space $\mathbb{R}^n \times \mathbb{U}_m$ are a priori discontinuous whereas the trajectories $X(.)$ are continuous in the state space. Furthermore,

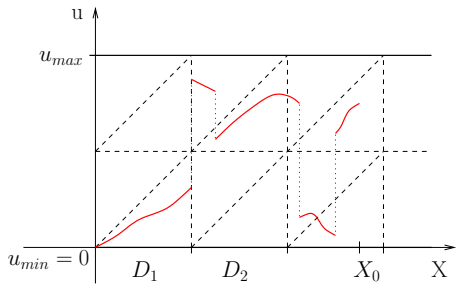


Figure 3: Example of discontinuous trajectory in the space $\mathbb{R} \times [0, u_{max}]$

as illustrated on figure 3, the notion of transitions between two cells of the mesh Δ has no meanings with respect to the control. Indeed, an admissible control function can switch at any time to a given cell Δ_i to another Δ_j, not necessarily adjacent to Δ_i, and without reaching one boundary of the system. In consequence the choice of the continuous variable, describing the state of the hybrid system, is restricted to the state variable $X \in \mathbb{R}^n$. We then introduce the following notations:

1. $\mathcal{Q}$ the countable set of indices of a simplicial mesh $(D_q)_{q \in \mathcal{Q}}$ of the state space $\mathbb{R}^n$.

2. $\mathcal{E} = \{(q, q') \in \mathcal{Q} \times \mathcal{Q} \ ; \ \partial D_q \cap \partial D_{q'} \neq \emptyset\}$ the transition set: a transition between two discrete modes q and q' is allowed if the associated cells D_q and $D_{q'}$ are adjacent.

3. $\mathcal{D} = \{D_q \ ; \ q \in \mathcal{Q}\}$ the collection of simplicial cells of the mesh of $\mathbb{R}^n$. By construction, each domain D_q induces a set of affine constraints on the state X in the mode q: $D_q = \{X \in \mathbb{R}^n \ ; \ N_q X + L_q \leq O\}$.

4. $\mathcal{U} = \{\mathbb{U}_m \ ; \ q \in \mathcal{Q}\} \subset \mathbb{U}_m$ the collection of control domains. Here, in each mode q the control domain is unchanged.

5. $\mathcal{F} = \{f_q \ ; \ q \in \mathcal{Q}\}$ the collection of piecewise affine field vectors defined by interpolation of the nonlinear field vector f (see section 2.1):

$$\begin{array}{rccc} \forall q \in \mathcal{Q},\ f_q : & D_q \times \mathbb{U}_m & \rightarrow & \mathbb{R}^n \\ & (X,u) & \mapsto & f_h(X,u) \end{array}$$

According to lemma 3, the vector f_q is locally defined by: $\forall q' \in \mathcal{K}(q), \forall (X,u) \in \Delta_{q'}, f_q(X,u) = A_{q'}X + B_{q'}u + c_{q'}$.

6. $\mathcal{G} = \{G_e \ ; \ e \in \mathcal{E}\}$ the collection of guards: $\forall e = (q,q') \in \mathcal{E}, G_e = \partial D_q \cap \partial D_{q'}$.

Remark 2 *The definition of hybrid systems proposed at first by M.S. Branicky and al. in [13], suggests the introduction of the collection $\mathcal{R} = \{R_e \ ; \ e \in \mathcal{E}\}$ of Reset functions here defined by: $\forall e = (q,q') \in \mathcal{E}, \forall X \in G_e, R_e(X) = \{X\}$. Indeed in our model, at any transition $e = (q,q') \in \mathcal{E}$, the corresponding cells D_q and $D_{q'}$ are adjacent so that we do not need to reinitialize the continuous variable X.*

We can now define our hybrid model and the associated notion of solution, extending that of [27, §2.1]:

Proposition 6 *$\mathcal{H} = (\mathcal{Q}, \mathcal{E}, \mathcal{D}, \mathcal{U}, \mathcal{F}, \mathcal{G})$ is a hybrid system approximating the nonlinear system (1).*

Definition 2 *A solution (or execution) of the hybrid system $\mathcal{H}$ is a hybrid trajectory $\chi = (\tau, q, X)$ where:*

- *$\tau = ([t_i, t_{i+1}])_{i=0\ldots r}$ ($t_i \leq t_{i+1}$ if $i < r$) is a sequence of real intervals.*
- *$q : \tau \rightarrow \mathcal{Q}$ describes the discrete state of $\mathcal{H}$ and $X : \tau \rightarrow \mathbb{R}^n$ the continuous behavior of $\mathcal{H}$.*

and satisfying, for all $i \in \{0, \ldots, r\}$ such that $t_i < t_{i+1}$,

- *For $t \in]t_i, t_{i+1}[$, $q(t)$ is constant and $X(t)$ remains inside $D_{q(t)}$.*
- *there is a measurable function $u :]t_i, t_{i+1}[\rightarrow \mathbb{U}_m$ such that: $\forall t \in]t_i, t_{i+1}[$, $\dot{X}(t) = f_{q(t)}(X(t), u(t))$.*

At any time t, the state of the hybrid system $\mathcal{H}$ is described by its position $X(t)$ in the state space $\mathbb{R}^n$ and the associated discrete mode $q(t) = q \in \mathcal{Q}$. For each nonempty time interval $]t_i, t_{i+1}[$, the system $\mathcal{H}$ is in mode $q(t) = q_i$ at the position $X(t) \in D_{q_i}$. In this mode, there exists an admissible control u such that $\mathcal{H}$ evolves continuously according to the dynamic: $\dot{X}(t) = f_{q_i}(X(t), u(t))$. As soon as the system reaches a guard $G_{(q_i, q_{i+1})}$ at time t_{i+1}, a discrete transition $e = (q_i, q_{i+1}) \in \mathcal{E}$ may occur from the mode q_i towards the mode q_{i+1} and we start again the same process from the position $X(t_{i+1})$ in mode q_{i+1} at time t_{i+1}. Note that t_i and t_{i+1} are the respective entering and exiting times in the cell D_{q_i}.

2.3.2 Computation of hybrid data on the fly

Let X_0 be the current position in the state space $\mathbb{R}^n$ of the hybrid system $\mathcal{H}$ defined in the previous section. In this section we first present an explicit algorithm to compute on the fly the current discrete mode associated to the position X_0 i.e. the cell D_q which contains the point X_0. We will assume in particular that the hybrid automaton is not Zeno (no infinite number of switch can occur in a finite time, see e.g. [27, §2.3] and [46]). Then we want to compute the local control constraints induced by each cell $\Delta_{q'}$ in the mode q.

Computation of the current state The principle of the algorithm is based on the definition of the implicit mesh Δ of the state and control space $\mathbb{R}^n \times \mathbb{U}_m$ (see section 2.2): we first compute the n-cube and then the simplex inside this cube that contain the given position X_0.
Computation of the n-cube C such that $X_0 \in C$: according to the definition 1, the n-cube C is defined by one origin point and the length h of its edges. Here the origin of the mesh Δ has to be chosen at the origin 0 of $\mathbb{R}^n \times \mathbb{U}_m$, so that the searched cube C can be written as: $C = kh + [0, h]^n$, where $kh = (k_1 h, \ldots, k_n h) \in h\mathbb{Z}^n$. Since we want to compute all the cubes C to which $X_0 = (x_1, \ldots, x_n) \in \mathbb{R}^n$ belongs, we necessarily obtain:

$$\forall i \in \{1, \ldots, n\}, \begin{cases} k_i = \left[\dfrac{x_i}{h}\right] & \text{if } x_i \notin h\mathbb{Z} \\ \\ k_i \in \{\dfrac{x_i}{h} - 1, \dfrac{x_i}{h}\} & \text{otherwise} \end{cases}$$

Computation of the simplex D_q such that $X_0 \in D_q$: let $C = kh + [0, h]^n$ be a n-cube containing X_0. According to the proposition 4, C can be divided into $n!$ simplices defined by:

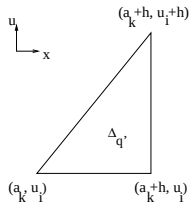
$$\begin{aligned} \forall \varphi \in \mathcal{S}_n, D_\varphi = \{X \in \mathbb{R}^n \ ; \ 0 \leq X_{\varphi(1)} - k_{\varphi(1)} h \leq \ldots \\ \cdots \leq X_{\varphi(n)} - k_{\varphi(n)} h \leq h\} \end{aligned} \tag{6}$$

To avoid the explicit computation of all the D_φ contained in the cube C, the trick is to directly compute the permutation φ which defines the searched simplex:

1. Form $L := [x_1 - k_1 h, \ldots, x_n - k_n h]$;
2. Apply a sort by selection algorithm to the list L. We so obtain the sorted list $\widetilde{L}$ and the permutation φ that performs the sort on the list L.
3. Deduce the simplex D_φ satisfying: $X_0 \in D_\varphi$

Computation of the local constraints on the control We consider a given cell D_q in the state space and again $\mathcal{K}(q) = \{q' \in I \ / \ p^{\perp}_{\mathbb{R}^n}(\Delta_{q'}) = D_q\}$ is the set of cells in Δ whose projection in $\mathbb{R}^n$ is exactly D_q. By construction, in each cell Δ_q, the control depends on the state. In this paragraph we want to compute the explicit control constraints. For that matter, we define $U_{q'}(X)$

to be the set of control constraints in the cell $\Delta_{q'}$ for a given $X \in \mathbb{R}^n$: $\forall X \in D_q,\ \forall q' \in \mathcal{K}(q),\ U_{q'}(X) = \{u \in \mathbb{R}^m \ / \ (X,u) \in \Delta_{q'}\}$. The problem now is to determine the geometrical structure of $U_{q'}(X)$ in $\mathbb{R}^m$. For instance, on figure 4, we can remark that the control domain $U_{q'}(x)$ is a segment (namely an 1-simplex) in $\mathbb{R}$ when $x \in]a_k, a_k + h]$, and is reduced to one point $\{u_i\}$, when $x = a_k$.



$$(x,u) \in \Delta_{q'}$$
$$\Updownarrow$$
$$x \in D_q = [a_k, a_k + h]$$
and
$$u \in U_{q'}(x) = \{u \in \mathbb{U}_1; 0 \leq u - u_i \leq x - a_k\}$$

Figure 4: Definition of D_q and $U_{q'}(x)$ in cell $\Delta_{q'}$ for $n = m = 1$

Let us take $X \in D_q$ and $q' \in \mathcal{K}(q)$. By definition, $\Delta_{q'}$ is a simplex in $\mathbb{R}^{n+m}$, so that it can be defined by a system of $(n+m+1)$ affine independent inequalities: $MY + d \geq 0$ where $M \in \mathbb{M}_{n+m+1,n+m}(\mathbb{R})$ and $d \in \mathbb{R}^{n+m+1}$. We now define the left and right parts of M as follows:

$$M_1 = M \begin{bmatrix} I_n \\ 0 \end{bmatrix} M_2 = M \begin{bmatrix} 0 \\ I_m \end{bmatrix}$$

so that: $M = [\ M_1 \mid M_2\ ]$. Thus we have $\forall (X,u) \in \Delta_{q'}, M_1 X + M_2 u + d \geq 0$. And we hence obtain a characterization of $U_{q'}$:

$$\begin{aligned} \forall X \in D_q, U_{q'}(X) &= \{u \in \mathbb{R}^m / (X,u) \in \Delta_{q'}\} \\ &= \{u \in \mathbb{R}^m / M_2 u + (M_1 X + d) \geq 0\} \end{aligned} \tag{7}$$

Therefore $U_{q'}(X)$ is determined by a finite number of affine inequalities and is a polyhedral set. Moreover, since Δ_q is bounded by construction, it follows that $U_{q'}(X)$ is a bounded polyhedral set, i.e. a polytope. Furthermore, it also has the following property:

Proposition 7 *For all $X \in \mathring{D}_q$. Then, for all $q' \in \mathcal{K}(q)$, $U_{q'}(X)$ is a m-simplex in $\mathbb{R}^m$.*

The proof is given in appendix A ; the idea is to count the number of intersections between the whole control space and the n-faces of the simplex $\Delta_{q'}$.

In this section, we have proposed a model of hybrid automaton for the approximation of nonlinear control systems $\dot{X} = f(X,u)$. The construction of

the hybrid model is performed on the fly, so that at any time we are able to determine the state and the associated local control constraints. Next section addresses the controllability problem, namely the existence of (non compulsorily optimal) solutions via this hybrid approximation.

3 Approximation of the controllable domain

In this section, we focus on the controllability of the nonlinear system (1) towards a given polyhedral target. For a given initial point $X_0 \in \mathbb{R}^n$ in the state space, we want to study the existence of admissible trajectories of the system that steers X_0 to the considered target at some finite time. The theory of linear control systems without constraints on the control have been extensively developed and provides some powerful results like the Kalman controllability criterion [31, 32] for autonomous linear systems like $\dot{X} = AX + Bu$. The use of this criterion can be extended to nonlinear control systems by using local controllability techniques an so provides local controllability results [33, 44]. Some global results can be obtained for a class of systems which depends affinely on the control [30, 34].

Our goal is to propose a constructive approach of nonlinear controllability by the study of the controllable domain (the set of initial points that can be steered to the target). We thus present now methods and algorithms to approximate the controllable domain of nonlinear systems by the way of the hybrid model defined in section 2.3. We first present our hybrid approach for the controllability of nonlinear systems: the nonlinear controllable domain is approximated by the hybrid one. Then, after stating some useful properties of the hybrid controllable domain, we develop a new algorithm computing a convex approximation of the hybrid controllable domain. Some experimental results will then validate our algorithm.

3.1 Hybrid approach to the nonlinear controllable domain

In section 2, we have built a hybrid model approximating every nonlinear control system like (1). We now need to verify that the use of our hybrid model enables the approximated solving of the initial control problem ; namely we analyze the approximation of the nonlinear controllable domain by the hybrid one. Let us specify the notion of controllability and controllable domain to a given target $\mathcal{T}_f$:

Definition 3 *$X_0 \in \mathbb{R}^n$ is controllable to $\mathcal{T}_f$ if and only if there exists $T \geq 0$ and $u : [0, T] \to \mathbb{U}_m$ measurable, such that the following system (S) admits a solution $X(.)$ over $[0, T]$:*

$$(S) \left\{ \begin{array}{lcl} \dot{X}(t) & = & f(X(t), u(t)) \\ X(0) & = & X_0, \quad X(T) \in \mathcal{T}_f \end{array} \right.$$

The set of controllable points in $\mathbb{R}^n$ to $\mathcal{T}_f$ is called the controllable domain *of the system (1).*

This definition can be applied to any control system, and particularly to the hybrid automaton previously described. Therefore, any $X_0 \in \mathbb{R}^n$ is controllable to the target $\mathcal{T}_f$ for the hybrid system $\mathcal{H}$ if and only if there exists an admissible hybrid trajectory, in the sense of definition 2, that steers X_0 to $\mathcal{T}_f$ in some finite time $T > 0$. In particular, for any controllable point $X_0 \in \mathbb{R}^n$, the hybrid system may accept several executions that steer X_0 to the target, and several different cell paths $q = (q_i)_i$ between X_0 and $\mathcal{T}_f$.

3.1.1 Global error of the hybrid approximation

Let $\mathcal{C}_{NL}(\mathbb{R}^n)$ and $\mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)$ be the respective controllable domains of the nonlinear system (1) and its hybrid model $\mathcal{H}$ towards a given target $\mathcal{T}_f$. To validate the hybrid approximation, we need to evaluate the distance between $\mathcal{C}_{NL}(\mathbb{R}^n)$ and $\mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)$, namely their Hausdorff distance defined by:

$$\begin{aligned} & d_H(\mathcal{C}_{NL}(\mathbb{R}^n), \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)) \\ & \quad = \max \left(\sup_{X \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)} \inf_{Y \in \mathcal{C}_{NL}(\mathbb{R}^n)} \|X - Y\|, \right. \\ & \qquad\qquad \left. \sup_{Y \in \mathcal{C}_{NL}(\mathbb{R}^n)} \inf_{X \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)} \|X - Y\| \right) \end{aligned}$$

Let $X_0 \in \mathcal{C}_{NL}(\mathbb{R}^n)$ be a controllable point to $\mathcal{T}_f$ by the nonlinear system (1). Hence, according to definition 3, there exist a finite time $T > 0$ and an admissible control $u_0 : [0, T] \to \mathbb{U}_m$ such that the system: $\dot{X}(t) = f(X(t), u(t)), \quad X(0) = X_0$ has a solution X satisfying: $X(T) \in \mathcal{T}_f$. Then we can easily build a new point $Y_{h,0}$ that is controllable by the hybrid system $\mathcal{H}$ in the same time T and control u_0 as for X_0: $Y_{h,0} = \widetilde{Y}(T)$ where $\widetilde{Y}$ is the unique solution of the system: $\dot{\widetilde{Y}}(t) = -f_h(\widetilde{Y}(t), u_0(T - t)), \quad \widetilde{Y}(0) = X(T)$ obtained from (1) by time reversal on the time interval $[0, T]$. We thereupon deduce: $Y_{h,0} \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)$ and $\inf_{Y \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)} \|X_0 - Y\| \leq \|X_0 - Y_{h,0}\|$. According to the convergence results established in proposition 3, and assuming that f is Lipschitz in X, uniformly in u, we thus obtain: $\|X_0 - Y_{h,0}\| \leq \dfrac{\varepsilon(h)}{L}(e^{LT} - 1)$, where $\varepsilon(h)$ denotes the interpolation error between f and its piecewise affine approximation f_h ; we consequently have shown that for all $X_0 \in \mathcal{C}_{NL}(\mathbb{R}^n)$:

$$\exists T > 0, \inf_{Y \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)} \|X_0 - Y\| \leq \frac{\varepsilon(h)}{L}(e^{LT} - 1).$$

Moreover, starting from a controllable point of the automaton $\mathcal{H}$, we prove in the same manner that: $\forall Y_{h,0} \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n), \ \exists T_h \geq 0, \ \inf_{X \in \mathcal{C}_{NL}(\mathbb{R}^n)} \|X - Y_0\| \leq \dfrac{\varepsilon(h)}{L}(e^{LT_h} - 1)$. Since the final T is a priori unbounded, we introduce the notion of controllable domain in time less than or equal to $T > 0$, denoted by $\mathcal{C}_{NL,[0,T]}(\mathbb{R}^n)$ in the nonlinear case and $\mathcal{C}_{\mathcal{H},[0,T]}(\mathbb{R}^n)$ in the hybrid one. We have shown:

Proposition 8 *Let the nonlinear function $f : \mathbb{R}^n \times \mathbb{U}_m \to \mathbb{R}^n$ be Lipschitz continuous in $X \in \mathbb{R}^n$, uniformly in $u \in \mathbb{U}_m$. Then:*

$$d_H\left(\mathcal{C}_{NL,[0,T]}(\mathbb{R}^n), \mathcal{C}_{\mathcal{H},[0,T]}(\mathbb{R}^n)\right) \leq \varepsilon(h)\frac{e^{LT}-1}{L},$$

where the interpolation error $\varepsilon(h)$ satisfies: $\lim_{h\to 0} \varepsilon(h) = 0$.

Using the second part of the proposition 3, it is then straightforward to generalize this result as:

Proposition 9 *Let $f : \mathbb{R}^n \times \mathbb{U}_m \to \mathbb{R}^n$ be Lipschitz continuous over any compact set in $X \in \mathbb{R}^n$, uniformly in $u \in \mathbb{U}_m$. Then there exists $L_\Omega > 0$ such that:*

$$d_H\left(\mathcal{C}_{NL,[0,T]}(\mathbb{R}^n), \mathcal{C}_{\mathcal{H},[0,T]}(\mathbb{R}^n)\right) \leq \varepsilon_\Omega(h)\frac{e^{L_\Omega T}-1}{L_\Omega},$$

where $\Omega \subset \mathbb{R}^n$ is the compact set defined by the union of $\mathcal{C}_{NL,[0,T]}(\mathbb{R}^n)$ and $\mathcal{C}_{\mathcal{H},[0,T]}(\mathbb{R}^n)$.

3.1.2 Exploration reduction: controllable domain in finite paths of modes

We have shown that the hybrid controllable domain is a valid approximation of the nonlinear one. From now on, we consider that a given point X_0 in the state space is approximatively controllable by the initial nonlinear system (1) to the considered target if it belongs to the controllable domain of its hybrid approximation. Unfortunately the controllable domain of a hybrid system, like its attainable set, are rarely computable. One of the main reasons is that the number of discrete transitions needed to reach some modes of the hybrid system is unbounded [27, chapter 7]: by time reversal, the computation of the controllable domain comes down to the computation of the attainable sets from the target. Then, if the given point X_0 is "far from the target", all the admissible paths of cells between X_0 and the target have to be explored. Hence an infinite number of discrete transitions are to be simulated and the computation can not be done in finite time. Therefore an approach that is more favorable to the simulation is required. We first assume that:

(A_2) *The target $\mathcal{T}_f$ is a subset of some cell D_q in the state space $\mathbb{R}^n$: $\exists q \in \mathcal{Q}, \mathcal{T}_f \subset D_q$*

The assumption (A_2) is non restrictive: indeed, if there are several cells D_q whose intersection with the target $\mathcal{T}_f$ is non-empty, then $\mathcal{T}_f$ can be decomposed into the union of its intersections with each cell: $\mathcal{T}_f = \bigcup_{q\in\mathcal{Q}/D_q\cap\mathcal{T}_f\neq\emptyset}(D_q \cap \mathcal{T}_f)$. There, we have several targets to reach (as many new targets as intersections) each of those satisfying the hypothesis (A_2).

We first consider a given mode $q \in \mathcal{Q}$ of our hybrid model $\mathcal{H}$ and a local target $\mathcal{T}_q$ (i.e.: $\mathcal{T}_q \subset D_q$). We introduce the concept of controllable domain in the given mode q:

Definition 4 (Controllable domain in a mode q) *Let $q \in \mathcal{Q}$ be a given discrete mode of our hybrid model $\mathcal{H}$ and $\mathcal{T}_q$ a subset of D_q. $X_0 \in D_q$ is controllable to $\mathcal{T}_q$ in mode q if there exist a finite time $T > 0$ and an admissible control u such that the hybrid problem:*

$$\dot{X}(t) = f_h(X(t), u(t)), \quad X(0) = X_0, \quad X(T) \in \mathcal{T}_q$$

has a solution, denoted by X_h, satisfying: $\forall t \in [0, T],\ X_h(t) \in D_q$.

The controllable domain in mode q is the set of all the controllable points in mode q and is denoted by $\mathcal{C}_{\mathcal{H}}(\mathcal{T}_q, D_q)$.

Hence, any point in D_q from which the local target $\mathcal{T}_q$ is attainable while remaining within the cell D_q, necessarily reaches $\mathcal{T}_{q'} \supset \mathcal{T}_q$; hence:

Lemma 5 $\mathcal{T}_q \subset \mathcal{T}_{q'} \subset D_q \Rightarrow \mathcal{C}_{\mathcal{H}}(\mathcal{T}_q, D_q) \subset \mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q'}, D_q)$

The definition 4 can be easily extended to any subset of the state space, and especially to the set Ω_γ associated to $\gamma = (q_i)_{i=1\ldots r}$. This set, shown on figure 5, is defined by $\Omega_\gamma = \bigcup_{i=1}^{r} D_{q_i}$ where $\gamma = (q_i)_{i=0\ldots r}$ is a finite sequence of adjacent modes of our hybrid automaton $\mathcal{H}$ satisfying $\forall i = 1, \ldots, r-1,\ D_{q_i} \cap D_{q_{i+1}} \neq \emptyset$.

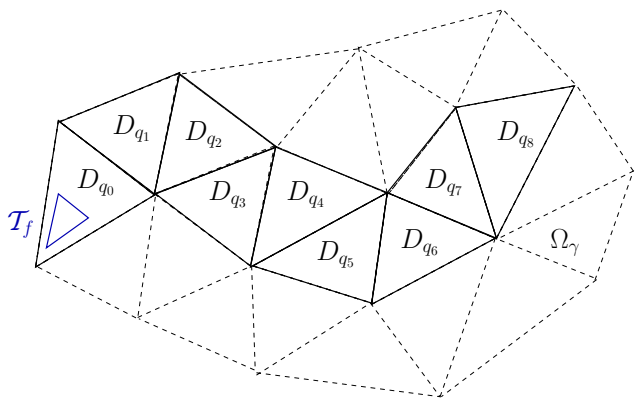


Figure 5: Definition of the domain Ω_γ along the path $\gamma = (q_i)_{i=0\ldots 8}$

Let $\mathcal{T}_f$ be a given subset of Ω_γ and the target for the controllability problem. We introduce the set $\mathcal{C}_{\mathcal{H}}(\mathcal{T}_f, \Omega_\gamma)$ of points that can be steered in finite time by the hybrid model $\mathcal{H}$ to the target $\mathcal{T}_f$ while remaining within Ω_γ, in the given sequence of modes. Our idea is to build an approximation of the controllable set inside Ω_γ, recursively defined along the path. Assuming without loss of generality that: $\mathcal{T}_f \subset D_{q_0}$, the principle consists in starting from the cell D_{q_0}: we first compute the local controllable domain in mode q_0. Then its intersection with the current guard $G_{(q_0,q_1)}$ defines a new target now in mode q_1 and we pursue the approximation the same way (the spreading rule is illustrated on

figure 6). The algorithm stops when the last cell q_r is reached or when the intersection between the current controllable domain and the next guard is empty.

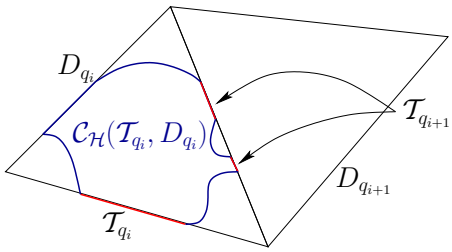


Figure 6: Spreading rule for the controllable domain from mode q_i to mode q_{i+1}

This induces a recursive definition of an under-approximation of the hybrid controllable domain inside Ω_γ:

Proposition 10 *We define:* $\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma) = \mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_0}, D_{q_0}) \cup \mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_1}, D_{q_1}) \cup \cdots \cup \mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_r}, D_{q_r})$, *where* $\mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_i}, D_{q_i})$ *denotes the set of points inside* D_{q_i} *from which the local target* $\mathcal{T}_{q_i} \subset D_{q_i}$ *is attainable in finite time while remaining within the cell* D_{q_i}. *The current target* $\mathcal{T}_{q_i}$ *is recursively defined by:*

$$\begin{cases} \tau_0 = \mathcal{T}_f \\ \tau_{i+1} = \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i}) \cap G_{(q_i, q_{i+1})}, \quad i = 0, \ldots, r-1 \end{cases}$$

$\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma)$ *is an under-approximation of* $\mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$, *hence :* $\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma) \subset \mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$.

Proof. We proceed by induction. Let us define for $i = 0, \ldots, r$, the property $\mathcal{P}(i) \ : \ \mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_0}, D_{q_0}) \cup \cdots \cup \mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_i}, D_{q_i}) \subset \mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$.

First, let $X_0 \in \mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_0}, D_{q_0})$. Since $D_{q_0} \subset \Omega_\gamma$ and $\mathcal{T}_{q_0} = \mathcal{T}_f$, the point X_0 is actually controllable to the $\mathcal{T}_f$ while remaining within Ω_γ. This proves that $X_0 \in \mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$ and $\mathcal{P}(0)$ is true.

Second, let us assume that the property $\mathcal{P}(i)$ is satisfied and $X_0 \in \bigcup_{k=0}^{i+1} \mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_k}, D_{q_k})$. By assumption, if $X_0 \in \bigcup_{k=0}^{i} \mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_k}, D_{q_k})$ then $X_0 \in \mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$. We can thus assume $X_0 \in \mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_{i+1}}, D_{q_{i+1}})$. By definition of the local controllable domain inside the cell $D_{q_{i+1}}$, there exists $Y_0 \in \mathcal{T}_{q_{i+1}}$ such that the hybrid system $\mathcal{H}$ steers X_0 to Y_0 while remaining inside $D_{q_{i+1}}$, and henceforth inside Ω_γ. Moreover by construction: $\mathcal{T}_{q_{i+1}} = G_{q_i, q_{i+1}} \cap \mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_i}, D_{q_i})$, so that Y_0 is steerable to the target $\mathcal{T}_f$ while remaining inside Ω_γ. By transitivity, we conclude that X_0 is controllable to the target $\mathcal{T}_f$ while remaining within Ω_γ: $X_0 \in \mathcal{C}_{\mathcal{H}}(\mathcal{T}_f, \Omega_\gamma)$ and the property $\mathcal{P}(i+1)$ is satisfied. □

The decomposition of the controllable domain in Ω_γ in local controllable sub-domains expressed in proposition 10 draws the structure of the algorithm presented in section 3.2 for the hybrid controllable domain computation. Actually, as shown next, it is also possible to use this spreading in different directions.

3.1.3 Expansion of the controllable domain in r iterations

In paragraph 3.1.2 we have proposed a constructive approach of the hybrid controllable domain to a given target Ω_γ. Its key points are described by the proposition 10. In practise, this process may be used in two different manner illustrated on figures 7 and 8.

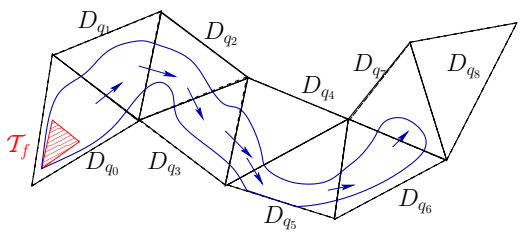


Figure 7: Controllable under-approximation in a given finite cells path $\gamma = (q_i)_{i=0,\ldots,8}$.

The first is an extension of the process described previously: we are given a finite sequence $\gamma = (q_i)_{i=0,\ldots,r-1}$ of adjacent modes of the hybrid model $\mathcal{H}$ and we compute the controllable under-approximation as expressed in proposition 10 and illustrated on figure 7. A weakness of this method is the arbitrary choice of the path in which we want to compute our controllable under-approximation. As we will see later a way to exhibit such a path is to start with a cell large enough to contain both the source and the target. Then such an approximation at a given level could guide the selection of an interesting path at a more refined approximation level.

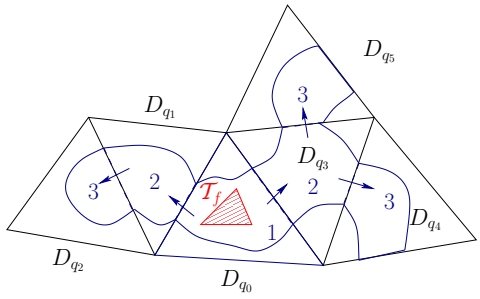


Figure 8: Controllable under-approximation in 3 iterations from the target $\mathcal{T}_f$.

The second idea is based on a more global approach of the hybrid controllable domain. There, we compute an under-approximation of the global hybrid controllable domain $\mathcal{C}_\mathcal{H}(\mathbb{R}^n)$ by time reversal starting from the cell containing the target $\mathcal{T}_f$. The difference is that we only impose a maximum distance to the target. The principle is illustrated on figure 8: we compute the cell D_{q_0} of

the state space containing the target ($\mathcal{T}_f \subset D_{q_0}$). Starting from D_{q_0} by time reversal, we compute the local controllable domain to $\mathcal{T}_f$ in this cell and then its intersections with *all the guards (i.e. facets) of* D_{q_0}. Each non empty intersection defines a new target and a new cell at the next iteration. This process is pursued at the p-th iteration from the controllable domains computed at the $(p-1)$-th iteration.

3.2 Computation of a convex approximation of the hybrid controllable domain

We have studied the approximation of the controllable domain to a target $\mathcal{T}_f$ of a nonlinear system by the controllable domain of its hybrid approximation. In view of simulations, we also developed a more constructive approach for the hybrid controllable domain computation. This approach is based on local controllable domains propagation and reduces the state space exploration. Now, we want to compute the set of controllable points (or attainable points by time reversal) in $\mathbb{R}^n$, i.e. the set of initial points that can be steered to a given target $\mathcal{T}_f$ in finite time. The considered class of systems is the one of piecewise affine hybrid systems defined in section 2.3.

A first method for the attainable sets computation consists in extracting, by abstraction, a simplified model equivalent to the initial one for the regarded property [2, 3, 43]. The main problem is to find the right abstraction that enables to conclude under a reasonable complexity. Another class of methods developed for safety verification is based on approximations by unions of polytopes. For instance, [4] compute an over-approximation of the attainable set and can thus certify that the system can not escape from an admissible set of states. On the contrary our idea is to guaranty as much as possible the controllability of a given initial point. In [22], we proposed an algorithm to compute an under-approximation in time $T > 0$ of the controllable set without state constraints. In [39], an extension of this algorithm to piecewise affine systems under constant control constraints is presented. Here, these techniques are adapted to our hybrid model $\mathcal{H}$ for bounded convex polyhedral targets. From now on, we assume:

(A_3) *The target $\mathcal{T}_f$ is a bounded polytope of $\mathbb{R}^n$ defined as the convex hull of its vertices.*

3.2.1 Convex approximation within a given state cell

Let us consider a given discrete mode $q \in \mathcal{Q}$ of our hybrid model, or equivalently a given cell D_q in the state space. Let $\mathcal{T}_q \subset D_q$ be a local target in mode q satisfying the hypothesis (A_3): $\mathcal{T}_q = Conv(s_{q,1}, \ldots, s_{q,k})$. In this paragraph, we address the problem of approximating the local controllable set $\mathcal{C}_{\mathcal{H}}(\mathcal{T}_q, D_q)$ i.e. the set of points in D_q steerable in finite time to $\mathcal{T}_q$ while remaining within D_q. Our main idea is to compute some specific controllable points in D_q ; their convex hull defines a convex approximation of the hybrid controllable domain in mode q.

Now, any point X_0 in D_q is controllable to a target $\mathcal{T}_q$ in mode q if it is attainable from $\mathcal{T}_q$ by time reversal. Consequently, a simple way to compute controllable points in a mode q is: for an admissible control u_j, we compute by time reversal the hybrid trajectories $X[s_{q,i}, u_j]$ coming from the vertex $s_{q,i}$ of $\mathcal{T}_q$. We then introduce their intersections, when they exist, with the boundary of D_q:

$$X_{i,j} = X[s_{q,i}, u_j](T_i) \tag{8}$$

where $T_i = \sup\{t > 0; X[s_{q,i}, u](t) \in \partial D_q\}$. As illustrated on figure 9, when the trajectory $X[s_{q,i}, u_j]$ remains inside D_q on a non empty time interval, all points between s_{q_i} and the corresponding intersection $X_{i,j}$ is controllable to the target $\mathcal{T}_q$. Our approximation will be the convex hull of all the points computed this way.

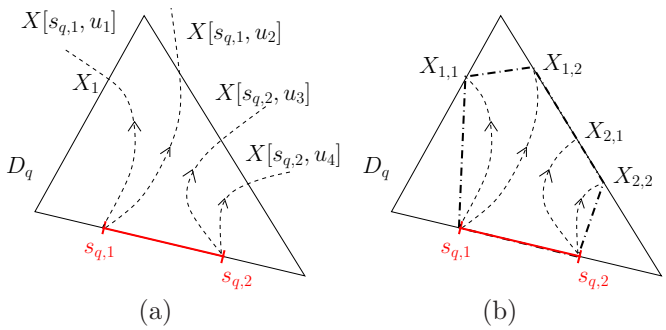


Figure 9: Principle of the convex approximation in a state cell D_q: (a) along the trajectory $X[s_{q,1}, u_1]$, only points between $s_{q,1}$ and X_1 are controllable to $s_{q,1} \in \mathcal{T}_q$ (b) Convex approximation

The main difficulty in this approach is the choice of control functions u. Indeed they have to be simple enough to enable the efficient computation of the related trajectories, but also significant enough to obtain a good quality approximation of the hybrid controllable domain. According to the construction of the mesh $\Delta = (\Delta_{q'})_{q' \in I}$ of the state and control space developed in section 2.2, there exists a column of cells $\{\Delta_{q'}; q' \in \mathcal{K}(q)\}$ in $\mathbb{R}^n \times \mathbb{U}_m$ whose projection on the state space $\mathbb{R}^n$ is exactly D_q. We recall that the cells of this column are defined by their indices this way: $\mathcal{K}(q) = \{i \in I \; ; \; p^{\perp}_{\mathbb{R}^n}(\Delta_i) = D_q\}$. By definition of our hybrid model, in each cell $\Delta_{q'}$ of this column, the hybrid dynamic f_h is affine. Our idea is then to choose a control function such that the control and state trajectory $(X(t), u(t))$ remains within only one cell $\Delta_{q'}$ $(q' \in \mathcal{K}(q))$ while $X(t)$ remains within the state cell D_q:

$$\exists T > 0, \; \forall t \in [0, T], X(t) \in D_q \;\; \text{and} \;\; (X(t), u(t)) \in \Delta_{q'} \tag{9}$$

We thus ensure that the hybrid dynamic is affine over the time interval $[0, T]$.

Our method is then to consider *feedback affine controls*:

$$u_j(t) = F_j X(t) + g_j \tag{10}$$

such that $(X(t), u(t))$ evolves on the edges of our state and control mesh Δ. Such controls at the position $X(t)$ correspond to the vertices of the control domains $U_{q'}(X(t))$ defined in paragraph 2.3.2 and their definition is illustrated on figure 10.

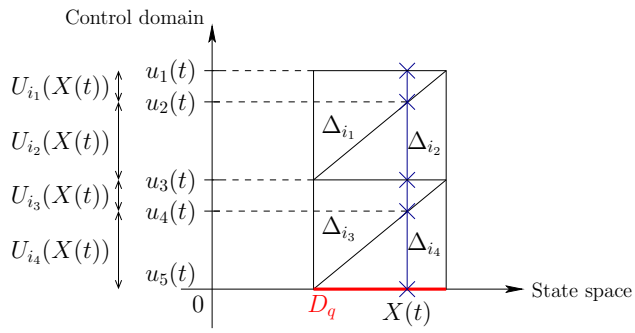


Figure 10: Control $u_i(t)$ chosen for the computation of a convex approximation of the hybrid controllable domain in mode q

The principle of our approximation algorithm 1 in mode q is as follows: for each previously defined control u_j and each vertex $s_{q,i}$ of the target $\mathcal{T}_q$,

1. Test if the trajectory by time reversal $X[s_{q,i}, u_j]$ remains inside D_q on a non empty time interval (notice that for this test, we do not need to explicitly compute the trajectory, see [40, chapter 5, paragraph 5.1.2]). If not, stop.

2. Computation of $X[s_{q,i}, u_j]$ by solving: $\dot{X}(t) = A_{q'}X(t) + B_{q'}u_j(t) + c_{q'}$, i.e. $\dot{X}(t) = (A_{q'} + B_{q'}F_j)X(t) + (B_{q'}g_j + c_{q'})$.

3. Computation of its first intersection with the boundary of D_q.

Let $\Lambda(\mathcal{T}_q, D_q)$ be the convex hull of the so computed controllable points $X_{i,j}$ defined by (8) and computed by the algorithm 1. By construction, we deduce:

Proposition 11 *$\Lambda(\mathcal{T}_q, D_q)$ is a convex approximation of the hybrid controllable domain $\mathcal{C}_{\mathcal{H}}(\mathcal{T}_q, D_q)$ in mode q. Moreover if $\mathcal{C}_{\mathcal{H}}(\mathcal{T}_q, D_q)$ is convex, $\Lambda(\mathcal{T}_q, D_q)$ is a* guaranteed under-approximation*:*

$$\mathcal{C}_{\mathcal{H}}(\mathcal{T}_q, D_q) \text{ convex } \Rightarrow \Lambda(\mathcal{T}_q, D_q) \subset \mathcal{C}_{\mathcal{H}}(\mathcal{T}_q, D_q).$$

Algorithm 1 ConvexApproximation: Convex approximation of the hybrid controllable domain in a mode

Require: $\mathcal{H}$, q a discrete mode of the hybrid automaton $\mathcal{H}$, $\mathcal{T}_q = Conv(s_{q,1}, \ldots, s_{q,k})$ the local target in mode q.

Ensure: A convex approximation of the controllable domain $\mathcal{C}_{\mathcal{H}}(\mathcal{T}_q, D_q)$.

Initialization: $\Lambda := \emptyset$;
{Computation of the indices $q' \in \mathcal{K}(q)$ of cells $\Delta_{q'}$ whose projection is exactly D_q :}
$\mathcal{K}(q) := \{q' \in I \ / \ p^{\perp}_{\mathbb{R}^n}(\Delta_{q'}) = D_q\}$
{Set of already tested controls:} $\Sigma := \emptyset$.
for all $q' \in \mathcal{K}(q)$ **do**
  {Computation of the polytope $U_{q'}(X)$ of control constraints (see §2.3.2):}
  $U_{q'}(X) := ControlConstraints(\Delta_{q'})$;
  {For each non tested vertex of $U_{q'}(X)$,}
  for all i from 1 to card($U_{q'}(X)$) such that $U_{q'}(X)[i] \notin \Sigma$ **do**
    {i-th vertex of Σ:} $u := t \rightarrow U_{q'}(X(t))[i]$; $\Sigma := \Sigma \cup \{u\}$;
    {For each vertex $s_{q,j}$ of the target $\mathcal{T}_q$,}
    for all j from 1 to k **do**
      if the trajectory $X[s_{q,j}, u]$ remains in D_q on a non empty time interval **then**
        {Computation of the intersection of $X[s_{q,j}, u]$ with ∂D_q:}
        $(X_f, ExitFace) := OutCell^{\dagger}(\mathcal{H}, q, s_{q,j}, u)$;
        {X_f, if it exists, is controllable,}
        if $X_f \neq \emptyset$ **then** $\Lambda := \Lambda \cup \{X_f\}$; **end if**
      end if
    end for
  end for
end for
Return Conv(Λ).

† $(X_f, ExitFace) := OutCell(\mathcal{H}, q, X_0, u)$ computes the intersection point X_f of the trajectory $X[X_0, u]$ of the hybrid automaton $\mathcal{H}$ with the state cell D_q. $ExitFace$ is the facet of D_q satisfying: $X_f \in ExitFace$.

3.2.2 Expansion of the convex approximation and approximation error

Let us consider a finite sequence of adjacent discrete modes of the hybrid automaton $\mathcal{H}$ satisfying $\forall i \in \{1, \ldots, r-1\},\ D_{q_i} \cap D_{q_{i+1}} \neq \emptyset$ and $\mathcal{T}_f \subset D_{q_0}$. As in section 3.1.2, we introduce the set Ω_γ associated to the path γ and defined by: $\Omega_\gamma = \bigcup_{i=0}^{r} D_{q_i}$. According to proposition 10, the hybrid controllable domain in Ω_γ can be under-approached by the union of local controllable domains successively defined in the cells D_{q_i} of the path γ:

$$\begin{aligned} \underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma) = \mathcal{C}_{\mathcal{H}}(\mathcal{T}_0, D_{q_0}) \cup \mathcal{C}_{\mathcal{H}}(\mathcal{T}_1, D_{q_1}) \cup \ldots \\ \cup \mathcal{C}_{\mathcal{H}}(\mathcal{T}_r, D_{q_r}) \subset \mathcal{C}_{\mathcal{H}}(\Omega_\gamma) \end{aligned} \tag{11}$$

where $\mathcal{T}_{q_i}$ denotes the local target in mode q_i. Similarly, we define an approximation $\Lambda(\Omega_\gamma)$ of $\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma)$ by successive local convex approximations:

$$\Lambda(\Omega_\gamma) = \Lambda(\tau_0, D_{q_0}) \cup \Lambda(\tau_1, D_{q_1}) \cup \cdots \cup \Lambda(\tau_r, D_{q_r}) \tag{12}$$

where $\Lambda(\tau_i, D_{q_i})$ denotes the convex approximation computed by the algorithm 1 and $(\tau_i)_{i=0\ldots r}$ is the sequence of local targets defined along the path γ by recurrence as follows:

$$\begin{cases} \tau_0 = \mathcal{T}_f \\ \tau_{i+1} = \Lambda(\tau_i, D_{q_i}) \cap G_{(q_i, q_{i+1})},\ i = 0, \ldots, r \end{cases} \tag{13}$$

We easily check that the targets τ_i, $i = 0, \ldots, r$ are polytopes (i.e. bounded and convex polyhedra). Furthermore, each local target τ_i is a convex approximation of $\mathcal{T}_{q_i}$, so that $\Lambda(\tau_i, D_{q_i})$ is actually a convex approximation of the controllable domain $\mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_i}, D_{q_i})$:

Proposition 12 *For all $i = 0, \ldots, r$, the local targets τ_i defined by the recursive relation (13), are respective convex approximations of targets $\mathcal{T}_{q_i}$ and $\Lambda(\tau_i, D_{q_i})$ is a convex approximation of order $\sqrt{n}h$ of the controllable domain $\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$ in mode q_i.*
Moreover, if the sets $\mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_i}, D_{q_i})$, $i = 0, \ldots, r$, are convex then: $\forall i = 0, \ldots, r,\ \tau_i \subset \mathcal{T}_{q_i}$ and $\Lambda(\tau_i, D_{q_i}) \subset \mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_i}, D_{q_i})$.

Now, we want to evaluate the approximation error, namely the Hausdorff distance $d_H(\mathcal{C}_{\mathcal{H}}(\Omega_\gamma), \Lambda(\Omega_\gamma))$ between the controllable domain $\mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$ and its piecewise convex approximation $\Lambda(\Omega_\gamma)$. Since $\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma)$ is an under-approximation of $\mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$ and according to the decompositions (11) and (12), we deduce:

$$\begin{aligned} d_H(\mathcal{C}_{\mathcal{H}}(\Omega_\gamma), \Lambda(\Omega_\gamma)) &\leq d_H\left(\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma), \Lambda(\Omega_\gamma)\right) \\ &\leq \max_{i=0\ldots r} d_H(\mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_i}, D_{q_i}), \Lambda(\tau_i, D_{q_i})) \end{aligned}$$

Moreover, for any cell D_{q_i}, $i = 0, \ldots, r$, the distance of the local controllable domain $\mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_i}, D_{q_i})$ to its convex approximation $\Lambda(\tau_i, D_{q_i})$ is bounded by the

size of the mesh $\mathcal{D}$ of the state space. Hence, according to 5, we have: $\forall i = 1, \ldots, r,\ d_H(\mathcal{C}_{\mathcal{H}}(\mathcal{T}_{q_i}, D_{q_i}), \Lambda(\tau_i, D_{q_i})) \leq \sqrt{n}h$ and we can conclude by:

$$d_H(\mathcal{C}_{\mathcal{H}}(\Omega_\gamma), \Lambda(\Omega_\gamma)) \leq \sqrt{n}h.$$

3.3 Experience on the non linear spring

In this section, we consider the small example of the nonlinear spring presented in [44, paragraph 7.3.1]:

$$\begin{cases} \dot{x}(t) &= y(t) \\ \dot{y}(t) &= -x(t) - 2x(t)^3 + u(t) \end{cases} \tag{14}$$

where the control constraints are: $\forall t \geq 0,\ |u(t)| \leq 1$. In this section, we want to apply the previously described algorithms to compute a controllable approximation of the set of initial positions $(x_0, y_0 = \dot{x}_0)$, from which the spring can be steered to its equilibrium position $(0, 0)$.
In this example, we perform the convex approximation algorithm on three successive adjacent cells. Figure 11 shows the three successive hybrid controllable approximations for the discretization step $h = 1$. Figure3.3 shows the result of our algorithm at three different discretization level $h \in \{2, 1, \frac{1}{2}\}$. The outside blue curve delimits the real nonlinear controllable domain over the three considered cells: as we can observe, the hybrid approximations tend to the exact nonlinear domain when h decreases.

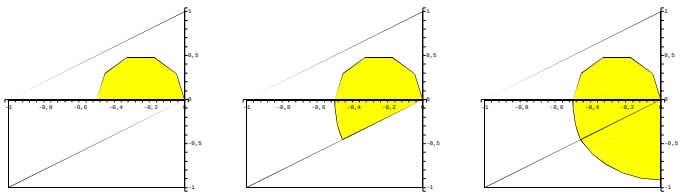


Figure 11: Successive controllable approximations inside three adjacent cells for $h = 1$.

3.4 Orbital transfer

We now consider a satellite in orbit around the Earth. Our control problem is the minimum time transfer of this satellite from an initial orbit to a geostationary one.

Let m be the mass of the satellite and u its thrust. The satellite dynamic is

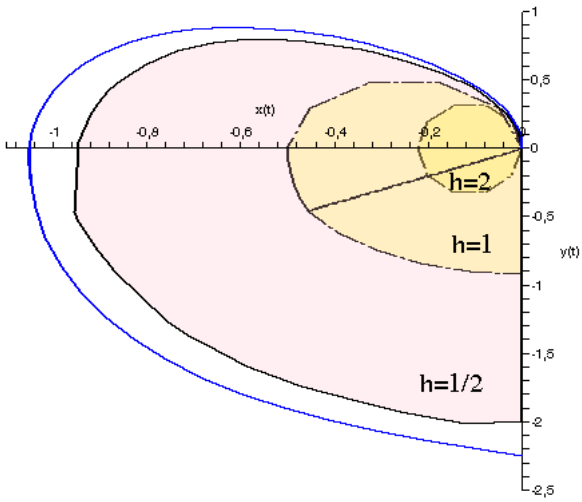


Figure 12: Convex controllable approximations at scales $h = 2$, $h = 1$ and $h = 1/2$. Our approximations tend to the exact controllable domain delimited by the blue curve.

described by the equations:

$$\ddot{r} = -\mu \frac{r}{\|r\|^3} + \frac{u}{m}, \tag{15}$$

$$\dot{m} = -\frac{\|u\|}{v_e}, \ v_e \in \mathbb{R}^+. \tag{16}$$

where μ denotes the gravitation constant. The control u is assumed bounded: $\|u\| \leq F_{max}$. The numerical values of the involved physical constants are given in Table 1.

Parameter	**Value**
μ	$5165,8620912$ Mm^3.h^{-2}
m_0	1500 kg
F_{max}	3 N

Table 1: Physical constants for the orbital transfer problem

The mathematical model of the satellite is a variable mass three dimensional model. Simplified models have been proposed such as coplanar transfer [14] and/or constant mass models [12]. For each model, geometrical studies have been proposed to analyze the controllability properties and the structure of extremals. For the sake of simplicity, we restrict our analysis to coplanar transfers at constant mass. The satellite dynamic is described by equations expressed in radial-orthoradial coordinates (P, e_x, e_y, L) as follows:

$$\begin{bmatrix} \dot{P} \\ \dot{e}_x \\ \dot{e}_y \\ \dot{L} \end{bmatrix} = \sqrt{\frac{\mu}{P}} \begin{bmatrix} 0 \\ 0 \\ 0 \\ W^2/P \end{bmatrix} + \sqrt{\frac{P}{\mu}} \begin{bmatrix} 0 \\ sin\ L \\ -cos\ L \\ 0 \end{bmatrix} u_1 + \sqrt{\frac{P}{\mu}} \begin{bmatrix} 2P/W \\ cos\ L + (e_x + cos\ L)/W \\ sin\ L + (e_y + sin\ L)/W \\ 0 \end{bmatrix} u_2 \tag{17}$$

where: $W = 1 + e_x cos\ L + e_y sin\ L$. Variables (P, e) are respectively the parameter and the eccentricity vector of the osculating ellipse and L the longitude. The variable change between the Cartesian and radial-orthoradial frames is defined by:

$$\begin{cases} r_1 = \frac{P}{W} cos\ L \\ r_2 = \frac{P}{W} sin\ L \end{cases} \qquad \begin{cases} v_1 = -\sqrt{\frac{\mu}{P}}(e_y + sin\ L) \\ v_2 = \sqrt{\frac{\mu}{P}}(e_x + cos\ L) \end{cases} \tag{18}$$

where $v = \dot{r}$ denotes the satellite velocity. From now on, the control constraints are expressed as:

$$|u_1| + |u_2| < F_{max}.$$

and the control domain $\mathbb{U}_m$ could be defined by: $\mathbb{U}_m = Conv\left([-F_{max}, 0], [0, -F_{max}], [F_{max}, 0], [0, F_{max}]\right).$

The control problem is to minimize the time transfer of the satellite from a given initial orbit to a given final one, the position of the satellite on this last orbit being free. The related initial and final conditions are summed up on Table 2.

Initial conditions	**Final conditions**
$P^0 = 11.625$ Mm	$P^f = 42.165$ Mm
$e_x^0 = 0.75$	$e_x^f = 0$
$e_y^0 = 0$	$e_y^f = 0$
$L^0 = \pi$ rad	L^f free

Table 2: Initial and final condition for the orbital transfer problem

Let us now consider an initial point $X_0 = [11.625, 0.75, 0, \pi]$ expressed in orbital coordinates. At the rougher scale $h = 6$, our algorithm 1 computes 451 controllable points in 4 minutes, building a four-dimensional approximation made of 104 vertices. By projection, we could obtain for example:

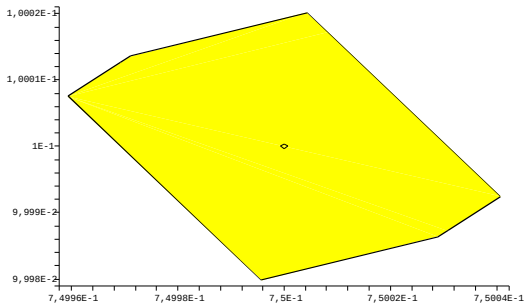
- the attainable set of the eccentricity e from the initial point $e_0 = (0.75, 0)$ (see figure 13-(a)).
- the three dimensional attainable set of parameters (P, e_y, L) (see figure 13-(b)).

Using variable change (18), these numerical results could be translated in Cartesian coordinates (r_1, r_2) and give an approximation of the attainable set from X_0 into the Earth Cartesian frame (see figure 14).

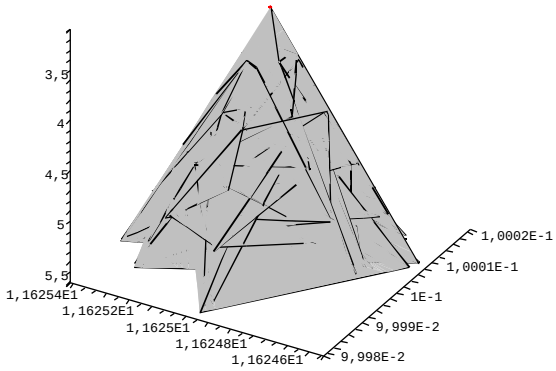
Observe that since the Earth radius $r_{\mathcal{E}}$is about 6 kilometers, collision with the Earth is possible. To avoid this configuration, we could add a constraint like $\|r\| > r_{\mathcal{E}}$: we would then have to redefine the involved cells of the state mesh and this way, to restrict the controllable domain.

4 Approaching nonlinear optimal solutions by optimal solutions of the hybrid control problem

In this section we focus on the resolution of the nonlinear optimal control problem via the hybrid approach. Let $\mathbb{U}_m$ be a convex and compact polytope in $\mathbb{R}^m$ such that: $0 \in \mathbb{U}_m$: in section 2.3, we shaw that any nonlinear control system



(a)



(b)

Figure 13: Attainable sets for (a) the eccentricity $e = (e_x, e_y)$ (b) parameters (P, e_y, L), at the rougher scale ($h = 6$).

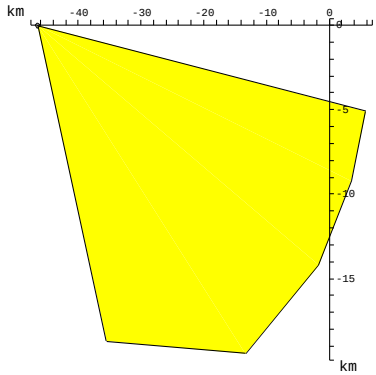


Figure 14: Approximation of the satellite attainable domain at scale $h = 6$ in the Cartesian frame (r_1, r_2)

like (1) can be approached by a piecewise affine hybrid model. This enables us to define two optimal control problems related to each control system:

$(\mathcal{P}_{NL})$ *Minimize the cost function* $J(X_0, u) = \int_0^{t_f} l(X(t), u(t))dt$ *with respect to the control* u *under the nonlinear dynamic:*

$$\begin{cases} \dot{X}(t) = f(X(t), u(t)) \\ X(0) = X_0 \end{cases} \qquad X(t_f) \in \mathcal{T}_f$$

and the constraints: $\forall t \in [0, t_f], u(t) \in \mathbb{U}_m$. *The final time* t_f *is unspecified.*

$(\mathcal{P}_{\mathcal{H}})$ *Minimize the cost function* $J(X_0, u) = \int_0^{t_f} l(X(t), u(t))dt$ *with respect to the control* u *under the piecewise affine hybrid dynamic:*

$$\begin{cases} \dot{X}(t) = f_h(X(t), u(t)) \\ X(0) = X_0 \end{cases} \qquad X(t_f) \in \mathcal{T}_f$$

and the constraints: $\forall t \in [0, t_f], u(t) \in \mathbb{U}_m$. *The final time* t_f *is unspecified.*

As for the controllability problem, a key point of our hybrid approach is first to prove that solutions of the hybrid problem $(\mathcal{P}_{\mathcal{H}})$ are satisfactory approximations of those of $(\mathcal{P}_{NL})$. We then address the problem of solving the hybrid optimal control problem $(\mathcal{P}_{\mathcal{H}})$: a powerful tool for optimal trajectories analysis is the (PMP) Pontryagin maximum principle [37]. First, in section 4.1, we state a suitable maximum principle for our hybrid optimal control problem $(\mathcal{P}_{\mathcal{H}})$ and then deduce in section 4.2 the general structure of hybrid optimal trajectories.

4.1 Hybrid maximum principle

In the classical context of smooth systems, the Pontryagin maximum principle provides first-order necessary optimality conditions and a pseudo-Hamiltonian formulation of the considered optimal control problem. It has been first stated by Pontryagin and al. in [37]. Stronger versions have been stated especially in the context on constraints on the control. References are numerous ; one can refer to [37, 33, 18, 1] for a proof of the maximum principle or to [11, paragraphe 1.5] and [44] for a detailed study of its applications.

The main difficulty of establishing a hybrid maximum principle is to manage discontinuities generated by the discrete component of the hybrid system. Today several formulations has been proposed depending on the structure and the regularity of the studied hybrid systems [42, 45]. Especially P. Riedinger, C. Iung and F. Kratz state and prove a hybrid maximum principle for a general class of hybrid systems in [38, theorem 2.3]. Next our hybrid maximum principle is inspired from their theorem.

Theorem 1 (Hybrid maximum principle) *We introduce the Hamiltonian H_h related to the hybrid model $\mathcal{H}$ defined in section 2.3:*

$$H_h(X, u, \lambda) = l(X, u) + \lambda^\top f_h(X, u).$$

If the control u is optimal on the time interval $[0, t_f]$, then there exists a piecewise absolute continuous application $\lambda : [0, t_f] \to \mathbb{R}^n$ such that:

- *For almost all $t \in [0, t_f]$, H reaches its minimum with regards to the control at point $u(t)$ i.e.:*

$$H_h(X(t), u(t), \lambda(t)) = \min_{v \in \mathbb{U}_m} H_h(X(t), v, \lambda(t)).$$

- $\dot{X}(t) = \dfrac{\partial H_h}{\partial \lambda}(X(t), u(t), \lambda(t))$
- $\dot{\lambda}(t)^\top = -\dfrac{\partial H_h}{\partial X}(X(t), u(t), \lambda(t))$ *(Euler-Lagrange equation)*
- $H_h(X(t), u(t), \lambda(t)) = 0$ *along the optimal trajectory.*
- *At each transition instant t_i between two modes q_i and q_{i+1}, we have the following transversality conditions:*

$$\begin{cases} \lambda(t_i^+) = \lambda(t_i^-) \\ \\ H_h(X(t_i^+), u(t_i^+), \lambda(t_i^+)) = H_h(X(t_i^-), u(t_i^-), \lambda(t_i^-)) \end{cases}$$

We first recall that by definition, the state variable is continuous at each transition time t_i: $X(t_i^+) = X(t_i^-)$, which together with the autonomous feature

of the hybrid system, implies the continuity of the hybrid Hamiltonian at each transition time.

Then transitions between discrete modes of our hybrid automaton are non constrained: at each transition time t_i between (at least) two modes q_i and q_{i+1}, our hybrid system could either switch to mode q_{i+1}, or stay in mode q_i, depending on the value of the field vector f_h at $X(t_i)$. This implies the continuity of the adjoint vector λ at each transition time.

4.2 Characterization of hybrid optimal trajectories

4.2.1 Necessary optimality conditions in mode q

Let $X_0 \in \mathbb{R}^n$ be a controllable point by our hybrid system $\mathcal{H}$ and q the corresponding discrete mode satisfying: $X_0 \in D_q$. We assume that the initial time $t = 0$ is not a transition time of the system, which means that X_0 does not belong to any guard of $\mathcal{H}$ (in other words $X_0 \in int\ D_q$). This assumption guarantees that any (state) trajectory steered from X_0 remains inside the cell D_q on a non empty time interval.

Our point is now the local resolution (inside D_q) of the optimization problem stated by the hybrid maximum principle. According to this principle, any state and control trajectory candidate to optimality, also called extremal, in mode q has to fulfill three conditions: to minimize the hybrid Hamiltonian, to enable the definition of an appropriate adjoint state vector λ and to cancel the Hamiltonian along this trajectory.

Minimization of the hybrid Hamiltonian H_h in mode q In paragraph 2.3.2, we defined the set of local control constraints at a given position X in one mode q by introducing the sets:

$$\forall X \in D_q, \forall q' \in \mathcal{K}(q), U_{q'}(X) = \{u \in \mathbb{U}_m; (X, u) \in \Delta_{q'}\}.$$

where $\mathcal{K}(q)$ denotes the set of indices of state-control cells of the mesh Δ whose projection is exactly D_q. Therefore $(U_{q'}(X))_{q' \in \mathcal{K}(q)}$ forms a partition of the control domain $\mathbb{U}_m$ i.e.:

$$\forall X \in D_q, \mathbb{U}_m = \bigcup_{q' \in \mathcal{K}(q)} U_{q'}(X) \tag{19}$$

Now, by definition of the hybrid system $\mathcal{H}$, between two successive transition instants t_i and t_{i+1}, the state trajectory $X(t)$ remains in cell D_q, the state and control trajectory $(X(t), v)$ belongs to $\Delta_{q'}$ for all $v \in U_{q'}(X(t))$ and the hybrid dynamic at any point $(X(t), v)$ for $v \in U_{q'}(X(t))$ is affine: $\forall t \in [t_i, t_{i+1}], \forall v \in U_{q'}(X(t)), f_h(X(t), u(t)) = A_{q'}X(t) + B_{q'}v + c_{q'}$. Together with (19), we then

have:

$$\begin{aligned}
&\inf_{v\in\mathbb{U}_m} H(X(t),v,\lambda(t)) \\
&\quad = \min_{q'\in\mathcal{K}(q)} \left(\min_{v\in U_{q'}(X(t))} H(X(t),v,\lambda(t)) \right) \\
&\quad = \min_{q'\in\mathcal{K}(q)} \left(\min_{v\in U_{q'}(X(t))} l(X(t)) + \lambda(t)^\top f_h(X(t),v) \right) \\
&\quad = \min_{q'\in\mathcal{K}(q)} \left(\min_{v\in U_{q'}(X(t))} l(X(t)) + \lambda(t)^\top A_{q'} X(t) \right. \\
&\qquad\qquad \left. + \lambda(t)^\top c_{q'} + \lambda(t)^\top B_{q'} v \right)
\end{aligned}$$

Let us consider the inner minimization problem for a given $q' \in \mathcal{K}(q)$. The objective is affine in the optimization variable v. Moreover, according to the proposition 7, the set $U_{q'}(X(t))$ is a m-simplex whose vertices are denoted by $\sigma_i(t) = F_i X(t) + g_i$, $i = 1,\ldots,m+1$. The induced constraints on the control v are therefore affine and the inner minimization problem actually is a linear program whose solution is well-known:

$$\begin{gathered}
v^\star(t) = F_i X(t) + g_i \\
\text{if } \ \forall j \neq i, \lambda(t)^\top B_{q'} \left[(F_i - F_j) X(t) + g_i - g_j \right] < 0.
\end{gathered}$$

We observe that the switching conditions explicitly depend both on the state X and the adjoint state λ.

Remark 3 (Switching curves and singular controls) *From now on, we assume that optimal controls are non singular or equivalently, that zeros of each switching function*

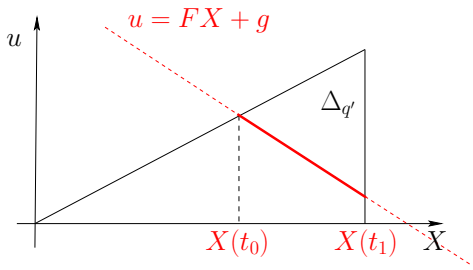
$$S_{i,j} : t \mapsto \lambda(t)^\top B_{q'} \left[(F_i - F_j) X(t) + g_i - g_j \right]$$

where $i,j = 1,\ldots,m, i \neq j$, are isolated.

Consequently our minimization problem in mode q could be replaced by the simultaneous solving of a finite number (exactly *card* $\mathcal{K}(q)$) of linear programs: the searched optimal controls are feedback (more precisely they are piecewise affine in the state variable X). Notice that the choice of each piece $FX + g$ also depends on the adjoint state variable λ.

The next step is to check that such controls fulfill the missing optimality necessary conditions given by the hybrid maximum principle. Let us now introduce a given affine feedback control $u = FX + g$ and the resulting trajectory in mode q. As illustrated on figure 15, we now focus on a time interval $[t_0, t_1]$ on which the considered trajectory $(\widetilde{X}, F\widetilde{X} + g)$ remains in a state-control cell $\Delta_{q'}$ for some $q' \in \mathcal{K}(q)$:

$$\forall t \in [t_0, t_1], (\widetilde{X}(t), F\widetilde{X}(t) + g) \in \Delta_{q'}.$$



- $u = FX+g$ a given affine feedback control.
- $\widetilde{X}$ the resulting trajectory solution of:

$$\dot{X}(t) = f_h(X(t), FX(t) + g).$$

Figure 15: Definition of an affine feedback control $u = FX + g$ in mode q

It follows that the hybrid dynamic is affine along the trajectory $(\widetilde{X}, F\widetilde{X} + g) \in \Delta_{q'}$ i.e. for all $t \in [t_0, t_1]$:

$$f_h(\widetilde{X}(t), F\widetilde{X}(t) + g) = (A_{q'} + B_{q'}F)\,\widetilde{X}(t) + B_{q'}g + c_{q'}.$$

Definition of the adjoint vector λ in mode q The adjoint parameter is then simply defined as solution of the following first order differential equation:

$$\dot{\lambda}(t)^\top = -\frac{\partial l}{\partial X}(\widetilde{X}(t)) - \lambda(t)^\top(A + BF) \tag{20}$$

for all $t \in]t_0, t_1[$. According to our regularity assumptions, the cost function l is continuously differentiable on $\mathbb{R}^n \times \mathbb{U}_m$, so that $t \mapsto \frac{\partial l}{\partial X}(\widetilde{X}(t))$ is continuous. This implies the expected absolute continuity of the solution λ. Notice that no initial condition on λ is required.

Hybrid Hamiltonian equal to 0 along the optimal trajectory Lastly we have to check that the hybrid Hamiltonian is equal to 0 along the considered trajectory $(\widetilde{X}, F\widetilde{X} + g)$. We write for all $t \in [t_0, t_1]$:

$$\begin{aligned}
&H(\widetilde{X}(t), F\widetilde{X}(t) + g, \lambda(t)) \\
&\qquad = l(\widetilde{X}(t)) + \lambda(t)^\top \left[(A + BF)\widetilde{X}(t) + Bg + c\right] \\
&\qquad = l(\widetilde{X}(t)) + \lambda(t)^\top \widetilde{X}(t).
\end{aligned}$$

Using $\ddot{X}(t) = (A + BF)\dot{X}(t)$ and relation (20), we prove

$$\begin{aligned}
&\frac{d}{dt}H(X(t), u(t), \lambda(t)) \\
&\qquad = \frac{\partial l}{\partial X}(X(t))\dot{X}(t) + \dot{\lambda}(t)^\top \dot{X}(t) + \lambda(t)^\top \ddot{X}(t) \\
&\qquad = \frac{\partial l}{\partial X}(X(t))\dot{X}(t) + \lambda(t)^\top (A + BF)\dot{X}(t) \\
&\qquad\qquad - \left(\frac{\partial l}{\partial X}(X(t)) + \lambda(t)^\top (A + BF)\right)\dot{X}(t) \\
&\qquad = 0
\end{aligned}$$

for all $t \in [t_0, t_1]$. That means that the Hamiltonian is constant along the considered trajectory $(\widetilde{X}, F\widetilde{X}+g, \lambda)$. Thus to fulfill the last optimality necessary condition required by our hybrid maximum principle, we just have to cancel the hybrid Hamiltonian at the initial time i.e.:

$$l(X(t_0)) + \lambda(t_0)^\top \left[(A + BF)X(t_0) + Bg + c\right] = 0. \tag{21}$$

This gives an explicit constraint on the adjoint state variable λ at the initial time t_0.

In conclusion that feedback affine controls are good local candidates to optimality. Next we propose a strategy to globally reconstruct the extremals of our hybrid optimal control problem $(\mathcal{P}_\mathcal{H})$.

4.2.2 Global structure of hybrid extremals

Let (X, u, λ) be an extremal of the hybrid optimal control problem $(\mathcal{P}_\mathcal{H})$ for the initial condition $X(0) = X_0$. By definition 2 of an hybrid trajectory, there exists a sequence of time intervals $([t_i, t_{i+1}])_{i=0\dots r}$ and a sequence of discrete modes $(q_i)_{i=0\dots r+1}$ such that:

$$\forall t \in [t_i, t_{i+1}], X(t) \in D_{q_i} \tag{22}$$

The transversality conditions stated by the hybrid maximum principle together with the definition of our hybrid model, guarantee the continuity of both the state X and the adjoint state λ at each switching time t_i between two discrete modes. There just remains to determine the structure of trajectories inside each mode q_i.

Structure in mode q_i We focus on one discrete mode q_i of our hybrid automaton $\mathcal{H}$ and introduce the set $\Sigma_{q_i}(X)$ of the vertices of the control domain $U_{q'}(X)$ induced at position X in mode q_i:

$$\begin{aligned}\Sigma_{q_i}(X) \quad = \quad & \{FX + g \in \mathbb{U}_m / \exists q' \in \mathcal{K}(q_i), \\ & (FX + g) \text{ is a vertex of } U_{q'}(X)\} \subset \mathbb{U}_m.\end{aligned}$$

According to results presented in paragraph 4.2.1, the optimal control u is locally the solution of a linear program and can be expressed as an explicit piecewise affine function of the state X, each piece's choice implicitly depending on the adjoint vector λ: therefore there exists a subdivision $(\widetilde{t}_j)_{j=0\dots N_i}$ of the time interval $]t_i, t_{i+1}[$ such that for all $j \in \{0, \dots, N_i - 1\}$:

$$\begin{gathered}\exists (F_j, g_j) \in \mathbb{M}_{m,n} \times \mathbb{R}^m, \forall t \in [\widetilde{t}_j, \widetilde{t}_{j+1}[, \\ u(t) = F_j X(t) + g_j \in \Sigma_{q_i}(X(t)).\end{gathered} \tag{23}$$

Hence: $(X, F_j X + g_j) \in \Delta_{q'}$ on $[\widetilde{t}_j, \widetilde{t}_{j+1}[$. In other words, as illustrated on figure 16, the optimal trajectory (X, u) evolves along the edges of the state and control mesh Δ, continuously in X and not in u.

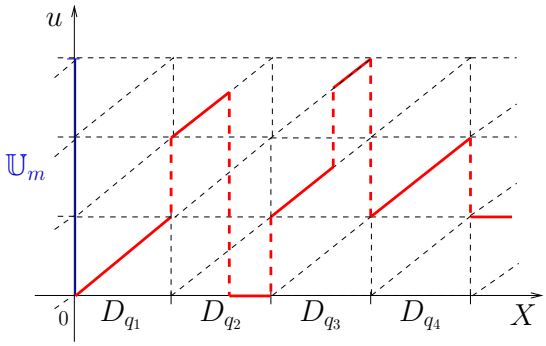


Figure 16: Hybrid extremals evolve along edges of the state and control mesh Δ

The main issue is so to efficiently compute both the optimal control and the adjoint state variable λ at any given position X. Indeed Euler-Lagrange equations determining λ depend on the control choice, which in turns depends on λ. Therefore we need to determine $u = FX+g$ and λ satisfying the optimality conditions without having to compute any trajectories.

Switching conditions and optimal control computation Let us consider our hybrid optimal control problem at a given time $\widetilde{t}_j$. The state $\widetilde{X}_j = X(\widetilde{t}_j)$ and the adjoint state $\widetilde{\lambda}_j = \lambda(\widetilde{t}_j)$ are supposed to be known. According to (23), we want to determine control candidate to optimality satisfying $u = FX + g$ in the set $\Sigma_{q_i}(X)$. We also need to compute the first time $\widetilde{t}_{j+1} > \widetilde{t}_j$ at which the optimality conditions are not fulfilled any more.

We assume that the control $u = FX+g$ satisfies all the optimality conditions required by the hybrid maximum principle at time $\widetilde{t}_j$:

$$\begin{aligned} H_h(\widetilde{X}_j, F\widetilde{X}_j + g, \widetilde{\lambda}_j) &= \min_{u \in \mathbb{U}_m} H_h(\widetilde{X}_j, u, \widetilde{\lambda}_j) \\ &= \min_{\substack{F'\widetilde{X}_j + g' \\ \in \Sigma_{q_i}(\widetilde{X}_j)}} H_h(\widetilde{X}_j, F'\widetilde{X}_j + g', \widetilde{\lambda}_j) \\ H_h(\widetilde{X}_j, F\widetilde{X}_j + g, \widetilde{\lambda}_j) &= 0 \end{aligned}$$

This induces that switching functions could be expressed via the hybrid Hamiltonian this way:

$$\begin{aligned} S_{FX+g,F'X+g'} &= \lambda^\top \left[f_h(X, FX+g) - f_h(X, F'X+g') \right] \\ &= -l(X) - \lambda^\top f_h(X, F'X+g') \\ &= -H_h(X, F'X+g', \lambda) \leq 0 \end{aligned}$$

Therefore switching conditions are obtained by studying the zeros of the application: $u \mapsto H_h(\widetilde{X}_j, u, \widetilde{\lambda}_j)$ over the finite set $\Sigma_{q_i}(\widetilde{X}_j)$. We thus compute:

$H_h(\widetilde{X}_j, F\widetilde{X}_j+g, \widetilde{\lambda}_j)$ for all $F\widetilde{X}_j+g \in \Sigma_{q_i}(\widetilde{X}_j)$. When the latter equals zero, the corresponding control is a good candidate to optimality. The main advantage of this approach is that we only need $card\ \Sigma_{q_i}(\widetilde{X}_j)$, i.e. at most $m+card\ \mathcal{K}(q_i)$, function evaluations of $H_h(\widetilde{X}_j, \cdot, \widetilde{\lambda}_j)$.

We then introduce:

$$\widetilde{t}_{j+1} = \min\Big(t_{i+1}; \inf\{t > t_j; H_h(\widetilde{X}(t), F\widetilde{X}(t)+g, \widetilde{\lambda}(t)) > 0\}\Big)$$

where $(\widetilde{X}, \widetilde{\lambda})$ denotes the state and adjoint state trajectory steered from $(\widetilde{X}_j, \widetilde{\lambda}_j)$ at time $\widetilde{t}_j$ according to the control $u = FX + g$. In other words, $\widetilde{t}_{j+1}$ denotes either the exit time, namely t_{i+1}, from the current mode q_i or the first instant at which optimality conditions are not satisfied anymore. At time $\widetilde{t}_{j+1}$, we have new initial conditions $(\widetilde{X}_{j+1}, \widetilde{\lambda}_{j+1}) = (\widetilde{X}(\widetilde{t}_{j+1}), \widetilde{\lambda}(\widetilde{t}_{j+1}))$.

4.3 Simulation algorithm of hybrid extremals

For a given initial condition of the adjoint state, we propose an algorithm to compute extremals of the hybrid optimal control problem i.e. trajectories satisfying the necessary optimality conditions of the hybrid maximum principle. Algorithm 2 details this computation of a hybrid extremal in a given mode q_i.

The global simulation algorithm of hybrid extremals relies on the computation of discrete modes reached by the trajectory satisfying the necessary optimality conditions and on the transversality conditions that guarantee the continuity of the adjoint state variable λ. The principle is as follows: starting from a given initial condition (t_0, X_0, λ_0) in mode q_0 and using algorithm 2, we compute the trajectory inside D_{q_0} satisfying the hybrid maximum principle and its exit time from the cell D_{q_0} towards an adjacent cell D_{q_1}. By continuity of the state and adjoint state, the exit conditions give new initial conditions (t_1, X_1, λ_1) in mode q_1 and we pursue the simulation the same way.

5 Conclusion

In this paper, we address the problem of hybrid optimal control of nonlinear dynamical systems. First we propose a hybrid approximation, by way of a hybrid automaton with piecewise affine dynamics, of complex systems. We therefore define a piecewise affine optimal control problem and then focus on two main problems in control theory: the controllability question and the computation of optimal solutions.

For the controllability, we have given a convex approximation of the controllable set which is at a guaranteed distance of the actual controllable set. Furthermore, when the controllable set is convex, this approximation is an under-approximation and thus guarantees controllability. We then proposed and implemented an algorithm to compute this approximation and its efficiency

Algorithm 2 Extremal computation in a given discrete mode q_i

Require: $t_i \geq 0$ initial time in mode q_i at the position $X_i \in D_{q_i}$ for the given initial condition: $\lambda(t_i) = \lambda_i$.

1: **Initialization.** $j = 0$; $\widetilde{t}_0 = t_i$; $\widetilde{X}_0 = X_i$; $\widetilde{\lambda}_0 = \lambda_i$;

2: **Control computation.** Find the control $F\widetilde{X}_j + g \in \Sigma_{q_i}(\widetilde{X}_j)$ satisfying the necessary optimality conditions at time $\widetilde{t}_j$ i.e.:

$$\begin{aligned} H_h(\widetilde{X}_j, F\widetilde{X}_j + g, \widetilde{\lambda}_j) &= 0 \\ H_h(\widetilde{X}_j, F'\widetilde{X}_j + g', \widetilde{\lambda}_j) &> 0 \end{aligned}$$

for all $F'\widetilde{X}_j + g' \in \Sigma_{q_i}(\widetilde{X}_j) \backslash \{F\widetilde{X}_j + g\}$.

3: **State and adjoint state trajectories.** Compute the state $X_j = X[\widetilde{X}_j, FX + g]$ and adjoint state $\lambda_j = \lambda[\widetilde{\lambda}_j, FX + g]$ trajectories respectively steered from $\widetilde{X}_j$ and $\widetilde{\lambda}_j$ according to the control $u(X) = FX + g$:

$$\dot{X}(t) = f_h(X(t), FX(t) + g), \quad X(t_i) = \widetilde{X}_j$$

$$\begin{aligned} \dot{\lambda}(t)^\top = -\frac{\partial H_h}{\partial X}(&X[\widetilde{X}_j, FX + g](t), \\ &FX[\widetilde{X}_j, FX + g](t) + g, \lambda(t)) \\ \lambda(t_i) = \widetilde{\lambda}_j \end{aligned}$$

4: Compute the exit time t_{i+1} of $X[\widetilde{X}_j, FX + g]$ from the cell D_{q_i} and the next cell $D_{q_{i+1}}$ at time t_{i+1}.

5: **Switching time.** Introducing the Hamiltonian along the considered trajectory, compute

$$\begin{aligned} \widetilde{t}_{j+1} = \inf\{t > t_j; H_h(&X[\widetilde{X}_j, FX + g](t), \\ &FX[\widetilde{X}_j, FX + g](t) + g, \lambda[\widetilde{\lambda}_j, FX + g](t)) > 0\} \end{aligned}$$

6: **if** $\widetilde{t}_{j+1}$ exists and $\widetilde{t}_{j+1} < t_{i+1}$ **then**

7: Go back to step 2 with new initial conditions: $\widetilde{t}_{j+1}$, $\widetilde{X}_{j+1} = X[\widetilde{X}_j, FX + g](\widetilde{t}_{j+1})$ and $\widetilde{\lambda}_{j+1} = \lambda[\widetilde{\lambda}_j, FX + g](\widetilde{t}_{j+1})$ until $\widetilde{t}_{j+1} = t_{i+1}$ i.e. until the state leaves the cell D_{q_i}.

8: $j = j + 1$;

9: **end if**

and accuracy are shown on the examples of the nonlinear spring and of the change of orbit.

Then we addressed the question of optimality. By developing an hybrid maximum principle we are able to characterize the hybrid optimal trajectories and to prove that the optimal control have the form of affine feedbacks. Moreover, the principle gives us a way to compute the switching times between modes and thus an algorithm to follow the optimal trajectories.

Further improvements include mixing this approach with shooting methods in order to get a better initialization of the adjoint vector.

References

[1] A.A. Agrachev and Y.L. Sachkov. *Control Theory from the Geometric Viewpoint*, volume 87 of *Encyclopaedia of Mathematical Sciences*. Springer, 2004.

[2] R. Alur, T. Dang, and F. Ivancic. Reachability analysis of hybrid systems via predicate abstraction. In *Fifth International Workshop on Hybrid Systems: Computation and Control*, 2002.

[3] E. Asarin and T. Dang. Abstraction by projection and application to multi-affine systems. In R. Alur and G.J. Pappas, editors, *Hybrid Systems: Computation and Control*, volume 2993 of *LNCS*, pages 32–47. Springer, 2004.

[4] E. Asarin, T. Dang, and A. Girard. Reachability of non-linear systems using conservative approximations. In O. Maler and A. Pnueli, editors, *Hybrid Systems: Computation and Control*, volume 2623 of *LNCS*, pages 22–35. Springer, 2003.

[5] Baoticy and Borrelliz. Hybrid systems modeling and control. *European Journal of Control*, 9(2-3):177–189, 2003.

[6] C.B. Barber, D.P. Dobkin, and H.T. Huhdanpaa. The Quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, 1996.

[7] M. Bardi and I. Capuzzo-Dolcetta. *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, volume 17 of *Systems & Control: Foundations & Applications*. Birkauser, 1997.

[8] G. Barles. *Solutions de viscosité des équations de Hamilton-Jacobi*, volume 17 of *Mathématiques et Applications*. Springer-Verlag, 1994.

[9] R. Bellman. *Dynamic Programming*. Princeton University Press, 1957.

[10] D.P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 1 & 2. Athena Scientific, 1984.

[11] B. Bonnard and J.-B. Caillau. *Advanced topics in control systems theory*, volume 328 of *Lecture Notes in Control and Information Sciences*, chapter 1: Introduction to nonlinear optimal control, pages 1–60. Springer-Verlag, A. Loria, F. Lamnabhi and E. Panteley edition, 2006.

[12] B. Bonnard, J.-B. Caillau, and E. Trélat. Geometric optimal control of elliptic keplerian orbits. *Discrete and Continuous Dynamical Systems - Series B*, 4(5):929–956, 2005.

[13] M.S. Branicky, V.S. Borkar, and S.K. Mitter. A Unified Framework for Hybrid Control: Model and Optimal Control Theory. *IEEE Transactions on Automatic Control*, 43(1):31–45, 1998.

[14] J.-B. Caillau and J. Noailles. Coplanar control of a satellite around the earth. *ESAIM: Control, Optimisation and Calculus of Variations*, 6:239–258, 2001.

[15] I. Capuzzo-Dolcetta. *On a Discrete Approximation of the Hamilton-Jacobi Equation of Dynamic Programming*, volume 10 of *Applied Mathematics and Optimization*, pages 367–377. Springer, 1983.

[16] M.-J. Chien and E.S. Kuh. Solving nonlinear resistive networks using piecewise-linear analysis and simplicial subdivision. *IEEE Transactions on Circuits and Systems*, 24(6):305–317, 1977.

[17] Y.C. Cho, C.G. Cassandras, and D.L. Pepyne. Forward decomposition algorithms for optimal control of a class of hybrid systems. *Int. Journal of Robust and Nonlinear Control*, 11(5):497–513, 2001.

[18] F.H. Clarke. *Optimization and Nonsmooth Analysis*. SIAM Classics in Applied Mathematics, 1990.

[19] M.G. Crandall and P.-L. Lions. Viscosity solutions of hamilton-jacobi equations. *Transactions of the American Mathematical Society*, 277(1):1–42, 1983.

[20] J. Della Dora, A. Maignan, M. Mirica-Ruse, and S. Yovine. Hybrid computation. In *Proceedings of the 2001 International Symposium on Symbolic and Algebraic Computation*. Bernard Mourrain editor, ACM Press, July 2001.

[21] J.-G. Dumas and A. Rondepierre. Modeling the electrical activity of a neuron by a continuous and piecewise affine hybrid system. In *Hybrid Systems: Computation and Control*, volume 2623 of *LNCS*, pages 156–171. Springer, 2003.

[22] J.-G. Dumas and A. Rondepierre. Algorithms for symbolic/numeric control of affine dynamical systems. In Manuel Kauers, editor, *Proceedings of the 2005 International Symposium on Symbolic and Algebraic Computation*, pages 277–284. ACM Press, New York, 2005.

[23] R. Fierro, A. K. Das, V.Kumar, and J. P. Ostrowski. Hybrid control of formations of robots. In *IEEE Proceedings of the International Conference on Robotics and Automation*, 2001.

[24] H. Freudenthal. Simplizialzerlegungen von beschraeukter flachkeit. *Annals of Math. in Science and Engin.*, 43:580–582, 1942.

[25] R.V. Gamkrelidze. Discovery of the maximum principle. *Journal of Dynamical and Control Systems*, 5(4):437–451, 1999.

[26] A. Girard. Approximate solutions of odes using piecewise linear vector fields. In *Proceedings of the 2002 Computer Algebra in Scientific Computing*, pages 107–120. Springer Verlag, September 2002.

[27] A. Girard. *Analyse algorithmique des systèmes hybrides*. PhD thesis, Institut National Polytechnique de Grenoble, 2004.

[28] L.C.G.J.M. Habets, P.J. Collins, and J.H. van Schuppen. Reachability and control synthesis for piecewise-affine hybrid systems on simplices. *IEEE Transactions on Automatic Control*, 51(6):938–948, June 2006.

[29] Sven Hedlund and Anders Rantzer. Optimal control of hybrid systems. In *Proceedings 38th IEEE Conference on Decision and Control*, August 24 1999.

[30] V. Jurdjevic. *Geometric Control Theory*. Cambrigde University Press, 1997.

[31] R.E. Kalman. Mathematical description of linear dynamical systems. *SIAM Journal on Control*, 1:152–192, 1963.

[32] R.E. Kalman, P.A. Falb, and M. Arbib. *Topics in Mathematical System Theory*. McGraw-Hill, 1969.

[33] M.M. Lee and L. Markus. *Foundations of Optimal Control Theory*. Whiley, New York, 1967.

[34] C. Lobry. Contrôlabilité des systèmes non linéaires. *SIAM Journal of Control*, 8:573–605, 1970.

[35] Christopher E. Long, Pradeep K. Polisetty, and Edward P. Gatzke. Deterministic global optimization for nonlinear model predictive control of hybrid dynamic systems. *International Journal of Robust and Nonlinear Control*, 2006.

[36] H.J. Pesch. A practical guide to the solutions of real-life optimal control problems. *Parametric Optimization. Control Cybernet*, 23:7–60, 1994.

[37] L. Pontriaguine, V. Boltiansky, R. Gamkrelidze, and E. Michtchenko. *The mathematical theory of optimal processes*. Editions de Moscou, 1962.

[38] P. Riedinger, C. Iung, and F. Kratz. An optimal control approach for hybrid systems. *European Journal of Control*, 9:449–458, 2003.

[39] A. Rondepierre. Piecewise affine systems controllability and hybrid optimal control. In *Proceedings of the 2005 International Conference on Informatics in Control, Automation and Robotics*, 2005.

[40] A. Rondepierre. *Algorithmes hybrides pour le contrôle optimal des systèmes non linéaires*. PhD thesis, Institut National Polytechnique de Grenoble, 2006.

[41] E. Sontag. *Mathematical Control Theory: Deterministic Finite Dimensional Systems*. Graduate Texts in Mathematics. Springer, second edition, 1998.

[42] H.J. Sussmann. A maximum principle for hybrid optimal control problems. In *IEEE Conference on Decision and Control*, pages 425–430, 1999.

[43] P. Tabuada and G.J. Pappas. Hybrid abstractions that preserve timed languages. In M.D. Di Benedetto and A.L. Sangiovanni-Vincentelli, editors, *Hybrid Systems : Computation and Control*, volume 2034 of *LNCS*, pages 501–514. Springer, 2001.

[44] E. Trélat. *Contrôle optimal: théorie et applications*. Mathématiques Concrètes. Vuibert, 2005.

[45] J. Zaytoon. *Systèmes dynamiques hybrides*. Systèmes automatisés. Hermès Science, 2001.

[46] J. Zhang, K.H. Johansson, J. Lygeros, and S. Sastry. Zeno hybrid systems. *International Journal of Robust and Nonlinear Control*, 11:435–451, 2001.

[47] G.M. Ziegler. *Lectures on Polytopes*. Graduate Texts in Mathematics. Springer-Verlag, 1994.

A Proof that the control constraints are m-simplices

We take $q' \in \mathcal{K}(q)$. The problem is to compute the intersection of the affine subspace $P = \{(X_0, u)/u \in \mathbb{R}^m\}$ in $\mathbb{R}^{n+m}$ and the simplex $\Delta_{q'}$ (see figure 17):

$$U_{q'}(X_0) = \{(X_0, u) \ / \ u \in \mathbb{R}^m\} \cap \Delta_{q'} = P \cap \Delta_{q'}$$

We know that the intersection of an affine subspace with a simplex is a simplex [47, Theorem 1.1]. Therefore $U_{q'}(X_0)$ is a simplex in $\mathbb{R}^m$. There remain to to compute its dimension. The idea is to consider the intersection of the subspace P of dimension m with the n-faces of $\Delta_{q'}$. These intersections, when they exist, are of dimension 0, and define the vertices of $U_{q'}(X_0)$. We then have to prove that there exists exactly $(m+1)$ different n-faces intersecting P to conclude.

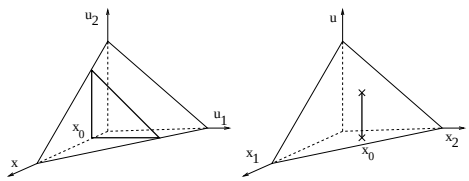


Figure 17: Intersection $P \cap \Delta_{q'}$ for $(n,m) = (1,2)$ on the left, $(n,m) = (2,1)$ on the right.

Let $\{X_1, \ldots, X_{n+1}\}$ be the vertices of D_q. By construction, we have: $p^{\perp}_{\mathbb{R}^n}(\Delta_{q'}) = D_q$. We can then introduce the vertices of $\Delta_{q'}$:

$$V_k = \begin{bmatrix} X_{i_k} \\ u_{j_k} \end{bmatrix}, i_k \in \{1, \ldots, n+1\}, j_k \in \{1, \ldots, m+1\},$$
$$k = 1, \ldots, n+m+1$$

and: $\{X_{i_k}; k = 1, \ldots, n+m+1\} = \{X_1, \ldots, X_{n+1}\}$.
We also assume: $X_0 \in \mathring{D}_q$, hence:

$$\exists!(\alpha_k)_{k=1\ldots n+1} \in]0,1[^{n+1}, \sum_{k=1}^{n+1} \alpha_k = 1 \text{ and } X_0 = \sum_{k=1}^{n+1} \alpha_k X_k \tag{24}$$

1st step: any n-face of $\Delta_{q'}$ has either one intersection with P, or none.
Let F be a n-face of $\Delta_{q'}$ defined by the list (renumbered to ease the readability) of its vertices: $(\begin{bmatrix} X_{i_k} \\ u_{j_k} \end{bmatrix})_{k=1,\ldots,n+1}$ where $i_k \in \{1, \ldots, n+1\}$, $j_k \in \{1, \ldots, m+1\}$.

- 1st *case: the i_k are all different.*
  We then have: $\{X_{i_k} \ / \ k = 1, \ldots, n+1\} = \{X_1, \ldots, X_{n+1}\}$. (24) thus becomes: $X_0 = \sum_{k=1}^{n+1} \alpha_{i_k} X_{i_k}$. We state: $X_F = \sum_{k=1}^{n+1} \alpha_{i_k} \begin{bmatrix} X_{i_k} \\ u_{j_k} \end{bmatrix} = \begin{bmatrix} X_0 \\ \sum_{k=1}^{n+1} \alpha_{i_k} u_{j_k} \end{bmatrix}$.
  Then: $X_F \in P$. Moreover, by construction of X_F, $X_0 \in \mathring{D}_q$ involves: $X_F \in \mathring{F}$, hence: $X_F \in P \cap F$.
  By uniqueness of the convex decomposition in a given simplex, we then deduce:
  $$X_F \in \mathring{F} \text{ and } P \cap F = \{X_F\}$$

- 2nd *case: the i_k are not all different.*
  We then deduce: $\{X_{i_k} \ / \ k = 1, \ldots, n+1\} \subsetneq \{X_1, \ldots, X_{n+1}\}$. For $Y \in F$,

we therefore have:

$$\exists!(\gamma_{i_k})_{k=1\ldots n+1} \in [0,1]^{n+1}, \begin{cases} \sum\limits_{k=1}^{n+1} \gamma_{i_k} = 1 \\ Y = \sum\limits_{k=1}^{n+1} \gamma_{i_k} \begin{bmatrix} X_{i_k} \\ u_{j_k} \end{bmatrix} \end{cases}$$

However, according to the hypothesis: $X_0 \in \mathring{D}_q$, X_0 depends on all the X_i, $i = 1, \ldots, n+1$. It follows: $\sum\limits_{k=1}^{n+1} \gamma_{i_k} X_{i_k} \neq X_0$, i.e.: $Y \notin P$, and: $F \cap P = \emptyset$.

2nd **step: there exist exactly** $(m+1)$ **n-faces intersecting** P**.**
By construction, we know that: $\{X_{i_k} \ / \ k = 1, \ldots, n+m+1\} = \{X_1, \ldots, X_{n+1}\}$; after renumbering the vertices of $\Delta_{q'}$, we can assume:

$$\forall l \in \{1, \ldots, n+1\}, \ V_l = \begin{bmatrix} X_l \\ u_{j_l} \end{bmatrix}$$

- We state: $F_0 = Conv(\{V_1, \ldots, V_{n+1}\})$. By affine independence of $X_1, \ldots, X_{n+1}$ of D_q, the vertices $V_1, \ldots, V_{n+1}$ are also affinely independent, so that F_0 is actually a n-face of $\Delta_{q'}$ intersecting P.
- For $n+2 \leq l \leq n+m+1$ (i.e. m possible values for l), we state:

  $$F_l = Conv\left(\left(\{V_1, \ldots, V_{n+1}\} - \{V_{i_l}\}\right) \cup \{V_l\}\right),$$
  $$1 \leq i_l \leq n+1$$

  where: $V_l = (X_{i_l}, u_{j_l})$ and $V_{i_l} = (X_{i_l}, u_{j_{i_l}})$. As for F_0, F_l is a n-face of $\Delta_{q'}$ which intersects P.

To conclude, we have found $(1+m)$ n-faces of $\Delta_{q'}$ which intersect P, i.e. $(m+1)$ vertices of $U_{q'}(X_0)$.