HAL open science

Implicit Euler numerical simulations of sliding mode systems

Vincent Acary, Bernard Brogliato

To cite this version:

Vincent Acary, Bernard Brogliato. Implicit Euler numerical simulations of sliding mode systems. [Research Report] RR-6886, INRIA. 2009, pp.39. inria-00374840v2

HAL Id: inria-00374840

https://inria.hal.science/inria-00374840v2

Submitted on 10 Apr 2009

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.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Implicit Euler numerical simulations of sliding mode systems

Vincent Acary — Bernard Brogliato

N° 6886

March 2009

Thème NUM

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6886--FR+ENG

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE
INRIA
centre de recherche GRENOBLE - RHÔNE-ALPES

Implicit Euler numerical simulations of sliding mode systems

Vincent Acary*, Bernard Brogliato†

Thème NUM — Systèmes numériques
Équipe-Projet Bipop

Rapport de recherche n° 6886 — March 2009 — 36 pages

Abstract: In this report it is shown that the implicit Euler time-discretization of some classes of switching systems with sliding modes, yields a very good stabilization of the trajectory and of its derivative on the sliding surface. Therefore the spurious oscillations which are pointed out elsewhere when an explicit method is used, are avoided. Moreover the method (an *event-capturing*, or *time-stepping* algorithm) allows for accumulation of events (Zeno phenomena) and for multiple switching surfaces (i.e., a sliding surface of codimension $\geqslant 2$). The details of the implementation are given, and numerical examples illustrate the developments. This method may be an alternative method for chattering suppression, keeping the intrinsic discontinuous nature of the dynamics on the sliding surfaces. Links with discrete-time sliding mode controllers are studied.

Key-words: Switching systems, Filippov's differential inclusions, complementarity problems, backward Euler algorithm, sliding modes, maximal monotone mappings, mixed linear complementarity problem, ZOH discretization.

* vincent.acary@inrialpes.fr
† Bernard.brogliato@inrialpes.fr

Centre de recherche INRIA Grenoble – Rhône-Alpes
655, avenue de l'Europe, 38334 Montbonnot Saint Ismier
Téléphone : +33 4 76 61 52 00 — Télécopie +33 4 76 61 52 52

Simulations numriques par la mthode d'Euler implicite des systmes modes glissants

Résumé : Dans ce rapport, on montre que la discrtisation en temps de type Euler implicite conduit une trs bonne stabilisation d'une classe de systmes commuts avec des modes glissants, et de leurs drives sur la surface de glissement. Les oscillations artificielles qui sont gnralement mentionnes pour l'implmentation discrte de ce type de systmes sont vites. De plus, la mthode (de type "event-capturing" ou "time–stepping") permet de traiter des accumulations d'vnements (Phnomne de Zenon) et des surfaces de commutations multiples (*i.e.* des surfaces de glissement de codimension $\geqslant 2$). Dans ce rapport, les dtails de l'implmentation sont donns et des exemples numriques illustrent ses proprits. Cette mthode peut tre une alternative aux mthodes complexes de suppression des oscillations, en gardant la nature intrinsquement discontinue de la dynamique sur les surfaces de glissement. Le lien avec les commandes modes glissants en temps discret est tudi.

Mots-clés : Systmes commuts, Inclusion Diffrentielles de Filippov, Problmes de complmentarit, Mthode d'Euler implicite, modes glissants, oprateurs, maximaux monotones, Problme linaire de complmentarit mixte, Discrtisation Bloqueur d'Ordre Zro (BOZ)

1 Introduction

Sliding mode controllers are widely used because of their intrinsic robustness properties [41, 23, 8, 51]. Some important fields of application are induction motors [43, 53, 7], aircraft control [44, 32, 54, 35], hard disk drives [33, 31], solar energy systems [28]. However they are known to generate chattering which renders their application delicate. Solutions to cope with chattering or reduce its effects have been proposed, see e.g. [4, 5, 12, 15, 51, 55], which also have their own limitations [55]. One drawback of these solutions is that they usually destroy the intrinsic discontinuous nature of sliding mode control. Fundamentally, these control schemes are of the switching discontinuous type and they yield closed-loop systems that can be recast into Filippov's differential inclusions. The numerical simulation of such nonsmooth dynamical systems is non trivial and it has received a lot of attention, see e.g. [50, 49, 16, 34, 22, 37, 20], to cite a few. Both event-driven methods and time-stepping methods have been developed, see e.g. [1] for a survey. In this paper we focus on time-stepping methods, which have an interest not only for the sake of numerical simulation, but also for the real implementations of sliding mode controllers on discrete-time systems [52]. Recently it has been shown that the *explicit* Euler method generates unwanted effects like spurious oscillations (also called chattering effects) around the switching surface [25, 26, 52, 57]. In parallel, the digital implementation of sliding mode controllers has been studied thoroughly in [27, 36], where the Zero–Order Holder (ZOH) discretization is used.

The purpose of this paper is to analyze the *implicit* (*backward*) Euler method for some particular classes of differential inclusions, that include sliding mode controllers. It is shown that, besides convergence and order results, the advantage of the implicit method is that it allows one to get a very accurate and smooth stabilization on the switching surface (of codimension one or larger than one). Roughly speaking, this is due to the fact that the switches are no longer monitored by the state at step k, but by a *multiplier* (a slack variable in a nonlinear programming language). The multivalued part of the $\mathrm{sgn}(\cdot)$ function, i.e. a multifunction, is then correctly taken into account, avoiding stiff problems. The advantage of such "dual" methods in terms of their accuracy on the sliding surface has already been noticed in [49, 50] in an event-driven context, where the motivation was the simulation of mechanical systems with Coulomb friction. From a numerical point of view, our study shows that convergence and order results may not be sufficient to guarantee that the derivative of the state is correctly approximated on the switching surface. The implicit method adapts naturally to an arbitrary large number of switching surfaces, that is not the case of most of the other methods which become quite cumbersome as soon as more than two switching surfaces are considered. A further advantage of the proposed method is that contrary to other methods that have been studied and which destroy the intrinsic discontinuous nature of sliding mode systems [1] (like the so-called *boundary layer control*, or various filtering techniques), our method keeps the multivalued discontinuity and consequently the fundamental aspects and properties of sliding mode control from a Filippov's systems point of view. Moreover, sampling rates need not be high to reduce chattering, contrary to other discrete sliding mode controllers. A second contribution of this paper is to show that the results that hold for the backward Euler scheme, extend to ZOH discretizations of sliding mode systems.

The paper is organized as follows: Section 2 presents a motivating example for using an implicit Euler implementation of the simplest sliding mode system. In Section 3, a class of differential inclusions is introduced and existence and uniqueness results are given under the maximal monotonicity assumption. Through several examples, the Equivalent–Control–Based Sliding–Mode–Control (ECB-SMC) and the Lyapunov–based discontinuous robust control are shown to fit well within this class of differential inclusion. In Section 4, some convergence and chattering free finite–time stabilization results are given. These central results of the paper show that the implicit Euler implementation of the differential inclusion yields a chattering free convergence in finite time on the sliding surface. Section 5 is devoted the study of Discrete–time Sliding Mode Control and the extension to ZOH discretization. Some hints on the numerical implementation of the implicit Euler scheme are given in Section 6 and the paper ends with some numerical experiments in Section 9.

[1] see [55] for a discussion on this point.

Notations and definitions: Let $A \in I\!R^{n\times m}$, then $A_{\bullet i}$ is the ith column and $A_{i\bullet}$ is the ith row. The open ball of radius $r > 0$ centered at a point $x \in I\!R^n$ is denoted by $B_r(x)$. For a set of indices $\alpha \subset \{1, \ldots, n\}$ and a column vector $x \in I\!R^n$, the column vector x_α will denoted the sub-vector of corresponding indices in α, that is $x_\alpha = [x_i, i \in \alpha]^T$.

2 A simple example

To start with we consider the simplest case:

$$\dot{x}(t) \in -\mathrm{sgn}(x(t)) = \begin{cases} 1 & \text{if} \quad x(t) < 0 \\ -1 & \text{if} \quad x(t) > 0 \\ \text{[-1,1]} & \text{if} \quad x(t) = 0 \end{cases}, \quad x(0) = x_0 \tag{1}$$

with $x(t) \in I\!R$. This system possesses a unique Lipschitz continuous solution for any x_0. The backward Euler discretization of (1) reads as:

$$\begin{cases} x_{k+1} - x_k = -hs_{k+1} \\ s_{k+1} \in \mathrm{sgn}(x_{k+1}) \end{cases} \tag{2}$$

This method converges with at least order $\frac{1}{2}$ (see Proposition 2 below). Let us now state a result which shows that once the iterate x_k has reached a value inside some threshold around zero for some k, then the dual variable s_{k+1} keeps its value and so does x_{k+n} for all $n \geqslant 1$.

Lemma 1 *For all $h > 0$ and $x_0 \in I\!R$, there exists k_0 such that $x_{k_0+n} = 0$ and $\dfrac{x_{k_0+n+1} - x_{k_0+n}}{h} = 0$ for all $n \geqslant 1$.*

Proof: The value k_0 is defined as the first time step such that $x_{k_0} \in [-h, h]$. If $x_0 \in [-h, h]$, then $k_0 = 0$. Otherwise, the solution of the time-discretization (2) is given by $x_k = x_0 - \mathrm{sgn}(x_0)kh, s_k = \mathrm{sgn}(x_o)$ while $x_k \notin [-h, h]$ for $k < k_0$, and $k_0 = \lceil \frac{|x(0)|}{h} \rceil - 1$. The symbol $\lceil x \rceil$ is the ceiling function which gives the smallest integer greater than or equal to x.

Let us now consider that $x_{k_0} \in [-h, h]$. The only possible solution for

$$\begin{cases} x_{k_0+1} - x_{k_0} = -hs_{k_0+1} \\ s_{k_0+1} \in \mathrm{sgn}(x_{k_0+1}) \end{cases} \tag{3}$$

is $x_{k_0+1} = 0$ and $s_{k_0+1} = \dfrac{x_{k_0}}{h}$. For the next iteration, we have to solve

$$\begin{cases} x_{k_0+2} = -hs_{k_0+2} \\ s_{k_0+2} \in \mathrm{sgn}(x_{k_0+2}) \end{cases} \tag{4}$$

and we obtain $x_{k_0+2} = 0$ and $s_{k_0+2} = 0$. The same holds for all x_{k_0+n},s_{k_0+n}, $n \geqslant 3$, redoing the same reasoning. Clearly then the terms $\dfrac{x_{k_0+n+1} - x_{k_0+n}}{h}$ approximating the derivative, are zero for any $h > 0$. □

This result is robust with respect to the numerical threshold that can be encountered in floating point operations. Indeed, let us assume that $x_{k_0} - h = \varepsilon \ll 1$, that is, $\varepsilon > 0$ is zero at the machine precision. We obtain $s_{k_0+1} = -1$ and $x_{k_0+1} = \varepsilon$ that is zero at the machine precision. For $n = 2$, we obtain $x_{k_0+2} = 0$ and $s_{k_0} = \dfrac{\varepsilon}{h}$. This robustness stems from the fact that the dynamics is not only monitored by the sign of x_k but also by the fact that the "dual" variable s_{k+1} belongs to $[-1, 1]$.

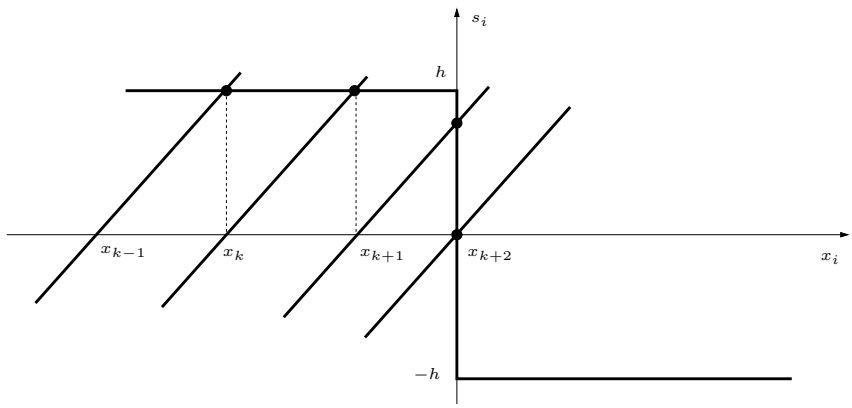


Figure 1: Iterations of the backward Euler method.

Consequently this result shows that there are no spurious oscillations around the switching surface, contrary to other time-stepping schemes like the explicit Euler method [25, 26]. Remarkably Lemma 1 holds for any $h > 0$, which means that even a large time step assures a smooth stabilization on the sliding surface. It is noteworthy that solving the system (2) with unknown x_{k+1} and s_{k+1} is equivalent to calculate the intersection between the graph of the multivalued mapping $x_{k+1} \mapsto -h\text{sgn}(x_{k+1})$ and the straight line $x_{k+1} \mapsto x_{k+1} - x_k$. This is illustrated on Figure 1, where few iterations are depicted until the state reaches zero.

From a control perspective the input is implemented on $[t_k, t_{k+1})$ as $u_k = -\text{sgn}(x_{k+1})$ as a function of x_k and h, where h is the sampling time. There is no problem of causality in such an implementation. It is noteworthy that in the implicit method there is absolutely no issue related to calculating sgn(0), or more exactly sgn(ϵ) where ϵ is a very small quantity whose sign is uncertain. The implicit method automatically computes a value inside the multivalued part of the sign multifunction and may be considered as *the* time-discretization of the multifunction sgn($\cdot$). It is easy to show that the explicit method yields an oscillation around $x = 0$, as shown in more general situations in [25, 26]. Other time-stepping methods exist, like the so-called *switched model* [1, 37], however it fails to correctly solve the integration problem when the number of switched surfaces is too large (see also [4] for similar issues when the so-called *sigmoid blending* mechanism is implemented). Moreover this method may yield a stiff system, and from a control point of view it introduces a high-gain feedback that may not be desirable in practical applications.

On Figure 2(a)-(c), the discrete state x_k and the control s_k are displayed for $x_0 = 1.01$ at $t_0 = 0$ and for various values of the time–step h that are sufficiently large to illustrate the behavior of the time–stepping scheme and its convergence.

Let us define two discrete function norms to measure the convergence:

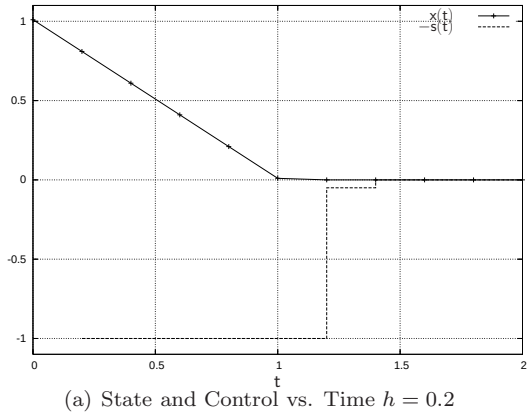
$$\begin{aligned} \|e_f\|_\infty &= \textstyle\sum_{i=0}^N |f_k - f(t_k)| \\ \|e_f\|_p &= (h \textstyle\sum_{i=0}^N |f_k - f(t_k)|^p)^{1/p}. \end{aligned} \tag{5}$$

We can compute that

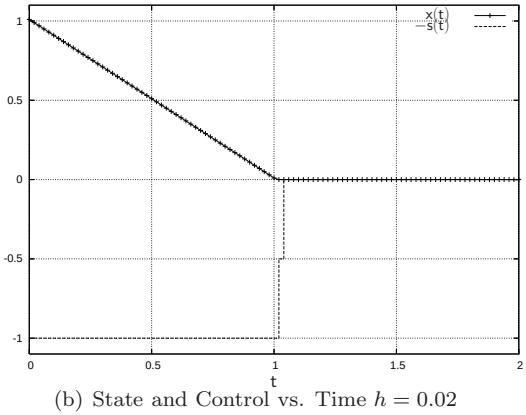
$$\|e_s\|_\infty = 1 \text{ for all } h > 0 \tag{6}$$

and therefore there is no convergence in infinite norm $\|.\|_\infty$ for $s = \text{sgn}(x)$. In $\|.\|_1$ and $\|.\|_2$, we can respectively observe the convergence with order 1 on Figure 2(d).

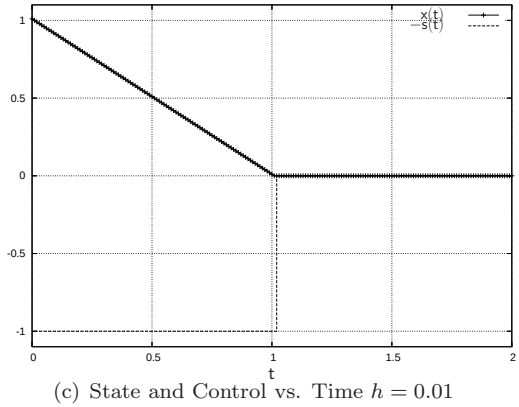
Complementarity framework Let us end this section by restating the systems (1) and (2) into the complementarity framework. Let us introduce equivalent formulations of the inclusion



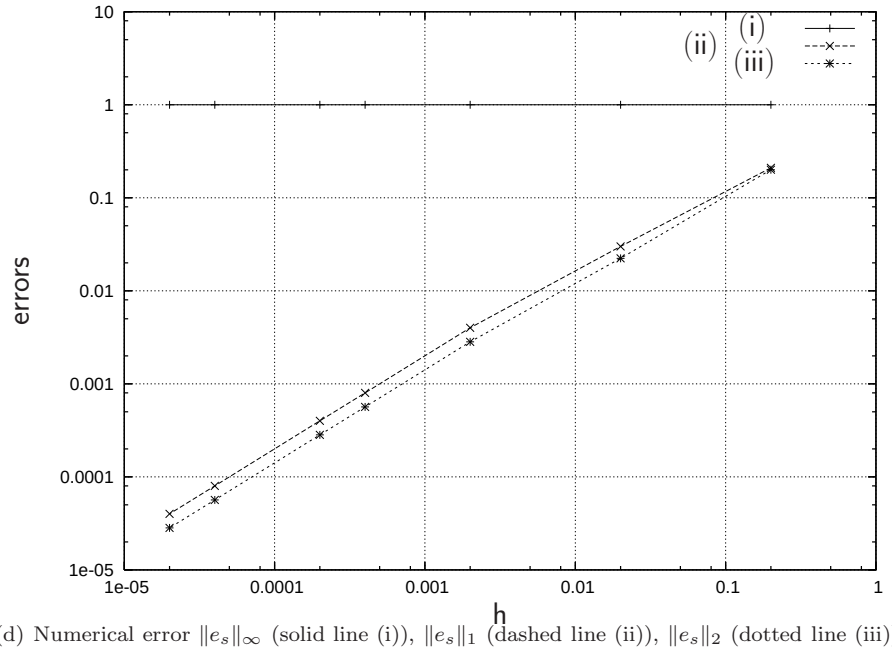
(a) State and Control vs. Time $h = 0.2$



(b) State and Control vs. Time $h = 0.02$



(c) State and Control vs. Time $h = 0.01$



(d) Numerical error $\|e_s\|_\infty$ (solid line (i)), $\|e_s\|_1$ (dashed line (ii)), $\|e_s\|_2$ (dotted line (iii)), with respect to h in logscale

Figure 2: A simple example for $x_0 = 1.01$ at $t_0 = 0$.

$s(t) \in \mathrm{sgn}(x(t))$ such that

$$s(t) \in \mathrm{sgn}(x(t)) \Leftrightarrow x(t) \in N_{[-1,1]}(s(t)) \Leftrightarrow s(t) \in [-1,1] \text{ and } \begin{cases} x(t) = 0 \text{ if } s(t) \in]-1,1[\\ x(t) \leqslant 0 \text{ if } x(t) = -1 \\ x(t) \geqslant 0 \text{ if } x(t) = 1 \end{cases} \tag{7}$$

where $N_{[-1,1]}$ is the normal cone in the sense of Convex Analysis to the interval $[-1,1]$. The definition of the normal cone in the present case,

$$N_{[-1,1]}(s) = \{-v_1 + v_2, 0 \leqslant v_1 \perp s+1 \geqslant 0, 0 \leqslant v_2 \perp 1-s \geqslant 0\} \tag{8}$$

yields the following complementarity representation of the sign multi-valued function

$$x(t) \in N_{[-1,1]}(s(t)) \Leftrightarrow \begin{cases} x(t) = -v_1(t) + v_2(t) \\ 0 \leqslant v_1(t) \perp s(t)+1 \geqslant 0 \\ 0 \leqslant v_2(t) \perp 1-s(t) \geqslant 0 \end{cases} \tag{9}$$

In order to directly substitute the value of $s(t)$ into the dynamics $\dot{x}(t) = -s(t)$, a other complementarity formulation can be defined. By setting $\lambda_1(t) = \frac{1}{2}(1-s(t))$ and $\lambda_2(t) = v_1(t)$, one gets

$$x(t) \in N_{[-1,1]}(s(t)) \Leftrightarrow \begin{cases} s(t) = 1 - 2\lambda_1(t) \\ 0 \leqslant \lambda_1(t) \perp x(t) + \lambda_2(t) \geqslant 0 \\ 0 \leqslant \lambda_2(t) \perp 1 - \lambda_1(t) \geqslant 0 \end{cases} \tag{10}$$

3 A class of differential inclusions

Let us now introduce the following class of differential inclusions, where $x(t) \in I\!R^n$:

$$\begin{cases} \dot{x}(t) \in -A(x(t)) + f(t, x(t)), \text{ a.e. on } (0,T) \\ \\ x(0) = x_0 \end{cases} \tag{11}$$

The following assumption is made:

Assumption 1 *The following hold:*

- **(i)** *$A(\cdot)$ is a multivalued maximal monotone operator from $I\!R^n$ into $I\!R^n$, with domain $D(A)$, i.e., for all $x \in D(A), y \in D(A)$ and all $x' \in A(x), y' \in A(y)$, one has*

$$(x' - y')^T (x - y) \geqslant 0 \tag{12}$$

- **(ii)** *There exists $L \geqslant 0$ such that for all $t \in [0,T]$, for all $x_1, x_2 \in I\!R^n$, one has $||f(t,x_1) - f(t,x_2)|| \leqslant L||x_1 - x_2||$.*
- **(iii)** *There exists a function $\Phi(\cdot)$ such that for all $R \geqslant 0$:*

$$\Phi(R) = \sup \left\{ \| \frac{\partial f}{\partial t}(\cdot, v) \|_{\mathcal{L}^2((0,T);I\!R^n)} \mid \| v \|_{\mathcal{L}^2((0,T);I\!R^n)} \leqslant R \right\} < +\infty$$

.

The following is proved in [10, 9].

Proposition 1 *Let Assumption 1 hold, and let $x_0 \in D(A)$. Then the differential inclusion (11) has a unique solution $x : (0,T) \to I\!R^n$ that is Lipschitz continuous.*

In this paper we shall focus on inclusions of the form:

$$\begin{cases} \dot{x}(t) \in f(t, x(t)) - B\mathrm{Sgn}(Cx(t) + D), \text{ a.e. on } (0, T) \\ \\ x(0) = x_0 \end{cases} \tag{13}$$

with $B \in I\!R^{n\times m}$, and $\mathrm{Sgn}(Cx(t) + D) \triangleq (\mathrm{sgn}(C_1x + D_1), ..., \mathrm{sgn}(C_mx + D_m))^T \in I\!R^m$. It will be shown how to recast (13) into (11).

Example 1 (Equivalent-control-based sliding-mode-control (ECB-SMC)) *Consider a system $\dot{x}(t) = Fx(t) + Gu$, with an equivalent-control-based sliding-mode-control (ECB-SMC) of the form $u(x) = -(HG)^{-1}HFx - \alpha(HG)^{-1}Sgn(Hx)$, $\alpha > 0$ (see e.g. [57]). Then the closed-loop system $\dot{x}(t) = (F - G(HG)^{-1}HF)x(t) - \alpha G(HG)^{-1}Sgn(Hx(t))$ fits within (13).*

Let us now state a well-posedness result which is a consequence of Proposition 1.

Corollary 1 *Consider the differential inclusion in (13). Suppose that* **(ii)** *and* **(iii)** *of Assumption 1) hold. If there exists an $n \times n$ matrix $P = P^T > 0$ such that*

$$PB_{\bullet i} = C_{i\bullet}^T \tag{14}$$

for all $1 \leqslant i \leqslant m$, then for any initial data the differential inclusion (13) has a unique solution $x : (0, T) \to I\!R^n$ that is Lipschitz continuous.

Proof: The proof uses a state variable change introduced in [13]. Let R be the symmetric square root of P, i.e. $R^2 = P$. Let us perform the state transformation $z = Rx$. Then we get

$$\dot{z}(t) \in Rf(t, R^{-1}z(t)) - RB\mathrm{Sgn}(CR^{-1}z(t) + D) \tag{15}$$

Notice that $B\mathrm{Sgn}(CR^{-1}z(t) + D) = \sum_{i=1}^m B_{\bullet i}\mathrm{sgn}(C_{i\bullet}R^{-1}z + D_i)$. Therefore $RB\mathrm{Sgn}(CR^{-1}z(t) + D) = \sum_{i=1}^m RB_{\bullet i}\mathrm{sgn}(C_{i\bullet}R^{-1}z + D_i) = \sum_{i=1}^m R^{-1}C_{i\bullet}^T\mathrm{sgn}(C_{\bullet i}R^{-1}z + D_i)$. We can rewrite the system as

$$\dot{z}(t) \in Rf(t, R^{-1}z(t)) - \sum_{i=1}^m R^{-1}C_{i\bullet}^T\mathrm{sgn}(C_{i\bullet}R^{-1}z(t) + D_i) \tag{16}$$

The multivalued mapping $\xi \mapsto \mathrm{sgn}(\xi)$ is monotone. By [46, Exercise 12.4] it follows that each multivalued mapping $z \mapsto R^{-1}C_{i\bullet}^T\mathrm{sgn}(C_{i\bullet}R^{-1}z(t) + D_i)$ is monotone. From [29, Proposition 1.3.11] it follows that $R^{-1}C_{i\bullet}^T\mathrm{sgn}(C_{i\bullet}R^{-1}z(t) + D_i) = \partial f_i(z)$ with $f_i(z) = |C_{i\bullet}R^{-1}z(t) + D_i|$. By [45, Theorem 5.7] it follows that $f_i(\cdot)$ is convex. Being the subdifferential of a convex function, the multivalued mapping $z \mapsto \partial f_i(z)$ is maximal (monotone) [45, Corollary 31.5.2]. Therefore by Proposition 1 the inclusion in (16) possesses a unique Lipschitz solution on $(0, T)$ for any $T > 0$ and since R is full–rank so does (13). □

Example 2 *Consider the sliding mode system in [25, Equ.(1)–(4)]. One has $B = (0 \quad 1)^T$, $C = (c_1 \quad 1)$, $D = 0$. Then the condition in (14) holds with $P = \begin{pmatrix} p_{11} & c_1 \\ c_1 & 1 \end{pmatrix}$ and $p_{11} > (c_1)^2$ assures that $P > 0$.*

Example 3 *Consider $B = \begin{pmatrix} 1 & 2 \\ 2 & -1 \end{pmatrix}$, $Sgn(Cx + D) = (sgn(x_1 + 2x_2), sgn(2x_1 - x_2))^T$. Trajectories may slide on codimension one surfaces $x_1 + 2x_2 = 0$ or $2x_1 - x_2 = 0$ and on the codimension 2 surface ($x_1 + 2x_2 = 0$ and $2x_1 - x_2 = 0$).*

Example 4 *One solution to reduce chattering is the observer based SMC. Let us consider the following example taken from [55], whose closed-loop dynamics is given by:*

$$\begin{pmatrix} \dot{x}(t) \\ \dot{e}(t) \\ \dot{x}_s(t) \\ \ddot{x}_s(t) \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ k & -k & -k & 0 \\ 0 & 0 & 0 & 1 \\ \frac{1}{\tau^2} & 0 & -\frac{1}{\tau^2} & -\frac{2}{\tau} \end{pmatrix} \begin{pmatrix} x(t) \\ e(t) \\ x_s(t) \\ \dot{x}_s(t) \end{pmatrix} - \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} sgn(Cx(t)) \tag{17}$$

with $C = (1\ \ -1\ 0\ 0)$. *For the notations see [55, §II.C]. This system satisfies the condition (14) with* $P = \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & p_{22} & 0 & 0 \\ 0 & 0 & p_{33} & 0 \\ 0 & 0 & 0 & p_{44} \end{pmatrix}$, $p_{22} > 1$, $p_{33} > 0$, $p_{22} > 0$.

Notice that the condition (14) implies that $B_{\bullet i}^T P B_{\bullet i} = B_{\bullet i}^T C_{i\bullet}^T = B_{i\bullet} C_{\bullet i} > 0$. When $m = 1$ this is a relative degree one condition. It is noteworthy that (14) does not imply that B has full column rank. In particular it does not preclude $m > n$. Dissipative systems with no feedthrough matrix satisfy an input-output constraint similar to (14) [14].

Example 5 (Lyapunov-based discontinuous robust control) *Let us show how the above material adapts to this type of feedback controller. The class of dynamical systems is*

$$\dot{x}(t) = f(x(t)) + Bu(t) + B\gamma(t), \quad x(0) = x_0 \tag{18}$$

where $x(t) \in I\!R^n$, $B \in I\!R^{n\times m}$, $f(\cdot)$ *satisfies assumption 1, and* $\gamma(\cdot) \in I\!R^m$ *is a bounded disturbance satisfying* $|\gamma_i(t)| < \rho_i$ *for all* $1 \leqslant i \leqslant m$, *all* $t \geqslant 0$ *and some finite* ρ_i. *The problem is the stabilization of the system at the origin* $x = 0$, *knowing that there exists a function* $V(\cdot)$ *such that the uncontrolled undisturbed system* $\dot{x}(t) = f(x(t))$ *admits* $V(\cdot)$ *as a Lyapunov function. In particular, one has* $\dot{V}(x(t)) = \nabla V(x(t))^T f(x(t)) \leqslant 0$ *along the trajectories of the free system. Let us rewrite the system in (18) as*

$$\dot{x}(t) = f(x(t)) + \sum_{i=1}^{m} B_{\bullet i} u_i + \sum_{i=1}^{m} B_{\bullet i}\gamma_i(t) \tag{19}$$

Let us propose the control input $u_i(x) = -\rho_i sgn(\nabla V^T(x) B_{\bullet i})$. *We obtain:*

$$\dot{x}(t) \in f(x(t)) - \sum_{i=1}^{m} \rho_i B_{\bullet i} sgn(\nabla V(x)^T B_{\bullet i}) + \sum_{i=1}^{m} B_{\bullet i}\gamma_i(t) \tag{20}$$

We can state the following result.

Corollary 2 *Suppose that* $V(x) = \frac{1}{2}x^T P x$, $P = P^T > 0$. *The system in (20) has a unique Lipschitz continuous solution on* $[0, +\infty)$ *for any* x_0.

Proof: *We have* $\nabla V(x)^T B_{\bullet i} = B_{\bullet,i}^T P x$. *Let* $z = Rx$, *where* $R > 0$ *is the symmetric square root of* P. *We may rewrite (20) as*

$$\dot{z}(t) \in Rf(R^{-1}z(t)) - \sum_{i=1}^{m} \rho_i R B_{\bullet i} sgn(B_{\bullet i}^T R z) + \sum_{i=1}^{m} R B_{\bullet i}\gamma_i(t)$$

Then following the same steps as for the proof of Corollary 1 we conclude that Proposition 1 applies to this system, hence to (20). □

Such a controller assures the global asymptotic stability of the equilibrium $x = 0$. *This is made possible because of the multivalued characteristic of the discontinuous input. The closed-loop system possesses the origin as its unique equilibrium, because of the multivaluedness property. The restriction to quadratic Lyapunov functions stems from monotonicity preserving conditions, and is not straightforwardly avoided.*

4 Convergence results and Chattering Free Finite–time Stabilization

The differential inclusion (11) is time-discretized on $[0, T]$ with a backward Euler scheme as follows:

$$\begin{cases} \dfrac{x_{k+1} - x_k}{h} + A(x_{k+1}) \ni f(t_k, x_k), \text{ for all } k \in \{0, ..., N-1\} \\ \\ x_0 = x(0) \end{cases} \tag{21}$$

where $h = \frac{T}{N}$. The fully implicit method uses $f(t_{k+1}, x_{k+1})$ instead of $f(t_k, x_k)$. The convergence and order results stated in Proposition 2 below have been derived for the semi-implicit scheme (21) in [10]. So the analysis in this section is based on such a discretization. However this is only a particular case of a more general $\theta-$method which is used in practical implementations. The next result is proved in [10].

Proposition 2 *Under assumption 1, there exists η such that for all $h > 0$ one has*

$$\text{For all } t \in [0,T],\ ||x(t) - x^N(t)|| \leqslant \eta\sqrt{h} \tag{22}$$

Moreover $\lim_{h\to 0^+} \max_{t\in[0,T]} ||x(t) - x^N(t)||^2 + \int_0^t ||x(s) - x^N(s)||^2 ds = 0.$

Thus the numerical scheme in (21) has at least order $\frac{1}{2}$, and convergence holds. The conditions of Assumption 1 for the existence and uniqueness results of Proposition 1 are sufficient only. Other criteria exist, like Filippov's criterion for uniqueness of solutions [18, Proposition 5]. Similarly it is possible that time-stepping methods converge for systems that satisfy such a criterion, despite no result seems to be available in the literature. As seen in Lemma 1, the precision of the method may be much better than what is to be expected from (22) on large portions of the trajectories.

The differential inclusion in (13) is therefore discretized as follows:

$$\begin{cases} \dfrac{x_{k+1} - x_k}{h} \in f(t_k, x_k) - B\mathrm{Sgn}(Cx_{k+1} + D), \ \text{ a.e. on } (0,T) \\ \\ x(0) = x_0 \end{cases} \tag{23}$$

One sees that advancing the implicit method from step k to step $k+1$ involves solving generalized equations with unknown x_{k+1}, of the form $0 \in F_s(x_{k+1}) + F_m(x_{k+1})$ where $F_s(\cdot)$ is singlevalued while $F_m(\cdot)$ is multivalued. h, t_k and x_k appear as parameters of the generalized equations. Solving such generalized equations thus boils down to computing the intersection between the graph of $F_s(\cdot)$ and the graph of $F_m(\cdot)$ as illustrated in section 2. The result of Proposition 2 applies to (23). As we shall see next, such an implicit method also assures a good estimate of the derivative $\dot{x}$ and a smooth stabilization of the discrete-time solution on the sliding surface.

Before stating the smooth stabilization result, let us consider a preliminary result. Let us denote the output of the dynamical as:

$$y(t) \triangleq Cx(t) + D \tag{24}$$

Lemma 2 *Let us assume that a sliding mode exists for some indices $i \in \alpha \subset \{1 \dots m\}$ such that*

$$\exists t_* > 0, \quad y_\alpha(t) = C_{\alpha\bullet}x(t) + D_\alpha = 0, \qquad \text{for all } t > t_*. \tag{25}$$

Then there exists $\rho > 0$ such that for all $t > t_*$ and for all $x(t)$ such that $C_{\alpha\bullet}x(t) + D_\alpha = 0$, one has

$$\|(C_{\alpha\bullet}f(x(t),t))\| \leqslant \rho \tag{26}$$

Furthermore, let Assumption 1.(ii) holds, then the following bound is satisfied in the neighborhood of the sliding subspace,

$$\exists r > 0, \exists \kappa > 0, \exists \rho > 0 \text{ such that } \forall t > t_*, \forall \bar{x} \in B_r(x), \|(C_{\alpha\bullet}f(\bar{x},t))\| \leqslant \kappa r + \rho \tag{27}$$

for all $x(t)$ such that $C_{\alpha\bullet}x(t) + D_\alpha = 0$.

Proof: From (25), we have $\dot{y}_\alpha(t) \in C_{\alpha\bullet}f(x(t),t) - hC_{\alpha\bullet}B\mathrm{Sgn}(y(t))$. For $t > t_*$, the sliding mode $y_\alpha(t) = 0$ implies that $\dot{y}_\alpha(t) = C_{\alpha\bullet}\dot{x}(t) = 0$ for all $t > t_*$ and therefore

$$C_{\alpha\bullet}f(x(t),t) \in C_{\alpha\bullet}B\mathrm{Sgn(y(t))} \tag{28}$$

The inclusion (28) yields

$$\exists \rho > 0, \|(C_{\alpha\bullet} f(x(t), t))\| \leqslant \rho \tag{29}$$

for all $x(t)$ such that $C_{\alpha\bullet} x(t) + D_\alpha = 0$. By the assumption 1.(ii), the Lipschitz continuity of $f(\cdot, \cdot)$ allows us to write for some $\kappa > 0$

$$\forall \bar{x}(t) \in B_r(x(t)), \quad \|C_{\alpha\bullet}(f(\bar{x}(t), t) - f(x(t), t))\| \leqslant \|C_{\alpha\bullet}\| Lr \triangleq \kappa r. \tag{30}$$

Combining (29) and (30) ends the proof. □

Lemma 1 extends to (23) as follows when the sliding surface of codimension $|\alpha|$ is attained.

Lemma 3 *Let us assume that a sliding mode occurs for the index $\alpha \subset \{1 \dots m\}$, that is $y_\alpha(t) = 0, t > t_*$. Let C and B be such that (14) holds and $C_{\alpha\bullet} B_{\bullet\alpha} > 0$. Then there exists $h_c > 0$ such that $\forall h < h_c$, there exists $k_0 \in I\!N$ such that $y_{k_0+n} = C x_{k_0+n+1} + D = 0$ for all integers $n \geqslant 1$.*

Proof: At each time–step, we have to solve for $y_{k+1} = C x_{k+1} + D$ and s_{k+1} the generalized equation

$$\begin{cases} y_{k+1} = y_k + hCf(t_k, x_k) - hCBs_{k+1} \\ s_{k+1} \in \mathrm{Sgn}(y_{k+1}) \end{cases} \tag{31}$$

Under condition (14), the convergence of the time–stepping scheme is ensured by Proposition 2. The convergence and the existence of the sliding mode ensure that

$$\exists k_0, \exists K_1 > 0, \exists K_2 > 0, \exists t_1 > t_* \text{ such that } \quad \|y_{\alpha,k_0}\| \leqslant K_1 \sqrt{h} \text{ and } \|x_{k_0} - x(t_1)\| \leqslant K_2 \sqrt{h} \tag{32}$$

for $C_{\alpha\bullet} x(t_1) + D_\alpha = 0$. Using (27) for $x(t_1)$ and a sufficiently small h such that $r = K_2\sqrt{h}$, we have the following bound

$$\|y_{\alpha,k_0} + hC_{\alpha,\bullet} f(t_{k_0}, x_{k_0})\| \leqslant \sqrt{h}(K_1 + h\kappa K_2 + \sqrt{h}\rho) \tag{33}$$

Introducing the complementary index set $\beta = \{i, y_i(t) = C_{i\bullet} x(t) + D_i \neq 0\}$, for $t > t_*$ almost everywhere and using (33) we obtain that there exists $\rho_1 > 0$ such that

$$\|y_{\alpha,k_0} + hC_{\alpha,\bullet} f(t_{k_0}, x_{k_0}) - hC_{\alpha\bullet} B_{\bullet\beta} \mathrm{Sgn}(y_{\beta,k_0+1})\| \leqslant \sqrt{h}(K_1 + h\kappa K_2 + \sqrt{h}(\rho + \rho_1)) \tag{34}$$

and therefore it is possible to choose h_1 such that for all $h < h_1$

$$\left| \left[-h(C_{\alpha\bullet} B_{\bullet\alpha})^{-1} \left[y_{\alpha,k_0} + hC_{\alpha,\bullet} f(t_{k_0}, x_{k_0}) - hC_{\alpha\bullet} B_{\bullet\beta} \mathrm{Sgn}(y_{\beta,k_0+1}) \right] \right]_i \right| \leqslant 1, \text{ for all } i \in \alpha. \tag{35}$$

If (35) is satisfied, the unique solution of (31) at the iteration $k_0 + 1$ is given by

$$y_{\alpha,k_0+1} = 0; s_{\alpha,k_0+1} = -h(C_{\alpha\bullet} B_{\bullet\alpha})^{-1} \left[y_{\alpha,k_0} + hC_{\alpha,\bullet} f(t_{k_0}, x_{k_0}) - hC_{\alpha\bullet} B_{\bullet\beta} \mathrm{Sgn}(y_{\beta,k_0+1}) \right] \tag{36}$$

The next iterate will by given by the solution of the generalized equation,

$$\begin{cases} y_{k_0+2} = hCf(t_{k_0+1}, x_{k_0+1}) - hCBs_{k_0+2} \\ s_{k_0+2} \in \mathrm{Sgn}(y_{k_0+2}) \end{cases} . \tag{37}$$

Using the fact that $y_{\alpha,k_0+1} = C_{\alpha\bullet} x_{k_0+1} + D_\alpha = 0$, we can use (29) to conclude that there exists h_2 such that for all $h < h_2$

$$\left| \left[-h(C_{\alpha\bullet} B_{\bullet\alpha})^{-1} \left[hC_{\alpha,\bullet} f(t_{k_0+1}, x_{k_0+1}) - hC_{\alpha\bullet} B_{\bullet\beta} \mathrm{Sgn}(y_{\beta,k_0+2}) \right] \right]_i \right| \leqslant 1, \text{ for all } i \in \alpha, \tag{38}$$

and therefore the solution of (37) is

$$y_{\alpha,k_0+2} = 0; s_{\alpha,k_0+2} = -h(C_{\alpha\bullet} B_{\bullet\alpha})^{-1} \left[hC_{\alpha,\bullet} f(t_{k_0+1}, x_{k_0+1}) - hC_{\alpha\bullet} B_{\bullet\beta} \mathrm{Sgn}(y_{\beta,k_0+2}) \right] \tag{39}$$

The bound (29) is uniform and can be applied for the next steps. Choosing h_c as the minimum of the considered time steps $h_1, h_2, \dots$, the proof is obtained for $y_{\alpha,k_0+n}, n \geqslant 1$. □

The finite-time convergence of the time-discretization of similar nonsmooth dynamical systems (essentially mechanical systems with dry friction) is proved in [6]. Our results may therefore be considered as the continuation of studies on the finite-time convergence for algorithms of the proximal type.

5 Discrete–time Sliding Mode Control (SMC)

This section is devoted to show how the above discretizations may be used in a digital control framework.

5.1 Example of an implicit Euler controller (IEC)

Let us come back to the inclusion in (1). For this simple system, the ZOH and the Euler discretization yield the same–discrete system. Assume the integrator $\dot{x}(t) = u(t)$ is sampled with sampling period $h > 0$. On the time interval $[t_k, t_{k+1})$ one has $x(t) = x_k + h_t u_k$, where $h_t = t - t_k$. The controller $u(x) = -\text{sgn}(x)$ is known as the equivalent control-based SMC [57]. Let us implement a "backward" controller $u_k = -\text{sgn}(x_{k+1})$ at time t_k, following the above lines. Suppose that $x_k \in [-h, h]$. Then following the same calculations as in the proof of Lemma 1, we obtain that $s_{k+1} = \frac{x_k}{h}$. Therefore on $[t_k, t_{k+1})$:

$$x(t) = x_k - \frac{h_t}{h} x_k \tag{40}$$

and it follows that $x(t_{k+1}) = x_{k+1} = 0$. On the next sampling interval $[t_{k+1}, t_{k+2})$ one obtains $s_{k+2} = 0$

$$x(t) = x_{k+1} - \frac{h_t}{h} x_{k+1} = 0 - \frac{h_t}{h} 0 = 0 \tag{41}$$

and so on on the next intervals, where the zero value is obviously some small value at the machine accuracy. if we suppose that $x_k \notin [-h, h]$, the value of s_{k+1} is 1 or -1 according to the sign of x_k. To summarize the control is given explicitly in terms of x_k and h by

$$u_k = -\text{proj}_{[-1,1]}(\frac{x_k}{h}) \tag{42}$$

where proj_C denotes the Euclidean projection operator onto the set C.

As alluded to above, such an "implicit" input is causal and can be computed at t_k with the values of the state at t_k by (42). It requires at each step to solve a rather simple multivalued problem which a Mixed Linear Complementarity Problem (MLCP, see Section 6 below). It is not of the high gain type.

Remark 1 *The fact that the function sgn$(\cdot)$ generates only binary values ($+1$ or -1) does not hamper the above method to work. Indeed the implicit Euler method allows us to compute values of the sign multifunction inside its multivalued part at $x_k = 0$.*

5.2 Extension to ZOH discretized systems

The ZOH discretization of linear time invariant systems $\dot{x}(t) = Fx(t) + Gu(t)$ with an ECB-SMC controller, $u(x) = -(CG)^{-1}(CFx + \alpha \text{Sgn}(Cx)), \alpha > 0$ results in a discrete-time system of the form:

$$x_{k+1} = \Phi x_k - \Gamma s_k \text{ for all } t \in [kh, (k+1)h) \tag{43}$$

where $h > 0$ is the sampling period, and

$$\Phi = \exp(Fh) - \int_0^h \exp(F\tau) d\tau G(CG)^{-1} CF \tag{44}$$

$$\Gamma = \int_0^h \exp(F\tau) G(CG)^{-1} d\tau \tag{45}$$

with $G \in I\!R^{n\times m}$, $C \in I\!R^{m\times n}$, when a explicit Euler implementation of the control is performed [52, 56]. For an implicit Euler implementation, let us set

$$\begin{cases} u_k = -(CG)^{-1}(CFx_k + s_{k+1}) \\ s_{k+1} = \text{Sgn}(Cx_{k+1}), \end{cases} \tag{46}$$

which corresponds to the implicit discrete time version of the ECB-SMC controller. We therefore get on each sampling period:

$$x_{k+1} = \Phi x_k - \Gamma s_{k+1} \text{ for all } t \in [kh, (k+1)h) \tag{47}$$

At each time–step, one has to solve

$$\begin{cases} x_{k+1} = \Phi x_k - \Gamma s_{k+1} \\ y_{k+1} = C x_{k+1} + D \\ s_{k+1} \in \mathrm{Sgn}(y_{k+1}) \end{cases} . \tag{48}$$

Inserting the first line of (48) into the second line we obtain the following one–step system

$$\begin{cases} y_{k+1} = C\Phi x_k + D - C\Gamma s_{k+1} \\ s_{k+1} \in \mathrm{Sgn}(y_{k+1}) \end{cases} . \tag{49}$$

Comparing with the time–discretized systems in (23) and (31) one sees that the term hCB is replaced in case of a ZOH method by the term $C\Gamma$. Provided the problem has a unique solution one can compute the controller in (46) with the knowledge of x_k, h, F, G and C. We will see in the next Section how the computation can be carried out in practice.

6 Implementation of Discrete–Time Systems

Let us consider in this section the following discrete–time system:

$$\begin{cases} x_{k+1} = R x_k + p - S s_{k+1} \\ y_{k+1} = C x_{k+1} + D \\ s_{k+1} \in \mathrm{Sgn}(y_{k+1}) \end{cases} \tag{50}$$

where $k \geqslant 0$ is an integer, x_k the discrete state, y_k the discrete output and s_k the discrete input. The discrete system (50) is a common representative for the discretization given by (23), (21) or (48) and the matrices $R \in I\!R^{n\times n}$, $S \in I\!R^{n\times m}$ and the vector $p \in I\!R^n$ are determined by the chosen time–discretization method and detailed in Section 6.2. The matrices C and D are given by their definition in (13).

6.1 Mixed Linear Complementarity Problem (MLCP)

The time–discretized system (50) appears to be a Mixed Linear Complementarity Problem (MLCP) that we have to solve at each time–step. Let us define what is a MLCP in its general form with bounds constraints as it has been proposed in [21]:

Definition 1 (MLCP) *Given a matrix $M \in I\!R^{m\times m}$, a vector $q \in I\!R^m$ and lower and upper bounds $l, u \in \overline{I\!R}^m$, find $z \in I\!R^m$, $w, v \in I\!R^m_+$ such that*

$$\begin{cases} Mz + q = w - v \\ l \leqslant z \leqslant u \\ (z-l)^T w = 0 \\ (u-z)^T v = 0 \end{cases} \tag{51}$$

where $\overline{I\!R} = I\!R \cup \{+\infty, -\infty\}$.

Note that the problem (51) implies that

$$-(Mz+q) \in N_{[l,u]}(z). \tag{52}$$

where the notation $N_C(x)$ is used for the normal cone in the Convex Analysis sense to a convex set C at the point x. The box $[l,u] \subset I\!R^m$ is defined by the Cartesian product of the intervals $[l_i, u_i], i \in \{1,\ldots,m\}$. The normal cone to a convex set is a standard instance of a multi–valued mapping [45]. The relation (52) is equivalent to the MCP (51) if we assume that w is the positive part of $Mz+q$, that is $w = (Mz+q)^+ = max(0, Mz+q))$ and v is the negative part of $Mz+q$, that is $v = (Mz+q)^- = max(0, -(Mz+q))$.

In order to state the problem (50) as a MLCP, the variable x_{k+1} is condensed into the second line such that

$$\begin{cases} y_{k+1} = CRx_k + Cp - CSs_{k+1} + D \\ s_{k+1} \in \mathrm{Sgn}(y_{k+1}) \end{cases} \tag{53}$$

and the following variable and parameters are defined as follows

$$\begin{cases} z = s_{k+1}; \quad y_{k+1} = w - v \\ M = CS, \quad q = -(CRx_k + Cp + D) \\ l_i = -1, u_i = 1, i = 1 \ldots m. \end{cases} \tag{54}$$

Finally, the problem (50) can be recast into a MLCP by observing that

$$\begin{array}{c} s_{k+1} \in \mathrm{Sgn}(y_{k+1}) \\ \Updownarrow \\ y_{k+1} \in N_{[-1,1]^m}(s_{k+1}) \\ \Updownarrow \\ s_{k+1} \in [-1,1]^m \text{ and } \begin{cases} y_{j,k+1} = 0 \text{ if } s_{j,k+1} \in]-1,1[\\ y_{j,k+1} \leqslant 0 \text{ if } s_{j,k+1} = -1 \\ y_{j,k+1} \geqslant 0 \text{ if } s_{j,k+1} = 1 \end{cases} , j \in \{1,\ldots,m\} \end{array} \tag{55}$$

The MLCP (51) is a well-known problem in the mathematical programming theory arising for instance from the Karush–Kuhn–Tucker optimality conditions of a quadratic program or from the primal/dual optimality conditions of a linear program. The MCLP enjoys a large number of numerical algorithms and several reliable solvers have been implemented. Several families of solvers may be cited: a) extensions of Lemke and principal pivotal techniques for LCP to MLCP [48, 47, 21, 17] b) extensions of projection/splitting techniques for MLCP [24, 19] and c) semi–smooth Newton methods [40]. In this paper, the computations are done with the help the Siconos/Numerics open source Library [3] and/or the PATH solver [21]. The results of existence and uniqueness of solutions of (51) are related to the properties of M (P-properties or coherent orientations of the associated affine map (normal map) for particular cases of bounds constraints). Without entering into further details, we refer to [30, 24] for the main results. The assumptions on the matrix M drives the choice of particular solvers that can be in polynomial time rather than standard exponential time for brute force enumerative solvers.

6.2 Some Time–Discretization Methods

In this Section, the formulation of the discrete–time system (50) is related to the continuous time system (13) through a given discretization method.

Explicit Euler discretization of $f(\cdot,\cdot)$ Let us start with the explicit Euler discretization method of the term $f(t, x(t))$ as it has been given in (21). At each time step, the matrices in (50) and in the MLCP (51) can be identified as

$$R = I, p = hf(t_k, x_k), S = hB, \quad M = hCB, q = -(hCf(t_k, x_k) + Cx_k + D) \tag{56}$$

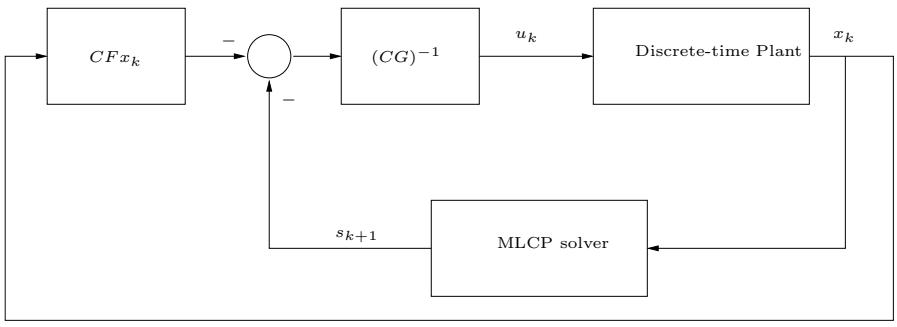


Figure 3: Control system schema with implicit Euler implementation.

Let the assumptions of Corollary 1 be satisfied with B full–column rank ($CB = B^T PB > 0$). This result ensures the existence and uniqueness of a solution of the MCLP. Furthermore, standard pivotal techniques such as Lemke's method or projection/splitting such as Projected Successive Over-Relation (PSOR) compute the solution.

Implicit Euler and θ- method In a more general way, we can choose to time–discretize the term $f(t, x(t))$ by a implicit Euler scheme or a θ-method. The main motivation for doing in this way is the higher accuracy and stability that we can obtain for such a numerical integration scheme (see [2] for an example of instability with the Explicit Euler method). Let us consider first that the mapping $f(\cdot,\cdot)$ is affine, that is $f(t, x(t)) = Fx(t) + g$. The matrices in (50) and in the MLCP (51) can be identified as

$$\begin{cases} R = (I - h\theta F)^{-1}(I + h(1-\theta)F), p = (I - h\theta F)^{-1}g, S = h(I - h\theta F)^{-1}B, \\ M = hC(I - h\theta F)^{-1}B, q = -((I - h\theta F)^{-1}(I + h(1-\theta)F)x_k + (I - h\theta F)^{-1}g + D) \end{cases} \tag{57}$$

for $\theta \in [0, 1]$. For $\theta = 0$, the explicit Euler case is retrieved. For $\theta = 1$, the implicit Euler scheme is used to discretize f. If the mapping $f(\cdot,\cdot)$ is nonlinear, a newton linearization can be invoked. In this case, the solution at each time step is sought as a limit of solutions of successive MLCPs. We refer to [2] for a detailed presentation of these developments.

Zero–Order Holder (ZOH) method The ZOH discretization presented in Section 5.2 can be also formalized into the form (50) and then (51) with

$$R = \Phi, p = 0, S = \Gamma, \quad M = C\Gamma, q = -(C\Phi x_k) \tag{58}$$

In practice, numerous methods are available to compute the ZOH discretization, *i.e.*, Φ and Γ which amounts to compute the matrix exponential and its time integral [39]. In this work, the numerical computation is performed using an explicit Runge–Kutta method with high order of accuracy and a numerical tolerance near the machine precision threshold. On the Figure 3, the control scheme is depicted showing that the controller is causal and computed form x_k.

7 Two other classes of differential inclusions

In this section, we introduce other classes of differential inclusions which extend (13). The second class of differential inclusions is:

$$\begin{cases} \dot{x}(t) \in f(t, x(t)) - \sum_{i=1}^{m}(A_i x(t) + B_i)\mathrm{sgn}(C_i x(t) + D_i), \text{ a.e. on } (0, T) \\ x(0) = x_0 \end{cases} \tag{59}$$

with $A_i \in I\!R^{n\times n}$, $B_i \in I\!R^{n\times 1}$, $C_i \in I\!R^{1\times n}$, D_i is a scalar.

The third class that we shall analyze is:

$$\begin{cases} \dot{x}(t) \in f(t, x(t)) - g(x(t))\mathrm{Sgn}(h(x(t))), \ \text{ a.e. on } (0,T) \\ \\ x(0) = x_0 \end{cases} \tag{60}$$

where $g : I\!R^n \to I\!R^{n\times m}$ and $h : I\!R^n \to I\!R^m$ are smooth functions, $\mathrm{Sgn}(h(x)) = [\mathrm{sgn}(h_1(x), ..., \mathrm{sgn}(h_m(x)]^T \in I\!R^m$.

Corollary 3 *Consider the differential inclusion in (59). Suppose that* **(ii)** *and* **(iii)** *of Assumption 1) hold. Suppose that the multivalued mappings $x \mapsto (A_i x + B_i) sgn(C_i x + D_i)$, $1 \leqslant i \leqslant m$, are hypomonotone. Then for any initial data the differential inclusion (59) has a unique solution $x : (0,T) \to I\!R^n$ that is Lipschitz continuous.*

The proof is straightforward and is omitted.

Example 6 *The mapping $I\!R \to I\!R$, $x \mapsto (x+1)sgn(x)$ is hypomonotone. Indeed $(x+1)sgn(x) = |x| + sgn(x)$ and $x \mapsto |x| + kx$ is monotone for any $k \geqslant 1$.*

Example 7 *Let k_1, k_2 be reals. The mapping $F : I\!R \to I\!R$, $x \mapsto \begin{cases} -k_1 x + 1 & if \quad x \geqslant 0 \\ -k_2 x - 1 & if \quad x \leqslant 0 \\ [-1,1] & if \quad x = 0 \end{cases}$, is hypomonotone with constant k for any $k \geqslant \max(|k_1|, |k_2|)$. Let $k_1 = -k_2 = k$. Then $F(x) = (kx+1)sgn(x)$. The linearized Stribeck friction model (with multivalued part at zero tangential velocity) [38] is hypomonotone.*

Let us state other cases where (59) fits within Proposition 1.

Lemma 4 *Let $B_i = \alpha C_i^T$ for some $\alpha > 0$, $D_i = 0$ and $Ker(C_i) \subseteq Ker(A_i)$. Then for any initial data the differential inclusion (59) has a unique solution $x : (0,T) \to I\!R^n$ that is Lipschitz continuous..*

Proof: First notice that the set-valued mapping $x \mapsto \alpha C_i^T \mathrm{sgn}(C_i x + D_i)$ is maximal monotone [46, Exercise 12.4]. Under the lemma's conditions, one sees that $x \mapsto A_i x \mathrm{sgn}(C_i x)$ is continuous on the surface $\Sigma_i = \{x \in I\!R^n \mid C_i x = 0\}$. Indeed the jump of the vector field is equal to $2A_i x = 0$ on Σ_i. Moreover it is Lipschitz continuous as it is piececewise linear. Hence Proposition 1 applies. □

Corollary 4 *Consider the differential inclusion in (60). Suppose that* **(ii)** *and* **(iii)** *of Assumption 1) hold. Suppose that the multivalued mappings $x \mapsto g_{\bullet i}(x) sgn(h_i(x))$, $1 \leqslant i \leqslant m$, are hypomonotone. Then for any initial data the differential inclusion (60) has a unique solution $x : (0,T) \to I\!R^n$ that is Lipschitz continuous.*

The proof is straightforward and is omitted.

Example 8 *The mapping $F : I\!R \to I\!R$, $x \mapsto \frac{1}{1+x^2} sgn(\arctan(x))$, is hypomonotone with any $k \geqslant \frac{9}{8\sqrt{3}}$.*

Remark 2 *As noted in [55] chattering may be due in sliding mode control appplications to the presence of parasitic dynamics. Simple modelling of these yield when inserted in (1) the differential inclusion (see [55, (7) (8)])*

$$\begin{pmatrix} \dot{x}(t) \\ \dot{x}_s(t) \\ \ddot{x}_s(t) \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ \frac{1}{\tau^2} & -\frac{1}{\tau^2} & -\frac{2}{\tau} \end{pmatrix} \begin{pmatrix} x(t) \\ x_s(t) \\ \dot{x}_s(t) \end{pmatrix} - \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} sgn(Cx(t)) \tag{61}$$

with $C = (0\ 1\ 0)$. *The relative degree of the triplet* (A, B, C) *of this system is* $r = 3$, *where* $B = (1\ 0\ 0)^T$ *and* $A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ \frac{1}{\tau^2} & -\frac{1}{\tau^2} & -\frac{2}{\tau} \end{pmatrix}$. *This system does not fit within the above classes of inclusions. Similar conclusions hold for the other form of parasitics in [55, (3) (4)]. Such parasitics may be seen as a* non collocation *issue, that is known to greatly influence the stability of systems and usually may yield instability. The mere existence and uniqueness of solutions for such relative degree 3 systems is not trivial. In [42] an example is given that possesses an infinity of absolutely continuous Filippov's solutions, but a unique so-called forward solution. One interesting question is to determine what kind of solution is approximated by the backward Euler method applied to (61) which, according to [42, Theorem 1] has a unique forward solution since its leading Markov parameter is* $CA^2B = \frac{1}{\tau^2} > 0$. *A possible solution for this non collocation issue is the observer design of Example 4.*

The differential inclusions in (59)–(60) are therefore discretized as follows:

$$\begin{cases} \frac{x_{k+1}-x_k}{h} \in f(t_k, x_k) - \rho x_k - \sum_{i=1}^m (A_i x_{k+1} + B_i)\mathrm{sgn}(C_i x_{k+1} + D_i) + \rho_i x_{k+1}, \text{ a.e. on } (0,T) \\ x(0) = x_0 \end{cases} \tag{62}$$

and

$$\begin{cases} \frac{x_{k+1}-x_k}{h} \in f(t_k, x_k) - \rho x_k - g(x_{k+1})\mathrm{Sgn}(h(x_{k+1}) + \rho x_{k+1}, \text{ a.e. on } (0,T) \\ x(0) = x_0 \end{cases} \tag{63}$$

where the ρ_i are the hypomonotonicity constants and $\sum_{i=1}^m \rho_i = \rho$. The result of Proposition 2 applies to (62) and (63).

7.1 A simple hypomonotone case

As shown in Section 2 on a simple monotone example, in practice the intersection may be computed as follows. Let us now illustrate this on the following system with hypomonotone multivalued part:

$$\dot{x}(t) \in -(x(t) + 1)\mathrm{sgn}(x(t)) + u(t) \tag{64}$$

with $x(t) \in I\!R$. We may discretize it as:

$$-(x_{k+1} - x_k - hu_k - h\rho x_k) \in h(x_{k+1} + 1)\mathrm{sgn}(x_{k+1}) + h\rho x_{k+1},\ k \geqslant 0,\ x_0 = x(0) \tag{65}$$

Notice that we may rewrite (65) as

$$0 \in x_{k+1} - (x_k + hu_k + h\rho x_k) + h(x_{k+1} + 1)\mathrm{sgn}(x_{k+1}) + h\rho x_{k+1} \tag{66}$$

Let us denote the mapping in the right-hand-side of (66) as $F(x_{k+1})$. The set-valued mapping $F(\cdot)$ is *strongly monotone* [24, Definition 2.3.1] for all $\rho \geqslant 1$. It follows from [24, Theorem 2.3.3] that the *generalized equation* $0 \in F(x_{k+1})$ has a unique solution.

For $u(t) = 0$, the following Lemma extends the Lemma 1.

Lemma 5 *For all* $1 > h > 0$ *and* $x_0 \in I\!R$, *there exists* k_0 *such that* $x_{k_0+n} = 0$ *and* $\dfrac{x_{k_0+n+1} - x_{k_0+n}}{h} = 0$ *for all* $n \geqslant 1$.

Proof: If $x_0 \in [-h, h]$, then $k_0 = 0$. Otherwise, for $k < k_0$ and $x_k \notin [-h, h]$, the solution is given by :

$$x_{k+1} = \frac{x_k - h\mathrm{sgn}(x_k)}{1 + h\mathrm{sgn}(x_k)}; s_{k+1} = \mathrm{sgn}(x_k) \tag{67}$$

From the solution (67), the step k_0 for which $x_{k_0} \in [-h, h]$ can be easily found. Let us now consider that $x_{k_0} \in [-h, h]$. The only possible solution for

$$\begin{cases} x_{k_0+1} - x_{k_0} = -h(x_{k_0+1} + 1)s_{k_0+1} \\ s_{k_0+1} \in \operatorname{sgn}(x_{k_0+1}) \end{cases} \tag{68}$$

is $x_{k_0+1} = 0$ and $s_{k_0+1} = \dfrac{x_{k_0}}{h}$. For the next iteration, we have to solve

$$\begin{cases} x_{k_0+2} = -hs_{k_0+2} \\ s_{k_0+2} \in \operatorname{sgn}(x_{k_0+2}) \end{cases} \tag{69}$$

and we obtain $x_{k_0+2} = 0$ and $s_{k_0+2} = 0$. The same holds for all x_{k_0+n},s_{k_0+n}, $n \geqslant 3$, redoing the same reasoning. Clearly then the terms $\dfrac{x_{k_0+n+1} - x_{k_0+n}}{h}$ approximating the derivative, are zero for any $h > 0$. □

We conclude that in this case also the system and its derivative are correctly approximated at the zero value on the sliding surface. There is no spurious oscillation around the switching surface.

8 Detailed Implementation of Implicit Euler Discretization of the general case (60)

This section is devoted to the implementation and the study of the numerical algorithm. The interval of integration is $[0, T]$, $T > 0$, and a grid $t_0 = 0$, $t_{k+1} = t_k + h$, $k \geqslant 0$, $t_N = T$ is constructed. The approximation of a function $f(\cdot)$ on $[0, T]$ is denoted as $f^N(\cdot)$, and is a piecewise constant function, constant on the intervals $[t_k, t_{k+1})$. We denote $f^N(t_k)$ as f_k. The time-step is $h > 0$.

8.1 Time–discretization

Starting from (60), let us introduce a new notation,

$$\begin{aligned} \dot{x}(t) &= f(x(t), t) - g(x(t))s(t) \\ y(t) &= h(x(t)) \\ s(t) &\in \operatorname{Sgn}(y(t)) \end{aligned} \tag{70}$$

where $s(t) \in I\!R^m$ and $y(t) \in I\!R^m$ are complementary variables related through the $\operatorname{Sgn}(\cdot)$ multi–valued mapping. According to the class of systems (13), (59) or (60) that we are studying the functions $f(\cdot)$ and $g(\cdot)$ are defined either in a fully nonlinear framework or by affine functions. We present the time-discretization in its full generality and specialize the algorithms for each case in Section 8.4.

Let us now proceed with the time discretization of (70) by a fully implicit scheme :

$$\begin{aligned} x_{k+1} &= x_k + hf(x_{k+\theta}, t_{k+\theta}) - hg(x_{k+\gamma})s_{k+1} \\ y_{k+1} &= h(x_{k+1}) \\ s_{k+1} &\in \operatorname{Sgn}(y_{k+1}) \end{aligned} \tag{71}$$

where $x_{k+\theta} = \theta x_{k+1} + (1-\theta)x_k$, $x_{k+\gamma} = \gamma x_{k+1} + (1-\gamma)x_k$, and $t_{k+\theta} = \theta t_{k+1} + (1-\theta)t_{k+1}$, with $\theta = [0, 1]$ and $\gamma \in [0, 1]$. As in [1], we call the problem (71) the "one–step nonsmooth problem".

This time-discretization is slightly more general than a standard implicit Euler scheme. The main discrepancy lies in the choice of a θ-method to integrate the nonlinear term. For $\theta = 0$, we

retrieve the explicit integration of the smooth and single valued term $f(\cdot)$. Moreover for $\gamma = 0$, the term $g(\cdot)$ is explicitly evaluated. The flexibility in the choice of θ and γ allows the user to improve and control the accuracy, the stability and the numerical damping of the proposed method. For instance, if the smooth dynamics given by $f(\cdot)$ is stiff, or if we have to use large step sizes for practical reasons, the choice of $\theta > 1/2$ offers better stability properties with respect to h.

8.2 Mixed Complementarity Problem

The so-called "one–step nonsmooth problem" (71) appears to be a Mixed Complementarity Problem (MCP) that we have to solve at each time–step. Let us define what is a MCP :

Definition 2 (MCP) *Given a function $f : I\!R^q \to I\!R^q$ and lower and upper bounds $l, u \in \bar{I\!R}^q$, find $z \in I\!R^q$, $w, v \in I\!R^q_+$ such that*

$$\begin{cases} F(z) = w - v \\ l \leqslant z \leqslant u \\ (z-l)^T w = 0 \\ (u-z)^T v = 0 \end{cases} \tag{72}$$

where $\bar{I\!R} = I\!R \cup \{+\infty, -\infty\}$.

Note that the problem (72) implies that

$$-F(z) \in N_{[l,u]}(z). \tag{73}$$

The relation (73) is equivalent to the MCP (72) if we assume that w is the positive part of $F(z)$, that is $w = F^+(z) = max(0, F(z))$ and v is the negative part of $F(z)$, that is $v = F^-(z) = max(0, -F(z))$.

The One–step nonsmooth problem as a MCP Let us define the MCP by

$$z = \begin{bmatrix} x_{k+1} \\ s_{k+1} \end{bmatrix}$$

$$F(z) = \begin{bmatrix} x_{k+1} - x_k - hf(x_{k+\theta}, t_{k+\theta}) + hg(x_{k+\gamma})s_{k+1} \\ -h(x_{k+1}) \end{bmatrix} \tag{74}$$

$$l_i = \begin{cases} -\infty, i = 1 \dots n \\ -1, i = n+1 \dots m \end{cases} \quad , \quad u_i = \begin{cases} +\infty, i = 1 \dots n \\ +1, i = n+1 \dots m \end{cases}$$

If z solves the MCP (74), the bounds u and l and the condition $(z-l)^T w = 0, (u-z)^T v = 0$ imply that

$$w_i = \begin{cases} 0, i = 1 \dots n \\ y_i^- \geqslant 0, i = n+1 \dots m \end{cases} \quad , \quad v_i = \begin{cases} 0, i = 1 \dots n \\ y_i^+ \geqslant 0, i = n+1 \dots m \end{cases} \tag{75}$$

The MCP is then given by

$$\begin{bmatrix} x_{k+1} - x_k - hf(x_{k+\theta}) + hg(x_{k+\gamma})s_{k+1} \\ -h(x_{k+1}) \end{bmatrix} = \begin{bmatrix} 0 \\ y^-(x_{k+1}) - y^+(x_{k+1}) = -y(x_{k+1}) \end{bmatrix}$$
$$-1 < s_{k+1} < 1$$
$$(s_{k+1}+1)^T y^-(x_{k+1}) = 0$$
$$(1 - s_{k+1})^T y^+(x_{k+1}) = 0 \tag{76}$$

It is clear that the problem (71) is equivalent to the MCP defined by (74). The results of existence and uniqueness of solution of (72) or equivalently (73) are related to the monotonicity properties

of F (P-properties or coherent orientations of the associated affine map for particular cases of bounds and affine function $F(\cdot)$. Without entering into further details, we refer to [30, 24] for the main results.

Numerical Solvers The MCP (72) can be solved by a large family of solvers based on Newton–type Methods and interior-points techniques. We refer to [11] for a comparison of several solvers based on Newton's method. The numerical implementation of the MCP solvers are often based on the computation of the Jacobian matrix of the function $F(\cdot)$ with respect to z. The Jacobian matrix is explicitly given in the case defined in (74) by

$$\nabla_z F(z) = \begin{bmatrix} I - h\theta\nabla_x f(x_{k+\theta}, t_{k+\theta}) + h\gamma\nabla_x g(x_{k+\gamma}) \bar{\otimes} s_{k+1} & hg(x_{k+\gamma}) \\ \nabla_x h(x_{k+1}) & 0 \end{bmatrix}, \tag{77}$$

where $\bar{\otimes}$ denotes the simple contracted tensor product and the third–order tensor $\nabla_x g(x)$ is the Jacobian of g with the respect x given by the following component:

$$(\nabla_x g(x))_{klp} = \frac{\partial g_{kl}(x)}{\partial x_p}. \tag{78}$$

8.3 Newton's linearization and Mixed Linear Complementarity Problems

Due to the fact that two of the systems that are studied in this paper involve affine functions $f(\cdot)$ and $g(\cdot)$, we propose to solve the "one–step nonsmooth problem" (71) by performing an external Newton linearization, which yields a Mixed Linear Complementarity Problems (MLCP).

Newton's linearization The first line of the problem (71) can be written under the form of a residue $\mathcal{R}$ depending only on x_{k+1} and s_{k+1} such that

$$\mathcal{R}(x_{k+1}, s_{k+1}) = 0 \tag{79}$$

with $\mathcal{R}(x, s) = x - x_k - hf(\theta x + (1-\theta)x_k, t_{k+\theta}) + hg(\gamma x + (1-\gamma)x_k)s$. The solution of this system of nonlinear equations is sought as a limit of the sequence $\{x^\alpha_{k+1}, s^\alpha_{k+1}\}_{\alpha\in I\!N}$ such that

$$\begin{cases} x^0_{k+1} = x_k \\ \\ \mathcal{R}_L(x^{\alpha+1}_{k+1}, s^{\alpha+1}_{k+1}) = x^\alpha_{k+1} - x_k - hf(x^\alpha_{k+\theta}) + \nabla_x\mathcal{R}(x^\alpha_{k+1}, s^\alpha_{k+1})(x^{\alpha+1}_{k+1} - x^\alpha_{k+1}) + hg(x^\alpha_{k+\gamma})s^{\alpha+1}_{k+1} = 0 \end{cases} \tag{80}$$

The computation of the Jacobian of $\mathcal{R}$ with respect to x, denoted by $M(x, s)$ leads to

$$M(x, s) = \nabla_x\mathcal{R}(x, s) = I - h\theta\nabla_x f(\theta x + (1-\theta)x_k, t_{k+\theta}) + h\gamma\nabla_x g(\gamma x + (1-\gamma)x_k) \bar{\otimes} s. \tag{81}$$

At each time–step, we have to solve the following linearized problem,

$$x^\alpha_{k+1} - x_k - hf(x^\alpha_{k+\theta}, t_{k+\theta}) + M(x^\alpha_{k+1}, s^\alpha_{k+1})(x^{\alpha+1}_{k+1} - x^\alpha_{k+1}) + hg(x^\alpha_{k+\gamma})s^{\alpha+1}_{k+1} = 0, \tag{82}$$

that is

$$x^{\alpha+1}_{k+1} = x^\alpha_{k+1} + M^{-1}(x^\alpha_{k+1}, s^\alpha_{k+1}) \left[x_k - x^\alpha_{k+1} + hf(x^\alpha_{k+\theta}, t_{k+\theta}) - hg(x^\alpha_{k+\gamma})s^{\alpha+1}_{k+1}\right]. \tag{83}$$

The matrix M is clearly non singular for small h. The same operation is performed with the second equation of (71) leading to the following linearized equation

$$y^{\alpha+1}_{k+1} = y^\alpha_{k+1} + \nabla_x h(x^\alpha_{k+1}) \left[x^{\alpha+1}_{k+1} - x^\alpha_{k+1}\right] \tag{84}$$

Inserting (83), we get the following linear relation between $y^{\alpha+1}_{k+1}$ and $s^{\alpha+1}_{k+1}$,

$$y^{\alpha+1}_{k+1} = y^\alpha_{k+1} + \nabla_x h(x^\alpha_{k+1}) \left[M^{-1}(x^\alpha_{k+1}, s^\alpha_{k+1})(x_k - x^\alpha_{k+1} + hf(x^\alpha_{k+\theta}, t_{k+\theta}) - hg(x^\alpha_{k+\gamma})s^{\alpha+1}_{k+1})\right] \tag{85}$$

Mixed linear complementarity problem (MLCP) To summarize, the problem to be solved in each Newton iteration is:

$$\begin{cases} y_{k+1}^{\alpha+1} = -W_{k+1}^{\alpha+1} s_{k+1}^{\alpha+1} + b_{k+1}^{\alpha+1} \\ s_{k+1}^{\alpha+1} \in \mathrm{Sgn}(y_{k+1}^{\alpha+1}) \end{cases} \tag{86}$$

with $W \in I\!R^{m\times m}$ and $b \in I\!R^m$ defined by

$$\begin{aligned} W_{k+1}^{\alpha+1} &= h\nabla_x h(x_{k+1}^{\alpha}) M^{-1}(x_{k+1}^{\alpha}, s_{k+1}^{\alpha}) g(x_{k+\gamma}^{\alpha}) \\ b_{k+1}^{\alpha+1} &= y_{k+1}^{\alpha} + \nabla_x h(x_{k+1}^{\alpha}) \left[M^{-1}(x_{k+1}^{\alpha}, s_{k+1}^{\alpha})(x_k - x_{k+1}^{\alpha} + hf(x_{k+\theta}^{\alpha}, t_{k+\theta})) \right] \end{aligned} \tag{87}$$

The problem (86) is equivalent to a MLCP which can be solved under suitable assumptions by many linear complementarity solvers such as pivoting techniques, interior point techniques and splitting/projection strategies. The reformulation into a standard MLCP follows the same line as for the MCP in the previous section. One obtains,

$$\begin{cases} y_{k+1}^{\alpha+1,+} - y_{k+1}^{\alpha+1} = -W_{k+1}^{\alpha+1} s_{k+1}^{\alpha+1} + b_{k+1}^{\alpha+1} \\ 0 \leqslant (s_{k+1}^{\alpha+1} + 1) \perp y_{k+1}^{\alpha+1,-} \geqslant 0 \\ 0 \leqslant (1 - s_{k+1}^{\alpha+1}) \perp y_{k+1}^{\alpha+1,+} \geqslant 0 \end{cases} \tag{88}$$

As for the MCP, there exists numerous methods to numerically solve MLCP. In the worst case when the matrix $W_{k+1}^{\alpha+1}$ has no special properties, the MCLP can be always solved by enumerative solvers for which various implementations can be found. With some positiveness properties [24], standard methods for LCP[19] can be straightforwardly extended. Among these methods, we can cite the family of projection/splitting methods, interior point methods and semi-smooth Newton methods (see [1] for an overview).

8.4 The special cases of the affine systems

In this section, we specify the time–discretization to the two other classes of systems (13) and (59) and for particular value of θ and γ.

8.4.1 Time–discretization of the system (13)

For the system (13), the function $g(x)$ is reduced to the matrix B and the function $h(x)$ is affine, that is $h(x) = Cx + D$. The matrix $W_{k+1}^{\alpha+1}$ and $b_{k+1}^{\alpha+1}$ are then given by

$$\begin{cases} W_{k+1}^{\alpha+1} = hCM^{-1}(x_{k+1}^{\alpha}, s_{k+1}^{\alpha}, t_{k+1})B \\ b_{k+1}^{\alpha+1} = y_{k+1}^{\alpha} + C\left[M^{-1}(x_{k+1}^{\alpha}, t_{k+1})(x_k - x_{k+1}^{\alpha} + hf(x_{k+\theta}^{\alpha}, t_{k+\theta})) \right] \end{cases} \tag{89}$$

with

$$M(x,t) = I - h\theta\nabla_x f(\theta x + (1-\theta)x_k, \theta t + (1-\theta)t_k) \tag{90}$$

If $CB > 0$ then the matrix $W_{k+1}^{\alpha+1}$ is also positive definite for sufficiently small h. This result ensures the existence and uniqueness of a solution of the MCLP. Furthermore, standard pivoting techniques such as Lemke's method or projection/splitting such as Projected Successive Over-Relation (PSOR) compute the solution.

Semi-implicit discretization with $\theta = 0$ The matrix M is then reduced to

$$M(x,t) = I \tag{91}$$

and

$$\begin{aligned} W &= hCB \\ b_{k+1} &= y_k + hCf(x_k, t_k) \end{aligned} \tag{92}$$

In this particular case, there is no need to perform some Newton iterations because the system to be solved at each time–step is linear. Furthermore, the MLCP has a solution for any $h > 0$ under the assumptions that $CB > 0$.

Fully implicit discretization with an affine function The same conclusion can be drawn if $\nabla_x f(\theta x + (1-\theta)x_k, \theta t + (1-\theta)t_k)$ is equal to a constant matrix E that is when $f(\cdot)$ is linear time-invariant and given by $f(x,t) = Ex(t) + F$. In this case, the matrix M reduces to

$$M(x,t) = I - h\theta E \tag{93}$$

and

$$\begin{aligned} W &= hC(I - h\theta E)^{-1}B \\ b_{k+1} &= y_k + hC(I - h\theta E)^{-1}\left[Ex_k + a\right] \end{aligned} \tag{94}$$

8.4.2 Time–discretization of the system (59)

For the system (59), we recall that $g(\cdot)$ and $h(\cdot)$ are given by

$$\begin{array}{rcl} g(x) & = & [g_{\bullet j}(x) = A_j x + B_j, j = 1 \dots m] \in I\!R^{n\times m}, \\ h(x) & = & [h_i(x) = C_i x + D_i, i = 1 \dots m] \in I\!R^{m}. \end{array} \tag{95}$$

The components of $g(\cdot)$ can be explicitly expressed by

$$g_{kl}(x) = \sum_{p=1}^{n} A_{l,kp} x_p + B_l, \tag{96}$$

and therefore, the Jacobian of $g(\cdot)$ is given by

$$(\nabla_x g(x))_{klp} = \frac{\partial g_{kl}(x)}{\partial x_p} = A_{l,kp}. \tag{97}$$

The Jacobian of h takes the following simple form:

$$\nabla h(x) \quad = \quad C = [C_i, i = 1 \dots m] \in I\!R^{m\times n}. \tag{98}$$

After the newton linearization, we have to solve at each Newton's loop the MLCP (86) with

$$\begin{aligned} W_{k+1}^{\alpha+1} &= hCM^{-1}(x_{k+1}^{\alpha}, s_{k+1}^{\alpha}, t_{k+1})g(x_{k+\gamma}^{\alpha}) \\ b_{k+1}^{\alpha+1} &= y_{k+1}^{\alpha} + C\left[M^{-1}(x_{k+1}^{\alpha}, t_{k+1})(x_k - x_{k+1}^{\alpha} + hf(x_{k+\theta}^{\alpha}, t_{k+\theta}))\right] \end{aligned} \tag{99}$$

Semi–implicit discretization with $\theta = \gamma = 0$ If γ and θ vanish, the residue $\mathcal{R}$ given by

$$\mathcal{R}(x,s) = x - x_k - hf(x_k, t_k) + hg(x_k)s \tag{100}$$

is linear in x and s. In this particular case, there is no need to perform Newton's iterations. The MCLP defined by (99) can be simplified to

$$W_{k+1} = hCg(x_k)b_{k+1} = y_k + C\left[hf(x_k, t_k))\right] \tag{101}$$

Algorithm 1 Implicit Euler time-discretization with a generic MCP solver

Require: System definition: $\mathsf{f}, \mathsf{g}, \mathsf{h}$
Require: $\mathsf{x}(0)$ the initial condition
Require: t_0, T time–integration interval
Require: h time–step
Require: θ, γ numerical integration parameters
Ensure: $(\{\mathsf{x}_k\}, \{\mathsf{s}_k\}, \{\mathsf{y}_k\}), \mathsf{k} \in \{1, 2, \ldots\}$

$\mathsf{k} \leftarrow 0; \quad \mathsf{x}_0 \leftarrow \mathsf{x}(0); \quad \mathsf{y}_0 \leftarrow \mathsf{y}(0) = \mathsf{h}(\mathsf{x}(0)); \quad \mathsf{tau}_0 \leftarrow 0$
// *Time integration loop*
while $\mathsf{t}_k < \mathsf{T}$ **do**
  Solve the MCP (76) for $\mathsf{x}_{k+1}, \mathsf{s}_{k+1}, \mathsf{y}_{k+1}$ with F, l and u given by (74) and the Jacobian $\nabla_z F(z)$ given by (77).
  //*Update*
  $\mathsf{x}_k \leftarrow \mathsf{x}_{k+1}; \quad \mathsf{s}_k \leftarrow \mathsf{s}_{k+1}; \quad \mathsf{y}_k \leftarrow \mathsf{y}_{k+1}$
  //*time iteration*
  $\mathsf{t}_k \leftarrow \mathsf{t}_{k+1}; \quad \mathsf{k} \leftarrow \mathsf{k} + 1$
end while

Algorithm 2 Implicit Euler time-discretization with an external Newton loop and a MLCP solver

Require: System definition: $\mathsf{f}, \mathsf{g}, \mathsf{h}$
Require: $\mathsf{x}(0)$ the initial condition
Require: t_0, T time–integration interval
Require: h time–step
Require: θ, γ numerical integration parameters
Require: ε Newton's method tolerance
Ensure: $(\{\mathsf{x}_k\}, \{\mathsf{s}_k\}, \{\mathsf{y}_k\}), \mathsf{k} \in \{1, 2, \ldots\}$

$\mathsf{k} \leftarrow 0; \quad \mathsf{x}_0 \leftarrow \mathsf{x}(0); \quad \mathsf{y}_0 \leftarrow \mathsf{y}(0) = \mathsf{h}(\mathsf{x}(0)); \quad \mathsf{tau}_0 \leftarrow 0$

// *Time integration loop*
while $\mathsf{t}_k < \mathsf{T}$ **do**
  $\alpha \leftarrow 0; \quad \mathsf{x}^0_{k+1} \leftarrow \mathsf{x}^0_k; \quad \mathsf{s}^0_{k+1} \leftarrow \mathsf{s}^0_k; \quad \mathsf{y}^0_{k+1} \leftarrow \mathsf{y}^0_k$
  //*Newton's loop*
  while $\|\mathcal{R}(\mathsf{x}^\alpha_{k+1}, \mathsf{s}^\alpha_{k+1})\| > \varepsilon$ **do**
    $\mathsf{M}^{-1}(\mathsf{x}^\alpha_{k+1}, \mathsf{s}^\alpha_{k+1}) \leftarrow (\mathsf{I} - \mathsf{h}\theta\nabla_x \mathsf{f}(\mathsf{x}^\alpha_{k+\theta}, \mathsf{t}_{k+1}) - \mathsf{h}\gamma\nabla_x \mathsf{g}(\mathsf{x}^\alpha_{k+\theta}) \bar{\otimes} \mathsf{s}^\alpha_{k+1})^{-1}.$
    $\mathsf{W}^{\alpha+1}_{k+1} \leftarrow \mathsf{h}\nabla_x \mathsf{h}(\mathsf{x}^\alpha_{k+1})\mathsf{M}^{-1}(\mathsf{s}^\alpha_{k+1}, \mathsf{s}^\alpha_{k+1})\mathsf{g}(\mathsf{x}^\alpha_{k+1})$
    $\mathsf{b}^{\alpha+1}_{k+1} \leftarrow \mathsf{y}^\alpha_{k+1} + \nabla_x \mathsf{h}(\mathsf{x}^\alpha_{k+1})\mathsf{M}^{-1}(\mathsf{s}^\alpha_{k+1}) \left[\mathsf{x}_k - \mathsf{x}^\alpha_{k+1} + \mathsf{h}\mathsf{f}(\mathsf{x}^\alpha_{k+1})\right]$

    Solve the MLCP (88) for $\mathsf{y}^{\alpha+1}_{k+1}, \mathsf{s}^{\alpha+1}_{k+1}$

    $\mathsf{s}^\alpha_{k+1} \leftarrow \mathsf{s}^{\alpha+1}_{k+1}; \quad \mathsf{y}^\alpha_{k+1} \leftarrow \mathsf{x}^{\alpha+1}_{k+1}$
    $\mathsf{x}^\alpha_{k+1} \leftarrow \mathsf{x}^\alpha_{k+1} + \mathsf{M}^{-1}(\mathsf{s}^\alpha_{k+1}) \left[\mathsf{x}_k - \mathsf{x}^\alpha_{k+1} + \mathsf{h}\mathsf{f}(\mathsf{x}^\alpha_{k+1}) + \mathsf{h}\mathsf{g}(\mathsf{x}^\alpha_{k+1})\mathsf{s}^{\alpha+1}_{k+1}\right]$
    $\alpha \leftarrow \alpha + 1$
  end while
  //*Update*
  $\mathsf{x}_{k+1} \leftarrow \mathsf{x}^\alpha_{k+1}; \quad \mathsf{s}_{k+1} \leftarrow \mathsf{s}^{\alpha+1}_{k+1}; \quad \mathsf{y}_{k+1} \leftarrow \mathsf{y}^{\alpha+1}_{k+1}$
  //*time iteration*
  $\mathsf{t}_k \leftarrow \mathsf{t}_{k+1}; \quad \mathsf{k} \leftarrow \mathsf{k} + 1$
end while

8.5 Algorithms

We propose in this section two algorithms to sum-up the numerical implementation of the implicit Euler time–stepping scheme. The Algorithm 1 describes the implementation with a generic MCP solver and the Algorithm 2 describes the numerical implementation of the algorithm with an external Newton linearization and a MCLP solver.

In the case of the system (13) with an affine function $f(\cdot)$ or $\theta = 0$, the MLCP matrix W can be computed before the beginning of the time loop, saving a lot of computing effort. In the case of the system (59) with $\theta = \gamma = 0$, the MLCP matrix W can be computed before the beginning of the Newton loop.

9 Numerical experiments

Let us illustrate the above developments with numerical integrations performed with the SICONOS software platform of the INRIA[2] [1, 3] which is designed for the simulation of multivalued nonsmooth systems.

9.1 Chattering free stabilization

Let us consider the following continuous–time closed loop system from [26] given by

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -c_1 \end{bmatrix} x - \begin{bmatrix} 0 \\ \alpha \end{bmatrix} \operatorname{sgn}(\begin{bmatrix} c_1 & 1 \end{bmatrix} x). \tag{102}$$

As it is shown in [26] the trajectories obtained by an explicit Euler discretization exhibit spurious oscillations which are described by period-2 cycle around the sliding manifold. On Figure 4, the trajectories obtained by implicit discretization are shown using the implicit Euler discretization with $h = 1$, $h = 0.3$, $h = 0.1$ and $h = 0.01$ and with $c_1 = 1$ and $\alpha = 1$. As it has been predicted by theoretical discussions of Section 4, the sliding manifold is reached in finite time and without any chattering. Indeed, the matrix $CB = \alpha = 1$ satisfies the assumptions of Lemma 3. Note that the algorithm is also very robust in the sense that the simulation can be performed with relatively large time–steps.

9.2 Example 3: Multiple sliding surfaces

Let us consider the example 3. The system can be defined in the form (13) with

$$B = \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix}, \quad D = 0, \quad f(x(t), t) = 0 \tag{103}$$

This example illustrates Lemma 3 since $CB = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}$. The results displayed on Figure 5 show that the system reaches firstly the sliding surface $2x_2 + x_1 = 0$ without any chattering, The system then slides on the surface up to reaching the second sliding surface $2x_1 - x_2 = 0$ and comes to rest at the origin.

9.3 Extensions to ZOH discretized systems

The extension to ZOH discretized systems is illustrated on a first example taken from [27]. In the notation of Section 5.2, the LTI system with an ECB-SMC controller is defined by the following data,

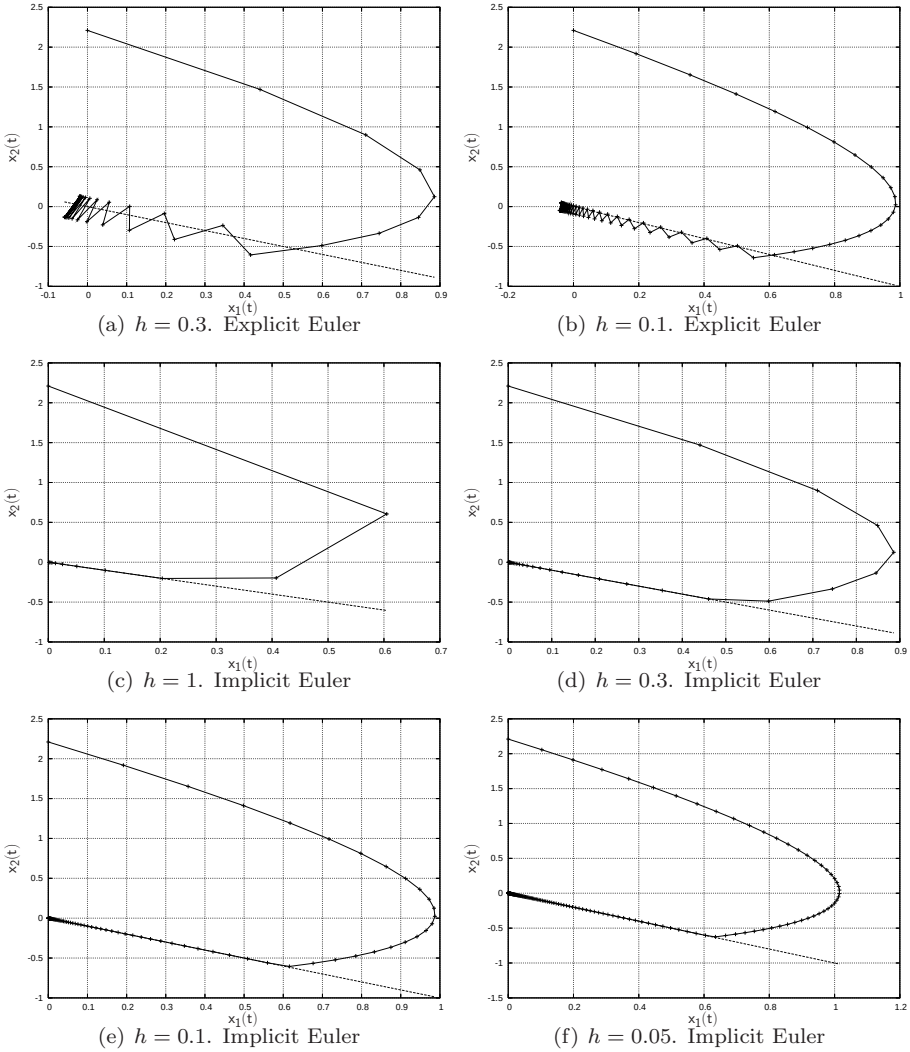
$$F = \begin{bmatrix} 0 & 1 \\ -a_1 & -a_2 \end{bmatrix}, \quad G = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C = \begin{bmatrix} c_1 & 1 \end{bmatrix}. \tag{104}$$

Starting from the initial data, $x_0 = [0.55, 0, 55]^T$, Galias and Yu [27] have shown that the Explicit ZOH discretization of the system with $a_1 = -2$, $a_2 = 2$, $c_1 = 1$ and $h = 0.3$ exhibits a period–2 orbit. The results are reproduced on Figure 6(a). On Figure 6(b), the Implicit ZOH discretization as proposed in Section 5.2 is free of chattering. On Figure 7, a comparison is given between the ZOH and the Euler discretization of the vector field $f(\cdot, \cdot)$. Another example taken from [57] in the MIMO case is given by the following parameters,

$$F = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ -1 & -3 & 1 \end{bmatrix}, \quad G = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}. \tag{105}$$

Similar results are depicted on Figure 8.

[2] `http://siconos.gforge.inria.fr/`

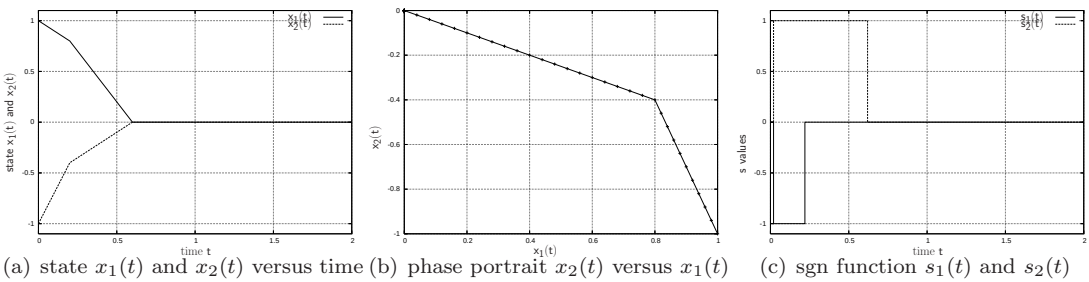


(a) $h = 0.3$. Explicit Euler (b) $h = 0.1$. Explicit Euler

(c) $h = 1$. Implicit Euler (d) $h = 0.3$. Implicit Euler

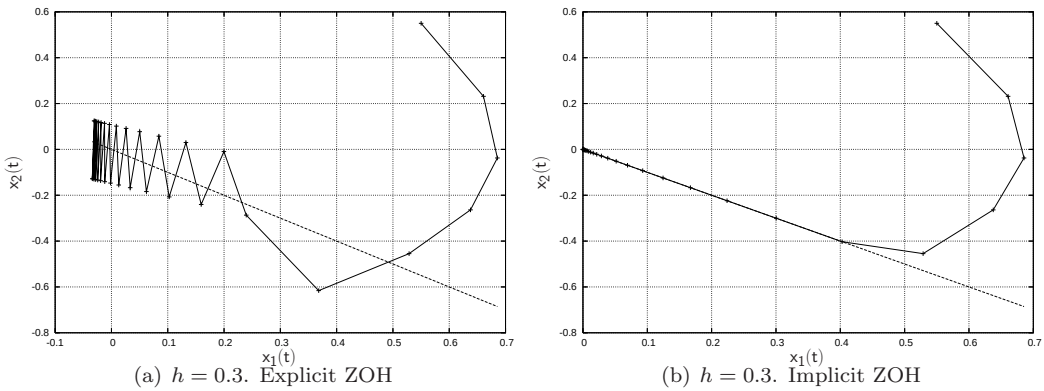
(e) $h = 0.1$. Implicit Euler (f) $h = 0.05$. Implicit Euler

Figure 4: Equivalent control based SMC, $c_1 = 1, \alpha = 1$ and $x_0 = [0, 2.21]^T$. State $x_1(t)$ versus $x_2(t)$.



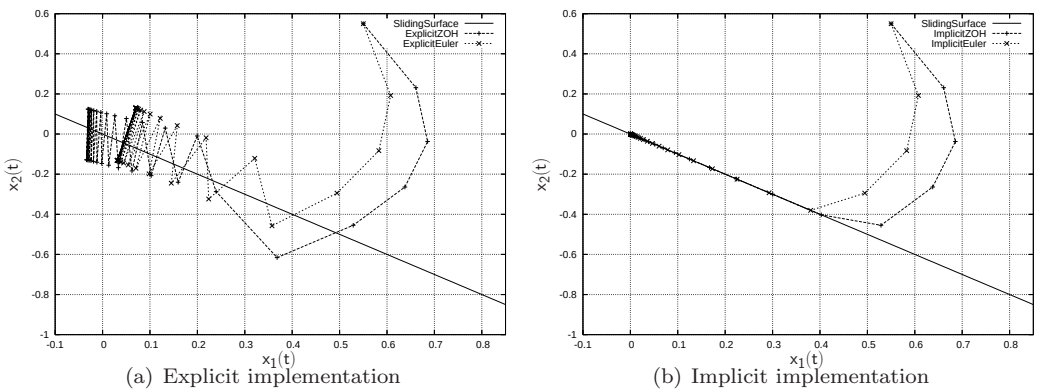
(a) state $x_1(t)$ and $x_2(t)$ versus time (b) phase portrait $x_2(t)$ versus $x_1(t)$ (c) sgn function $s_1(t)$ and $s_2(t)$

Figure 5: Multiple Sliding surface. $h = 0.02$, $x(0) = [1.0, -1.0]^T$



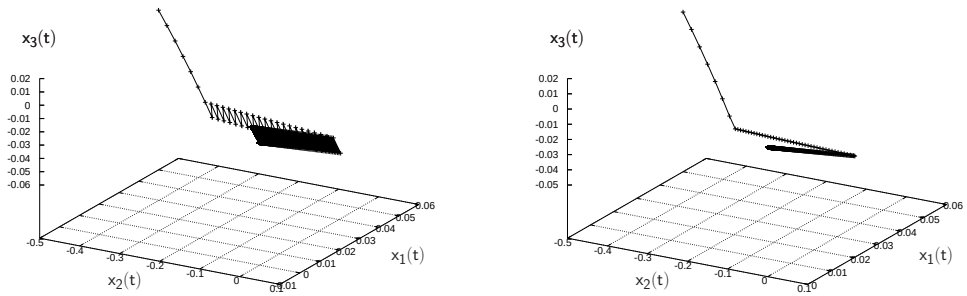
(a) $h = 0.3$. Explicit ZOH (b) $h = 0.3$. Implicit ZOH

Figure 6: Equivalent control based SMC, $a_1 = -2$, $a_2 = 2$, $c_1 = 1$ and $h = 0.3$. $x_0 = [0.55, 0, 55]^T$ State $x_1(t)$ versus $x_2(t)$.



(a) Explicit implementation (b) Implicit implementation

Figure 7: Comparison of Euler and ZOH discretizations of ECB-SMC system, $a_1 = -2$, $a_2 = 2$, $c_1 = 1$ and $h = 0.3$. $x_0 = [0.55, 0, 55]^T$ State $x_1(t)$ versus $x_2(t)$.



(a) $h = 0.3$. Explicit ZOH (b) $h = 0.3$. Implicit ZOH

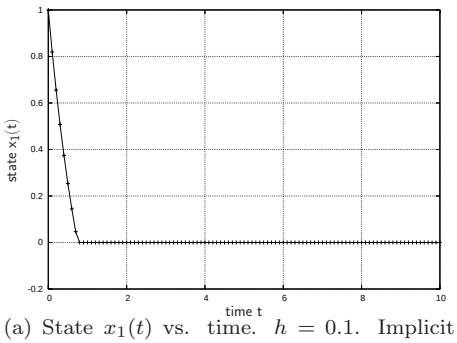
Figure 8: Equivalent control based SMC, Trajectory with initial point $x_0 = [0.05, -0.5, 0.02]$

9.4 Lyapunov-based robust control

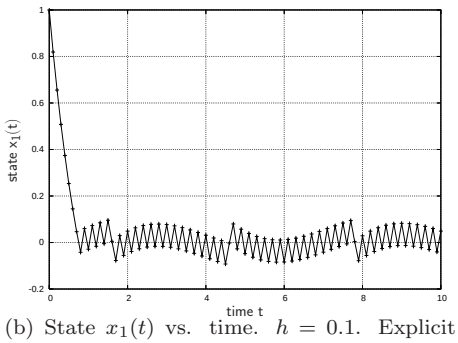
We propose in this section to give an numerical example which fits with the example (18) of a Lyapunov-based discontinuous robust control. Let us consider the following system

$$\dot{x}(t) = -x(t) - u(t) + \gamma(t) \tag{106}$$

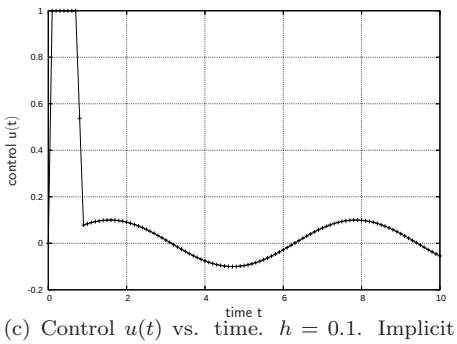
with $\gamma(t) = \alpha \sin(t)$ and $u(t) = \mathrm{sgn}(x(t))$. Is is obvious that, as expected, the implicit method yields a smooth stabilization at $x = 0$ whereas the explicit Euler has significant chattering. Figure 9(c) illustrates the fact that the controller varies inside the multivalued part of the sgn function in order to assure the existence of an equilibrium point.



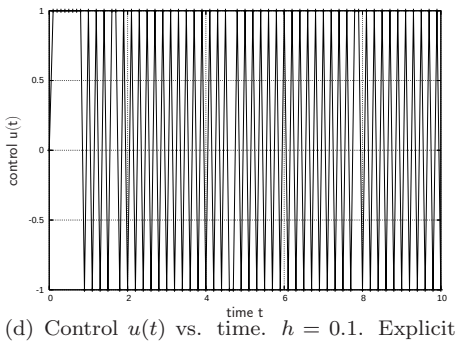
(a) State $x_1(t)$ vs. time. $h = 0.1$. Implicit Euler



(b) State $x_1(t)$ vs. time. $h = 0.1$. Explicit Euler



(c) Control $u(t)$ vs. time. $h = 0.1$. Implicit Euler



(d) Control $u(t)$ vs. time. $h = 0.1$. Explicit Euler

Figure 9: Lyapunov-based discontinuous robust control. $h = 0.1$ $\alpha = 0.1$

9.5 The Filippov example

Example 9 *Let us consider now the well known Filippov example which can be defined in the form (13) with*

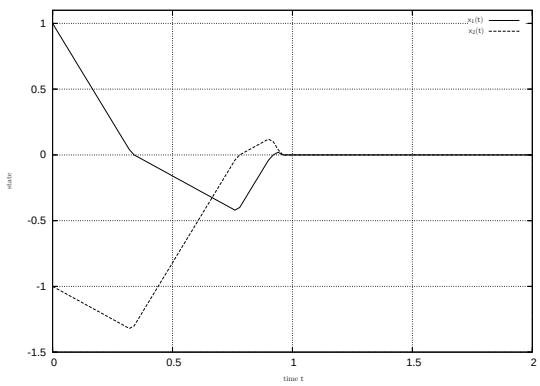
$$B = \begin{bmatrix} 1 & -2 \\ 2 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad D = 0, \quad f(x(t), t) = 0. \tag{107}$$

The trajectories may slide on the codimension 2 surface given by $Cx = 0$, that $x = 0$.

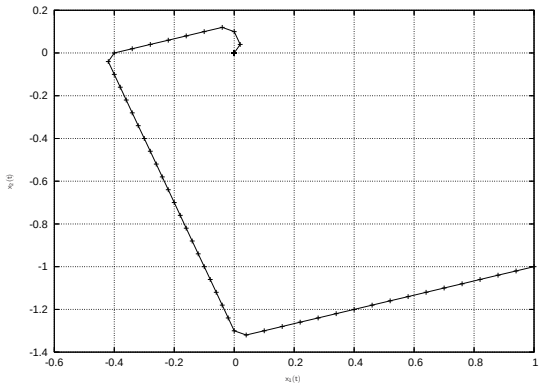
The results displayed on Figure 9.5 show that the system reaches the origin without any chattering. The sufficient conditions of the Lemma 3 are not satisfied when seems to indicated that these conditions has to be improved.

9.6 Example 4: Observer based SMC

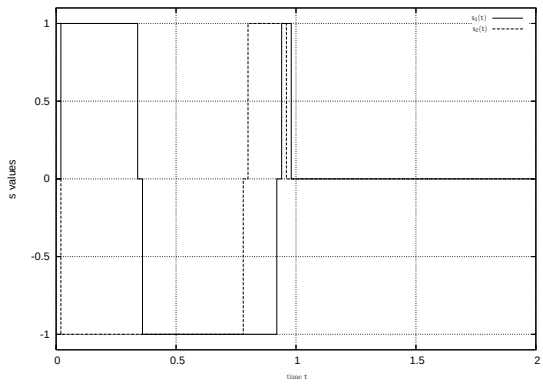
Let us illustrate the performance of our implementation on the observer based SMC described by the Example 4. The dynamics is given by (17) with $k = 1$ and $\tau = 0.001$. The initial conditions



(a) state $x_1(t)$ and $x_2(t)$ versus time



(b) phase portrait $x_2(t)$ versus $x_1(t)$



(c) sgn function $s_1(t)$ and $s_2(t)$

Figure 10: Multiple Sliding surface. Filippov Example. $h = 0.002$, $x(0) = [1.0, -1.0]^T$

are chosen as $[2.0, 0, 0, 0]^T$. The numerical parameters are given by $h = 0.1$ that is a sampling of $10Hz$ and $\theta = 1, \gamma = 1$. On Figure 11, the error between the reference command and the observer state is given. On Figure 12, we can observe the behavior of the command without any chattering.

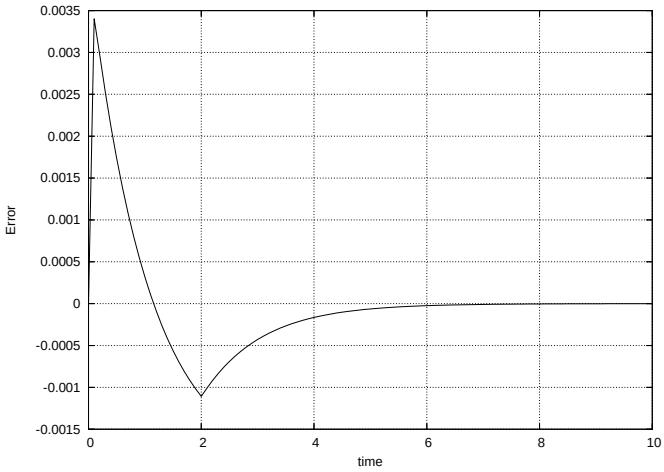
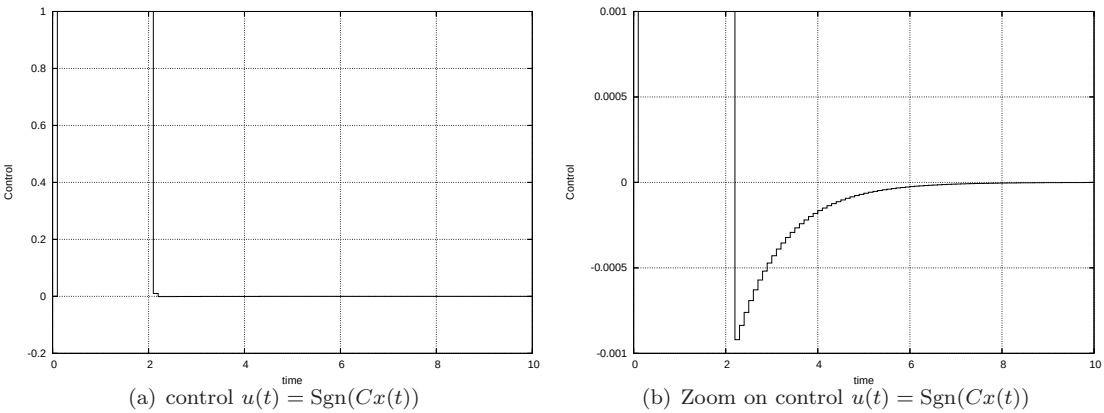


Figure 11: Observer based SMC: Error $e(t)$. $k = 1$ and $\tau = 0.001$. $h = 0.1$ $\theta = 1, \gamma = 1$



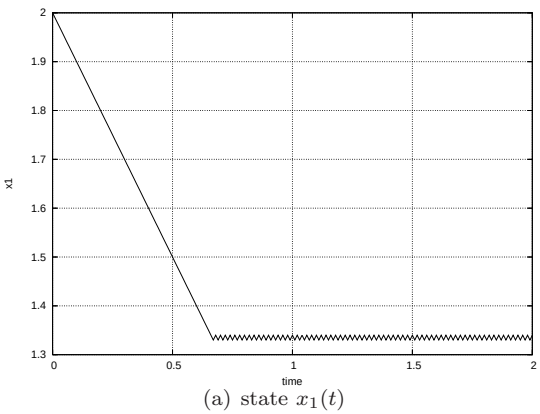
(a) control $u(t) = \mathrm{Sgn}(Cx(t))$ (b) Zoom on control $u(t) = \mathrm{Sgn}(Cx(t))$

Figure 12: Observer based SMC: Control. $k = 1$ and $\tau = 0.001$. $h = 0.1$ $\theta = 1, \gamma = 1$

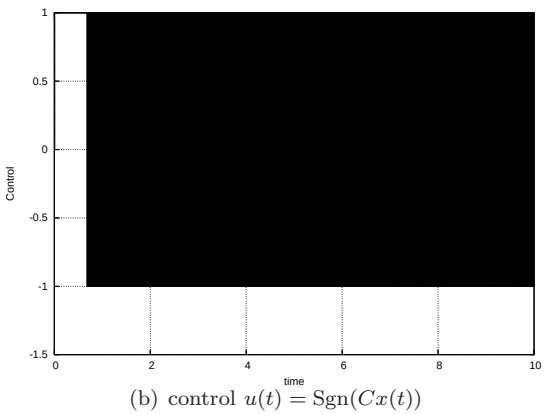
On the three previous examples, one sees a very accurate and smooth stabilization on the sliding surface, even for values of h not so small.

Influence of the integration parameters θ and γ In the following numerical experiments, we discuss the role of the numerical parameters θ and γ. Due to the fact that the function $g(\cdot)$ is linear and reduced to a matrix $B = [1, 0, 0, 0]^T$, the parameter γ has no influence on the numerical time–integration. On the contrary, the parameter θ has a huge influence on the stability of the integration. Indeed, the implicit Euler integration ($\theta = 1$) of the smooth term $f(\cdot)$ is unconditionally stable. This is not the case for the explicit Euler $\theta = 0$ and for the chosen parameters k and τ, the instability of the scheme for $h = 0.1$ does not allow to proceed to integration. On Figure 13, the instability of the scheme is illustrated and appears as a chattering on the state x. The stability is retrieved for $h < 0.005$.

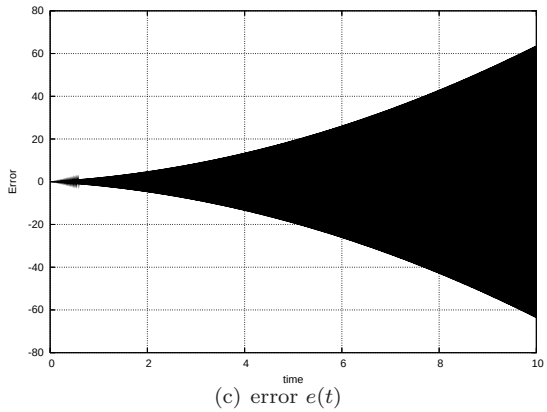
Other choices of θ can be made to improve the numerical time–integration of the smooth dynamics. For instance, $\theta = 1/2$ yields a second order scheme for the integration of f. Unfortunately, the scheme is not of second order due to the fully implicit integration of the multi-valued part.



(a) state $x_1(t)$



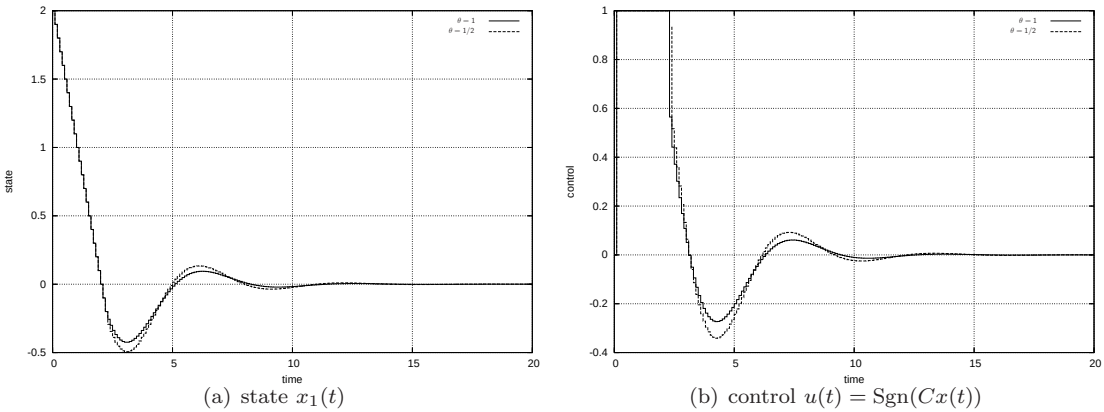
(b) control $u(t) = \mathrm{Sgn}(Cx(t))$



(c) error $e(t)$

Figure 13: Observer based SMC. $k = 1$ and $\tau = 0.001$. $h = 0.01$ $\theta = 0, \gamma = 1$

Nevertheless, it can be interested to use of $\theta = 1/2$ to decrease the numerical damping of the scheme on the smooth term. On the Figure 14, two simulations of the observer based SMC are presented for $\theta = 1$ and $\theta = 1/2$. The parameter τ has been modified to 0.5 to correctly integrate the parasitic dynamics with the same time–step $h = 0.1$.



(a) state $x_1(t)$ (b) control $u(t) = \mathrm{Sgn}(Cx(t))$

Figure 14: Observer based SMC with $\theta = 1$ and $\theta = 1/2$. $k = 1$ and $\tau = 0.5$. $h = 0.01, \gamma = 1$

10 Conclusions

In this paper the backward Euler method is studied on specific classes of Filippov's systems that encompass sliding mode control systems. It is shown that such implict schemes allow a smooth accurate stabilization on the sliding surface, even in case of codimension larger than one. Despite the backward Euler method has been studied and used for a long time in other fields like contact mechanics and electric circuits simulation [1], it seems it has not yet been used in the sliding mode control community. This work therefore constitutes the introduction of a new discretization method for EBC-SMC systems. The novelty compared to numerical simulation is that this time one has to consider not only the numerical simulation, but also the implementation on real processes. Perhaps one obstacle to the dissemination of the method is that at first sight, the controller designed from a backward philosophy looks like a non causal controller. However as shown in this paper this is not the case. This paper paves the way towards the study of a new family of discrete-time sliding mode controllers.

References

[1] V. Acary and B. Brogliato. *Numerical Methods for Nonsmooth Dynamical Systems: Applications in Mechanics and Electronics*, volume 35 of *Lecture Notes in Applied and Computational Mechanics*. Springer Verlag, 2008.

[2] V. Acary and B. Brogliato. Implicit Euler numerical simulation of sliding mode systems. Technical Report RR 1234567890, INRIA, 2009.

[3] V. Acary and F. Pérignon. An introduction to SICONOS. Technical Report TR-0340, INRIA, http://hal.inria.fr/inria-00162911/en/, 2007.

[4] J.C. Alexander and T.I. Seidman. Sliding modes in intersecting switching surfaces. I: Blending. *Houston J. Math.*, 24(3):545–569, 1998.

[5] J.C. Alexander and T.I. Seidman. Sliding modes in intersecting switching surfaces. II: Hysteresis. *Houston J. Math.*, 25(1):185–211, 1999.

[6] B. Baji and A. Cabot. An inertial proximal algorithm with dry friction: finite convergence results. *Set-Valued Anal.*, 14(1):1–23, 2006.

[7] O. Barambones, P. Alkorta, A.J. Garrido, I. Garrido, and F.J. Maseda. An adaptive sliding mode control scheme for induction motor drives. *Int. J. Circuits, Systems and Signal Processing*, 1(1):73–78, 2007.

[8] G. Bartolini, L. Fridman, A. Pisano, and E. Usai, editors. *Modern Sliding Mode Control Theory. New Perspectives and Applications*, volume 375 of *LNCIS*. Springer Verlag, 2008.

[9] J. Bastien and C.H. Lamarque. Persoz' gephyroidal model described by a maximal monotone differential inclusion. *Arch. Appl. Mech.*, 78(5), 2008.

[10] J. Bastien and M. Schatzman. Numerical precision for differential incusions with uniqueness. *ESAIM M2AN: Mathematical Modelling and Numerical Analysis*, 36(3):427–460, 2002.

[11] S.C. Billups, S.P. Dirkse, and M.C. Ferris. A comparison of large scale mixed complementarity problem solvers. *Computational Optimization and Applications*, 7:3–25, 1997.

[12] I. Boiko, L. Fridman, A. Pisano, and E. Usai. A comprehensive analysis of chattering in second order sliding mode control systems. Bartolini, Giorgio (ed.) et al., Modern sliding mode control theory. New perspectives and applications. Berlin: Springer. Lecture Notes in Control and Information Sciences 375, 23-49 (2008)., 2008.

[13] B. Brogliato. Absolute stability and the Lagrange-Dirichlet theorem with monotone multivalued mappings. *Systems and Control Letters*, 51:343–353, 2004.

[14] B. Brogliato, R. Lozano, B. Maschke, and O. Egeland. *Dissipative Systems Analysis and Control. Theory and Applications*. Springer Verlag, London, 2nd edition, 2007.

[15] B. Brogliato and A. Trofino Neto. Practical stabilization of a class of nonlinear systems with partially known uncertainties. *Automatica*, 31(1):145–150, 1995.

[16] M. K. Camlibel, W.P.M.H. Heemels, and J.M. Schumacher. Consistency of a time-stepping method for a class of piecewise-linear networks. *IEEE Trans. Circuits and systems I*, 49:349–357, 2002.

[17] M. Cao and M.C. Ferris. A pivotal method for affine variational inequalities. *Mathematics of Operations research*, 21(1):44–64, 1996.

[18] J. Cortes. Discontinuous dynamical systems. a tutorial on solutions, nonsmooth analysis, and stability. *IEEE Control Systems Magazine*, pages 36–73, June 2008.

[19] R. W. Cottle, J. Pang, and R. E. Stone. *The Linear Complementarity Problem*. Academic Press, Inc., Boston, MA, 1992.

[20] M.-F. Danca. Numerical approximations of a class of switch dynamical systems. *Chaos Solitons Fractals*, 38(1):184–191, 2008.

[21] S. P. Dirkse and M. C. Ferris. The PATH solver: A non-monotone stabilization scheme for mixed complementarity problems. *Optimization Methods and Software*, 5:123–156, 1995.

[22] A.L. Dontchev and F. Lempio. Difference methods for differential inclusions: a survey. *SIAM reviews*, 34(2):263–294, 1992.

[23] C. Edwards, E. Fossas Colet, and L. Fridman, editors. *Advances in Variable Structure and Sliding Mode Control*, volume 334 of *LNCIS*. Springer Verlag, 2006.

[24] F. Facchinei and J. S. Pang. *Finite-dimensional Variational Inequalities and Complementarity Problems*, volume I & II of *Springer Series in Operations Research*. Springer Verlag NY. Inc., 2003.

[25] Z. Galias and X. Yu. Complex discretization behaviors of a simple sliding-mode control system. *IEEE Transactions on Circuits and Systems – II: Express Briefs*, 53(8):652–656, August 2006.

[26] Z. Galias and X. Yu. Euler's discretization of single input sliding-mode control systems. *IEEE Transactions on Automatic Control*, 52(9), September 2007.

[27] Z. Galias and X. Yu. Analysis of zero–order holder discretiation of two–dimensional sliding–mode control systems. *IEEE Transactions on Circuits and Systems – II: Express Briefs*, 55(12), December 2008.

[28] W. Garcia-Gabin, D. Zambrano, and E.F. Camacho. Sliding mode predictive control of a solar air conditioning plant. *Control Engineering Practice*, 2009. in press.

[29] D. Goeleven, D. Motreanu, Y. Dumont, and M. Rochdi. *Variational and Hemivariational Inequalities: Theory, Methods and Applications; Volume I: Unilateral Analysis and Unilateral Mechanics*. Nonconvex Optimization and its Applications. Kluwer Academic Publishers, 2003.

[30] P.T. Harker and J.-S. Pang. Finite-dimensional variational inequality and complementarity problems: a survey of theory, algorithms and applications. *Mathematical Programming*, 48:160–220, 1990.

[31] G. Herrmann, S.S. Ge, and G. Guo. Practical implementation of a neural network controller in a hard disk drive. *IEEE Transactions on Control Systems Technology*, 13(1):146–154, January 2005.

[32] R.A. Hess and S.R. Wells. Sliding mode control applied to reconfigurable flight control design. *J. of Guidance, Control, and Dynamics*, 26(3):452–462, May–June 2003.

[33] Q. Hu, C. Du, L. Xie, and Y. Wang. Discrete-time sliding mode control with time-varying surface for hard disk drives. *IEEE Transactions on Control Systems Technology*, 17(1):175–183, January 2009.

[34] A.E. Kastner-Maresch. The implicit midpoint rule applied to discontinuous differential equations. *Computing*, 49:45–62, 1992.

[35] A. Koren and M. Idan. Integrated sliding mode guidance and control for a missile with on-off actuators. *J. of Guidance, Control, and Dynamics*, 31(1):204–214, January–February 2008.

[36] A.J. Koshkouei and A.S.I. Zinober. Sliding mode control of discrete-time systems. *Journal of Dynamic Systems, Measurement, and Control*, 122:793–802, December 2000.

[37] R. Leine and H. Nijmeijer. *Dynamics and Bifurcations of Non-Smooth Mechanical Systems*. Springer Verlag, Lecture Notes in Applied and Computational Mechanics 18, 2004.

[38] L. Marton and B. Lantos. Control of mechanical systems with stribeck friction and backlash. *Systems and Control Letters*, in press. doi:10.1016/j_sysconle.2008.10.001.

[39] C. Moler and C. Van Loan. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM Rev.*, 45(1):3–49, 2003.

[40] T. S. Munson, F. Facchinei, M. C. Ferris, A. Fischer, and C. Kanzow. The semismooth algorithm for large scale complementarity problems. *INFORMS Journal on Computing*, 13:294–311, 2001.

[41] Y. V. Orlov. *Discontinuous Systems. Lyapunov Analysis and Robust Synthesis under Uncertainty Conditions*. Communications and Control Engineering. Springer verlag, London, 2009.

[42] A. Y. Pogromski, W.P.M.H. Heemels, and H. Nijmeijer. On solution concepts and well-posedness of linear relay systems. *Automatica*, 39:2139–2147, 2003.

[43] A.B. Proca, A. Keyhani, and J.M. Miller. Sensorless sliding-mode control of induction motors using operating condition dependent models. *IEEE Transactions on Energy Conversion*, 18(2):205–212, June 2003.

[44] E. Promtun and S. Seshagiri. Sliding mode control of a pitch-rate of an f-16 aircraft. *Int. J. of Applied Science, Eng. and Technology*, 5(1):1–7, 2009.

[45] R.T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

[46] R.T. Rockafellar and R.J.-B. Wets. *Variational Analysis*, volume 317. Springer Verlag, New York, 1997.

[47] T. Rutherford. Miles: A mixed inequality and nonlinear equation solver, 1993.

[48] R. W. H. Sargent. An efficient implementation of the Lemke algorithm and its extension to deal with upper and lower bounds. *Mathematical Programming Study*, 7:36–54, 1978.

[49] D. Stewart. A high accuracy method for solving ODEs with discontinuous rigth–hand–side. *Numerische Mathematik*, 58:299–328, 1990.

[50] D. Stewart. A numerical method for friction problems with multiple contacts. *J. Austral. Math. Soc. Ser. B*, 37:288–308, 1996.

[51] Y.C. Tsai and A.C. Huang. Multiple-surface sliding controller design for pneumatic servo systems. *Mechatronics*, 18(9):506–512, November 2008.

[52] B. Wang, X. Yu, and G. Chen. ZOH discretization effect on single-input sliding mode control systems with matched uncertainties. *Automatica*, 45:118–125, 2009.

[53] W.J. Wang and J.Y. Chen. Passivity-based sliding mode position control for induction motor drives. *IEEE Transactions on Energy Conversion*, 20(2):316–321, June 2005.

[54] S.R. Wells and R.A. Hess. Multi-input/multi-output sliding mode control for a tailles fighter aircraft. *J. of Guidance, Control, and Dynamics*, 26(3):463–473, May–June 2003.

[55] K.D. Young, V.I. Utkin, and U. Ozguner. A control engineer's guide to sliding mode control. *IEEE Transactions on Control Systems Technology*, 7(3):328–342, May 1999.

[56] X. Yu and G. Chen. Discretization behaviors of equivalent control based sliding-mode control systems. *IEEE Transactions on Automatic Control*, 48(9):1641–1646, September 2003.

[57] X. Yu, B. Wang, Z. Galias, and G. Chen. Discretization effect on equivalent control-based multi-input sliding-mode control systems. *IEEE Transactions on Automatic Control*, 53(6):1563–1569, July 2008.

Contents

1 Introduction **3**

2 A simple example **4**

3 A class of differential inclusions **7**

4 Convergence results and Chattering Free Finite–time Stabilization **9**

5 Discrete–time Sliding Mode Control (SMC) 12
5.1 Example of an implicit Euler controller (IEC) 12
5.2 Extension to ZOH discretized systems 12

6 Implementation of Discrete–Time Systems 13
6.1 Mixed Linear Complementarity Problem (MLCP) 13
6.2 Some Time–Discretization Methods 14

7 Two other classes of differential inclusions 15
7.1 A simple hypomonotone case 17

8 Detailed Implementation of Implicit Euler Discretization of the general case (60) 18
8.1 Time–discretization 18
8.2 Mixed Complementarity Problem 19
8.3 Newton's linearization and Mixed Linear Complementarity Problems 20
8.4 The special cases of the affine systems 21
8.4.1 Time–discretization of the system (13) 21
8.4.2 Time–discretization of the system (59) 22
8.5 Algorithms 24

9 Numerical experiments 25
9.1 Chattering free stabilization 25
9.2 Example 3: Multiple sliding surfaces 25
9.3 Extensions to ZOH discretized systems 25
9.4 Lyapunov-based robust control 28
9.5 The Filippov example 28
9.6 Example 4: Observer based SMC 28

10 Conclusions 32

Centre de recherche INRIA Grenoble – Rhône-Alpes
655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399