Implicit Euler numerical scheme and chattering-free implementation of sliding mode systems

Vincent Acary, Bernard Brogliato

To cite this version:


HAL Id: inria-00423576
https://inria.hal.science/inria-00423576

Submitted on 29 Oct 2017

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.
Implicit Euler numerical scheme and chattering-free implementation of sliding mode systems

Vincent Acary*, Bernard Brogliato

Keywords: Switching systems, Filippov’s differential inclusions, Complementarity problems, Backward Euler algorithm, Sliding modes, Maximal monotone mappings, Mixed linear complementarity problem, ZOH discretization

In this paper 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 multiple switching surfaces (i.e., a sliding surface of codimension ≥ 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.

1. Introduction

Sliding mode controllers are widely used because of their intrinsic robustness properties [1,2]. Some important fields of application are induction motors [3], aircraft control [4], hard disk drives [5], solar systems [6], and autonomous robots [7]. 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. [8–10], which also have their own limitations [10]. 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 simulations of such nonsmooth dynamical systems is non trivial and it has received a lot of attention, see [11] and references therein. 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. Recently it has been shown that the explicit Euler method generates unwanted effects like spurious oscillations (also called chattering effects) around the switching surface [12,13]. In parallel, the digital implementation of sliding mode controllers has been studied in [14], 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 \( \text{sgn}(\cdot) \) 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 [15] 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

* Corresponding author. Tel.: +33 4 61 52 29; fax: +33 4 61 54 77.
E-mail addresses: vincent.acary@inrialpes.fr (V. Acary), bernard.brogliato@inrialpes.fr (B. Brogliato).
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 system 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 this paper are shown to fit well within this class of differential inclusions. In Section 5, a class 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 7.

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

2. A simple example

To start with we consider the simplest case:

\[
\dot{x}(t) = -\text{sgn}(x(t)) \quad \text{if } x(t) \neq 0 \\
\dot{x}(t) = 0 \quad \text{if } x(t) = 0
\]

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

\[
\begin{align*}
\Delta k \geq 1 & - x_k = -h s_{k+1} \\
\Delta k \geq 1 & - s_{k+1} = \text{sgn}(x_{k+1}) \quad (2)
\end{align*}
\]

This method converges with at least order \( \frac{1}{2} \) (see Proposition 2). 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 \( x_{k+1} \) for all \( n \geq 1 \).

**Lemma 1.** For all \( h > 0 \) and \( x_0 \in \mathbb{R} \), there exists \( k_0 \) such that \( x_{k_0} = 0 \) and \( s_{k_0+1} = -\text{sgn}(x_{k_0}) \) for all \( n \geq 1 \).

**Proof.** The value \( k_0 \) is defined as the first time step such that \( x_{k_0} \in [-h, h] \). If \( x_k \in [-h, h] \), then \( k_0 = 0 \). Otherwise, the solution of the time-discretization (2) is given by \( x_k = x_0 - h s_{k_0} \) while \( s_k = \text{sgn}(x_k) \) while \( x_k \neq [-h, h] \) for \( k < k_0 \), and \( k_0 = \left\lfloor \frac{\text{sgn}(x_0)}{h} \right\rfloor + 1 \). The symbol \( \lfloor \cdot \rfloor \) is the ceiling function which gives the smallest integer greater than or equal to \( \cdot \). Let us now consider that \( x_{k_0} \in [-h, h] \). The only possible solution for

\[
\begin{align*}
x_{k_0+1} &= x_{k_0} - h s_{k_0+1} \\
\Delta k \geq 1 & - s_{k_0+1} = \text{sgn}(x_{k_0+1})
\end{align*}
\]

is \( x_{k_0+1} = 0 \) and \( s_{k_0+1} = \frac{\text{sgn}(x_0)}{h} \). For the next iteration, we have to solve

\[
\begin{align*}
x_{k_0+2} &= x_{k_0+1} - h s_{k_0+2} \\
\Delta k \geq 1 & - s_{k_0+2} = \text{sgn}(x_{k_0+2})
\end{align*}
\]

and we obtain \( x_{k_0+2} = 0 \) and \( s_{k_0+2} = 0 \). The same holds for all \( n \geq 3 \) and so the same reasoning clearly then the terms \( (s_{k+1} - x_{k+1})/h \) approximating the derivative are zero for any \( h > 0 \). \( \square \)

This result is robust with respect to the numerical threshold that can be encountered in floating point operations. Indeed, let us assume that \( x_0 = 0 \) and \( h = \varepsilon \ll 1 \), that is \( \varepsilon > 0 \) is zero at the machine’s precision. We obtain \( s_{k_0+1} = -1 \) and \( x_{k_0+1} = \varepsilon \) that is zero at the machine’s precision. For \( n = 2 \), we obtain \( x_{k_0+2} = 0 \) and \( s_{k_0+2} = 0 \). This robustness stems from the fact that the dynamics is not only monitored by the sign of \( x_k \) but also by the belongingness to the interior of \([-1,1]\) of the “dual” variable \( s_{k+1} \).

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 [12,13]. 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 \( x_{k+1} \mapsto -\text{sgn}(x_{k+1}) \) and the straight line \( x_{k+1} \mapsto x_{k+1} - x_k \). This is illustrated on Fig. 1, where few iterations are depicted until the state reaches zero.

From a control perspective the input is implemented on \([I_0, I_{k+1}] \) as \( u_k = - \text{sgn}(x_{k+1}) \) as a piecewise affine 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 \( \text{sgn}(0) \), or more exactly \( \text{sgn}(\varepsilon) \) where \( \varepsilon \) 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 \( \text{sgn}(\cdot) \). It is easy to show that the explicit method yields an oscillation around \( x = 0 \), as shown in more general situations in [12,13]. Other time-stepping methods like the so-called switched model [11,16] fail to correctly solve the integration problem when the number of switched surfaces is too large (see also [8] 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.

---

\(^1\) See [10] for a discussion on this point.
3. A class of differential inclusions

Let us now introduce the following class of differential inclusions, where \( x(t) \in \mathbb{R}^n \):

\[
\begin{align*}
\dot{x}(t) &\in -A(x(t)) + f(t, x(t)), \quad \text{a.e. on } (0, T) \\
x(0) &= x_0.
\end{align*}
\]  

(5)

The following assumption is made:

**Assumption 1.** The following items hold:

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

\[
(x' - y')^T(x - y) \geq 0.
\]

(6)

- **(ii)** There exists \( L \geq 0 \) such that for all \( t \in [0, T], \) for all \( x_1, x_2 \in \mathbb{R}^n \), one has \( \|f(t, x_1) - f(t, x_2)\| \leq L\|x_1 - x_2\|. \)

- **(iii)** There exists a function \( \Phi(\cdot) \) such that for all \( R \geq 0 \):

\[
\Phi(R) = \sup \left\{ \frac{\|f(t, \cdot)\|}{\|\nabla f(t, \cdot)\|}, \quad \|v\|_{L^2(0, T); \mathbb{R}^n} \leq R \right\} < +\infty.
\]

**Proposition 1** (Bastien & Schatzman [17], Prop. 2.6). Let Assumption 1 hold, and let \( x_0 \in D(A) \). Then the differential inclusion (5) has a unique solution \( x : (0, T) \rightarrow \mathbb{R}^n \) that is Lipschitz continuous with an essentially bounded derivative.

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

\[
\dot{x}(t) = f(t, x(t)) - B\text{sgn}(Cx(t) + D), \quad \text{a.e. on } (0, T) \\
x(0) = x_0
\]  

(7)

with \( B \in \mathbb{R}^{n \times m} \), and \( \text{sgn}(Cx + D) \triangleq (\text{sgn}(C_1x + D_1), \ldots, \text{sgn}(C_mx + D_m))^T \in \mathbb{R}^m \). It will be shown how to recast (7) into (5).

**Example 1** (Equivalent-Control-Based Sliding-Mode-Control (ECB-SMC)). Consider a system \( x(t) = Fx(t) + Gu \), with an equivalent-control-based sliding-mode-control (ECB-SMC) of the form \( u(x) = -(H_G)^{-1}H_x - \alpha(H_G)^{-1}\text{sgn}(H_x), \alpha > 0 \). Then the closed-loop system \( \dot{x}(t) = (F - G(H_G)^{-1}H)\dot{x}(t) - \alpha G(H_G)^{-1}\text{sgn}(Hx(t)) \) fits within (7).

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

**Corollary 1.** Consider the differential inclusion in (7). Suppose that (ii) and (iii) of Assumption 1 hold. If there exists an \( n \times n \) matrix \( P = P^T > 0 \) such that
Proof. The proof uses a state variable change introduced in [18]. 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

\[
\dot{z}(t) \in Rf(t, R^{-1}z(t)) - RBSgn(C_1 R^{-1}z(t) + D_1).
\]

Notice that \( BSgn(C_1 R^{-1}z(t) + D_1) = \sum_{i=1}^{m} B_{*i} sgn(C_1 R^{-1}z + D_i) \). Therefore \( RBSgn(C_1 R^{-1}z(t) + D_1) = \sum_{i=1}^{m} R_i C_1 i sgn(C_1 R^{-1}z + D_i) \). We can rewrite the system as

\[
\dot{z}(t) \in Rf(t, R^{-1}z(t)) - \sum_{i=1}^{m} R_i C_1 i sgn(C_1 R^{-1}z(t) + D_i).
\]

The multivalued mapping \( \xi \mapsto sgn(\xi) \) is monotone. By [19, Exercise 12.4] it follows that each multivalued mapping \( z \mapsto R_i C_1 i sgn(C_1 R^{-1}z(t) + D_i) \) is monotone. From [20, Proposition 1.3.11] it follows that \( R_i C_1 i sgn(C_1 R^{-1}z(t) + D_i) = \delta f(\xi) \) with \( f(\xi) = (*) \). By [21, Theorem 5.2] it follows that \( \dot{f}(\cdot) \) is convex. Being the subdifferential of a convex function, the multivalued mapping \( z \mapsto \delta f(\xi) \) is maximal (monotone) [21, Corollary 31.5.2]. Therefore by Proposition 1 the inclusion in (10) possesses a unique Lipschitz solution on \((0, T)\) for all \( T > 0 \) and since \( R \) is full-rank so does (7). \( \square \)

Example 2. Consider the sliding mode system in [12, Equ. (1)–(4)]. One has \( B = (0, 0)^T \), \( C = (c_1, 1) \), \( D = 0 \). Then the condition in (8) 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 \( B = \begin{pmatrix} 1 \\ 2 \\ -1 \end{pmatrix} \). \( Sgn(Cx + D) = (s(x_1 + 2x_2), s(x_2 - x_1)) \). Trajectories may slide on codimension one surfaces \( x_1 + 2x_2 = 0 \) or \( x_2 = x_1 = 0 \) 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 [10], whose closed-loop dynamics is given by:

\[
\begin{aligned}
\dot{x}(t) &= f(x(t)) + Bu(t) + By(t), \\
x(0) &= x_0
\end{aligned}
\]

where \( x(t) \in \mathbb{R}^n \), \( B \in \mathbb{R}^{n \times m} \), \( f(\cdot) \) satisfies Assumption 1, and \( y(\cdot) \in \mathbb{R}^m \) is a bounded disturbance satisfying \( |y(t)| < \rho \) for all \( i \leq m \), all \( t > 0 \) and some finite \( \rho \). 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 \( V(x(t)) = \nabla V(x(t))^T f(x(t)) \leq 0 \) along the trajectories of the free system. Let us rewrite the system in (12) as

\[
\dot{x}(t) = f(x(t)) + \sum_{i=1}^{m} B_{*i} u_i + \sum_{i=1}^{m} B_{*i} y_i.
\]

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

\[
\dot{x}(t) = f(x(t)) - \sum_{i=1}^{m} \rho_i B_{*i} sgn(\nabla V(x)^T B_{*i}) + \sum_{i=1}^{m} B_{*i} y_i.
\]

We can state the following result.

Corollary 2. Suppose that \( V(x) = \frac{1}{2} x^T P x \). Then the system in (14) has a unique Lipschitz continuous with an essentially bounded derivative solution on \([0, +\infty)\) for any \( x_0 \).

Proof. We have \( \nabla V(x)^T B_{*i} = B_{*i}^T P x \). Let \( z = Rx \), where \( R > 0 \) is the symmetric square root of \( P \). We may rewrite (14) as

\[
\dot{z}(t) \in Rf(R^{-1}z(t)) - \sum_{i=1}^{m} \rho_i R B_{*i} sgn(B_{*i}^T R z) + \sum_{i=1}^{m} R B_{*i} y_i.
\]

Then following the same steps as for the proof of Corollary 1 we conclude that Proposition 1 applies to this system, hence to (14). \( \square \)

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 (5) is time-discretized on \([0, T]\) with a backward Euler scheme as follows:

\[
\frac{x_{k+1} - x_k}{h} + A(x_{k+1}) = f(t_k, x_k),
\]

for all \( k \in \{0, \ldots, N - 1\} \),

\[
x_0 = x(0)
\]

where \( h = T/N \). 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 (15) in [17]. 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.

Proposition 2 (Bastien & Schatzman [17], Prop. 2.7 and 4.4). Under Assumption 1, there exists \( \eta \) such that for all \( h > 0 \) one has

\[
\text{for all } t \in [0, T], \quad \|x(t) - x^N(t)\| \leq \eta \sqrt{h}
\]

where the function \( x^N(\cdot) \) is defined by the linear interpolation of the \( x_0 \)'s at \( t_k \).

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 (15) has at least order $\frac{1}{2}$, and convergence holds. As seen in Lemma 1, the precision of the method may be much better than what is to be expected from (16) on large portions of the trajectories.

The differential inclusion in (7) is therefore discretized as follows:

\[
\frac{X_{k+1} - X_k}{h} \in f(t_k, x_k) - B\operatorname{Sgn}(C_{X_{k+1}} + D),
\]

a.e. on $(0, T)$, $x(0) = x_0$. \hfill (17)

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 $0 \in F_{t_k}(X_{k+1}) + F_m(x_{k+1})$ where $F_m(\cdot)$ is singlevalued while $F_m(\cdot)$ is multivalued. The values $h, t_k$, 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 $F_t(\cdot)$ and the graph of $F_m(\cdot)$ as illustrated in Section 2. The result of Proposition 2 applies to (17). As we shall see next, such an implicit method also assures a good estimate of the derivative $x$ 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 system as:

\[
y(t) \hat{=} C x(t) + D.
\] \hfill (18)

**Lemma 2.** Let us assume that a sliding mode exists for some indices $i \in \{1, \ldots, m\}$, i.e.,

\[
\exists \tau > 0 \text{ such that } y_{\tau}(t) = C_{\alpha} x(t) + D_\alpha = 0 \quad \text{for all } t > t_\tau.
\] \hfill (19)

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

\[
\|C_{\alpha} \cdot f((x(t), t))\| \leq \rho.
\] \hfill (20)

Furthermore let Assumption 1(ii) hold. Then the following bound is satisfied in the neighborhood of the sliding subspace:

\[
\exists \kappa > 0, \exists \kappa > 0, \exists \rho > 0 \text{ such that } \forall t > t_\tau, \forall x \in B_r(x), \|C_{\alpha} \cdot f((x(t), t))\| \leq \kappa r + \rho
\] \hfill (21)

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

**Proof.** From (19), we have $y_{\tau}(t) = C_\alpha x(t) + D_\alpha = 0$ implies that $y_{\tau}(t) = C_{\alpha} x(t) = 0$ for all $t > t_\tau$, and therefore

\[
C_{\alpha} \cdot f((x(t), t)) \in C_{\alpha} \cdot B \operatorname{Sgn}(y(t)).
\] \hfill (22)

The inclusion (22) yields

\[
\exists \rho > 0 \text{ such that } \|C_{\alpha} \cdot f((x(t), t))\| \leq \rho
\] \hfill (23)

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

\[
\forall x(t) \in B_r(x(t)), \|C_{\alpha} \cdot f((x(t), t)) - f(x(t), t))\| \leq \|C_{\alpha} \| L_r \leq \kappa r.
\] \hfill (24)

Combining (23) and (24) ends the proof. \hfill $\square$

Lemma 1 extends to (17) when the sliding surface of codimension $\kappa$ is attained.

**Lemma 3.** Let us assume that a sliding mode occurs for the index $\alpha \in \{1, \ldots, m\}$, that is $y_{\alpha}(t) = 0, t > t_\tau$. Let $C$ and $B$ be such that (8) holds and $C_{\alpha} B_{\alpha} > 0$, then there exists $h_{\tau} > 0$ such that for all $h < h_{\tau}$, there exists $k_0 \in \mathbb{N}$ such that $y_{\alpha_{k_0+n}} = C_{\alpha_{k_0+n}} + D = 0$ for all integers $n \geq 1$.

**Proof.** At each time-step, we have to solve for $y_{\alpha+1} = C_{\alpha+1}$$+ D$ and $s_{k+1}$ the generalized equation

\[
\begin{align*}
\{y_{\alpha+1} &= y_k + hC_{\alpha} f(t_k, x_k) - hCB_{\alpha_{k+1}} + D, \\
\|y_{\alpha_{k+1}} - x(t_k)\| &\leq \kappa \sqrt{h},
\end{align*}
\] \hfill (25)

Under condition (8), the convergence of the time-stepping scheme is ensured by Proposition 2. The convergence and the existence of the sliding mode ensure that

\[
\exists \kappa_0, \exists \kappa_1 > 0, \exists \kappa_2 > 0, \exists \kappa_3 > 0, \exists \kappa_4 > 0, \exists \kappa_5 > 0, \exists \kappa_6 > 0,
\] \hfill (26)

which implies the convergence of the time-stepping scheme.

**5. Discrete-time Sliding Mode Control (SMC)**

This section is devoted to show how the above time-discretization 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-time
system. Assume that the integrator \( \dot{x}(t) = u(t) \) is sampled with a sampling period \( h > 0 \). On the time interval \([t_k, t_{k+1})\) one has \( x(t) = x_k + h \cdot u_k \), where \( t_k = t - t_k \). The controller \( u(t) = -\text{sgn}(x) \) is known as the equivalent control-based SMC [10]. 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 \([t_k, t_{k+1})\):

\[
x(t) = x_k - \frac{h}{h} x_k,
\]

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 \( s_{k+2} = 0 \) and

\[
x(t) = x_{k+1} - \frac{h}{h} x_k = 0 - \frac{h}{h} 0 = 0,
\]

and so on on the next intervals, where the zero value is obviously some small value at the machine accuracy. If we suppose that \( x_k \not\in [-h, h] \), 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

\[
u_k = -\text{proj}_{[-1,1]} \left( \frac{x_k}{h} \right),
\]

where \( \text{proj}_c \) 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 (36). It requires at each step to solve a rather simple multi-valued problem: a Mixed Linear Complementarity Problem (MLCP). Let us define what is an MLCP in its general form with bounds constraints as it has been proposed in [25]:

**Definition 1** (MLCP). Given a matrix \( M \in \mathbb{R}^{m \times m} \), a vector \( q \in \mathbb{R}^m \) and lower and upper bounds \( l, u \in \mathbb{R}^m \) such that

\[
\begin{align*}
Mz + q &= w - v \\
l \leq z \leq u \\
(z - l)^T w &= 0 \\
(u - z)^T v &= 0
\end{align*}
\]

where \( \mathbb{R} = \mathbb{R} \cup \{+\infty, -\infty\} \).

Note that the problem (45) implies that

\[
-(Mz + q) \in N_c(x)(z)
\]

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 \([l, u] \subset \mathbb{R}^m \) 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 [21]. The relation (46) is equivalent to the MLCP (45) 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 (44) as an MLCP, the variable \( x_{k+1} \) is condensed into the second line such that

\[
y_{k+1} = CRx_k + Cp - CSS_{k+1} + D,
\]
\[ s_{k+1} \in \text{Sgn}(y_{k+1}) \]
\[ y_{k+1} \in N[-1, 1] = (s_k + 1) \in \mathbb{Z} \]
\[ s_{k+1} \in [-1, 1] \]
\[
\begin{align*}
\begin{cases}
    y_{j,k+1} = 0 & \text{if } s_{j,k+1} \in [-1, 1] \\
    y_{j,k+1} = -1 & \text{if } s_{j,k+1} = -1 \\
    y_{j,k+1} = 1 & \text{if } s_{j,k+1} = 1
\end{cases}
\end{align*}
\] (49)

The MLCP (45) is a well-known problem in the mathematical programming theory that enjoys a large number of numerical algorithms. In this paper, the computations are done with the help of the SICONOS/NUMERIC library [26] and/or the PATH solver [25]. The results of existence and uniqueness of solutions of (45) are related to the properties of \( M \)-properties or coherent orientations of the associated affine map (normal map) for particular cases of bounds constraints (see [27]). The assumptions on the matrix \( M \) drive 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 (44) 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 (15). At each time step, the matrices in (44) and in the MLCP (45) can be identified as
\[
R = I, \quad p = h f(t_k, x_k), \quad S = hB, \quad M = hCB, \quad q = -(hCf(t_k, x_k) + Cx_k + D).
\] (50)

Let the assumptions of Corollary 1 be satisfied with \( B \) full-column rank (then \( CB = B^T B > 0 \)). This result ensures the existence and uniqueness of a solution of the MLCP. Furthermore, standard pivot algorithms such as Lemke’s method or projection splitting like the Projected Successive Over-Relation (PSOR) method compute the solution.

Implicit Euler and \( \theta \)-method. In a more general way, we can choose to time-discretize the term \( f(t, x(t)) \) by an implicit Euler scheme or a \( \theta \)-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 [28] 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 (44) and in the MLCP (45) can be identified as
\[
\begin{align*}
R &= (I - h\theta F)^{-1}(I + h(1 - \theta)F), \quad p = (I - h\theta F)^{-1}g, \\
S &= h(I - h\theta F)^{-1}B, \\
M &= hC(I - h\theta F)^{-1} - B, \quad q = -(I - h\theta F)^{-1} \\
\end{align*}
\] (51)

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 \( f(\cdot, \cdot) \). 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 [28] 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 (44) and then (45) with
\[
R = \Phi, \quad p = 0, \quad S = \Gamma, \quad M = C \Gamma, \quad q = -(C\Phi x_k).
\] (52)

In practice, numerous methods are available to compute the ZOH discretization, i.e. \( \Phi \) and \( \Gamma \). This amounts to computing the matrix exponential and its time integral [29]. In this work, the numerical computation is performed using an explicit Runge–Kutta method with a high order of accuracy and a numerical tolerance near the machine precision threshold. On Fig. 3, the control scheme is depicted showing that the controller is causal and computed from \( x_{k} \).

7. Numerical experiments

Let us illustrate the above developments with several numerical integrations performed with the siconos software of INRIA (see [11,26] and http://siconos.gforge.inria.fr which is designed for the simulation of multivalued nonsmooth systems.

7.1. Chattering free stabilization

Let us consider the following continuous-time closed loop system from [13] given by:
\[
\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -c_1 \end{bmatrix} x - \frac{1}{\alpha} \text{sgn}(\begin{bmatrix} c_1 & 1 \end{bmatrix} x).
\] (53)

As shown in [13] the trajectories obtained by an explicit Euler discretization exhibit spurious oscillations which are described by period-2 cycles around the sliding surface. On Fig. 4, the trajectories obtained by an implicit Euler discretization are shown with \( h = 1, \alpha = 0.3, h = 0.1 \) and \( h = 0.05 \) and with \( c_1 = 1 \) and \( \alpha = 1 \). As it has been predicted by the 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 (e.g. \( h = 1 \)).

7.2. Example 3: multiple sliding surfaces

Let us consider the Example 3. The system can be defined in the form (7) with
\[
B = \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix}, \quad D = 0, \quad f(x(t), t) = 0.
\] (54)

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

7.3. Extensions to ZOH discretized systems

The extension to ZOH discretized systems is illustrated on a first example taken from [14]. In the notation of Section 5.2, the LTI system with an ECB-SMC controller is defined by the following data,
\[
F = \begin{bmatrix} 0 & 1 \\ -a_1 & -a_2 \end{bmatrix}, \quad G = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C = \begin{bmatrix} c_1 \end{bmatrix}.
\] (55)

Starting from the initial data, \( x_0 = [0.55, 0.55]^T \), Galis and Yu [14] 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 Fig. 6(a). On Fig. 6(b), the Implicit ZOH discretization as proposed in Section 5.2 is free of chattering.

7.4. Lyapunov-based robust control

We propose in this section to give a numerical example which fits with (12). Let us consider the following system
\[
\dot{x}(t) = -A(x(t) - u(t) + y(t)),
\] (56)
with $y(t) = \alpha \sin(t)$ and $u(t) = \text{sgn}(x(t))$. As expected, the implicit method yields a smooth stabilization at $x = 0$ (see Fig. 7(a)) whereas the explicit Euler has significant chattering (see Fig. 7(b) and 7(d)). Fig. 7(c) illustrates the fact that the controller varies inside the multivalued part of the sign function in order to assure the existence of an equilibrium point.
Fig. 5. Multiple sliding surface. $h = 0.02, x(0) = [1.0, -1.0]^T$.

Fig. 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 $x_2(t)$.

Fig. 7. Lyapunov-based discontinuous robust control. $h = 0.1, \alpha = 0.1$. 
More simulation results may be found in the report [28], where in particular it is illustrated that the implicit Euler method can handle the Zeno phenomenon.

8. Conclusions

In this paper the backward Euler method is studied in specific classes of Filippov’s systems that encompass sliding mode control systems. It is shown that such implicit schemes permit a smooth accurate stabilization on the sliding surface, even for codimensions 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 [11], 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