On the Discretization of a Class of Homogeneous Differentiators

J. E. Carvajal-Rubio and A. G. Loukianov
Dept. of Electrical Engineering
CINVESTAV-IPN, Guadalajara
Zapopan, México
{jecarvajal,louk}@gdl.cinvestav.mx

J. D. Sánchez-Torres
Dept. of Mathematics and Physics
ITESO
Tlaquepaque, México
dsanchez@iteso.mx

M. Defoort
LAMIH, CNRS UMR 8201
Polytechnic University of Hauts-de-France
Valenciennes, France
michael.defoort@uphf.fr

***Abstract*—This paper proposes explicit and implicit discrete-time realizations for a class of homogeneous sliding-mode-based differentiators. The proposed approach relies on the exact discretization of the continuous differentiator. Also, it is demonstrated that the proposed implicit discretization always exists, is non-anticipative and unique. A numerical simulation shows the better performance of the implicit scheme over the proposed and the referenced explicit implementations.**

***Index Terms*—Exact Discretization, Implicit Discretization, Differentiator, Sliding-Mode**

I. Introduction

Sliding-modes are widely used to design and implement observer [1]–[4] due to their finite-time convergence, accuracy and robustness properties with respect to uncertainties [5]. One of its main disadvantages is the chattering effect [5]. An observer-based on sliding-modes is the high-order sliding mode homogeneous differentiator presented in [6]. It estimates the first n derivatives of a signal if its $(n+1)$ order derivative has a known bound. Moreover, this differentiator has shown good robustness properties in the presence of noises, and exact finite-time convergence in the absence of noises. The structure of this observer has been implemented in different areas in [7]–[9].

The controllers and observers are usually discretized to be implemented in a digital device. There are plenty of reasons to use those discrete-time approximations, in the form of the discrete-time algorithms, instead of the direct application of continuous ones. However, for the case of high-gain and sliding mode based methods, an inadequate discretization can lead to numerical chattering [10], [11] i.e., spurious oscillations due only to the numerical methods used in the discretization procedure. In order to avoid the numerical chattering effect, the design of discrete-time sliding modes has been analyzed and applied in several works [10]–[15]. Recently, concerning the homogeneous differentiator [6], some explicit discrete-time observers have been presented [16]–[18]. These schemes have been proposed such that the discrete-time differentiator preserves the estimation accuracy properties. In [17], a discrete-time differentiator is shown to be less sensitive to gain overestimation. In [18], a discrete-time system has been proposed, where nonlinear higher-order terms are used. Furthermore, it preserves the asymptotic accuracy properties for sampling and noise known from the continuous-time algorithms.

Other schemes of discretization rely on the implicit discretization [19]–[22], which have been shown to guarantee a smooth stabilization of the discrete sliding surface in the disturbance-free case, hence avoiding the chattering effects due to the time discretization.

In this paper, in order to obtain a higher accuracy performance when a low sampling frequency is used, two discretization algorithms for the high-order sliding mode homogeneous differentiator are derived. The first one is obtained by using the explicit exact discretization [23]. The second one is based on the implicit discretization, and it is shown to be able to avoid the chattering effects.

This paper is organized as follows. In Section II, important notations for the next sections are given. Also, it gives a summary of the continuous-time differentiator presented in [6]. In Section III, an explicit discrete-time differentiator and an implicit discrete-time differentiator are presented. It is also demonstrated that, for the implicit discretization, its well-posedness (non-anticipative, existence, and uniqueness of the controller). Section IV shows a comparison of the two discrete-time differentiators proposed in this work against the explicit discrete differentiator proposed in [16]. The conclusions are shown in Section V.

II. Preliminaries

A. Notation

For $x \in \mathbb{R}$, the absolute value of x, denoted by $|x|$, is defined as $|x| = x$ if $x \geq 0$ and $|x| = -x$ if $x < 0$. The set-valued function $\operatorname{sign}(x)$ is defined as $\operatorname{sign}(x) = 1$ if $x > 0$, $\operatorname{sign}(x) = -1$ if $x < 0$, and $\operatorname{sign}(x) \in [-1, 1]$ if $x = 0$. For $\gamma \geq 0$, the signed power γ of x is defined as $\lfloor x \rceil^{\gamma} = |x|^{\gamma} \operatorname{sign}(x)$. In particular, if $\gamma = 0$ then $\lfloor x \rceil^{\gamma} = \operatorname{sign}(x)$.

For the closed non empty convex set $[-1, 1] \subseteq \mathbb{R}$, the normal cone at $s \in [-1, 1]$, denoted as $\mathcal{N}_{[-1,1]}(s)$, is defined as $\mathcal{N}_{[-1,1]}(-1) = \mathbb{R}^-$, $\mathcal{N}_{[-1,1]}(1) = \mathbb{R}^+$, and $\mathcal{N}_{[-1,1]}(s) = 0$ for $s \in (-1, 1)$. Note that, by using a convex analysis, it follows

The authors sincerely thank CONACyT for the scholarship provided during this investigation to the student with No. CVU 555845, and to CINVESTAV for the provided resources.

$\mathcal{N}_{[-1,1]}(s)$ is the inverse of the sign set-valued function. With $y \in \mathbb{R}$, it can be represented as:

$$x \in \text{sign}(y) \Leftrightarrow y \in \mathcal{N}_{[-1,1]}(x).$$

A diagonal matrix with the elements on the main diagonal equal r_i is represented as $\text{diag}\{r_i\}_{i=0}^{n}$. For strictly positive numbers r_i, $i = 0, 1, 2, ..., n$ and $\lambda > 0$, the vector of weights $\boldsymbol{r}$ is defined as $\boldsymbol{r} = (r_0, r_1, ..., r_n)^T$ where the elements r_i are positive. The dilation matrix function is defined as $\boldsymbol{\Lambda_r}(\lambda) = \text{diag}\{\lambda^{r_i}\}_{r_i=0}^{n}$, where the elements r_i correspond to the element of the vector of weight $\boldsymbol{r}$. Furthermore $\forall \boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{\Lambda}_r(\lambda)\boldsymbol{x} = (\lambda^{r_0}x_0, ..., \lambda^{r_n}x_n)^T$.

As in [24] the following definitions is presented. A vector field $\boldsymbol{g} : \mathbb{R}^n \to \mathbb{R}^n$ is r-homogeneous with degree $v \in \mathbb{R}$, where $v \geq -(\min\{r_i | 0 \leq i \leq n\})$, if $\forall \boldsymbol{x} \in \mathbb{R}^n$ and $\forall \lambda > 0$ the following equation is satisfied:

$$\lambda^{-v}\boldsymbol{\Lambda_r}^{-1}(\lambda)\,\boldsymbol{g}(\boldsymbol{\Lambda_r}(\lambda)\boldsymbol{x}) = \boldsymbol{g}(\boldsymbol{x}) \tag{1}$$

.

B. High-order sliding Mode homogeneous differentiator

The objective of a high-order sliding mode differentiator is to obtain the first n derivatives of a function online. In this paper, this function is represented as $f_0(t)$, where $f_0 : \mathbb{R} \to \mathbb{R}$. It is also assumed that this function is at least $(n+1) - th$ differentiable and $|f_0^{(n+1)}(t)| \leq L$ for a known real number $L > 0$. Furthermore, the input of the differentiator is defined as $f(t) = f_0(t) + \Delta(t)$. It is also assumed that $\Delta(t)$ is a Lebesgue-measurable bounded noise with $|\Delta(t)| \leq \delta$ for an unknown real number δ.

In order to calculate the derivatives $f_0^{(1)}(t)$, $f_0^{(2)}(t)$, $\cdots$, $f_0^{(n)}(t)$ a state space representation is used. To obtain this representation, the states variables are defined as $x_i = f_0^{(i)}(t)$ and $\boldsymbol{x} = \begin{bmatrix} x_0 & x_1 & x_2 & \cdots & x_n \end{bmatrix}^T \in \mathbb{R}^{n+1}$. Therefore, we obtain the following representation for the differentiation problem in the state space:

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{e}_{n+1}f_0^{(n+1)}(t); \quad y_o = \boldsymbol{e}_1^T\boldsymbol{x} \tag{2}$$

with the canonical vectors $\boldsymbol{e}_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \end{bmatrix}^T$, $\boldsymbol{e}_{n+1} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix}^T$ and $\boldsymbol{A}$ is the following nilpotent matrix:

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}$$

The representation (2) is interesting in the sense that the successive time derivatives of $f_0(t)$ can be obtained through the design of a state observer. Previously in [6], the following observer has been proposed to estimate the first n derivatives of $f_0(t)$:

$$\begin{aligned} \dot{z}_0 &= -\lambda_n L^{\frac{1}{n+1}} \lfloor z_0 - f_0(t) \rceil^{\frac{n}{n+1}} + z_1 \\ \dot{z}_1 &= -\lambda_{n-1} L^{\frac{2}{n+1}} \lfloor z_0 - f_0(t) \rceil^{\frac{n-1}{n+1}} + z_2 \\ &\vdots \\ \dot{z}_n &= -\lambda_0 L \lfloor z_0 - f_0(t) \rceil^0 \end{aligned} \tag{3}$$

where $\boldsymbol{z} = \begin{bmatrix} z_0 & z_1 & z_2 & \ldots & z_n \end{bmatrix}^T$ estimates the state vector $\boldsymbol{x}$ in finite time for adequate parameters $\lambda_i > 0$. Moreover, in the presence of noise, the differentiator (3) is implemented by using $f(t)$ instead of $f_0(t)$.

For observer (3), sequences of parameters λ_i are presented in [25] for any n integer such that $0 \leq n \leq 7$. On the other hand, the parameters λ_i are not unique due to the fact that the sequences are built by using any $\lambda_0 > 1$ [6]. Different sequences are presented in [26], which are defined for $1 \leq n \leq 10$. Defining the estimation errors as $\sigma_i = z_i - x_i$ for $i = 0, 1, \ldots, n$. The differentiator (3) can be rewritten as:

$$\dot{\boldsymbol{z}} = \boldsymbol{A}\boldsymbol{z} + \boldsymbol{B}\boldsymbol{u}(\sigma_0) \tag{4}$$

where $\boldsymbol{B}$ is the $((n+1) \times (n+1))$ identity matrix, $\boldsymbol{u}(\sigma_0)$ is considered as the input vector of the observer and is defined as follows:

$$\begin{aligned} \boldsymbol{u}(\sigma_0) &= [\Psi_0(\sigma_0)\ \Psi_1(\sigma_0)\ \ldots\ \Psi_n(\sigma_0)]^T \\ \Psi_i(\sigma_0) &= -\lambda_{n-i} L^{\frac{i+1}{n+1}} \lfloor \sigma_0 \rceil^{\frac{n-i}{n+1}} \end{aligned} \tag{5}$$

with $\sigma_0 = z_0 - f_0(t)$. The error vector is defined as $\boldsymbol{\sigma} = \boldsymbol{z} - \boldsymbol{x}$ and the error dynamics can be represented as the following system:

$$\dot{\boldsymbol{\sigma}} = \boldsymbol{A}\boldsymbol{\sigma} + \boldsymbol{B}\boldsymbol{u}(\sigma_0) - \boldsymbol{e}_{n+1}f_0^{n+1}(t) = \boldsymbol{g}(\boldsymbol{\sigma})$$

Notice that the vector field $\boldsymbol{g}(\boldsymbol{\sigma})$, with $v = -1$ and the vector of weights $\boldsymbol{r} = (n+1, n, \ldots, 2, 1)$, satisfies Equation (1). Hence $\boldsymbol{g}(\boldsymbol{x})$ is r-homogeneous with degree -1.

III. Main Results

In this section, two new discretization algorithms are proposed for the homogeneous differentiator (3): the explicit exact one and the implicit exact one. For the implicit exact discretization algorithm, the methodology is based on [20]. Until now, for sliding mode controllers, this technique has been only implemented for first order sliding mode controllers [19], sliding mode twisting controllers [21] [22] and super-twisting controllers [20] [22].

A. Explicit Exact Discretization Algorithm

First, signal $f_0(t)$ is measured at the time instants $t_k = k\tau$ for $k = 0, 1, 2, 3, \cdots$, where τ is the sampling time and is assumed to be constant. Let us denote $x_i(t_k) = x_{i,k}$, $z_i(t_k) = z_{i,k}$ and $\sigma_i(t_k) = \sigma_{i,k}$. Furthermore, the vector of inputs of the observer is considered constant over the time interval $[t_k, t_{k+1})$ and is defined by using (5) as $\boldsymbol{u}_k = \boldsymbol{u}(\sigma_{0,k})$. Under these assumptions, an explicit exact discretization is performed for system (3) (see [23] for detailed explanations).

As $\boldsymbol{A}$ is a nilpotent matrix, then the following discrete-time observer is obtained:

$$\begin{aligned} \boldsymbol{z}_{k+1} &= \boldsymbol{\Phi}(\tau)\boldsymbol{z}_k + \boldsymbol{B}^*(\tau)\boldsymbol{u}_k \\ \boldsymbol{u}_k &= \left[\Psi_0(\sigma_{0,k}) \;\; \Psi_1(\sigma_0,k) \;\; \cdots \;\; \Psi_n(\sigma_0,k)\right]^T \\ \Psi_i(\sigma_{0,k}) &= -\lambda_{n-i}L^{\frac{i+1}{n+1}}\lfloor\sigma_{0,k}\rceil^{\frac{n-i}{n+1}} \end{aligned} \tag{6}$$

where $\boldsymbol{\Phi}(\tau)$ and $\boldsymbol{B}^*(\tau)$ have the following representation:

$$\boldsymbol{\Phi}(\tau) = \begin{bmatrix} 1 & \tau & \frac{\tau^2}{2!} & \cdots & \frac{\tau^{n-1}}{(n-1)!} & \frac{\tau^n}{n!} \\ 0 & 1 & \tau & \cdots & \frac{\tau^{n-2}}{(n-2)!} & \frac{\tau^{n-1}}{(n-1)!} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & \tau \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix}$$

$$\boldsymbol{B}^*(\tau) = \begin{bmatrix} \tau & \frac{\tau^2}{2!} & \frac{\tau^3}{3!} & \cdots & \frac{\tau^n}{n!} & \frac{\tau^{n+1}}{(n+1)!} \\ 0 & \tau & \frac{\tau^2}{2!} & \cdots & \frac{\tau^{n-1}}{(n-1)!} & \frac{\tau^n}{n!} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \tau & \frac{\tau^2}{2!} \\ 0 & 0 & 0 & \cdots & 0 & \tau \end{bmatrix}$$

The discrete-time differentiator (6) has a similar structure to those proposed in [16]–[18]. The main difference is that in [16], $\boldsymbol{B}^*(\tau)$ is given as $\boldsymbol{B}^*(\tau) = \tau\boldsymbol{B}$ and in [17], $\boldsymbol{B}^*(\tau)\boldsymbol{u}_k$ are injection terms obtained by placing the eigenvalues of the discrete error system. In [18], the explicit discrete-time differentiator can be in the form of (6) with a different matrix $\boldsymbol{B}^*(\tau)$.

Since function $f_0^{(n+1)}(t)$ can be variable over the time interval $[t_k, t_{k+1})$, then a discrete-time system of (2) is obtained by using Taylor series expansion with Lagrange's remainders [27], similarly to [17]:

$$\boldsymbol{x}_{k+1} = \boldsymbol{\Phi}(\tau)\boldsymbol{x}_k + \boldsymbol{h} \tag{7}$$

with

$$\boldsymbol{h} = \begin{bmatrix} \frac{\tau^{n+1}}{(n+1)!}f_0^{(n+1)}(\xi_n) & \frac{\tau^n}{n!}f_0^{(n+1)}(\xi_{n-1}) \cdots \tau f_0^{(n+1)}(\xi_0) \end{bmatrix}^T$$

where $\xi_i \in (t_k, t_{k+1})$ and $|f_0^{(n+1)}(\xi_i)| \le L$.

By using Equations (6) and (7), the discrete form of the error system is given as:

$$\boldsymbol{\sigma}_{k+1} = \boldsymbol{\Phi}(\tau)\boldsymbol{\sigma}_k + \boldsymbol{B}^*(\tau)\boldsymbol{u}_k - \boldsymbol{h}$$

B. Implicit discretization algorithm

Let us consider the same assumptions that in the previous subsection. However, here, the law control uses $\sigma_{0,k+1}$ instead of $\sigma_{0,k}$. A discretization is performed on system (4) by defining the control law as:

$$\begin{aligned} \boldsymbol{u}_{k+1} &= \left[\Psi_{0,k+1} \;\; \Psi_{1,k+1} \;\; \cdots \;\; \Psi_{n,k+1}\right]^T \\ \Psi_{i,k+1} &\in -\lambda_{n-i}L^{\frac{i+1}{n+1}}\lfloor\sigma_{0,k+1}\rceil^{\frac{n-i}{n+1}} \end{aligned}$$

In this manner, the following discretizations are obtained:

$$\begin{aligned} \boldsymbol{z}_{k+1} &= \boldsymbol{\Phi}(\tau)\boldsymbol{z}_k + \boldsymbol{B}^*(\tau)\boldsymbol{u}_{k+1} \\ \boldsymbol{\sigma}_{k+1} &= \boldsymbol{\Phi}(\tau)\boldsymbol{\sigma}_k + \boldsymbol{B}^*(\tau)\boldsymbol{u}_{k+1} - \boldsymbol{h} \end{aligned} \tag{8}$$

Notice that $\sigma_{0,k+1}$ cannot be calculated by using the second equation of (8) due to the impossibility to measure the states $x_{1,k}$, $x_{2,k}$, $\cdots$, $x_{n,k}$, vector $\boldsymbol{h}$ and $f_0^{(n+1)}(\xi_i)$. Therefore these terms are considered as perturbations for the estimation process of $\sigma_{0,k+1}$. In order to implement observer (8), the intermediate variable $\widetilde{\sigma}_{0,k+1}$ is defined as in [20] and the following unperturbed equation is proposed:

$$\begin{aligned} \widetilde{\sigma}_{0,k+1} &= \sigma_{0,k} + \tau\widetilde{\Psi}_{0,k} + \sum_{m=1}^{n}\frac{\tau^m}{m!}\left(z_{m,k} + \frac{\tau}{m+1}\widetilde{\Psi}_{m,k}\right) \\ \widetilde{\Psi}_{i,k+1} &\in -\lambda_{n-i}L^{\frac{i+1}{n+1}}|\widetilde{\sigma}_{0,k+1}|^{\frac{n-i}{n+1}}\operatorname{sign}(\widetilde{\sigma}_{0,k+1}) \end{aligned} \tag{9}$$

To compute $\widetilde{\sigma}_{0,k+1}$, Equation (9) is rewritten as follows:

$$\begin{aligned} &\widetilde{\sigma}_{0,k+1} + a_n\lfloor\widetilde{\sigma}_{0,k+1}\rceil^{\frac{n}{n+1}} + \cdots + a_1\lfloor\widetilde{\sigma}_{0,k+1}\rceil^{\frac{1}{n+1}} + \ldots \\ &\quad + b_k \in -a_0\operatorname{sign}(\widetilde{\sigma}_{0,k+1}) \end{aligned}$$

where $b_k = -\sigma_{0,k} - \sum_{m=1}^{n}\frac{\tau^m}{m!}z_{m,k}$ and $a_l = \frac{\tau^{n-l+1}}{(n-l+1)!}\lambda_l L^{\frac{n-l+1}{n+1}}$. A new support variable is introduced: $\xi_{k+1} \in \operatorname{sign}(\widetilde{\sigma}_{0,k+1})$. Then $\widetilde{\sigma}_{0,k+1}$ and ξ_{k+1} are the solutions of the generalized equations:

$$\begin{aligned} G(\widetilde{\sigma}_{0,k+1}) &\in -a_0\operatorname{sign}(\widetilde{\sigma}_{0,k+1}) \\ F(-a_0\xi_{k+1}) &\in \mathcal{N}_{[-1,1]}(\xi_{k+1}) \end{aligned}$$

with $G(x)$ and $F(y)$, for $x \in \mathbb{R}$ and $y \in \mathbb{R}$, defined as:

$$\begin{aligned} G(x) &= x + a_n\lfloor x\rceil^{\frac{n}{n+1}} + \cdots + a_1\lfloor x\rceil^{\frac{1}{n+1}} + b_k \\ F(y) &= G^{-1}(y) \end{aligned}$$

- If $y > b_k$, then $F(y) = (r_0)^{n+1}$

$$p(r) = r^{n+1} + a_n r^n + \cdots + a_1 r + (b_k - y) \tag{12}$$

- If $y < b_k$ then $F(y) = -(r_0)^{n+1}$

$$p(r) = r^{n+1} + a_n r^n + \cdots + a_1 r - (b_k - y) \tag{13}$$

- If $y = b_k$ then $F(y) = 0$.

where r_0 is the positive root of Equations (12) and (13) respectively. Notice that (12) and (13) are polynomial equations without zero coefficients and their signal patterns have one sign change due to the positive parameters a_i and the conditions given above. One sign change in Equations (12) and (13) indicates that each equation has only one positive root due to the Descartes' rule of signs [28].

Lemma 1: For parameters $a_i \in \mathbb{R}^+$ and $b_k \in \mathbb{R}$, there is an unique pair $\widetilde{\sigma}_{0,k+1} \in \mathbb{R}$ and $\xi_{k+1} \in [-1,1]$ which are solution of the generalized Equations (10) and (11), given as follows:

- **Case 1**: $b_k > a_0$

$\xi_{k+1} = -1$, $\widetilde{\sigma}_{0,k+1} \in \mathbb{R}^-$ and is determined by $\widetilde{\sigma}_{0,k+1} = -(r_0)^{n+1}$, where r_0 is the unique positive root of the polynomial equation (14) given hereafter.

- **Case 2**: $b_k \in [-a_0, a_0]$

$\widetilde{\sigma}_{0,k,+1} = 0$ and $\xi_{k+1} = -\frac{b_k}{a_0}$.

- **Case 3**: $b_k < -a_0$

$\xi_{k+1} = 1$, $\widetilde{\sigma}_{0,k+1} \in \mathbb{R}^+$ and is determined by $\widetilde{\sigma}_{0,k+1} = r_0^{n+1}$, where r_0 is the positive root of the polynomial equation (15) given hereafter.

The polynomial equations are the following:

$$p(r) = r^{n+1} + a_n r^n + \cdots + a_1 r + (-b_k + a_0) \quad (14)$$

$$p(r) = r^{n+1} + a_n r^n + \cdots + a_1 r + (b_k + a_0) \quad (15)$$

The proof is given in Appendix A. Notice that the positive root of the polynomial (14), $r_0 \in \left[0, (b_k - a_0)^{\frac{1}{n+1}}\right]$ and $r_0 \in \left[0, (-b_k - a_0)^{\frac{1}{n+1}}\right]$ for Case 3. Therefore, the discrete-time observer is expressed as follows:

$$\begin{aligned} \boldsymbol{z}_{k+1} &= \boldsymbol{\Phi}(\tau)\,\boldsymbol{z}_k + \boldsymbol{B}^*(\tau)\,\boldsymbol{v}_{k+1} \\ \boldsymbol{v}_{k+1} &= \left[\widetilde{\Psi}_{0,k+1} \;\; \widetilde{\Psi}_{1,k+1} \;\; \cdots \;\; \widetilde{\Psi}_{n,k+1}\right]^T \\ \widetilde{\Psi}_{i,k+1} &= -\lambda_{n-i} L^{\frac{i+1}{n+1}} \left|\widetilde{\sigma}_{0,k+1}\right|^{\frac{n-i}{n+1}} \xi_{k+1} \end{aligned} \quad (16)$$

As ξ_{k+1} and $\widetilde{\sigma}_{0,k+1}$ are calculated using only a_i, $\sigma_{0,k}$ and $z_{i,k}$, then the differentiator (16) is non-anticipative. Furthermore, r_0 is unique for the case 1 and 3 therefore $\widetilde{\sigma}_{0,k+1}$ and ξ_{k+1} always exist $\forall b_k \in \mathbb{R}$ and $a_0 \neq 0$. In order to implement observer (16), for each iteration k, the variables $\widetilde{\sigma}_{0,k+1}$ and ξ_{k+1} are calculated as in Lemma 1. The estimation error of observer (16) is expressed as follows:

$$\boldsymbol{\sigma}_{k+1} = \boldsymbol{\Phi}(\tau)\,\boldsymbol{\sigma}_k + \boldsymbol{B}^*(\tau)\,\boldsymbol{v}_{k+1} - \boldsymbol{h} \quad (17)$$

An attractive characteristic of this discrete-time system is that by using the variables $\epsilon_{i,k} = z_{i,k+1} - x_{i,k}$, for $0 \leq i \leq n$ and defining $\boldsymbol{\epsilon}_k = [\epsilon_{0,k} \; \epsilon_{1,k} \; \cdots \; \epsilon_{n,k}]^T$ the following equations are obtained:

$$\begin{aligned} \boldsymbol{\epsilon}_k &= (\boldsymbol{\Phi}(\tau) - I)\boldsymbol{z}_k + \boldsymbol{B}^*(\tau)\,\boldsymbol{v}_{k+1} + \boldsymbol{\sigma}_k \\ \boldsymbol{\epsilon}_k &= \boldsymbol{\Phi}(\tau)\,\boldsymbol{\sigma}_k + \boldsymbol{B}^*(\tau)\,\boldsymbol{v}_{k+1} + (\boldsymbol{\Phi}(\tau) - I)\,\boldsymbol{x}_k \end{aligned} \quad (18)$$

By using Equations (17) and (18), the following implicit discrete-time error system is obtained:

$$\begin{aligned} \boldsymbol{\epsilon}_{k+1} &= \boldsymbol{\Phi}(\tau)\,\boldsymbol{\epsilon}_k + \boldsymbol{B}^*(\tau)\,\boldsymbol{v}_{k+2} - \boldsymbol{h} \\ \boldsymbol{v}_{k+2} &= \left[\widetilde{\Psi}_{0,k+2} \;\; \widetilde{\Psi}_{1,k+2} \;\; \cdots \;\; \widetilde{\Psi}_{n,k+2}\right]^T \\ \widetilde{\Psi}_{i,k+2} &= -\lambda_{n-i} L^{\frac{i+1}{n+1}} \left|\epsilon_{0,k+1}\right|^{\frac{n-i}{n+1}} \xi_{k+2} \\ \xi_{k+2} &\in \operatorname{sign}(\epsilon_{0,k+1}) \end{aligned} \quad (19)$$

IV. Simulation Results

In this section, some simulations are performed for $n = 3$ using the implicit discrete-time differentiator (16), the explicit discrete-time differentiator (6) and the n^{th}−order homogeneous discrete-time differentiator (HDD), which was proposed in [16]. These simulations were done in MATLAB and the command `roots` was used for the implementation of the implicit differentiator. For (16), $z_{i,k+1}$ is compared against its corresponding $x_{i,k}$, due to the implicit discrete-time system (19). In contrast to (16) for the differentiators (6) and the HDD, each $z_{i,k}$ is compared against its corresponding $x_{i,k}$. For these simulations, the noise-free input $f_0(t) = \sin(t) - \cos(0.5t)$ is used. Furthermore, the same parameters $L = 2$, $\tau = 0.2\,\text{sec}$, $\lambda_0 = 1.1$, $\lambda_1 = 3.06$, $\lambda_2 = 4.16$, $\lambda_3 = 3$ are used for each differentiator. The above parameters λ_i are obtained from [25]. Furthermore, the initial condition for each differentiator is given as $\boldsymbol{z}_0 = [0, 0, 0, 0]^T$.

In Figures 1 2, 3 and 4, the best performance is shown using the implicit differentiator, followed by the explicit differentiator and the HDD with similar performances. In order to shown this in a clearer way, the estimation errors are depicted in Figures 5, 6, 7 and 8.

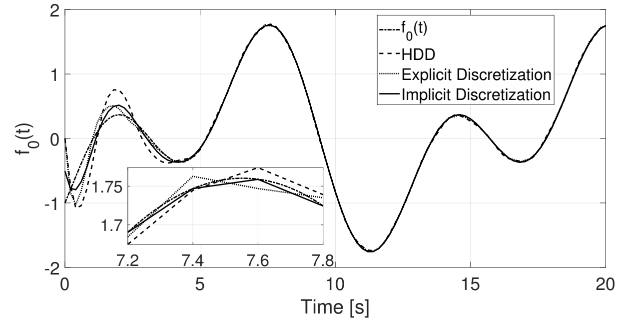

Fig. 1. Estimation of the noise-free input.

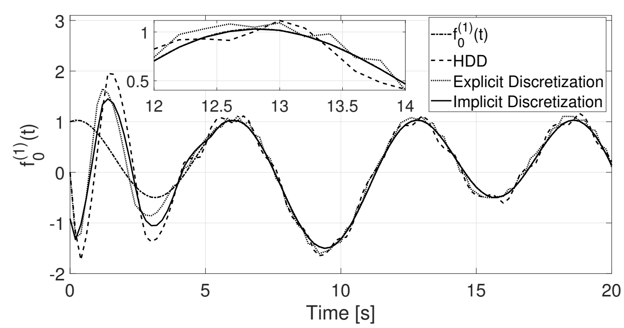

Fig. 2. Estimation of the first derivative of the noise-free input.

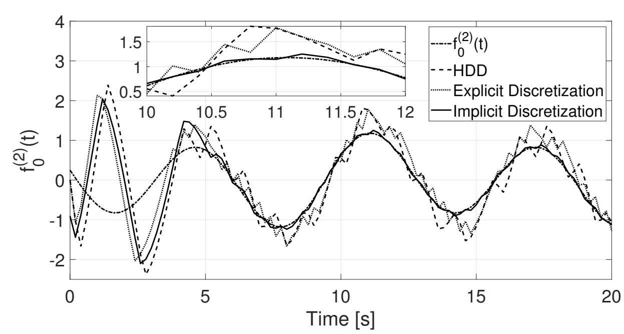

Fig. 3. Estimation of the second derivative of the noise-free input.

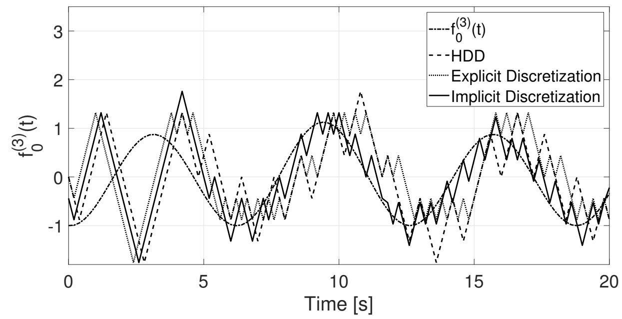

Fig. 4. Estimation of the third derivative of the noise-free input.

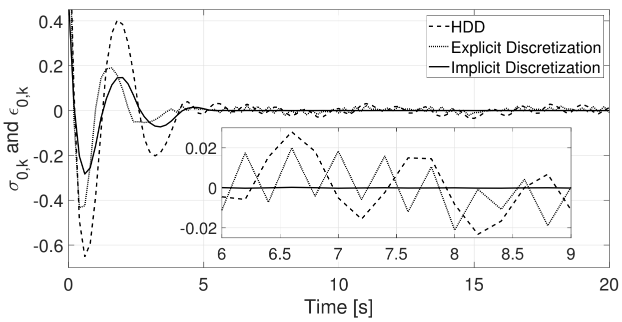

Fig. 5. Estimation error for the noise-free input.

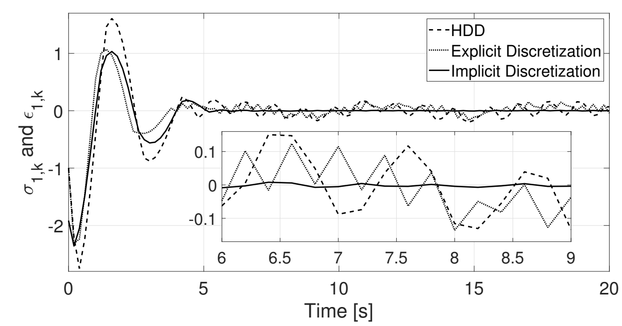

Fig. 6. Estimation error for the first derivative of the noise-free input.

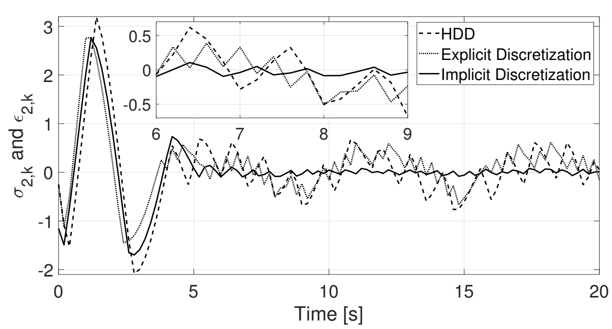

Fig. 7. Estimation error for the second derivative of the noise-free input.

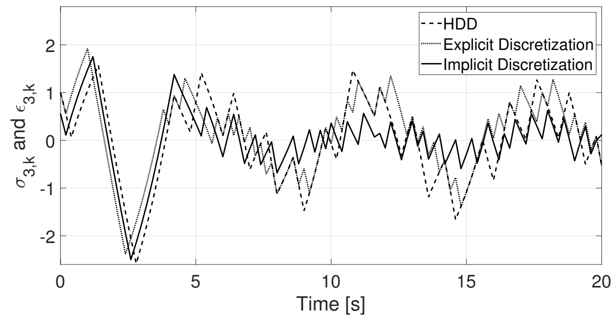

Fig. 8. Estimation error for the third derivative of the noise-free input.

Finally, from these simulations, three performance indexes are considered to compare the differentiators. The first index is defined as $y_i = \max\{|\sigma_{i,k}| \in \mathbb{R} | 10\,\text{sec} \leq t_k \leq 20\,\text{sec}\}$, for (16), the terms y_i are calculated by replacing $\sigma_{i,k}$ by $\epsilon_{i,k}$. The second index Y_i is the standard deviation of $\sigma_{0,k+1}$ or $\epsilon_{0,k+1}$, in the time interval $10\,\text{sec} \leq t_k \leq 20\,\text{sec}$. The third index is the arithmetic mean μ_i for $10 \leq t_k \leq 20$ and $\sigma_{i,k}$ or $\epsilon_{i,k}$, respectively. These variables are summarized in Table I. From this table, the proposed implicit differentiator shows the lowest maximum error, standard deviation and arithmetic mean for each state. Meanwhile, the proposed explicit differentiator shows a lower maximum error and standard deviation than the HDD. On the other hand, the proposed explicit differentiator shows the highest arithmetic mean.

TABLE I
PERFORMANCE INDEXES FOR $10\,\text{sec} \leq t_k \leq 20\,\text{sec}$.

Maximum Error	**Implicit**	**Explicit**	**HDD**
y_0	0.000139	0.023474	0.034911
y_1	0.005502	0.163731	0.201379
y_2	0.082027	0.667947	0.766403
y_3	0.623977	1.383014	1.66018
Standard Deviation			
Y_0	5.79107×10^{-5}	0.010852	0.016775
Y_1	0.002759	0.074249	0.099873
Y_2	0.042919	0.294402	0.37107
Y_3	0.303047	0.645008	0.743021
Arithmetic Mean			
μ_0	6.776782×10^{-6}	0.00359	0.000227
μ_1	2.5372×10^{-4}	0.034687	0.001264
μ_2	0.004209	0.151629	0.016899
μ_3	0.035425	0.30986	0.081065

V. CONCLUSION

This paper presented an exact discretization approach for the high-order sliding mode homogeneous differentiator. This procedure allows the obtention of two realizations of such differentiator; namely, explicit and implicit expressions. As a consequence of this method and an additional analysis, it follows the main result of this paper. It is an implicit discretization method as an implementation strategy of that continuous differentiator. The proposed implicit and exact discretization for the differentiator shows better accuracy properties than the proposed exact explicit differentiator and existing algorithms. Future works will address a demonstration of convergence for the proposed differentiators in the presence of measurement noise.

References

[1] J. Davila, L. Fridman, and A. Levant, "Second-order sliding-mode observer for mechanical systems," *IEEE Transactions on Automatic Control*, vol. 50, no. 11, pp. 1785–1789, 2005.

[2] I.-S. Kim, "A technique for estimating the state of health of lithium batteries through a dual-sliding-mode observer," *IEEE Transactions on Power Electronics*, vol. 25, no. 4, pp. 1013–1022, 2010.

[3] A.-M. Lienhardt, G. Gateau, and T. A. Meynard, "Digital sliding-mode observer implementation using fpga," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 4, pp. 1865–1875, 2007.

[4] C. Ren, X. Li, X. Yang, and S. Ma, "Extended state observer based sliding mode control of an omnidirectional mobile robot with friction compensation," *IEEE Transactions on Industrial Electronics*, 2019.

[5] C. Edwards and S. Spurgeon, *Sliding mode control: theory and applications*. Crc Press, 1998.

[6] A. Levant, "Higher-order sliding modes, differentiation and output-feedback control," *International journal of Control*, vol. 76, no. 9-10, pp. 924–941, 2003.

[7] P. Kaveh and Y. B. Shtessel, "Blood glucose regulation using higher-order sliding mode control," *International Journal of Robust and Nonlinear Control*, vol. 18, no. 4-5, pp. 557–569, 2008.

[8] Y. B. Shtessel, I. A. Shkolnikov, and A. Levant, "Smooth second-order sliding modes: Missile guidance application," *Automatica*, vol. 43, no. 8, pp. 1470–1476, 2007.

[9] M. Iqbal, A. I. Bhatti, S. I. Ayubi, and Q. Khan, "Robust parameter estimation of nonlinear systems using sliding-mode differentiator observer," *IEEE Transactions on industrial electronics*, vol. 58, no. 2, pp. 680–689, 2010.

[10] S. V. Drakunov and V. Utkin, "On discrete-time sliding modes," in *Nonlinear Control Systems Design*. Elsevier, 1990, pp. 273–278.

[11] V. I. Utkin, "Sliding mode control in discrete-time and difference systems," in *Variable Structure and Lyapunov Control*. Springer, 1994, pp. 87–107.

[12] A. Šabanović, N. Šabanović, and K. Jezernik, "Sliding modes in sampled-data systems," in *Automatika*. Publisher KoREMA-IFAC, 2003, vol. 44, pp. 3–4.

[13] W.-C. Su, S. V. Drakunov, and U. Ozguner, "An $o(t^2)$ boundary layer in sliding mode for sampled-data systems," *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 482–485, 2000.

[14] T. Nguyen, V. Azimi, W.-C. Su, and C. Edwards, "Improvement of control signals in output feedback sliding mode control of sampled-data systems," in *2017 American Control Conference (ACC)*. IEEE, 2017, pp. 5762–5767.

[15] K. Abidi, J.-X. Xu, and Y. Xinghuo, "On the discrete-time integral sliding-mode control," *IEEE Transactions on Automatic Control*, vol. 52, no. 4, pp. 709–715, 2007.

[16] M. Livne and A. Levant, "Proper discretization of homogeneous differentiators," *Automatica*, vol. 50, no. 8, pp. 2007–2014, 2014.

[17] S. Koch and M. Reichhartinger, "Discrete-time equivalent homogeneous differentiators," *2018 15th International Workshop on Variable Structure Systems (VSS)*, pp. 354–359, July 2018.

[18] S. Koch, M. Reichhartinger, M. Horn, and L. Fridman, "Discrete-time implementation of homogeneous differentiators," *IEEE Transactions on Automatic Control*, vol. 65, no. 2, pp. 757–762, Feb 2020.

[19] V. Acary, B. Brogliato, and Y. V. Orlov, "Chattering-free digital sliding-mode control with state observer and disturbance rejection," *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1087–1101, 2011.

[20] B. Brogliato, A. Polyakov, and D. Efimov, "The implicit discretization of the super-twisting sliding-mode control algorithm," *2018 15th International Workshop on Variable Structure Systems (VSS)*, pp. 349–353, July 2018.

[21] O. Huber, V. Acary, B. Brogliato, and F. Plestan, "Implicit discrete-time twisting controller without numerical chattering: Analysis and experimental results," *Control Engineering Practice*, vol. 46, pp. 129–141, 2016.

[22] D. Luo, X. Xiong, S. Jin, and W. Chen, "Implicit euler implementation of twisting controller and super-twisting observer without numerical chattering: Precise quasi-static mems mirrors control," in *MATEC Web of Conferences*, vol. 256. EDP Sciences, 2019, p. 03004.

[23] N. Kazantzis and C. Kravaris, "Time-discretization of nonlinear control systems via taylor methods," *Computers & chemical engineering*, vol. 23, no. 6, pp. 763–784, 1999.

[24] D. Efimov, A. Polyakov, A. Levant, and W. Perruquetti, "Realization and discretization of asymptotically stable homogeneous systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 5962–5969, 2017.

[25] A. Levant, "Filtering differentiators and observers," *15th International Workshop on Variable Structure Systems (VSS)*, pp. 174–179, July 2018.

[26] M. F. M. Reichhartinger, S.K. Spurgeon and M. Wipfler, "A robust exact differentiator toolbox for Matlab® Simulink®," *IFAC World Congress*, vol. 50, pp. 1711–1716, July 2017.

[27] W. J. Firey, "Remainder formulae in taylor's theorem," *The American Mathematical Monthly*, vol. 67, no. 9, pp. 903–905, 1960.

[28] H. Cheriha, Y. Gati, and V. P. Kostov, "On Descartes' rule of signs," *arXiv preprint arXiv:1905.01836*, 2019.

Appendix

- **Case 1**: $b_k > a_0$

In this case, if $\widetilde{\sigma}_{k+1} = 0$ then the inclusion (10) becomes $b_k \in -a_0 \operatorname{sign}(0)$, but $b_k > a_0$ then $b_k \notin [-a_0, a_0]$. If $\widetilde{\sigma}_{k+1} \in \mathbb{R}^+$ then the left side of the inclusion (10) is rewritten as:

$$\widetilde{\sigma}_{0,k+1} + \cdots + a_1 \left(\widetilde{\sigma}_{0,k+1}\right)^{\frac{1}{n+1}} + b_k = -a_0 \tag{20}$$

Since $b_k > a_0$ and $\left(\widetilde{\sigma}_{0,k+1}\right)^{\frac{n-i}{n+1}} > 0$, then the left-side of the above equation is greater than a_0, which is a contradiction. If $\widetilde{\sigma}_{0,k+1} \in \mathbb{R}^-$ the inclusion (10) is rewritten as follows:

$$\widetilde{\sigma}_{0,k+1} - \cdots - a_1 \left(-\widetilde{\sigma}_{0,k+1}\right)^{\frac{1}{n+1}} + b_k = a_0$$

Using the variable $r = \left(-\widetilde{\sigma}_{0,k+1}\right)^{\frac{1}{n+1}}$, $p\left(r\right) = 0$ is obtained, where $p\left(r\right)$ is the polynomial (14). It has only one positive root due to its signal pattern with one sign change and the coefficients $a_i > 0$ [28]. Notice that $r > 0$ because as $\widetilde{\sigma}_{0,k+1} \in \mathbb{R}^-$ and $|\widetilde{\sigma}_{0,k+1}|^{\frac{1}{n+1}} \in \mathbb{R}^+$ then $|\widetilde{\sigma}_{0,k+1}|^{\frac{1}{n+1}} = \left(-\widetilde{\sigma}_{0,k+1}\right)^{\frac{1}{n+1}} = r$. Hence, $\widetilde{\sigma}_{0,k+1} = -\left(r_0\right)^{n+1}$, where r_0 is the positive root of Equation (14). As $b_k > -a_0\xi_{k+1}$ for all $\xi_{k+1} \in [-1, 1]$ then $F\left(-a_0\xi_{k+1}\right) = c$ for some $c \in \mathbb{R}^-$. The inclusion (11) becomes $\xi_{k+1} \in \operatorname{sign}(c)$, accordingly, $\xi_{k+1} = -1$.

- **Case 3**: $b_k < -a_0$

As in the above case, a demonstration can be performed, where in this case $\xi_{k+1} = 1$ and $\widetilde{\sigma}_{0,k+1} = \left(r_0\right)^{n+1}$, r_0 is the positive root of Equation (15).

- **Case 2**: $b_k \in [-a_0, a_0]$

In this case $-\left(b_k + a_0\right) \leq 0$ and $a_0 - b_k \geq 0$. If $\widetilde{\sigma}_{0,k+1} \in \mathbb{R}^+$ then Equation (20) is obtained. Since $\left(\widetilde{\sigma}_{0,k+1}\right)^{\frac{n-i}{n+1}} > 0$, $-\left(a_0 + b_k\right) > 0$, which is a contradiction. In the same manner, a contradiction is obtained for $\widetilde{\sigma}_{0,k+1} < 0$. As $\widetilde{\sigma}_{0,k+1} = 0$ satisfies the inclusion (10), thereby, $\widetilde{\sigma}_{0,k+1} = 0$, the following inclusion is equivalent to (11):

$$\xi_{k+1} \in \operatorname{sign}(F\left(-a_0\xi_{k+1}\right)) \tag{21}$$

when $\xi_{k+1} > -\frac{b_k}{a_0}$ or $\xi_{k+1} < -\frac{b_k}{a_0}$ contradictions are obtained. $\xi_{k+1} = -\frac{b_k}{a_0}$ is the unique solution for the inclusion (21) for $\xi_{k+1} \in [-1, 1]$, it becomes $-\frac{b_k}{a_0} \in \operatorname{sign}(0)$.