

Continuous Optimization for Control of Hybrid Systems With Hysteresis via Time-Freezing

Armin Nurkanović^{ID} and Moritz Diehl^{ID}

Abstract—This letter regards numerical optimal control of a class of hybrid systems with hysteresis using solely techniques from nonlinear optimization, without any integer variables. Hysteresis is a rate independent memory effect which often results in severe nonsmoothness in the dynamics. These systems are not simply Piecewise Smooth Systems (PSS); they are a more complicated form of hybrid systems. We introduce a time-freezing reformulation which transforms these systems into a PSS. From the theoretical side, this reformulation opens the door to study systems with hysteresis via the rich tools developed for Filippov systems. From the practical side, it enables the use of the recently developed Finite Elements with Switch Detection (Nurkanović *et al.*, 2022), which makes high accuracy numerical optimal control of hybrid systems with hysteresis possible. We provide a time optimal control problem example and compare our approach to mixed-integer formulations from the literature.

Index Terms—Hybrid systems, optimal control, numerical algorithms, software.

I. INTRODUCTION

HYSTERESIS occurs in many physical systems, e.g., ferromagnetism, plasticity, superconductivity, phase transitions, but also in feedback control, e.g., thermostats [2], [3]. Hysteresis effects in dynamic systems are modeled with nonsmooth differential equations. This letter focuses on transforming some classes of systems with hysteresis into piecewise smooth system (PSS) and numerically solving optimal control problems (OCP) with PSS. We leverage recent advances in numerical optimal control of PSS, namely we use the finite elements with switch detection (FESD) method [1].

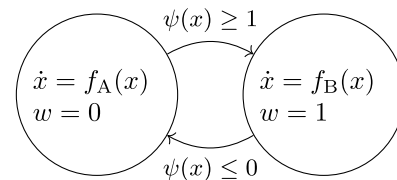
A hybrid system with hysteresis can be represented as a *finite automaton* [3] which has two modes of operation described by $f_A(x)$ and $f_B(x)$, cf. Fig. 1(a). If the system operates in mode A with $\dot{x} = f_A(x)$ and if $\psi(x) \geq 1$, it switches

Manuscript received 21 March 2022; revised 14 May 2022; accepted 31 May 2022. Date of publication 9 June 2022; date of current version 27 June 2022. This work was supported in part by the Deutsche Forschungsgemeinschaft (DFG) through Research Unit FOR 2401 and Project 424107692, and in part by the European Union (EU) through ELO-X under Grant 953348. Recommended by Senior Editor L. Zhang. (Corresponding author: Armin Nurkanović.)

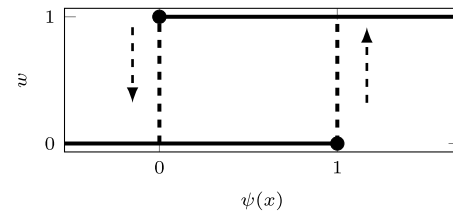
Armin Nurkanović is with the Department of Microsystems Engineering, University of Freiburg, 79110 Freiburg, Germany (e-mail: armin.nurkanovic@imtek.uni-freiburg.de).

Moritz Diehl is with the Department of Microsystems Engineering and the Department of Mathematics, University of Freiburg, 79110 Freiburg, Germany (e-mail: moritz.diehl@imtek.uni-freiburg.de).

Digital Object Identifier 10.1109/LCSYS.2022.3181888



(a) Finite automaton of a hybrid system with hysteresis



(b) Example hysteresis characteristic $(w, \psi(x))$

Fig. 1. Hybrid system with hysteresis.

to mode B with $\dot{x} = f_B(x)$. On the other hand, if it operates in mode B and if $\psi(x) \leq 0$, it switches to mode A. This is a typical hysteresis behavior given by the characteristic in Fig. 1(b), which is often called the *delayed relay operator* [4]. The dynamics of the system depend on the value of $w(t)$ and the scalar *switching function* $\psi(x)$. Notably, for $\psi(x) \in [0, 1]$ the function $w(t)$ can be 0 or 1.

There are several other related characteristics, e.g., the dashed lines in Fig. 1(b) could be solid, or the resulting polygon in the middle of the plot might be tilted. In all these cases the characteristic can be readily represented via a linear complementarity problem [5] and the nonsmooth dynamic system recast into a Dynamic Complementarity System (DCS). However, it is an open question if this DCS is a PSS.

Control of systems with hysteresis relying on Filippov solutions was studied in, e.g., [6], [7]. In control theory, systems with hysteresis are often studied via the hybrid systems framework which uses integer state and control variables [3], [8], [9]. Hence, in an optimal control context this requires solving Mixed Integer Optimization Problems (MIOP). They can be solved efficiently in case of discrete time linear hybrid systems [8] where MILP or MIQP formulations can be found. However, as soon as the junction times need to be determined precisely or non-linearity is present, e.g., in time optimal control problems, solving MIOP can become

arbitrarily difficult. On the other hand, the nonsmoothness can be modeled with complementarity constraints [10] and one must solve only nonsmooth Nonlinear Programs (NLP). However, standard time-stepping schemes for DCS have only first order accuracy and result necessarily in wrong numerical sensitivities and artificial local minima [10], [11].

The *time-freezing* reformulation transforms systems with state jumps into PSS and was first introduced in [12], [13]. This letter introduces a time-freezing reformulation to transform systems represented with the finite automaton in Fig. 1(a) into PSS. Here, the main idea is to regard $w(t)$ as a continuous differential state. However, $w(t)$ exhibits jump discontinuities in time at $(0, 1)$ and $(1, 0)$, which can be interpreted as a *state jump law*. As in [12], [13], we introduce *auxiliary dynamic systems* and a *clock state*. The auxiliary dynamic systems evolve in regions which are prohibited for the initial system and their trajectory endpoints satisfy the state jump law. Additionally, the evolution of the clock state is frozen during the evolution of the auxiliary systems. By regarding only the parts of $w(\cdot)$ when the clock state was evolving, we recover the original discontinuous solution. Note that the resulting time-freezing system is now a PSS, since the only remaining jump discontinuities are in the system's dynamics but not in the state anymore. For high accuracy numerical optimal control of PSS we use the FESD method [1]. An implementation is available in the open source software package NOSNOC [14], [15].

Contribution: We present a time-freezing reformulation for a class of hybrid systems with hysteresis, which transforms them into PSS. Constructive ways for finding the auxiliary dynamics needed in time-freezing are provided. Solution equivalence between the initial hybrid and time-freezing PSS are proven. From the theoretical side, this contribution enables one to treat hybrid systems with hysteresis with the tools for PSS and Filippov systems [16]. From the practical side, the highlight of this letter is that we can solve OCP with systems with hysteresis with high accuracy and without the use of any integer variables. The OCP discretized via FESD result in Mathematical Programs with Complementarity Constraints (MPCC). With appropriate reformulations the MPCC can often be solved by only a few NLP solves [17], i.e., the highly nonsmooth and nonlinear OCP are solved by purely derivative based algorithms. A time optimal control problem of a hybrid system with hysteresis and illustrates theoretical and algorithmic developments. We compare the continuous optimization-based FESD method to mixed integer solution strategies.

Outline: Section II gives some basic definitions on hybrid systems with hysteresis and PSS. In Section III we develop the time-freezing reformulation for a class of hybrid systems with hysteresis and provide a simple tutorial example. Section IV formalizes the relation between time-freezing PSS and hysteresis systems. Finally, Section V contains a numerical example and Section VI concludes this letter.

Notation: For the *physical* time derivative of a function $x(t)$ we use $\dot{x}(t) := \frac{dx}{dt}(t)$ and for the *numerical* time derivative of $y(\tau)$ we use $y'(\tau) := \frac{dy}{d\tau}(\tau)$. The matrix $I_n \in \mathbb{R}^{n \times n}$ is the identity matrix, and $\mathbf{0}_{m,n} \in \mathbb{R}^{m \times n}$ is the zero matrix. The concatenation of two column vectors $a \in \mathbb{R}^m$, $b \in \mathbb{R}^n$ is

denoted by $(a, b) := [a^\top, b^\top]^\top$. The concatenation of several column vectors is defined in an analogous way. The closure of a set C is denoted by \bar{C} , its boundary as ∂C and $\text{conv}(C)$ is its convex hull.

II. BASIC DEFINITIONS: HYBRID SYSTEMS WITH HYSTERESIS AND FILIPPOV SYSTEMS

In this section we provide some of the basic definitions and notation for PSS and hybrid systems.

A. PSS and Filippov Systems

We regard PSS of the following form

$$\dot{x} = f_i(x), \text{ if } x \in R_i \subset \mathbb{R}^{n_x}, \quad i \in \mathcal{I} := \{1, \dots, m\}, \quad (1)$$

with regions $R_i \subset \mathbb{R}^{n_x}$ and associated dynamics $f_i(\cdot)$, which are smooth functions on an open neighborhood of \bar{R}_i and m is a positive integer. Note that in general the right hand side (r.h.s.) of (1) is discontinuous in x . We assume that R_i are disjoint, nonempty, connected and open sets. They have piecewise-smooth boundaries ∂R_i . Moreover, we assume that $\bigcup_{i \in \mathcal{I}} R_i = \mathbb{R}^{n_x}$ and that $\mathbb{R}^{n_x} \setminus \bigcup_{i \in \mathcal{I}} R_i$ is a set of measure zero. Note that the dynamics are not defined on ∂R_i and to have a meaningful solution concept for the PSS (1) we regard the Filippov convexification of it [16]. The ODE (1) with a discontinuous r.h.s. is replaced by a Differential Inclusion (DI) whose r.h.s. is a convex and bounded set. Due to the assumed structure of the sets R_i , if \dot{x} exists, functions $\theta_i(\cdot)$ which serve as convex multipliers can be introduced and the Filippov DI for (1) reads as [1]

$$\begin{aligned} \dot{x} \in F_F(x) = \left\{ \sum_{i \in \mathcal{I}} f_i(x) \theta_i \mid \sum_{i \in \mathcal{I}} \theta_i = 1, \theta_i \geq 0, \right. \\ \left. 0 = \theta_i \text{ if } x \notin \bar{R}_i, \forall i \in \mathcal{I} \right\}. \end{aligned} \quad (2)$$

Note that in the interior of the regions R_i the *Filippov set* $F_F(x)$ is equal to $\{f_i(x)\}$ and on the boundary between regions we have a convex combination of the neighboring vector fields. The evolution of $x(\cdot)$ on region boundaries ∂R_i are called *sliding modes*. The sliding mode dynamics in Filippov's setting are implicitly defined by Differential Algebraic Equations (DAE) [16].

B. Hybrid Systems With Hysteresis

We consider dynamic systems represented with the finite automaton in Fig. 1(a):

$$\dot{x} = f(x, w) = (1 - w)f_A(x) + wf_B(x), \quad (3)$$

where the $(w, \psi(x))$ characteristic is illustrated in Fig. 1(b).

For a uniformly continuous function $x(t)$ on $t \in [0, T]$ and a smooth $\psi(\cdot)$, there can be only finitely many oscillations between 0 and 1. Consequently, the function $w(t)$ is piecewise constant and has only finitely many jumps between 0 and 1 [2].

The system in (3) has two modes of operation denoted by A and B. In order to be able to simulate (3) for $t \in [0, T]$ with a given $x(0) = x_0$ we must know $w(0)$ as well. This property is typical for systems with hysteresis. Furthermore, $w(\cdot)$ jumps between 0 and 1, hence we can describe it by an ODE with

the state vector $z := (x, w) \in \mathbb{R}^{n_x+1}$ which is associated with a state jump law.

$$\dot{z} = (f(x, w), 0), \quad (4)$$

accompanied by a state-jump law for $w(\cdot)$ at time-point t_s which covers two scenarios:

- 1) if $w(t_s^-) = 0$ and $\psi(x(t_s^-)) = 1$, then $x(t_s^+) = x(t_s^-)$ and $w(t_s^+) = 1$,
- 2) if $w(t_s^-) = 1$ and $\psi(x(t_s^-)) = 0$, then $x(t_s^+) = x(t_s^-)$ and $w(t_s^+) = 0$.

Clearly, due to the state jump law the ODE (4) is not simply a PSS as (1). Throughout this letter we assume, given $x(0)$ and $w(0)$ that there exists a solution to the Initial Value Problem (IVP) associated with (4). A way to define a meaningful notion of solution for hybrid system as (4) is given in e.g., [3, Sec. 5.4] and sufficient conditions for well-posedness are provided [3, Th. 5.4].

III. THE TIME-FREEZING REFORMULATION FOR HYBRID SYSTEMS WITH HYSTERESIS

This section introduces the time-freezing reformulation for the system (4). We define step-by-step the corresponding regions R_i of the time-freezing PSS and give constructive ways to find vector fields associated to them. The section finishes with a tutorial example.

A. The Time-Freezing System

The main idea is to transform the state $w(t)$ which is a piecewise constant function of time into a continuous differential state on a different time domain. We call this new time domain the *numerical time* and denote it by τ . Instead of t as in (1), τ will now be the time of the time-freezing PSS. Moreover, we introduce a clock state $t(\tau)$ in the time-freezing PSS which we call *physical time*. It grows whenever the systems evolves according to $f_A(x)$ or $f_B(x)$, i.e., $\frac{dt}{d\tau}(\tau) = 1$. Otherwise the physical time is frozen, i.e., $\frac{dt}{d\tau}(\tau) = 0$. In other words, the time is frozen whenever $w \notin \{0, 1\}$. Consequently, the $w(\cdot)$ takes only discrete values in physical time, i.e., when $t(\tau)$ is evolving.

The time-freezing PSS has the following state vector $y := (x, w, t) \in \mathbb{R}^{n_y}$, $n_y = n_x + 2$. In the sequel, we define its regions $R_i \subset \mathbb{R}^{n_y}$ and the associated vector fields $f_i(y)$. Some key observations can be made from Fig. 1(b). First, everything except the solid curve is prohibited for the system (4) in the (ψ, w) -plane. We use this prohibited part of the state space to define auxiliary dynamics. Second, the evolution happens in a lower-dimensional subspace since $\dot{w} = 0$. This corresponds in Filippov's setting to sliding modes. Hence, we define the regions such that the evolution of the initial system (4) corresponds to sliding modes of the time-freezing PSS, i.e., it happens on region boundaries ∂R_i .

A suitable partition of the (ψ, w) -plane can be achieved with *Voronoi regions*. The regions are defined as $R_i = \{z \mid \|z - z_i\|^2 < \|z - z_j\|^2, j = 1, \dots, 4, j \neq i\}$, $z = (\psi(x), w)$ with the points: $z_1 = (\frac{1}{4}, -\frac{1}{4})$, $z_2 = (\frac{1}{4}, \frac{1}{4})$, $z_3 = (\frac{3}{4}, \frac{3}{4})$ and $z_4 = (\frac{3}{4}, \frac{5}{4})$. An illustration of the regions is given in Fig. 2, where the black solid lines denote the region boundaries. This choice

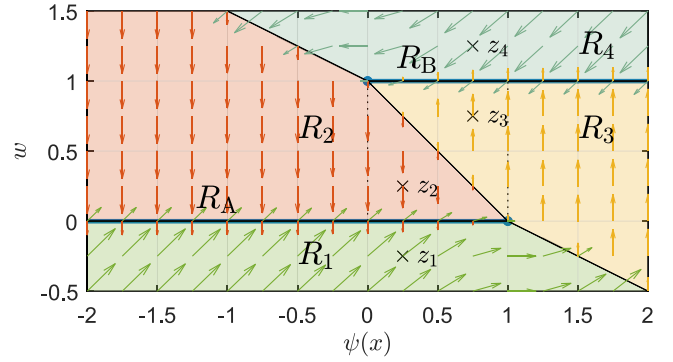


Fig. 2. Illustration of the partitioning of the state space in $(\psi(x), w)$ -plane for the time-freezing PSS via Voronoi regions with the corresponding auxiliary and DAE-forming dynamic's vector fields. The Voronoi points $z_i, i = 1, \dots, 4$, are marked by the crosses.

of z_i defines regions such that their boundaries correspond to the feasible set of the original system (4). Moreover, the space is split by the diagonal line between R_2 and R_3 such that we can define different auxiliary dynamics for the state jumps in both directions. One can make other choices for the points z_i with the exact same properties. The proposed choice partitions the space symmetrically, cf. Fig. 2. The figure illustrates also the vector fields in the regions R_i whose meaning is detailed below. It is important to note that the original system can only evolve at region boundaries $R_A := \{y \in \mathbb{R}^{n_y} \mid w = 0, \psi(x) \leq 1\} = \partial R_1 \cap \partial R_2$ and $R_B := \{y \in \mathbb{R}^{n_y} \mid w = 1, \psi(x) \geq 0\} = \partial R_3 \cap \partial R_4$.

We exploit the interior of the regions $R_i, i = 1, \dots, 4$ to define the needed auxiliary ODE. In what follows, in the regions R_2 and R_3 we define auxiliary dynamic systems whose trajectory endpoints satisfy the state jump law of (4). In the regions R_1 and R_4 we will define so-called DAE-forming ODE [13], which make sure that we obtain appropriate sliding modes on R_A and R_B , which are described by index 2 DAE [16] and which match the dynamics of the original system. The next definition formalizes the desired properties of an auxiliary ODE.

Definition 1 (Auxiliary ODE): The auxiliary ODE in regions R_2 and R_3 are denoted by $y' = f_{\text{aux},A}(y)$ and $y' = f_{\text{aux},B}(y)$, respectively. For every initial value $y(\tau_s) = y_s$ such that $(w(\tau_s), \psi(x(\tau_s))) = (1, 0)$, for $y_s \in R_B$, (and $(w(\tau_s), \psi(x(\tau_s))) = (0, 1)$ for $y_s \in R_A$, respectively) and for every well-defined and finite time interval $\overline{\mathcal{T}}_{\text{jump}} := (\tau_s, \tau_r)$ with the length $\tau_{\text{jump}} := \tau_r - \tau_s$, the auxiliary ODE satisfy the following properties: (i) $w(\tau) \in (0, 1), \forall \tau \in \overline{\mathcal{T}}_{\text{jump}}$, (ii) $x(\tau_s) = x(\tau_r)$, and (iii) $w(\tau_r) = 0$ (or $w(\tau_r) = 1$).

In other words, we define an ODE whose trajectory endpoints on $\overline{\mathcal{T}}_{\text{jump}}$ satisfy the state jump law associated with Eq. (4), cf. Fig. 2. The next proposition provides a constructive way to find an ODE with the above described properties.

Proposition 1 (Auxiliary ODE): Given an initial value $y(\tau_s) = y_s$ such that $w(\tau_s) = 1$ and $\psi(x(\tau_s)) = 0$, the ODE given by

$$y'(\tau) = f_{\text{aux},A}(y) := (\mathbf{0}_{n_x,1}, -\gamma(\psi(x) - 1), 0), \quad (5)$$

where $\gamma: \mathbb{R} \rightarrow \mathbb{R}$ and $\gamma(x) = \frac{ax^2}{1+x^2}$ with $a > 0$, is an auxiliary ODE defined in R_2 . Similarly, for $y(\tau_s) = y_s$ with $w(\tau_s) = 0$

and $\psi(x(\tau_s)) = 1$, the ODE

$$y'(\tau) = f_{\text{aux},B}(y) := (\mathbf{0}_{n_x,1}, \gamma(\psi(x)), 0). \quad (6)$$

is an auxiliary ODE in R_3 . In both cases $\tau_{\text{jump}} = \frac{1}{\gamma(-1)}$. *Proof:* We prove the assertion for (5), since the second part follows similar lines. Since $x'(\tau) = \mathbf{0}_{n_x,1}$ and $t'(\tau) = 0$ these two variables do not change their value, thus $\psi(x(\tau)) = \psi(x(\tau_s)) = 0$ and $t(\tau) = t(\tau_s)$ for $\tau \geq \tau_s$. Hence, we have $w'(\tau) = -\gamma(-1) < 0$. By explicitly solving the ODE we obtain $w(\tau_r) = 0$ for $\tau_r = \tau_s + \tau_{\text{jump}}$, where $\tau_{\text{jump}} = \frac{1}{\gamma(-1)}$. All conditions of Definition 1 are satisfied thus the proof is complete. ■

We briefly discuss some of the proprieties of such an auxiliary ODE, since there are several ways to construct similar ODE. Loosely speaking, in Fig. 2 in R_2 the vector field should point in the negative w -direction and in R_3 in the positive w -direction, and be zero in all other directions. Note that for $\psi(x) \in (0, 1)$ the vector fields of the auxiliary ODE in both cases point away from the manifold defined $\mathcal{M} = \{y \in \mathbb{R}^{n_y} \mid w + \psi(x) - 1 = 0\}$. In such scenarios, there is usually locally no unique solution to the associated Filippov DI, as the trajectory can leave \mathcal{M} at any point in time [16]. However, the system should never be initialized in this region, since this state is infeasible for the original system. We show later that it can never reach this undesired state if initialized appropriately. Furthermore, the auxiliary ODE from Proposition 1 have by construction the favorable property that they do not point away in both directions from \mathcal{M} at the junction points $(0, 1)$ and $(1, 0)$. This is why the function $\gamma(\cdot)$ was introduced in the auxiliary ODE. Another favorable property is, if the system is initialized with the wrong value for $w(\cdot)$ for $\psi(x) \notin (0, 1)$ the auxiliary ODE will automatically reinitialize $w(\cdot)$ while the physical time is frozen, cf. Fig. 2.

We still need to define DAE-forming vector fields for the regions R_1 and R_4 . These vector fields should be such that, together with the auxiliary dynamics in their respective regions, they result in sliding modes on R_A and R_B which match the dynamics of the initial system (4).

In a general PSS the vector fields are not defined on the region boundaries, thus we use Filippov's convexification [16] as defined in Eq. (2), and denote the Filippov set associated to the time-freezing PSS by $F_{\text{TF}}(\cdot)$. The next proposition gives a constructive way to find the desired vector fields.

Proposition 2 (DAE-forming ODE): Suppose the regions R_2 and R_3 are equipped with the vector fields $f_{\text{aux},A}(\cdot)$ and $f_{\text{aux},B}(\cdot)$ from Proposition 1, respectively. Let the region R_1 be equipped with the ODE

$$y' = f_{\text{DF},A}(y) := 2(f_A(x), 0, 1) - f_{\text{aux},A}(y), \quad (7)$$

then for $y \in R_A$ it holds that $(f_A(x), 0, 1) \in F_{\text{TF}}(y) = \overline{\text{conv}}\{f_{\text{aux},A}(y), f_{\text{DF},A}(y)\}$. Similarly, let the region R_4 be equipped with the following ODE

$$y' = f_{\text{DF},B}(y) := 2(f_B(x), 0, 1) - f_{\text{aux},B}(y), \quad (8)$$

then for $y \in R_B$ it holds that $(f_B(x), 0, 1) \in F_{\text{TF}}(y) = \overline{\text{conv}}\{f_{\text{aux},B}(y), f_{\text{DF},B}(y)\}$.

Proof: We prove the assertion for Eq. (7) and the second part follows similar lines. Note that for $y \in R_A =$

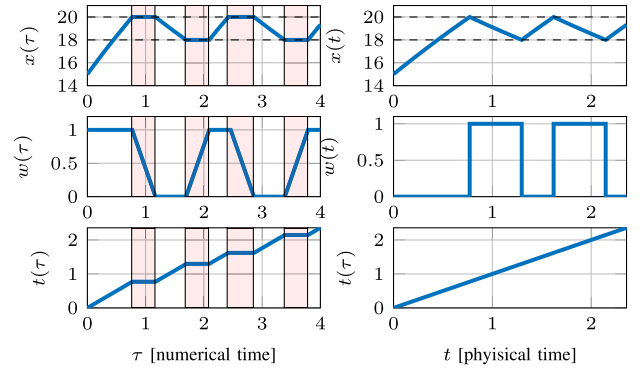


Fig. 3. Trajectories of the time-freezing PSS for a thermostat example in numerical time τ (left plot) and physical time t (right plot).

$\{y \mid c(y) := w = 0, \psi(x) < 1\}$ we have that $\nabla c(y)^\top f_{\text{aux},A}(y) < 0$ and $\nabla c(y)^\top f_{\text{DF},A}(y) > 0$. Hence, we have a sliding mode on $w = 0$ with $\frac{dw}{d\tau} = 0$ [16]. From (2) we have that $F_{\text{TF}}(y) = \{\theta_1(2f_A(x), 0, 1) - f_{\text{aux},A}(y) + \theta_2 f_{\text{aux},A}(y) \mid \theta_1 + \theta_2 = 1, \theta_1, \theta_2 \geq 0\}$. From this relation and $w' = 0$ we obtain that $\theta_1 - \theta_2 = 0$. Thus we can solve for θ_1 and θ_2 , i.e., $\theta_1 = \theta_2 = \frac{1}{2}$, which yields $(f_A(x), 0, 1) \in F_{\text{TF}}(y)$. This completes the proof. ■

Note that by construction the two sliding modes on R_A and R_B agree with the r.h.s. of Eq. (4) augmented by the dynamics of the clock state. Now we have defined vector fields in all regions of the time-freezing PSS which corresponds to the original system (4). Another favorable property of the chosen auxiliary and DAE forming ODE is: since $w'(\tau)$ is bounded by $a > 0$ it cannot make the sliding mode DAE arbitrarily stiff, especially if constraint drift happens.

B. A Tutorial Example

To illustrate the theoretical development we construct a time-freezing PSS for a thermostat system with hysteresis. The source code of the example is available in the repository of NOSNOC [15]. The system has a single state $x(\cdot)$ which models the temperature of a room which should stay inside the interval $x \in [18, 20]$. As soon as the temperature drops below $x = 18$ the heater is switched on and when the temperature grows above $x = 20$ it is switched off. The two modes of operation are given by $\dot{x} = f_A(x) = -0.2x + 5$ when the heater is on and $\dot{x} = f_B(x) = -0.2x$ when the heater is off. One can see that for $\psi(x) = 0.5(x - 18)$ we have a hybrid system that matches the finite automaton in Fig. 1(a). For a time-freezing PSS we define the regions R_i via the Voronoi points as in the last section. The auxiliary ODE's r.h.s. according to Proposition 1 read as $f_{\text{aux},A}(y) = (0, -\gamma(0.5(x - 18) - 1), 0)$ and $f_{\text{aux},B}(y) = (0, \gamma(0.5(x - 18)), 0)$ with $a = 1$. Similarly, the DAE-forming ODE r.h.s. according to Proposition 2 read as $f_{\text{DF},A}(y) = (-0.4x + 10, \gamma(0.5(x - 18) - 1), 2)$ and $f_{\text{aux},B}(y) = (-0.4x, -\gamma(0.5(x - 18)), 2)$.

We simulate now the time-freezing PSS with a FESD Radau-IIA integrator of order 3 [1] with $x(0) = 15$ and $w(0) = 0$. The left plot in Fig. 3 illustrates the evolution of the time-freezing PSS in numerical time. The red shaded

areas indicate the phases when the auxiliary ODE is active with $w \notin \{0, 1\}$ while the time is frozen, cf. bottom left plot. In the middle left plot we can see that $w(\tau)$ is now a continuous function in numerical time. The right plot in Fig. 3 shows the differential state in physical time $t(\tau)$. Clearly, in the middle right plot $w(t(\tau))$ is now a discontinuous function, hence the state jumps are successfully recovered in physical time.

IV. SOLUTION EQUIVALENCE

From the developments in the last section, the solution equivalence is nearly apparent. We formalize it in the next theorem.

Theorem 1: Regard the IVP corresponding to: (i) the Filippov DI of the time-freezing PSS equipped with the vector fields from Proposition 1 and 2 with a initial value $y(0) = (z_0, 0)$ with $z_0 = (x_0, w_0)$ and $w_0 \in \{0, 1\}$, on a time interval $[0, \tau_f]$, (ii) the ODE with state jumps from Eq. (4) with $z(0) = z_0$ on a time interval $[0, t_f] = [0, t(\tau_f)]$. Suppose solutions exist to both IVP.

Then the solutions of the two IVPs $z(t; z_0)$ and $y(\tau; y_0)$ fulfill at any $\frac{dt}{d\tau} = t'(\tau) \neq 0$:

$$z(t(\tau); z_0) = My(t(\tau); y_0), \text{ with } M = [I_{n_x+1} \quad \mathbf{0}_{n_x} + 1, 1]. \quad (9)$$

Proof: Denote the solution of IVP (i) by $y_1(\tau; y_0)$ for $\tau \in (0, \hat{\tau})$ and for (ii) and $t(\tau) \in (0, t(\hat{\tau}))$ by $z_1(t(\tau); z_0)$. For a given $w(0) = 0$ (or 1) we have from Proposition 2 that $y' = (f_A(x), 0, 1)$ (or $y' = (f_B(x), 0, 1)$). Note that if there is no $\tau_s \in (0, \hat{\tau})$ for the IVP (i) such that an auxiliary ODE becomes active, then $t(\tau) = \int_0^\tau d\tau_1 = \tau$. Since $(f_A(x), 0) = M(f_A(x), 0, 1)$, $(f_B(x), 0) = M(f_B(x), 0, 1)$ and $z_0 = My_0$ by setting $\hat{\tau} = \tau_f$, it follows that (9) holds.

Suppose now that we have a $\tau_s \in (0, \tau_f)$ such that for $w(\tau_s) = 1$ the auxiliary ODE $y' = f_{\text{aux},A}(y)$ becomes active (or similarly for $w(\tau_s) = 0$, $y' = f_{\text{aux},B}(y)$ becomes active). From the first part of the proof we have that (9) holds for $\tau \in (0, \tau_s)$ and hence for all $t(\tau) \in (0, t_s^-)$, where $t_s^- = t(\tau_s)$. From Proposition 1 we have that the solution satisfies $x(\tau_s) = x(\tau_f)$ and $w(\tau_f) = 0$ (or $w(\tau_f) = 1$) with $t'(\tau) = 0$ for $\tau \in [\tau_s, \tau_f]$. Hence, we have also $t(\tau_f) = t_s^+ = t(\tau_s)$. Denote by $y_s = (x(\tau_f), w(\tau_f), t(\tau_f))$. Using this we have $y_1(\tau - \tau_f, y_s) = y(\tau, y_0)$ for $\tau \in (\tau_f, \tilde{\tau})$ and denoting $z_s = My_s$ we see that $z_1(t(\tau) - t_s; z_s) = x(t(\tau), z_0)$ for $t(\tau) \in (t_s^+, \tilde{\tau})$. Assume that a single activation of an auxiliary ODE takes place and set $\tilde{\tau} = \tau_f$. Since the intervals (t_s, t_f) and (τ_f, τ_f) have the same length and $z_s = My_s$ from the definitions of the corresponding IVP, we conclude that relation (9) holds. If the auxiliary ODE becomes active multiple times we simply apply the same argument on the corresponding sub-intervals. This completes the proof. ■

The last theorem opens the door to study the regarded hybrid system with hysteresis as a Filippov system and to apply their rich theory, e.g., solution existence results [16]. From the practical side, we can use numerical methods for Filippov systems which allows us to avoid the use of integer variables.

V. NUMERICAL EXAMPLE: TIME OPTIMAL PROBLEM OF A CAR WITH TURBO CHARGER

In this section we apply the theoretical developments in a numerical example of a time optimal control problem of a car with turbo from [9]. We consider a double-integrator car model equipped with a turbo accelerator which follows a hysteresis characteristic as in Fig. 1(b). This makes the seemingly simple model severely nonlinear and nonsmooth.

The car is described by its position $q(t)$, velocity $v(t)$ and turbo charger state $w(t) \in \{0, 1\}$. The control variable is the car acceleration $u(t)$. The turbo accelerator is activated when the velocity exceeds $v \geq 15$ and is deactivated when it falls below $v \leq 10$. When it is on, it makes the nominal acceleration $u(t)$ three times greater. One can see that $\psi(x) = \frac{v-10}{5}$. In summary, the state vector reads as $z = (q, v, w) \in \mathbb{R}^3$ with two modes of operation described by $f_A(z) = (v, u, 0)$ and $f_B(z) = (v, 3u, 0)$. The acceleration is bounded by $|u| \leq \bar{u}$, $\bar{u} = 5$ and the velocity by $|v| \leq \bar{v}$, $\bar{v} = 25$.

In the OCP we consider the time-freezing PSS associated to the car model on a numerical time interval $\tau \in [0, \tau_f]$. The car should reach the goal $q(t(\tau_f)) = q_f = 150$ with $v(t(\tau_f)) = v_f = 0$, whereby $z(0) = z_0 = \mathbf{0}_{3,1}$. The auxiliary and DAE-forming dynamics are chosen according to Propositions 1 (with $a = 1$) and 2, respectively. The OCP reads as:

$$\min_{y(\cdot), u(\cdot), s(\cdot)} t(\tau_f) \quad (10a)$$

$$\text{s.t. } y(0) = (z_0, 0), \quad (10b)$$

$$y'(\tau) \in s(\tau)F_{\text{TF}}(y(\tau), u(\tau)), \quad \tau \in [0, \tau_f], \quad (10c)$$

$$-\bar{u} \leq u(\tau) \leq \bar{u}, \quad \tau \in [0, \tau_f], \quad (10d)$$

$$\bar{s}^{-1} \leq s(\tau) \leq \bar{s}, \quad \tau \in [0, \tau_f], \quad (10e)$$

$$-\bar{v} \leq v(\tau) \leq \bar{v}, \quad \tau \in [0, \tau_f], \quad (10f)$$

$$(q(\tau_f), v(\tau_f)) = (q_f, v_f). \quad (10g)$$

The objective consist of minimizing the final physical time. Since a time optimal control problem is considered, we introduce the scalar *speed-of-time* control variable $s(\cdot)$ which introduces a time-transformation and enables to have a variable terminal physical time $T_f = t(\tau_f)$. It is bounded by (10e) with $\bar{s} = 10$. NOSNOC ensures equidistant control grids in numerical and physical time τ .

The OCP is discretized with a FESD Radau-IIA scheme of order 3 with $N = 10$ control intervals and $N_{fe} = 3$ additional integration steps on every control interval, with $\tau_f = 5$. The controls are taken to be piecewise constant over the control intervals. The OCP discretization and MPCC homotopy is carried out via the open source tool NOSNOC, which has IPOPT [18] and CasADi [19] as a back-end.

Additionally, we compare our approach to the mixed integer formulation of [9]. We take the same control and state discretization as in NOSNOC which results 56 binary variables. Switches in the integer formulation are allowed only at the control interval boundaries, as a switch detection formulation requires significantly more integer variables and introduces more nonlinearity.

The problem is solved with the dedicated mixed integer nonlinear programming (MINLP) solver Bonmin [20]. Note that the only nonlinearity in the MINLP is due to the

TABLE I
COMPARISON OF NOSNOC TO MIXED INTEGER FORMULATIONS

| Solver | T_f | CPU Time [s] | $E(T_f)$ |
|-----------------------|--------------|--------------|-----------------|
| NOSNOC | 10.26 | 8.87 | 9.49e-02 |
| Gurobi with bisection | 11.21 | 5.31 | 7.88e+01 |
| Bonmin | 11.28 | 1481.58 | 7.88e+01 |

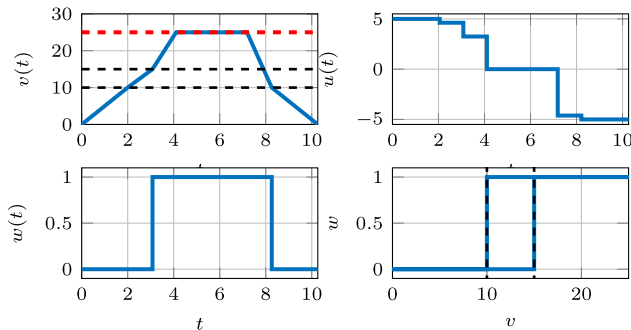


Fig. 4. Solutions of the OCP (10) in physical time. The top left and right plots show the velocity $v(t)$ and optimal controls $u(t)$, respectively. The bottom left and right plots show the hysteresis state $w(t)$ and the solution trajectory in the (v, w) -plane, respectively.

time-transformation for the optimal time T_f . Therefore, in a second experiment we fix T_f and solve the resulting MILP with the commercial solver Gurobi. We make a bisection-type search in T_f . The MILP with the smallest T_f that is still feasible, delivers the optimal time T_f . In this experiment 22 MILP were solved for an accuracy of 10^{-6} . To determine the solution quality, we additionally perform a high accuracy solution with the computed optimal controls and obtain $x_{\text{sim}}(t)$. We compare the terminal constraint satisfactions: $E(T_f) = \|x_{\text{sim}}(T_f) - (q_f, v_f)\|_2$. The source code for the simulation and the two MIOP approaches are provided in the NOSNOC repository [15].

The results are summarized in Table I. All three approaches provide a similar objective value. Gurobi is the fastest solver, NOSNOC is only slightly slower and Bonmin is significantly slower. The smallest terminal error is achieved via NOSNOC. This is due to the underlying FESD discretization, cf. [1]. Gurobi and Bonmin have the same discretization without switch detection and result in the same terminal error. On the other hand, Gurobi provides the most robust approach, as NOSNOC (i.e., IPOPT as underlying NLP solver) fails to converge in some variations of the discretization.

The results computed by NOSNOC is depicted in Fig. 4. One can see an intuitive behavior as the car uses the turbo accelerator as much as possible to reach the goal time optimally, with $T_f = 10.26$.

VI. CONCLUSION

In this letter we introduced a novel time-freezing reformulation for a class of hybrid systems with hysteresis. It transforms the systems with state jumps into PSS for which we leverage the recently developed FESD method which enables high accuracy optimal control by solving only smooth NLP. Thus, we can avoid use of computationally expensive mixed integer strategies in numerical optimal control and obtain quickly good and accurate nonsmooth solutions. In the theoretical part,

constructive ways to find auxiliary and DAE-forming ODE are provided and solution equivalence is proven. In future time-freezing for other types of finite automaton and hysteresis systems, as, e.g., described in the introduction should be investigated as well.

ACKNOWLEDGMENT

The authors thank Costas Pantelides from the Imperial College London and PSE Enterprise for inspiring discussions and providing the example in Section V.

REFERENCES

- [1] A. Nurkanović, M. Sperl, S. Albrecht, and M. Diehl, "Finite elements with switch detection for direct optimal control of nonsmooth systems," submitted for publication.
- [2] A. Visintin, *Differential Models of Hysteresis*, vol. 111. New York, NY, USA: Springer, 2013.
- [3] J. Lunze and F. Lamnabhi-Lagarrigue, *Handbook of Hybrid Systems Control: Theory, Tools, Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2009.
- [4] B. Brogliato and A. Tanwani, "Dynamical systems coupled with monotone set-valued operators: Formalisms, applications, well-posedness, and stability," *SIAM Rev.*, vol. 62, no. 1, pp. 3–129, 2020.
- [5] L. Vandenberghe, B. L. De Moor, and J. Vandewalle, "The generalized linear complementarity problem applied to the complete analysis of resistive piecewise-linear circuits," *IEEE Trans. Circuits Syst.*, vol. 36, no. 11, pp. 1382–1391, Nov. 1989.
- [6] F. Ceragioli, C. De Persis, and P. Frasca, "Discontinuities and hysteresis in quantized average consensus," *Automatica*, vol. 47, no. 9, pp. 1916–1928, 2011.
- [7] F. Bagagiolo and R. Mazzino, "Hybrid thermostatic approximations of junctions for some optimal control problems on networks," *SIAM J. Control Optim.*, vol. 57, no. 4, pp. 2415–2442, 2019.
- [8] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, 1999.
- [9] M. Avraam, "Modelling and optimisation of hybrid dynamic processes," Ph.D. dissertation, Dept. Electronic, Imperial Coll. London, London, U.K., 2000.
- [10] A. Nurkanović, S. Albrecht, and M. Diehl, "Limits of MPCC formulations in direct optimal control with nonsmooth differential equations," in *Proc. Eur. Control Conf. (ECC)*, 2020, pp. 2015–2020.
- [11] D. E. Stewart and M. Anitescu, "Optimal control of systems with discontinuous differential equations," *Numerische Mathematik*, vol. 114, no. 4, pp. 653–695, 2010.
- [12] A. Nurkanović, T. Sartor, S. Albrecht, and M. Diehl, "A time-freezing approach for numerical optimal control of nonsmooth differential equations with state jumps," *IEEE Control Syst. Lett.*, vol. 5, pp. 439–444, 2021.
- [13] A. Nurkanović, S. Albrecht, B. Brogliato, and M. Diehl, "The time-freezing reformulation for numerical optimal control of complementarity lagrangian systems with state jumps," 2021, *arXiv:2111.06759*.
- [14] A. Nurkanović and M. Diehl, "NOSNOC: A software package for numerical optimal control of nonsmooth systems," *IEEE Control Syst. Lett.*, early access, Jun. 2022. [Online]. Available: <https://doi.org/10.1109/LCSYS.2022.3181800>
- [15] "NOSNOC." 2022. [Online]. Available: <https://github.com/nurkanovic/nosnoc>
- [16] A. F. Filippov, *Differential Equations with Discontinuous Right-Hand Side*. Dordrecht, The Netherlands: Kluwer, 1988.
- [17] M. Anitescu, P. Tseng, and S. J. Wright, "Elastic-mode algorithms for mathematical programs with equilibrium constraints: Global convergence and stationarity properties," *Math. Program.*, vol. 110, no. 2, pp. 337–371, 2007.
- [18] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, pp. 25–57, 2006.
- [19] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi: A software framework for nonlinear optimization and optimal control," *Math. Program. Comput.*, vol. 11, no. 1, pp. 1–36, 2019.
- [20] P. Bonami *et al.*, "An algorithmic framework for convex mixed integer nonlinear programs," *Discrete Optim.*, vol. 5, no. 2, pp. 186–204, 2008.