2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Continuous Optimization for Control of Finite-State Machines with Cascaded Hysteresis via Time-Freezing

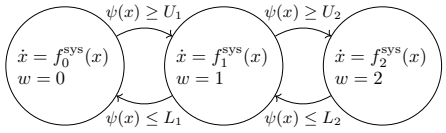
Wim Van Roy[1], Armin Nurkanović[2], Ramin Abbasi-Esfeden[1], Jonathan Frey[2], Anton Pozharskiy[2], Jan Swevers[1,3] and Moritz Diehl[2,4]

***Abstract*— Control problems with Finite-State Machines (FSM) are often solved using integer variables, leading to a mixed-integer optimal control problem (MIOCP). This paper proposes an alternative method to describe a subclass of FSMs using complementarity constraints and time-freezing. The FSM from this subclass is built up by a sequence of states where a transition between the states is triggered by a single switching function. This can be looked at as a cascade of hysteresis loops where a memory effect is used to maintain the active state of the state machine. Based on the reformulation for hybrid systems with a hysteresis loop [13], a method is developed to reformulate this subclass in a similar fashion. The approach transforms the original problem into a Piecewise Smooth System (PSS), which can be discretized using the recently developed Finite Elements with Switch Detection [15], allowing for high-accuracy solutions. The reformulation is compared to a mixed-integer formulation from the literature on a time-optimal control problem. This work is a first step towards the general reformulation of FSMs into nonsmooth systems without integer states.**

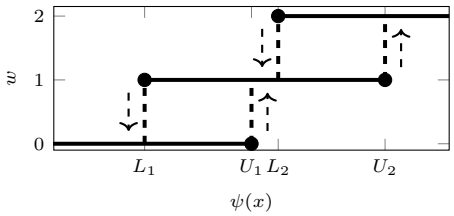
I. Introduction

A Finite-State Machine (FSM) or a hybrid automaton can be found in many control applications such as traffic control, heating systems [12], [19], and other energy management systems [8]. Additionally, FSMs are often used for formulating piecewise smooth systems (PSS) with smooth continuous motions that are interrupted by discrete events [17]. For these FSMs, there exist several specification formalisms whose semantics are clearly described and can serve as operational models for these hybrid systems, such as hybrid automata [3] and hybrid problems [16]. These formalisms provide an easy-to-understand and clearly defined modelling framework.

This paper focuses on a special form of FSM with cascaded hystersis cycles. The general FSM in this framework has n_s modes. The FSM and the cascaded hysteresis for such an automation with $n_s = 3$, is given in Figure 1. The active mode is modelled using an auxiliary variable $w(t) \in \{0, 1, ..., n_s - 1\}$ going from 0 to $n_s - 1$. If the system is in operating mode i, the ODE is given by $\dot{x} = f_i^{\text{sys}}(x)$. When the system reaches a upper limit given by $\psi(x) \geq U_{i+1}$, it switchs to a higher mode $i+1$, if it exist. In the same fashion, the system also switches to a lower mode $i-1$ when the lower limit $\psi(x) \leq L_{i-1}$. For these systems, the bounds are ordered so that $U_i < U_{i+1}$, $L_i < L_{i+1} \forall i \in \{0, 1, ..., n_s - 2\}$ and $L_i < U_i, \forall i \in \{0, 1, ..., n_s - 1\}$.



(a) FSM of a system with three states



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

Fig. 1. Example FSM.

In the context of optimal control problems (OCPs), this kind of system leads to a mixed-integer optimization problem. In the case of linear systems in discrete time, these problems can be solved efficiently using mixed-integer linear programming or mixed-integer quadratic programming solvers such as Gurobi [10], or CPLEX [11]. However, when nonlinearities are present, e.g. for time-optimal control, this OCP becomes a Mixed-Integer Nonlinear Program (MINLP), which can be arbitrarily difficult to solve in practice due to the combination of integer variables, nonlinearities and nonconvexities.

This paper proposes a method that converts a FSM into a PSS. We discretize the OCP subject to this PSS with the Finite Elements with Switch Detection (FESD) method [15] to obtain high-accuracy solution approximations. The conversion from the FSM to a PSS is based on the hysteresis characteristics that occur when looking at two adjacent modes of the FSM. This hysteresis characteristic can be reformulated into a PSS using *time-freezing* [13]. This approach converts the auxiliary state $w(t)$ into a continuous-time differential state, which exhibits jump discontinuities in physical time. This jump is interpreted as a *state jump law*. Time-freezing introduces *auxiliary dynamics* and a *clock*

This research was supported by the DFG via Research Unit FOR 2401 and project 424107692 and by the EU via ELO-X 953348 and by a Baekeland scholarship (Grant 182076) funded by Atlas Copco Airpower NV, Wilrijk, Belgium and the Institute for the Promotion of Innovation through Science and Technology in Flanders (VLAIO), Belgium.

[1]MECO research team, Dept. ME, KU Leuven, Belgium {wim.vanroy,ramin.abassi-esfeden,jan.swevers}@kuleuven.be

[2]Department of Microsystems Engineering (IMTEK), University of Freiburg, Germany {armin.nurkanovic,jonathan.frey,moritz.diehl}@imtek.uni-freiburg.de, anton.pozharskiy@merkur.uni-freiburg.de

[3]Flanders Make@KU Leuven, Leuven, Belgium

[4]Department of Mathematics, University of Freiburg, Germany

state to handle the state jump. The main idea is to define auxiliary dynamics in the regions of the state space that are otherwise prohibited for the system and freeze the clock state evolution. The auxiliary state variable $w(t)$ is frozen in the regions where the actual dynamics are present, but the clock state does evolve. The solution for the problem can be reconstructed by only looking at the regions where the clock state evolves.

Contribution: We extend the system with hysteresis [13] to a limited class of hybrid systems. The hybrid systems should consist of a finite number of ordered states where there is a single switch function to transition between the states. For these systems, we introduce two novel time-freezing reformulations with one and with two auxiliary variables, respectively. We propose a slight adaptation for the auxiliary Ordinary Differential Equation (ODE) that was suggested by [13] to improve numerical stability. A time-optimal control problem illustrates the approach. The numerical efficiency as well as the accuracy are compared with state-of-the-art mixed-integer solution strategies. This work is a first step towards a general method to describe general FSM as a PSS system. It allows studying these problems using the tools developed for Filippov systems [9]. The examples in this paper are provided in the software package `nosnoc_py` [1], [2], [14].

Outline: Section II gives some basic definitions of PSS and time-freezing. Section III develops the time-freezing reformulation for a class of hybrid systems with a sequence of states using two different approaches. Section IV compares the two approaches with approaches from the literature. Section V concludes the paper and highlights the advantages and disadvantages of the proposed approach.

Notation: The *physical* time derivative of a function $x(t)$ is given by $\dot{x}(t) := \frac{\mathrm{d}x}{\mathrm{d}t}(t)$. For the *numerical* time derivative of $y(\tau)$, we use $y'(\tau) := \frac{\mathrm{d}y}{\mathrm{d}\tau}(\tau)$. For a given set C, the closure is denoted by $\overline{C}$, its boundary by ∂C and its convex hull by $\text{conv}(C)$. All vector inequalities are to be understood element-wise. The regions of a PSS are denoted by $R_{i,j}$, and B_i corresponds to the parts of the state space where the FSM dynamics $f_i^{\text{sys}}(x)$ evolve. This corresponds to the sliding modes of the PSS.

II. Filippov Systems and Time-Freezing

This section explains the required background information regarding Filippov systems and time-freezing reformulations for a hybrid system with a hysteresis.

A. PSS and Filippov Systems

A general PSS is given by the set of ODEs

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

with n_f a positive integer, the regions $R_i \subset \mathbb{R}^{n_x}$ and associated dynamics $f_i(\cdot)$. These dynamics are smooth functions on an open neighborhood of $\overline{R}_i$. The regions are disjoint, nonempty and open sets. It is assumed that $\overline{\bigcup_{i\in\mathcal{I}} R_i} = \mathbb{R}^{n_x}$ holds and $\mathbb{R}^{n_x} \setminus \bigcup_{i\in\mathcal{I}} R_i$ is a set of measure zero.

Since the dynamics for this PSS are undefined on ∂R_i and to achieve a meaningful solution, a Filippov Differential Inclusion (DI) [9] treats the discontinuities. The Filipov DI [15] is given by:

$$\dot{x} \in F_{\mathrm{F}}(x) = \Big\{ \sum_{i\in\mathcal{I}} f_i(x)\,\theta_i \mid \sum_{i\in\mathcal{I}} \theta_i = 1,\ \theta_i \geq 0, \ \theta_i = 0 \text{ if } x \notin \overline{R_i},\ \forall i \in \mathcal{I} \Big\}, \quad (2)$$

where $\theta = (\theta_1, ..., \theta_{n_f}) \in \mathbb{R}^{n_f}$ are the Filippov mutltipliers. The r.h.s. of Eq. (2) is a convex and bounded set. The resulting dynamics on ∂R_i are now defined as a convex combination of the dynamics from the neighboring regions and are called *Sliding Modes*. For the interior of a region R_i, the *Filippov set* $F_{\mathrm{F}}(x)$ is equal to $\{f_i(x)\}$. Using Stewart's approach [18], this Filippov system is reformulated into a Dynamic Complementarity System (DCS).

B. Time-Freezing Reformulation for Hybrid System with a single Hysteresis characteristic

This subsection summarizes the reformulation of a hybrid system with a hysteresis into a PSS, as introduced in [13], and gives an improvement on defining the auxiliary dynamics. The main idea for this reformulation is transforming the integer state $w(t)$ into a continuous differential state $w(\tau)$ on a new time domain τ, the *numerical time*. This is the time of the PSS as the original time is retrieved from a clock state $t(\tau)$, the *physical time*. When the system evolves, this clock state is evolving as well, i.e., $\frac{dt}{d\tau}(\tau) = 1$. The (usually) discrete state $w(t)$, evolves continuously in numerical time $w(\tau)$ in parts of the state space via auxiliary dynamics, in this regions of the state space the evolution of the clock state is frozen, i.e. $\frac{dt}{d\tau}(\tau) = 0$. The state vector for the time-freezing PSS is thus $y := (x, w, t) \in \mathbb{R}^{n_y}$ with $n_y = n_x + 2$.

For this application, the Voronoi regions of the PSS are defined in a new space using a projection $\Pi(y) = (\psi(x), w)$. The regions have piecewise smooth boundaries ∂R_i and are given by

$$R_i = \{y \in \mathbb{R}^{n_y} | g_i(y) \leq \min_{j\in\mathcal{I}, j\neq i} g_j(y)\}, \quad (3)$$

with *discriminant function* $g_i(\cdot)$, defined as the distance to a point z_i:

$$g_i(y) = \|\Pi(y) - z_i\|. \quad (4)$$

For the hysteresis, four different regions are defined 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 $z_4 = (\frac{3}{4}, \frac{5}{4})$. The original system only evolves along the boundaries $B_0 := \{y \in \mathbb{R}^{n_y} \mid w = 0,\ \psi(x) \leq 1\} = \partial R_1 \cap \partial R_2$ for $f_0^{\text{sys}}(x)$ and $B_1 := \{y \in \mathbb{R}^{n_y} \mid w = 1,\ \psi(x) \geq 0\} = \partial R_3 \cap \partial R_4$ for $f_1^{\text{sys}}(x)$. These are the sliding modes of the PSS and are defined by the Filippov's convexification [9] using Eq. (2). Figure 2 illustrates the vector fields for this system.

For clarity, the ODEs have the notation $\underline{f}$ for the equations governing the lower part of the hysteresis when $w = 0$ and

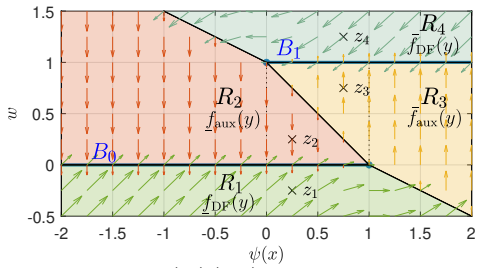


Fig. 2. Partitioning of the $(\psi(x), w)$-plane of a time-freezing PSS obtained by reformulating a hybrid system with a single hysteresis characteristic as proposed in [13]. The Voronoi points $z_1, \dots, z_4$, are marked as crosses.

have the notation $\bar{f}$ for the upper part of the hysteresis when $w = 1$. The regions R_2 and R_3 are equipped with auxiliary dynamics $\underline{f}_{\mathrm{aux}}(y)$ and $\bar{f}_{\mathrm{aux}}(y)$, respectively, transition between the two modes of the system. Their auxiliary ODEs are defined as:

$$\underline{f}_{\mathrm{aux}}(y) \coloneqq (\mathbf{0}_{n_x,1}, -\gamma(\psi(x) - 1), 0), \tag{5}$$
$$\bar{f}_{\mathrm{aux}}(y) \coloneqq (\mathbf{0}_{n_x,1}, \gamma(\psi(x)), 0), \tag{6}$$

where $\gamma : \mathbb{R} \to \mathbb{R}$ and $\gamma(x) = \frac{ax^2}{1+x^2}$ with $a > 0$ as proposed by [13]. This function is zero at x equal to 0 and smoothly transitions to 1 further away from zero.

As the auxiliary dynamics on the boundaries of the sliding mode, i.e. B_0 and B_1, should vanish, the ODEs for R_1 and R_4 are defined as $\underline{f}_{\mathrm{DF}}(y)$ and $\bar{f}_{\mathrm{DF}}(y)$ respectively by

$$\underline{f}_{\mathrm{DF}}(y) \coloneqq 2(f_0^{\mathrm{sys}}(x), 0, 1) - \underline{f}_{\mathrm{aux}}(y), \tag{7}$$
$$\bar{f}_{\mathrm{DF}}(y) \coloneqq 2(f_1^{\mathrm{sys}}(x), 0, 1) - \bar{f}_{\mathrm{aux}}(y). \tag{8}$$

This set of ODEs has some favorable properties. First, the set of vector fields points to the manifold defined by $\mathcal{M} = \{y \in \mathbb{R}^n_y | w + \psi(x) - 1 = 0\}$ apart from the points $(0, 1)$ and $(1, 0)$, where a linear combination of the ODEs can be made to leave the points and transition to another state. Second, wrong initializations for $w(\cdot)$ when $\psi(x) \notin (0, 1)$ will be corrected via to the auxiliary ODE while the physical time is frozen. Third, the sliding mode Differential Algebraic Equation (DAE) is not arbitrarily stiff since $w'(\tau)$ is bounded by $a > 0$. This property is important when constraint drift occurs. This is the drift between the physically invariant motion constraints and the numerical results due to the integration method.

Note that for $y \in B_0 = \{y \mid c(y) \coloneqq w = 0, \psi(x) < 1\}$, the system has $\nabla c(y)^\top \underline{f}_{\mathrm{aux}}(y) < 0$ and $\nabla c(y)^\top \underline{f}_{\mathrm{DF}}(y) > 0$. This results in a sliding mode on $w = 0$ with $\frac{dw}{d\tau} = 0$. From Eq. (2), the system equations result in $F_{\mathrm{F}}(y) = \{\theta_1(2(f_0^{\mathrm{sys}}(x), 0, 1) - \underline{f}_{\mathrm{aux}}(y)) + \theta_2 \underline{f}_{\mathrm{aux}}(y) \mid \theta_1 + \theta_2 = 1, \theta_1, \theta_2 \geq 0\}$. As $w' = 0$, we obtain $\gamma(\psi(x) - 1)\theta_1 - \gamma(\psi(x) - 1)\theta_2 = 0$. Solving these equations normally results in $F_{\mathrm{F}}(y) = (f_0^{\mathrm{sys}}(x), 0, 1)$ and $\theta_1 = \theta_2 = \frac{1}{2}$. This leads to a unique sliding mode. However, when $\gamma(\psi(x) - 1) \to 0$, numerical issues might appear when solving this system. Therefore, we propose to change $\gamma(x) = \frac{ax^2}{1+x^2} + b$ with $0 < b << 1$. This small change improved numerical stability without affecting the theorems from [13].

III. Two Finite-state Machine Reformulations

The previous approach can be extended to the FSM from the introduction that consist of a sequence of n_s states with a single switching function $\psi(x)$. This section introduces two variants to model such an FSM.

A. Variant 1: With one auxiliary variable

If $\psi(x) \geq U_i$, the system transitions from state $i - 1$ to i and when $\psi(x) \leq L_i$, the system transitions from $i + 1$ to i, given the system is in state i. Each transition between two states can be seen as a hysteresis curve with the levels L_i and U_i. If $U_{i-1} < L_i$ for all $i = 2, \dots, n_s$, a minor extension to the previous formulation can be made, where a similar set of ODE equations and Voronoi regions are created by introducing multiple hysteresis characteristics. The points for each hysteresis i can be reconstructed using a linear translation with (L_i, i) and a scaling of $U_i - L_i$. They are thus given as:

$$\begin{aligned} z_{i,1} &= \left(a_i - b_i, i - \frac{5}{4}\right), z_{i,2} = \left(a_i - b_i, i - \frac{3}{4}\right), \\ z_{i,3} &= \left(a_i + b_i, i - \frac{1}{4}\right), z_{i,4} = \left(a_i + b_i, i + \frac{1}{4}\right), \end{aligned} \tag{9}$$

with $a_i = \frac{U_i - L_i}{2} + L_i$ and $b_i = \frac{1}{4(U_i - L_i)}$. The regions $R_{i,1}$, $R_{i,2}$, $R_{i,3}$ and $R_{i,4}$, created using Eq. 3 and Eq. 4, correspond to the ODEs $\underline{f}_{\mathrm{DF},i}(y)$, $\underline{f}_{\mathrm{aux},i}(y)$, $\bar{f}_{\mathrm{aux},i}(y)$ and $\bar{f}_{\mathrm{DF},i}(y)$, respectively. These ODEs are similar to the case with one hysteresis-loop and given by,

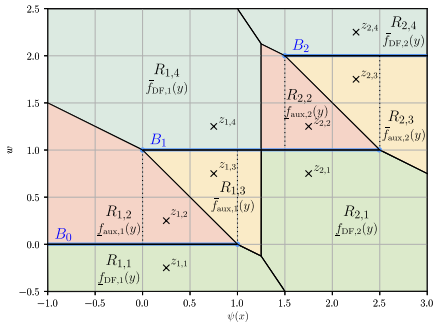
$$\underline{f}_{\mathrm{DF},i}(y) \coloneqq 2(f_{i-1}^{\mathrm{sys}}(x), 0, 1) - \underline{f}_{\mathrm{aux},i}(y), \tag{10a}$$
$$\underline{f}_{\mathrm{aux},i}(y) \coloneqq \left(\mathbf{0}_{n_x,1}, -\gamma\left(\frac{\psi(x) - U_i}{U_i - L_i}\right), 0\right), \tag{10b}$$
$$\bar{f}_{\mathrm{aux},i}(y) \coloneqq \left(\mathbf{0}_{n_x,1}, \gamma\left(\frac{\psi(x) - L_i}{U_i - L_i}\right), 0\right), \tag{10c}$$
$$\bar{f}_{\mathrm{DF},i}(y) \coloneqq 2(f_i^{\mathrm{sys}}(x), 0, 1) - \bar{f}_{\mathrm{aux},i}(y), \tag{10d}$$

where $f_i^{\mathrm{sys}}(x)$ is the ODE of FSM state i.

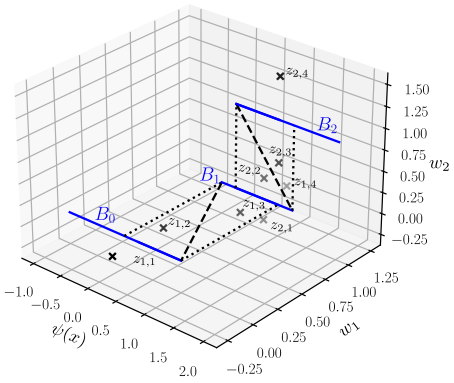
B. Variant 2: Reformulation with two auxiliary state variables for overlapping hysteresis curves

If $[L_i, U_i] \cap [L_{i+1}, U_{i+1}] \neq \emptyset$, the transition hysteresis characteristic between three subsequent states overlap. For $x \in R_{i,4} \cap R_{i+1,4}$, the established PSS reformulation would result in a sliding mode that is a convex combination of the dynamics $f_{i-1}^{\mathrm{sys}}(x)$ and $f_i^{\mathrm{sys}}(x)$. This cannot be related to the original system, thus the equivalence does not hold anymore, and we need to adapt the reformulation.

We suggest to mitigate this issue by separating the hysteresis curves in different planes. Therefore, we introduce two discrete variables w_1 and w_2 representing the original discrete variable w. When the state of the FSM changes, the discrete variables alternatingly change while the other one stays constant. The projection $\Pi(y)$ is now redefined as $\Pi(y) = (\psi(x), w_1, w_2)$ to create a three dimensional space. Figure 3 illustrates the change from the $(\psi(x), w)$-space to the $(\psi(x), w_1, w_2)$-space for an FSM with three states. Note



(a) Variant 1 - Partitioning in the $(\psi(x), w)$-space for non-overlapping hysteresis characteristics.



(b) Variant 2 - Partitioning in the $(\psi(x), w_1, w_2)$-space to handle overlapping hysteresis characteristics. This approach can also be applied to non-overlapping hysteresis characteristics.

Fig. 3. Illustration of the partitioning of the state space for a FSM with three states within different spaces for the time-freezing PSS via Voronoi regions with their corresponding auxiliary and DAE-forming dynamic's vector fields. The Voronoi points $z_{i,j}$ are marked with a cross.

that the points $z_{1,3}$ and $z_{2,1}$ would coincide in the $(\psi(x), w)$-plane for this example. For the $(\psi(x), w_1, w_2)$-hyperplane, the hysteresis is alternatingly placed in the plane w_2 for the even numbers i and in the plane w_1 for the uneven numbers of i. Visually, these planes create a 'staircase'. At every time point, the projection of the system dynamics and the states to this space should fall on these planes and thus follow the 'staircase'.

For the definition of the Voronoi regions, there is no unique expression. However, once the points are constructed in the two-dimensional (2D) $(\psi(x), w)$-space, they can be converted to the three-dimensional (3D) $(\psi(x), w_1, w_2)$-space. Constructing the points in the two-dimensional space needs some attention. There are two different cases. When the transitions do not overlap, Eq. (9) can be reused. Otherwise, when there is overlap, i.e. $\exists i \in \{0, 1, ..., n_s-1\} : L_i < U_{i-1}$, the 2D Voronoi regions should be constructed such that the points $z_{i,3}$ and $z_{i+1,1}$ coincide as well as $z_{i,4}$ and $z_{i,2}$.

Once these points are defined in two dimensions, converting them to three dimensions can be done by applying the transformations

$$\hat{z}_{i,j} = (z_{i,j,1}, z_{i,j,2} - \left\lfloor \frac{i}{2} \right\rfloor, \left\lfloor \frac{i}{2} \right\rfloor), \tag{11}$$
$$\text{for } i = 2n+1, n = 0, ..., \left\lfloor \frac{n_s - 1}{2} \right\rfloor, j \in \{1, ..., 4\},$$
$$\hat{z}_{i,j} = (z_{i,j,1}, z_{i,j,2} - \left\lfloor \frac{i+1}{2} \right\rfloor, \left\lfloor \frac{i+1}{2} \right\rfloor), \tag{12}$$
$$\text{for } i = 2n, n = 0, ... \left\lfloor \frac{n_s}{2} \right\rfloor, j \in \{1, ..., 4\},$$

with $\hat{z}_{i,j}$ the j-th points of the transition between state i and $i+1$ in the $(\psi(x), w_1, w_2)$-space. This projection folds the original 2-D plane upon the 'staircase' and adds auxiliary points around each edge to achieve sliding modes. The ODEs that belong to these Voronoi regions are alternatingly defining dynamics in the horizontal plane and the vertical plane. The next theorem allows constructing the ODEs for the Voronoi regions.

Theorem 1 (ODEs for an FSM with two state variables w_1 and w_2)**.** *Let us define regions $R_{i,j}$ as*

$$R_{i,j} = \tag{13}$$
$$\{y \in \mathbb{R}^{n_x+3} \mid \Pi(y) - \hat{z}_{i,j} \leq \min_{\substack{k \in \{1,...,n_s-1\},\\ l \in \{1,...,4\}}} (\Pi(y) - \hat{z}_{k,l})\},$$

with $\Pi(y) = (\psi(x), w_1, w_2)$ and the points $\hat{z}_{i,j}$ from Eq. (11) *and* (12) *equipped with the following ODEs*

$$\underline{f}_{\mathrm{DF},i}(y) := 2(f_{i-1}^{\mathrm{sys}}(x), 0, 0, 1) - \underline{f}_{\mathrm{aux},i}(y), \tag{14a}$$
$$\underline{f}_{\mathrm{aux},i}(y) := \left(\boldsymbol{0}_{n_x,1}, -M_i \gamma \left(\frac{\psi(x) - U_i}{U_i - L_i}\right), 0\right), \tag{14b}$$
$$\bar{f}_{\mathrm{aux},i}(y) := \left(\boldsymbol{0}_{n_x,1}, M_i \gamma \left(\frac{\psi(x) - L_i}{U_i - L_i}\right), 0\right), \tag{14c}$$
$$\bar{f}_{\mathrm{DF},i}(y) := 2(f_i^{\mathrm{sys}}(x), 0, 0, 1) - \bar{f}_{\mathrm{aux},i}(y), \tag{14d}$$

where $M_i \in \mathbb{R}^{2\times 1}$ with $M_i = [1, 0]^\top$ for even i and $M_i = [0, 1]^\top$ for odd i.
Then for $y \in B_i := (R_{i,3} \cap R_{i,4}) \cup (R_{i+1,1} \cap R_{i+1,2})$ with $i \in \{1, ..., n_s - 1\}$, it holds that:

$$F_{\mathrm{F}}(y) = \{(f_i^{\mathrm{sys}}(x), 0, 0, 1)\} \tag{15}$$
$$= \overline{\mathrm{conv}}\{\bar{f}_{\mathrm{aux},i}(y), \bar{f}_{\mathrm{DF},i}(y), \underline{f}_{\mathrm{DF},i+1}(y), \underline{f}_{\mathrm{aux},i+1}(y)\}$$

Proof. The region B_i can be divided into three parts:

- $y \in (R_{i,3} \cap R_{i,4}) \backslash (R_{i+1,1} \cap R_{i+1,2})$
- $y \in (R_{i+1,1} \cap R_{i+1,2}) \backslash (R_{i,3} \cap R_{i,4})$
- $y \in (R_{i+1,1} \cap R_{i+1,2}) \cap (R_{i,3} \cap R_{i,4})$.

For the first two regions, the proof is similar to Proposition 2 of [13]. For the last region

$$y \in (R_{i+1,1} \cap R_{i+1,2}) \cap (R_{i,3} \cap R_{i,4}) \tag{16}$$
$$= \{y \mid c(y) := w_1 + w_2 = i, \psi(x) \in (L_{i+1}, U_i)\}$$

a sliding mode is activate [9], since the following holds

$$\nabla c(y)^\top \underline{f}_{\mathrm{aux},i+1}(y) < 0,$$
$$\nabla c(y)^\top \underline{f}_{\mathrm{DF},i+1}(y) > 0,$$
$$\nabla c(y)^\top \bar{f}_{\mathrm{aux},i}(y) < 0,$$
$$\nabla c(y)^\top \bar{f}_{\mathrm{DF},i}(y) > 0$$

with $w_1 = \lfloor \frac{i+1}{2} \rfloor$, $w_2 = \lfloor \frac{i}{2} \rfloor$ and $\frac{dw_1}{d\tau} = \frac{dw_2}{d\tau} = 0$. From Eq. (2), we have

$$\begin{aligned} F_{\mathrm{F}}(y) = &\{\theta_{3,i}(2(f_i^{\mathrm{sys}}(x),0,0,1) - \bar{f}_{\mathrm{aux},i}(y)) \\ &+ \theta_{4,i}\bar{f}_{\mathrm{aux},i}(y) + \theta_{2,i+1}\underline{f}_{\mathrm{aux},i+1}(y) \\ &+ \theta_{1,i+1}(2(f_i^{\mathrm{sys}}(x),0,0,1) - \underline{f}_{\mathrm{aux},i+1}(y)) \\ &\text{with } \theta_{3,i} + \theta_{4,i} + \theta_{1,i+1} + \theta_{2,i+1} = 1, \\ &\theta_{3,i}, \theta_{4,i}, \theta_{1,i+1}, \theta_{2,i+1} > 0\}. \end{aligned}$$

From this relation, $w_1' = 0$ and $w_2' = 0$, we obtain that $\theta_{3,i} - \theta_{4,i} = 0$ and $\theta_{1,i+1} - \theta_{2,i+1} = 0$. Solving for $\theta_{3,i}$, $\theta_{4,i}$, $\theta_{1,i+1}$, $\theta_{2,i+1}$ results in $\theta_{3,i} = \frac{1}{2} - \theta_{1,i+1}$ with $\theta_{3,i} = \theta_{4,i}$ and $\theta_{1,i+1} = \theta_{2,i+1}$, which yields $F_{\mathrm{F}}(y) = \{(f_i^{\mathrm{sys}}(x),0,0,1)\}$. This completes the proof. □

Based on Theorem 1 and [13, Proposition 2], we can conclude that the system has a uniquely defined sliding mode for every stage such that the region B_i is equipped with $F_{\mathrm{F}}(y) = \{(f_i^{\mathrm{sys}}(x),0,0,1)\}$ for $i \in \{0,...,n_s-1\}$. Since the sliding mode is equal to the original dynamics, Eq. (15) can be used to model the original system with only continous variables.

IV. Comparison for a Numerical Example: Time Optimal Problem of a Car with Gearbox

In this section, we apply the developed methodology to a numerical example of the time-optimal control problem of a car with a gearbox using four different gears. The implementation is based on the double-integrator car model with turbo from [6]. This OCP is nonlinear and nonsmooth due to the gearbox together with the time-optimal control. The car model is described using a position $q(t)$, a velocity $v(t)$ and a gearbox state $w(t) \in \{0,1,2,3\}$. The car is controlled using an accelaration $u(t)$ which is bounded by $|u| \leq \bar{u}$, $\bar{u} = 2.5\mathrm{m/s}^2$. The velocity is bounded by $|v| \leq \bar{v}$, $\bar{v} = 30\mathrm{m/s}$. The gearbox multiplies the controlled accelaration and automatically shifts up or down when a speed limit is reached. These speed limits are given by U_i and L_i for gear i. We consider two operating modes to shift gears:

- Operating mode (a) has non-overlapping hysteresis curves and is given by $L^a = [5, 12.5, 20]$ m/s, and $U^a = [10, 17.5, 25]$ m/s..
- Operating mode (b) has overlapping hysteresis curves and is given by $L^b = [5, 7.5, 10]$ m/s and $U^b = [10, 12.5, 15]$ m/s.

The state vector is given by $z = (q, v, w) \in \mathbb{R}^3$ and the ODE is given by $f_i^{\mathrm{sys}}(z) = (v, (i+1)u, 0)$ for gear $i = 0, 1, 2, 3$. The OCP considers the numerical time interval $\tau \in [0, \tau_{\mathrm{f}}]$ and aims to minimize final physical time $t(\tau_{\mathrm{f}})$. It uses a time-freezing PSS associated with the car model based on the reformulations of Section III-A and III-B. At the end of the numerical time τ, the car should reach a final position $q(t(\tau_{\mathrm{f}})) = q_{\mathrm{f}} = 300$ m with $v(t(\tau_{\mathrm{f}})) = v_{\mathrm{f}} = 0$ m/s at τ_{f}. The

TABLE I
Comparison of different methods for (17) operating mode (a).

Solver	T_{f}	CPU Time [s]	$E(T_{\mathrm{f}})$
`nosnoc_py` Var. 1	21.96	428.69	2.85e-71
`nosnoc_py` Var. 2	**16.69**	905.84	8.5e-60
`Gurobi` with bisection	20.00	**731.38**	0
`Bonmin`	23.27	7200+	2.67e-12

OCP is given by

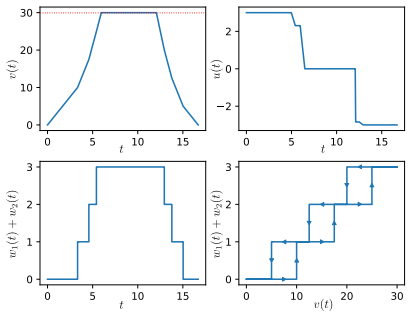
$$\begin{aligned} \min_{y(\cdot),u(\cdot),s(\cdot)} \quad & t(\tau_{\mathrm{f}}) && \text{(17a)} \\ \text{s.t.} \quad & y(0) = (z_0, 0), && \text{(17b)} \\ & y'(\tau) \in s(\tau) F_{\mathrm{F}}(y(\tau), u(\tau)),\ \tau \in [0, \tau_{\mathrm{f}}], && \text{(17c)} \\ & -\bar{u} \leq u(\tau) \leq \bar{u},\ \tau \in [0, \tau_{\mathrm{f}}], && \text{(17d)} \\ & \bar{s}^{-1} \leq s(\tau) \leq \bar{s},\ \tau \in [0, \tau_{\mathrm{f}}], && \text{(17e)} \\ & -\bar{v} \leq v(\tau) \leq \bar{v},\ \tau \in [0, \tau_{\mathrm{f}}], && \text{(17f)} \\ & (q(\tau_{\mathrm{f}}), v(\tau_{\mathrm{f}})) = (q_{\mathrm{f}}, v_{\mathrm{f}}). && \text{(17g)} \end{aligned}$$

where a variable terminal physical time $T_{\mathrm{f}} = t(\tau_{\mathrm{f}})$ is achieved using time transformation. This time transformation is achieved using a scalar *speed-of-time* control variable $s(\cdot)$. This *speed-of-time* is bounded by Eq. (17e) with $\bar{s} = 20$. The problem is discretized using a FESD Radau-IIA scheme of order 3 with $N = 15$ control intervals and $N_{\mathrm{FE}} = 3$ intermediate integration steps. In total, there are 45 grid points where the gear of the gearbox can change. The discretization and MPCC homotopy are carried out using the open-source tool `nosnoc_py`, which uses `IPOPT` [20] and `CasADi` [4]. The homotopy mode was "two-sided elastic" [5].

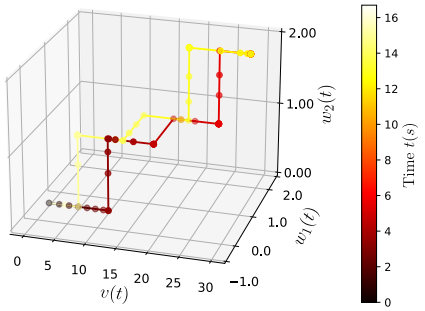
The two approaches from above are compared to a mixed-integer formulation based on [6], which results in 16 binary variables per interval. In order to simplify the problem, a switch in this formulation is only allowed at the interval boundaries. For a fair comparison, a lower speed limit for each gear avoids using a gear together with speed lower than the lower switching speed with a tolerance $v_{\mathrm{tol}} = 1\mathrm{m/s}$. The problem is solved using a MINLP solver `Bonmin` [7].

A second method uses a bisection-type search on the optimal time T_f for the shortest feasible time up to the accuracy of 10^{-5} s. This method reduces the OCP to a mixed-integer linear program which is solved by the commercial solver `Gurobi` [10].

All algorithms run on a single CPU core to have a fair comparison and have a calculation time limit of 7200 s. The source code is located in the example section of the `nosnoc_py` repository [2]. The results for the two cases are summarized in Table I and II, which includes the calculation time, the time to reach the endpoint and the terminal constraint satisfaction $E(T_{\mathrm{f}}) = \|x_{\mathrm{sim}}(T_{\mathrm{f}}) - (q_{\mathrm{f}}, v_{\mathrm{f}})\|_2$. Since for both `Gurobi` and `Bonmin` the switches need to coincide with the control interval boundaries, the required time to achieve the endpoint is always higher than the result of `nosnoc_py` using two auxiliary variables. The switch detection within `nosnoc_py` allows computing the optimal point for a switch between gears resulting in a lower objective value. The `nosnoc_py` implementation with



(a) Solution of the OCP in physical time. The top left and right plots show the velocity and control input respectively in function of the actual time. The bottom left and right show the sum of auxiliary variables in function of time and in function of the velocity. The arrows indicate the direction in time.



(b) 3D view of the speed and the auxiliary variables w_1 and w_2

Fig. 4. Solution of the gearbox example in operating mode (a). The image below shows a 3D view of the speed and auxiliary variables

TABLE II

COMPARISON OF DIFFERENT METHODS FOR (17) OPERATING MODE (b).

Solver	T_f	CPU Time [s]	$E(T_\mathrm{f})$
`nosnoc_py` Var. 2	**20.80**	**501.60**	2.36e-36
`Gurobi` with bisection	27.75	3501.44	1.99
`Bonmin`	34.28	7200+	7.91e-11

a single auxiliary variable fails to use all existing gears. The solution is stuck in a local minima that can be caused by the homotopy iterations. The implementation with two auxiliary variables is more robust against these local minima. The results of the `nosnoc_py` implementation with two auxiliary variables are shown in Figure 4. The computation time of `nosnoc_py` is similar to `Gurobi` while `Bonmin` is significantly slower and can not reach the global minimum due to the computation time limit. For operating mode (b), `nosnoc_py` is faster.

V. CONCLUSION

The paper introduces a novel time-freezing reformulation for a class of hybrid systems with Finite-state Machines (FSMs). This class has an ordered set of modes and needs to be operated using one single switching function. The reformulation uses state jumps to reformulate the system into a Piecewise Smooth System (PSS) and leverages the Finite Elements reformulation with Switch Detection (FESD) method to achieve an optimal control result where the transition between modes of the FSM can be determined accurately.

This approach avoids solving complex mixed-integer iterations and only relies on NLP iterates. The paper gives a theoretical background on improvements of the hysteresis reformulation [13] and a theoretical justification for the Differential Algebraic Equations (DAE)-forming Ordinary Differential Equations (ODEs). Future work includes generalizing this approach to a generic FSM and allowing the solver to handle multiple FSM within a single Optimal Control Problem (OCP).

REFERENCES

[1] NOSNOC. https://github.com/nurkanovic/nosnoc, 2022.

[2] nosnoc_py. https://github.com/FreyJo/nosnoc_py, 2023.

[3] Rajeev Alur, Costas Courcoubetis, Thomas A Henzinger, and Pei-Hsin Ho. Hybrid automata: An algorithmic approach to the specification and verification of hybrid systems. Technical report, Cornell University, 1993.

[4] Joel A. E. Andersson, Joris Gillis, Greg Horn, James B Rawlings, and Moritz Diehl. CasADi – a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11(1):1–36, 2019.

[5] Mihai Anitescu, Paul Tseng, and Stephen J. Wright. Elastic-mode algorithms for mathematical programs with equilibrium constraints: global convergence and stationarity properties. *Mathematical Programming*, 110(2):337–371, 2007.

[6] Marios Avraam. *Modelling and optimisation of hybrid dynamic processes.* PhD thesis, Imperial College London (University of London), 2000.

[7] Pierre Bonami, Lorenz T Biegler, Andrew R Conn, Gérard Cornuéjols, Ignacio E Grossmann, Carl D Laird, Jon Lee, Andrea Lodi, François Margot, Nicolas Sawaya, et al. An algorithmic framework for convex mixed integer nonlinear programs. *Discrete optimization*, 5(2):186–204, 2008.

[8] Miguel Carrión and José M Arroyo. A computationally efficient mixed-integer linear formulation for the thermal unit commitment problem. *IEEE Transactions on power systems*, 21(3):1371–1378, 2006.

[9] Aleksei Fedorovich Filippov. *Differential Equations with Discontinuous Righthand Sides*, volume 18. Springer Science & Business Media, Series: Mathematics and its Applications (MASS), 2013.

[10] Gurobi Optimization. `http://www.gurobi.com`.

[11] IBM Corp. *IBM ILOG CPLEX V12.1, User's Manual for CPLEX*, 2009.

[12] Jan Lunze and Françoise Lamnabhi-Lagarrigue. *Handbook of hybrid systems control: theory, tools, applications*. Cambridge University Press, 2009.

[13] Armin Nurkanović and Moritz Diehl. Continuous optimization for control of hybrid systems with hysteresis via time-freezing. *IEEE Control Systems Letters*, 2022.

[14] Armin Nurkanović and Moritz Diehl. NOSNOC: A software package for numerical optimal control of nonsmooth systems. *IEEE Control Systems Letters (L-CSS)*, 6:3110–3115, 2022.

[15] Armin Nurkanović, Mario Sperl, Sebastian Albrecht, and Moritz Diehl. Finite elements with switch detection for direct optimal control of nonsmooth systems. *arXiv preprint arXiv:2205.05337*, 2022.

[16] André Platzer. Differential dynamic logic for hybrid systems. *Journal of Automated Reasoning*, 41(2):143–189, 2008.

[17] Andrew Sogokon, Khalil Ghorbal, and Taylor T Johnson. Operational models for piecewise-smooth systems. *ACM Transactions on Embedded Computing Systems (TECS)*, 16(5s):1–19, 2017.

[18] David E. Stewart. A high accuracy method for solving ODEs with discontinuous right-hand side. *Numerische Mathematik*, 58(1):299–328, 1990.

[19] Augusto Visintin. *Differential models of hysteresis*, volume Appl. Math. Sci. 111. Springer-Verlag, Berlin, 1994.

[20] Andreas Wächter and Lorenz T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.