A Switched Reference Governor for High Performance Trajectory Tracking Control under State and Input Constraints

Nan Wang[1], Stefano Di Cairano[2], and Ricardo G. Sanfelice[1]

***Abstract*— This paper proposes a switched reference governor (RG) algorithm to achieve rapid and non-oscilliatory convergence to a given reference signal while satisfying the imposed constraints by switching between a fast controller and a slow controller. The proposed algorithm computes the set of state and admissible reference pairs for both controllers offline. At each iteration, it computes the admissible reference sets for each controller at the current state and activates one of the controllers based on the distance between the state and the reference. After a controller is activated, a lightweight optimization problem is solved to find an admissible reference that is closest to the reference signal. The solution, which is referred to as the virtual reference, is used as the reference signal. Recursive feasibility and convergence of the virtual reference to the given reference signal, among other key properties of the proposed switched RG, are shown and illustrated in a system.**

I. Introduction

Enforcing constraints in control systems by design is a new challenge in many control applications. Constraints often manifest as actuator magnitude and rate limitations, allowed ranges of process variables for safe, efficient system function, and requirements for collision avoidance. Constraint-handling methodologies in the context of feedback control include model predictive control (MPC) [1], control barrier functions (CBFs) [2], [3], [4], and reference governor (RG) [5]. MPC necessitates a comprehensive controller redesign incorporating constraints and it is known to result in significant computational burden. Similarly, CBF poses design challenges especially for recursive feasibility when both input and output constraints are present, and requires solving optimization problems that typically involve nonlinearities. RG is a state-feedback control law that modulates the reference signal of a pre-stabilized plant. Compared to MPC, RG utilizes existing or legacy controllers and supplements them with constraint management capacities. Different from CBF, RG may not require optimization and systematic methods to achieve recursive feasibility, leading to a streamlined design and enhanced computational efficiency, albeit at the cost of potentially reduced closed-loop performance [6].

[1]Nan Wang and Ricardo G. Sanfelice are with the Department of Electrical and Computer Engineering, University of California, Santa Cruz, CA 95064, USA; nanwang@ucsc.edu, ricardo@ucsc.edu

[2]Stefano Di Cairano is with the Mitsubishi Electric Research Laboratories, Cambridge, MA 02139-1955 dicairano@merl.com

[1]Nan Wang was a research intern at MERL during this work.

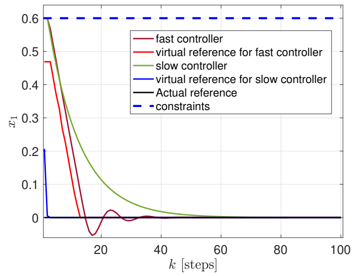
[1]Research by Ricardo G. Sanfelice is partially supported by NSF Grants no. CNS-2039054 and CNS-2111688, by AFOSR Grants nos. FA9550-19-1-0169, FA9550-20-1-0238, FA9550-23-1-0145, and FA9550-23-1-0313, by AFRL Grant nos. FA8651-22-1-0017 and FA8651-23-1-0004, by ARO Grant no. W911NF-20-1-0253, and by DoD Grant no. W911NF-23-1-0158.

Switching control strategies have garnered significant attention for their potential to improve performance when using RG. In [7], a switching RG mechanism for varying system operating points is introduced, alongside a finite state machine for transitioning between RG paradigms. The framework presented in [8] utilizes remote control of nonlinear discrete-time systems and dynamical transitions between two feedback controllers based on remotely computed virtual reference commands. The approach in [9] employs a supervisory RG method for load/frequency control in networked multi-area power systems, enhancing disturbance rejection performance by switching between multiple RGs that consider different configurations of disturbance. Specialized schemes are also emerging, such as a switched RG designed specifically for linear motor-driven systems in [10] and a data-driven switched RG for constrained braking systems in [11], based on the methodology presented in [12]. However, most of the switching mechanism in the references above are designed from experience or driven by data, which do not lead to formal guarantees of their properties.

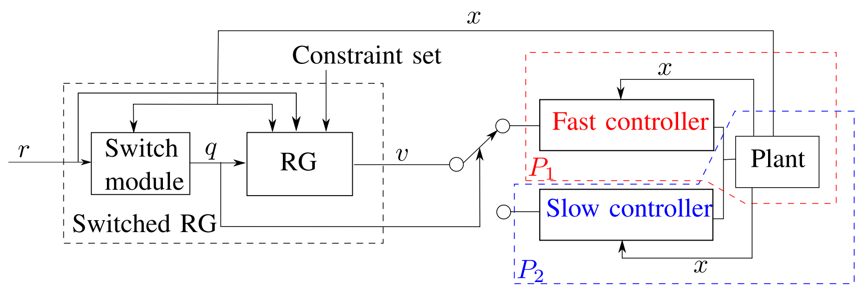
This paper presents a switched reference governor underpinned by Lyapunov-based techniques that transitions between a fast (potentially oscillatory) controller and a slow (and non-oscillatory) controller. The controllers with both characteristics are commonly encountered in control system design and can be exemplified using the double-integrator system

$$x^+ = \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \tag{1}$$

where $x := (x_1, x_2) \in \mathbb{R}^2$ and $u \in \mathbb{R}$. Two controllers, denoted $\gamma_1 : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}$ and $\gamma_2 : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}$, are designed for (1) as follows: $\gamma_1(x, v) := \begin{bmatrix} -0.5 & -0.5 \end{bmatrix} x + 0.5v$, $\gamma_2(x, v) := \begin{bmatrix} -0.25 & -1.5 \end{bmatrix} x + 0.25v$, where $v \in \mathbb{R}$ is the exogenous reference. With the same reference, illustrated in Figure 1a, the state trajectory resulting from applying γ_1 achieves very close proximity ($\approx 1\%$ error) to the actual reference roughly after 17 seconds, classifying γ_1 as a *fast controller*. Conversely, when γ_2 is applied, it takes approximately 50 seconds for the state trajectory to first reach very close proximity ($\approx 1\%$ error) to the reference, with no subsequent oscillations observed. Consequently, γ_2 is categorized as a *slow* controller. The individual controllers are not satisfactory due to oscillatory behavior of γ_1 and slow convergence of γ_2. In this paper, we design a switched RG algorithm that combines controllers to exploit their individual advantages, such as a fast convergence rate and nonoscillatory behavior, and to ensure that the constraints are



(a) The trajectory of x_1 in (1) controlled by fast controller γ_1 and slow controller γ_2.



(b) The proposed switched RG diagram. The switch RG consists of two modules: 1) a switch module that decides which controller to activate from the current state x and the reference r and 2) an RG module to compute the virtual reference v from the selected controller q, the actual reference r, the current state x, and the constraints Y.

Fig. 1: State trajectories of double integrator example and the proposed switched RG diagram.

satisfied. The proposed algorithm is designed by determining the set of state-reference pairs for the closed-loop systems controlled by the fast controller and by the slow controller. When the switched RG is running online, at each of its iterations, it determines the sets of admissible references for the fast controller and the slow controller at the current state. The decision to change the controller depends on whether these reference sets are nonempty and also on the value of a Lyapunov function. After the selection of an appropriate controller, the proposed switched RG selects a virtual reference within the corresponding admissible reference set that has minimal distance to the given reference (r) and applies it as the reference to the closed-loop system to steer its behavior while satisfying the constraints.

The remainder of the paper is structured as follows. Section II presents notation and preliminaries. Section III presents the problem statement. Section IV presents the switched RG algorithm. Section V presents the analysis of the theoretical guarantees. Section VI presents the simulation results. Due to space limits, the proofs are removed and will be published elsewhere.

II. Preliminaries

A. Notation

Given a point $x \in \mathbb{R}^n$ and a set $S \subset \mathbb{R}^n$, the distance between x and S is denoted $|x|_S := \inf_{s \in S} |x - s|$. The notation ∂S denotes the boundary of the set S. The notation $\overline{S}$ denotes the closure of the set S. The set of natural numbers including zero is denoted as $\mathbb{N}$ and the set of positive numbers as $\mathbb{N}_+$, i.e., $\mathbb{N}_+ = \{1, 2, ...\}$. Given a function $\phi : \mathbb{R}^n \to \mathbb{R}^m$, $\operatorname{dom} \phi$ denotes the domain of ϕ. The closed unit ball in the Euclidean norm is denoted by $\mathbb{B}$.

B. Closed-loop Tracking System

A discrete-time system P is modeled as

$$P : x^+ = G(x, v), \quad y = h(x, v), \quad z = E(x) \tag{2}$$

where $x \in \mathbb{R}^n$ is the state, $y \in \mathbb{R}^p$ is the constrained output that is constrained to be in the output constraint set $Y \subset \mathbb{R}^p$, $z \in \mathbb{R}^m$ is the performance output that is to track a reference signal denoted $r \in \mathcal{R} \subseteq \mathbb{R}^m$, where $\mathcal{R}$ is a given set of references, and $v \in \mathbb{R}^m$ is the virtual reference. For any virtual reference $k \mapsto v(k)$, the solution to P in (2) is a function $\phi : \operatorname{dom} \phi \to \mathbb{R}^n$ such that $\operatorname{dom} \phi = \operatorname{dom} v$ and $\phi(k+1) = G(\phi(k), v(k))$ for each $k \in \operatorname{dom} \phi \backslash \{\sup \operatorname{dom} \phi\}$. In addition, a solution ϕ is nontrivial if $\operatorname{dom} \phi$ contains at least two points, maximal if there does not exist another solution ϕ' such that $\operatorname{dom} \phi \subset \operatorname{dom} \phi'$ and $\phi(k) = \phi'(k)$ for each $k \in \operatorname{dom} \phi$, and complete if $\operatorname{dom} \phi$ is unbounded. As in this paper we propose an algorithm that switches between two controllers, γ_1 and γ_2, and properly determines the virtual reference v, we model the closed-loop system controlled by γ_1 as in (2) and denote it as P_1. Similarly, the closed-loop system resulting from using γ_2 is modeled as in (2) and denoted as P_2.

C. Stability, Attractivity, and Asymptotic Stability

For the analysis of properties, we provide definitions for stability, attractivity, and asymptotic stability. We first introduce a set-valued map $\mathcal{A}^P : \mathbb{R}^m \rightrightarrows \mathbb{R}^n$, which, given a virtual reference v, collects the equilibria of P in (2). Mathematically,

$$v \mapsto \mathcal{A}^P(v) := \{x_p \in \mathbb{R}^n : E(x_p) = v, G(x_p, v) = x_p\}. \tag{3}$$

Definition 2.1 (Asymptotic stability): Given $v \in \mathbb{R}^m$ and P as in (2) such that $\mathcal{A}^P(v) \subset \mathbb{R}^n$ is nonempty, the set $\mathcal{A}^P(v)$ is said to be

1) Lyapunov stable for P if for every $\epsilon > 0$, there exists $\delta > 0$ such that each solution ϕ to P with $|\phi(0)|_{\mathcal{A}^P(v)} \leq \delta$ satisfies $|\phi(k)|_{\mathcal{A}^P(v)} \leq \epsilon$ for each $k \in \operatorname{dom} \phi$.
2) Attractive for P with basin of attraction $\mathcal{U} \subset \mathbb{R}^n$ if each maximal solution ϕ to P with $\phi(0) \in \mathcal{U}$ is complete and satisfies $\lim_{k \to \infty} |\phi(k)|_{\mathcal{A}^P(v)} = 0$.
3) Asymptotically stable for P with basin of attraction $\mathcal{U}$ if it is Lyapunov stable and attractive with basin of attraction $\mathcal{U}$.

For any virtual reference $v \in \mathbb{R}^m$, the set $\mathcal{A}^{P_1}(v)$ is assumed to be attractive for P_1 and $\mathcal{A}^{P_2}(v)$ is assumed to be asymptotically stable for P_2, as explicitly articulated later in Assumption 3.1. This facilitates the designing of a switching scheme between P_1 and P_2.

III. Problem Formulation

In this paper, we design a switched RG that is equipped with a fast controller and a slow controller to track the given reference while satisfying the constraints. The problem is formulated as follows.

Problem 1: Given two discrete-time closed-loop systems obtained from controllers γ_1 and γ_2, denoted as P_1 and P_2, respectively, modeled as in (2), develop a reference governor to switch between P_1 and P_2 utilizing their distinct characteristics–specifically, slow convergence without overshoot and fast convergence with overshoot–while ensuring that the constrained output y belongs to the set Y.

The switched RG unites the fast controller, used when far from the reference to converge rapidly to it, with the slow controller, used when close to the reference to avoid oscillations. As is shown in Figure 1b, the switched RG consists of two key modules: a switch module and an RG module. The switch module is responsible for switching between the fast controller and slow controller, while the RG module is designed to compute the virtual reference for the currently selected controller. We make the following assumption.

Assumption 3.1: There exist a set-valued map $\mathcal{U}_0 : \mathbb{R}^m \rightrightarrows \mathbb{R}^n$ such that, given reference $r \in \mathcal{R}$, for each $v \in \mathbb{R}^m$, both $\mathcal{A}^{P_1}(v)$ and $\mathcal{A}^{P_2}(v)$ are nonempty, $\mathcal{U}_0(r)$ is an open set and contains an open neighborhood of $\mathcal{A}^{P_2}(r)$, and

1) for each $v \in \mathbb{R}^m$, $\mathcal{A}^{P_1}(v)$ is attractive for P_1 with basin of attraction $\mathbb{R}^n$;
2) for each $v \in \mathbb{R}^m$, $\mathcal{A}^{P_2}(v)$ is asymptotically stable for P_2, and, particularly, $\mathcal{A}^{P_2}(r)$ is asymptotically stable for P_2 with open basin of attraction containing $\overline{\mathcal{U}_0(r)}$.

Remark 3.2: Items 1 and 2 in Assumption 3.1 assert that the fast controller γ_1 induces global attractivity to $\mathcal{A}^{P_1}(v)$ for any fixed v produced by RG and, as the RG drives v to converge to r, eventually steering the state to nearby $\mathcal{A}^{P_1}(r)$ for P_1, which is situated within the basin of attraction of $\mathcal{A}^{P_2}(r)$ for P_2. The existence of the map $\mathcal{U}_0$ is for free since the basin of attraction is open. This map will be used later in the construction of the switching logic.

The proposed RG is designed to track a discrete-time exogenous desired reference r for the performance output z. RG is formulated to compute the virtual reference v derived based on the current state x and the reference r, which is then applied to system (2). Consequently, by inputting the virtual reference v, instead of the reference r into the closed-loop system (2), the constraint $y \in Y$ is maintained.

Most RG methodologies update the virtual reference, v, at every time instance. If v is constantly applied from a particular time instant onward, the resulting output will always adhere to the imposed constraints. A *maximal output admissible set* O_∞^P collects all states x and constant virtual references v such that the response initiating from state x and using a constant virtual reference v satisfies the constraints at all future time instances, namely,

$$O_\infty^P := \{(v, x) \in \mathbb{R}^m \times \mathbb{R}^n : \hat{y}_P(k|v, x) \in Y \ \forall k \in \mathbb{N}_+\},$$

where $\hat{y}_P(k|v, x)$ denotes the (predicted) constrained output y of the closed-loop system P at time k when starting from the initial state x with constant virtual reference v. The computation of O_∞^P is typically performed offline, requiring no calibration, and, in most cases, involves polynomial computational complexity. In certain cases it may be more convenient to compute an invariant subset of O_∞^P, denoted S_∞^P and defined as follows:

$$S_\infty^P := \{(v, x) \in S^P : (v, G(x, v)) \in S_\infty^P\} \subseteq O_\infty^P \qquad (4)$$

The set S_∞^P is a subset of O_∞^P, implying the admissibility of all points in S_∞^P. For each $(v, x) \in S_\infty^P$, we have $(v, G(x, v)) \in S_\infty^P$, ensuring forward invariance. The following assumption posits that the pairs of reference r and its equilibria are contained in both $S_\infty^{P_1}$ and $S_\infty^{P_2}$.

Assumption 3.3: The set of references $\mathcal{R} \subset \mathbb{R}^m$ is such that $\{(r, x) \in \mathcal{R} \times \mathbb{R}^n : x \in \mathcal{A}^{P_i}(r)\} \subset S_\infty^{P_i}$, for each $i \in \{1, 2\}$.

IV. Methodology

The switched RG algorithm relies on the output admissible sets of P_1 and P_2, respectively, $S_\infty^{P_1}$ and $S_\infty^{P_2}$, as outlined in (4). The construction of these sets can be accomplished offline using pre-existing methodologies for computing invariant sets, including the iterative method, the optimization-based method, and the polyhedral method. In this paper, the polyhedral method is employed.

The switched RG algorithm performs the following steps:

1) At the current state x, compute the admissible reference
$$\mathcal{V}_q(x) = \{v \in \mathbb{R}^m : (v, x) \in S_\infty^{P_q}\} \quad \forall q \in \{1, 2\}. \ (5)$$
2) Evaluate the switching logic (defined below) and update the mode q if necessary.
3) Solve the following optimization problem:
Problem 2: Given the reference $r \in \mathcal{R}$, the current mode $q \in \{1, 2\}$, and the current state $x \in \mathbb{R}^n$, solve
$$\underset{v \in \mathcal{V}_q(x)}{\arg\min} \quad |v - r|^2. \qquad (6)$$
The optimization process for solving Problem 2 simply involves a single-dimensional search within the set $\mathcal{V}_q(x)$, which can be performed numerically quite efficiently.

A. Switching Logic based on Lyapunov Function

We propose the following switching logic.

Switching Logic: Given the current state x, the current mode $q \in \{1, 2\}$, where $q = 1$ corresponds to the fast controller and $q = 2$ corresponds to the slow controller, and the reference r,

1) If $\mathcal{V}_q(x)$ is empty and $\mathcal{V}_{3-q}(x)$ is nonempty, then q is reset to $3 - q$, making P_{3-q} the active closed-loop system.
2) If $q = 1$, namely, the active closed-loop system is P_1, and
   a) both $\mathcal{V}_1(x)$ and $\mathcal{V}_2(x)$ are nonempty;

b) the state x is "close" to the reference r;

c) the virtual reference provided for P_2 is closer to r than the virtual reference at the previous time step,

then reset q to 2, so that the active closed-loop system is P_2 to avoid oscillations during convergence.

3) If $q = 2$, namely, the active closed-loop system is P_2, and

a) both $\mathcal{V}_1(x)$ and $\mathcal{V}_2(x)$ are nonempty;

b) the state x is "far away" from the reference r,

c) the virtual reference provided for P_1 is closer to r than the virtual reference at the previous time step,

then reset q to 1, so that the active closed-loop system is P_1 to achieve fast convergence.

In items 2.b and 3.b of the switching logic, the distance between the state $x \in \mathbb{R}^n$ and the reference $r \in \mathcal{R} \subset \mathbb{R}^m$ needs to be properly defined. Note that x and r may have different dimensions, making the norm of $x - r$ not suitable as distance. In this paper, a Lyapunov function is employed to capture this "closeness." To define Lyapunov function properly, we start with defining the positive definite functions.

Definition 4.1 (Positive definite function): A function $\rho : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is positive definite, also written $\rho \in \mathcal{PD}$, if $\rho(s) > 0$ for all $s > 0$ and $\rho(0) = 0$.

With this definition in place, we define a Lyapunov function notion that parameterizes the function by the reference r.

Definition 4.2 (Lyapunov function): Given P in (2) and the reference $r \in \mathcal{R}$, a function $V : \mathbb{R}^n \times \mathcal{R} \to \mathbb{R}$ is called a Lyapunov function for P relative to $\mathcal{A}^P(r)$ within the set $S \subset \mathbb{R}^n$ if

1) $V(x, r) > 0$ for all $x \in S \backslash \mathcal{A}^P(r)$, and $V(x, r) = 0$ for all $x \in \mathcal{A}^P(r)$,
2) $V(G(x, r), r) - V(x, r) \leq -\rho(|x|_{\mathcal{A}^P(r)})$ for all $x \in S$, where $\rho \in \mathcal{PD}$ is continuous.

Remark 4.3: In Assumption 3.1, for each $r \in \mathcal{R}$, P_2 is assumed to have the set $\mathcal{A}^{P_2}(r)$ asymptotically stable with a basin of attraction containing $\overline{\mathcal{U}_0(r)}$. By the converse theorem in [13], for each r, there exists a Lyapunov function $V : \mathbb{R}^n \times \{r\} \to \mathbb{R}$ for P_2 relative to $\mathcal{A}^{P_2}(r)$ within $\overline{\mathcal{U}_0(r)}$.

The update of q is performed by the `SwitchingLogic` function in Algorithm 1. In addition to x, q, $S_\infty^{P_1}$, $S_\infty^{P_2}$ and r, we also employ c_1 and c_2 such that $c_1 > c_2$ as the threshold of the Lyapunov function to trigger the switch. Setting $c_1 > c_2$ avoids chattering between the controllers. We also define v_{prev} as the value of the virtual reference at the previous time step. In Lines 3-7, Algorithm 1 initially handles the corner case in item 1 in **Switching Logic** where no admissible input is available. Item 2 in **Switching Logic** is implemented in Lines 8-9. Item 3 in **Switching Logic** is implemented in Lines 10-12. If no switch is triggered, the current value of q does not change; see Line 13.

To ensure robust switching, we construct the set-valued map $\mathcal{T}_{1,0} : \mathcal{R} \rightrightarrows \mathbb{R}^n$ such that $\mathcal{T}_{1,0}(r)$ is a closed set,

$$\mathcal{A}^{P_1}(r) + \delta_0^c \mathbb{B} \subset \mathcal{T}_{1,0}(r), \quad \mathcal{T}_{1,0}(r) + 2\delta_0 \mathbb{B} \subset \mathcal{U}_0(r)$$

for positive constants δ_0 and δ_0^c, and each solution to the closed-loop system P_2 with initial condition in $\mathcal{T}_{1,0}(r)$ resulting from applying γ_2 remains in $\mathcal{U}_0(r)$, where the set-valued map $\mathcal{U}_0$ comes from Assumption 3.1. From the design of the switching logic, the sets $\mathcal{U}_0(r)$ and $\mathcal{T}_{1,0}(r)$ are constructed as

$$\mathcal{T}_{1,0}(r) := \{x \in \mathbb{R}^n : V(x, r) \leq c_2\} \tag{7}$$

and $\mathcal{U}_0(r) := \{x \in \mathbb{R}^n : V(x, r) \leq c_1\}$. Note that with an arbitrary $c_2 > 0$, we cannot ensure $\mathcal{A}^{P_1}(r) + \delta_0^c \mathbb{B} \subset \mathcal{T}_{1,0}(r)$ because the set $\mathcal{A}^{P_1}(r)$ may not be a subset of $\mathcal{T}_{1,0}(r)$. Therefore, an additional assumption needs to be placed on c_2. Specifically, for sufficiently large c_2, $\mathcal{T}_{1,0}(r)$ has to contain $\mathcal{A}^{P_1}(r)$. This assumption is imposed in Assumption 5.4 below. $\mathcal{T}_{1,0}$ and $\mathcal{U}_0$ are captured by the second conditions in Line 8 and Line 10 in Algorithm 1, respectively.

Algorithm 1 Switching Logic

1: **function** $q \leftarrow$ SWITCHINGLOGIC($x, q, S_\infty^{P_1}, S_\infty^{P_2}, r, c_1, c_2, v_{prev}$)
2: Compute $\mathcal{V}_1(x) \leftarrow \{v \in \mathbb{R}^m : (v, x) \in S_\infty^{P_1}\}$ and $\mathcal{V}_2(x) \leftarrow \{v \in \mathbb{R}^m : (v, x) \in S_\infty^{P_2}\}$.
3: **for** $q = 1, 2$ **do**
4: **if** $\mathcal{V}_q(x) = \emptyset$ and $\mathcal{V}_{3-q}(x) \neq \emptyset$ **then**
5: **return** $3 - q$.
6: **end if**
7: **end for**
8: **if** $q = 1$ and $V(x, r) \leq c_2$ and $\min_{v \in \mathcal{V}_2(x)} |v - r| \leq |v_{prev} - r|$ **then**
9: **return** 2.
10: **else if** $q = 2$ and $V(x, r) \geq c_1$ and $\min_{v \in \mathcal{V}_1(x)} |v - r| \leq |v_{prev} - r|$ **then**
11: **return** 1.
12: **end if**
13: **return** q.
14: **end function**

Algorithm 2 Switched Reference Governor

Require: r, $c_1, c_2 \in \mathbb{R}_{>0}$ such that $c_1 > c_2$, $S_\infty^{P_1}$ and $S_\infty^{P_2}$
1: $k \leftarrow 0$, $x_0 \leftarrow x(0)$, $q_0 \leftarrow q(0)$.
2: $v_{prev} \leftarrow r(0)$.
3: **while** `true` **do**
4: $r_0 \leftarrow r(k)$.
5: $q \leftarrow$ `SwitchingLogic`($x_0, q_0, S_\infty^{P_1}, S_\infty^{P_2}, r_0, c_1, c_2, v_{prev}$)
6: Compute $\mathcal{V}_q(x_0) \leftarrow \{v \in \mathbb{R}^m : (v, x_0) \in S_\infty^{P_q}\}$.
7: Solve Problem 2 for r_0, q and x_0 to obtain the virtual reference v.
8: Apply v to P_q to generate the trajectory x.
9: $k \leftarrow k + 1$.
10: $x_0 \leftarrow x(k)$, $q_0 \leftarrow q$, $v_{prev} \leftarrow v$.
11: **end while**

B. A Switched RG Implementation

The proposed switched RG is summarized in Algorithm 2. The discrete time, state, and mode are initialized in Line 1. For each iteration, the current reference is updated in Line 4. The current mode is then updated by a call to the

SwitchingLogic function in Line 5. Using this updated mode, the virtual reference is computed by solving Problem 2 in Lines 6-7. The trajectory $k \mapsto x(k)$ is obtained by applying the computed virtual reference v to the selected closed-loop system P_q in Line 8. Data updates occur at the end of each iteration, setting the stage for the algorithm to progress to the next iteration.

V. Analysis of Theoretical Properties

This section introduces the hybrid system modeling of the proposed switched RG, following the theoretical guarantees.

A. Discrete-time Hybrid System Model

The given reference $r \in \mathcal{R}$ acts as a constant parameter in this model. In practice, if the given reference is piecewise constant and RG detects that the value of the reference changes, then the switched RG is restarted. By adding a logic variable q, we obtain a hybrid system model in discrete time, which we formulate next.

For $q \in \{1,2\}$, the RG function, defined as $\kappa_q : \mathbb{R}^n \times \mathcal{R} \to \mathbb{R}^m$, is such that given $r \in \mathcal{R}$ and $x \in \mathbb{R}^n$, $\kappa_q(x,r)$ equals the solution to Problem 2 for r and $\mathcal{V}_q(x)$. We define the extended state $\chi := (x, q, v_{prev}) \in \mathbb{R}^n \times \{1,2\} \times \mathbb{R}^m$. Given a constant $r \in \mathcal{R}$, the switched RG system is modeled as

$$\mathcal{H}_s : \begin{cases} \chi^+ = \begin{bmatrix} G_q(x, \kappa_q(x,r)) \\ q \\ \kappa_q(x,r) \end{bmatrix} =: f_s(\chi, r) & (\chi, r) \in C_s \\ \chi^+ = \begin{bmatrix} x \\ 3-q \\ v_{prev} \end{bmatrix} =: g_s(\chi, r) & (\chi, r) \in D_s \end{cases} \tag{8}$$

where V is defined in Definition 4.2 and exists due to the discussion in Remark 4.3, $D_s := D_1 \cup D_2$ and $C_s := (\mathbb{R}^n \times \{1,2\} \times \mathbb{R}^m \times \mathbb{R}^m) \backslash D_s$ with $D_1 := D^{(1)}_{1\to 2} \cup D^{(2)}_{1\to 2}$, $D_2 := D^{(1)}_{2\to 1} \cup D^{(2)}_{2\to 1}$, and

$$D^{(1)}_{1\to 2} := \{(x, q, v_{prev}, r) \in \mathbb{R}^n \times \{1,2\} \times \mathbb{R}^m \times \mathcal{R} : \mathcal{V}_i(x) \neq \emptyset \ \forall i \in \{1,2\}, V(x,r) \le c_2 \ q = 1, \min_{v \in \mathcal{V}_2(x)} |v - r| \le |v_{prev} - r|\},$$

$$D^{(2)}_{1\to 2} := \{(x, q, v_{prev}, r) \in \mathbb{R}^n \times \{1,2\} \times \mathbb{R}^m \times \mathcal{R} : \mathcal{V}_1(x) = \emptyset, \mathcal{V}_2(x) \neq \emptyset\},$$

$$D^{(1)}_{2\to 1} := \{(x, q, v_{prev}, r) \in \mathbb{R}^n \times \{1,2\} \times \mathbb{R}^m \times \mathcal{R} : \mathcal{V}_i(x) \neq \emptyset \ \forall i \in \{1,2\}, V(x,r) \ge c_1 \ q = 2, \min_{v \in \mathcal{V}_1(x)} |v - r| \le |v_{prev} - r|\},$$

$$D^{(2)}_{2\to 1} := \{(x, q, v_{prev}, r) \in \mathbb{R}^n \times \{1,2\} \times \mathbb{R}^m \times \mathcal{R} : \mathcal{V}_2(x) = \emptyset, \mathcal{V}_1(x) \neq \emptyset\}.$$

Given the initial state $x_0 \in \mathbb{R}^n$ and initial mode $q_0 \in \{1,2\}$, the initial extended state is set as $\chi_0 = (x_0, q_0, r)$. We denote the x, q, and v_{prev} component of a solution ϕ to (8) as ϕ_x, ϕ_q, and $\phi_{v_{prev}}$, respectively.

Remark 5.1: Note that $C_s \cup D_s = \mathbb{R}^n \times \{1,2\} \times \mathbb{R}^m \times \mathcal{R}$. Then, for any $\chi \in \mathbb{R}^n \times \{1,2\} \times \mathbb{R}^m$ and $r \in \mathcal{R}$, we have $(f_s(\chi, r), r) \in C_s \cup D_s$ and $(g_s(\chi, r), r) \in C_s \cup D_s$. Therefore, for any initial $\chi_0 \in \mathbb{R}^n \times \{1,2\} \times \mathbb{R}^m$, a nontrivial solution to (8) starting from χ_0 is guaranteed to exist, and every maximal solution to (8) is complete if $k \mapsto v(k)$ is complete.

Solutions to $\mathcal{H}_s$ are parameterized by the pair $(k,j) \in \mathbb{N} \times \mathbb{N}$. From [14, Definition 2.1], k represents the number of discrete time instances elapsed according to $\chi^+ = f_s(\chi, r)$ and j represents the number of jumps at which the state is updated according to $\chi^+ = g_s(\chi, r)$.

B. Recursive Feasibility

Recursive feasibility guarantees that if there exists a feasible control at the first time instance, then a feasible control will be found at all the following time instances. The following assumption posits that a feasible control solution can be derived given the initial condition.

Assumption 5.2: Given $r \in \mathcal{R}$, the initial state $x_0 \in \mathbb{R}^n$ and initial mode $q_0 \in \{1,2\}$ are such that $\mathcal{V}_{q_0}(x_0)$ is nonempty.

Next, Theorem 5.3 ensures a feasible solution to Problem 2 at each time.

Theorem 5.3: (Recursive feasibility) Given a reference $r \in \mathcal{R}$ and $(x_0, q_0) \in \mathbb{R}^n \times \{1,2\}$ satisfying Assumption 5.2, the infinite horizon admissible reference set $\mathcal{V}_{\phi_q(k,j)}(\phi_x(k,j))$ is nonempty for all $(k,j) \in \operatorname{dom}\phi$, where ϕ denotes the maximal solution to (8) starting from (x_0, q_0, r). Hence, there exists a feasible solution to Problem 2 for r, $\phi_q(k,j)$, and $\phi_x(k,j)$.

C. Finite-time Reachability

Next, we introduce a result demonstrating that the system is assured to enter the neighborhood of reference r within finite time, guaranteeing the activation of γ_2. The following assumption asserts that c_2 is large enough such that the sublevel set $\mathcal{T}_{1,0}(r)$ contains the equilibria set for P_1.

Assumption 5.4: For any $r \in \mathcal{R}$, the algorithm parameter c_2 is such that there exists a positive constant δ_0^c such that $\mathcal{A}^{P_1}(r) + \delta_0^c \mathbb{B} \subset \mathcal{T}_{1,0}(r)$.

Then, we show that when P_1 is active, the value of the Lyapunov function for P_2 decreases to c_2 within finite time.

Theorem 5.5: (Finite-time reachability to $\mathcal{T}_{1,0}$) Suppose Assumptions 3.1, 3.3 and 5.4 are satisfied. Given a reference $r \in \mathcal{R}$ and $(x_0, q_0) \in \mathbb{R}^n \times \{1,2\}$ satisfying Assumption 5.2, there exists a discrete time instance $k^* \in \mathbb{N}$ such that $\phi_x(k^*, j^*) \in \mathcal{T}_{1,0}(r)$, where ϕ denotes the maximal solution to (8) starting from (x_0, q_0, r), $j^* \in \{0,1\}$ such that $(k^*, j^*) \in \operatorname{dom}\phi$, and $\mathcal{T}_{1,0}(r)$ is defined in (7).

D. Forward Invariance

Next, we present the result showing that if the system starts within $\mathcal{T}_{1,0}(r)$, then γ_2 helps maintain the system within $\mathcal{T}_{1,0}(r)$ forever, preventing any future jumps.

Theorem 5.6: (Forward invariance) Suppose Assumptions 3.1, 3.3 and 5.4 are satisfied. Given a reference $r \in \mathcal{R}$

and $(x_0, q_0) \in \mathbb{R}^n \times \{1, 2\}$ *satisfying Assumption 5.2 and* $x_0 \in \mathcal{T}_{1,0}(r)$*, where* $\mathcal{T}_{1,0}(r)$ *is defined in (7), then* $\phi_x(k, j) \in \mathcal{T}_{1,0}(r)$ *for all* $(k, j) \in \operatorname{dom} \phi$*, where* ϕ *denotes the maximal solution to (8) starting from* (x_0, q_0, r)*.*

E. Finite Number of Jumps

The following result states that a maximum of two jumps can occur during the control process, preventing chattering phenomenon by design.

Theorem 5.7: (Finite number of jumps) Suppose Assumptions 3.1, 3.3 and 5.4 are satisfied. Given a reference $r \in \mathcal{R}$ *and* $(x_0, q_0) \in \mathbb{R}^n \times \{1, 2\}$ *satisfying Assumption 5.2, the maximal solution to (8) starting from* (x_0, q_0, r)*, denoted* ϕ*, satisfies* $J \leq 2$ *where* $(K, J) = \max \operatorname{dom} \phi$*.*

F. Convergence of Virtual Reference to Reference

The following theorem guarantees that the virtual reference v converges to r. Given that z converges to v, it ensures that z converges to r.

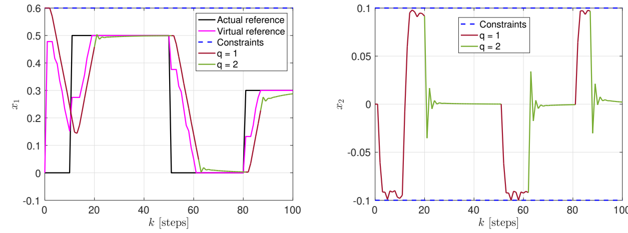
Theorem 5.8: (Convergence of v *to* r*) Suppose Assumptions 3.1, 3.3 and 5.4 are satisfied. Given a reference* $r \in \mathcal{R}$ *and* $(x_0, q_0) \in \mathbb{R}^n \times \{1, 2\}$ *satisfying Assumption 5.2, then* $\lim_{k+j \to \infty} \kappa_{\phi_q(k,j)}(\phi_x(k, j), r) = r$*, where* ϕ *denotes the maximal solution to (8) starting from* (x_0, q_0, r)*.*

VI. Simulation Results

In this simulation, the switched RG algorithm is used to track the piecewise-constant reference signal for the double-integrator system in (1). The Lyapunov function used in the simulation is $V(x, v) = \left(x - \begin{bmatrix} v \\ 0 \end{bmatrix}\right)^\top \begin{bmatrix} 6.8283 & 2.2626 \\ 2.2626 & 2.1010 \end{bmatrix} \left(x - \begin{bmatrix} v \\ 0 \end{bmatrix}\right)$. The algorithm parameters c_1 is set as 0.2 and c_2 is set as 0.1. The maximal output admissible sets $S_\infty^{P_1}$ and $S_\infty^{P_2}$ are computed using the MPT3 toolbox [15]. The constraint imposed on x_1 is $x_1 \in [-0.6, 0.6]$. The constraint imposed on x_2 is $x_2 \in [-0.1, 0.1]$. The system initiates at the point $(0.6, 0)$, posing a challenge given that the initial state nearly breaches the constraints. Compared to using a single RG, the switched RG achieves rapid and non-oscillatory convergence to the reference signal, as depicted in Figures 3a and 3b. Feasible control is consistently available, confirming recursive feasibility (Theorem 5.3). The Lyapunov function's value always decreases to c_2 within finite time (Theorem 5.5), even when initialized far from the reference. The switched RG switches between γ_1 and γ_2 as expected, ensuring a finite number of switches (Theorem 5.7). Furthermore, the virtual reference v converges to r (Theorem 5.8), and robust switching is observed without chattering.

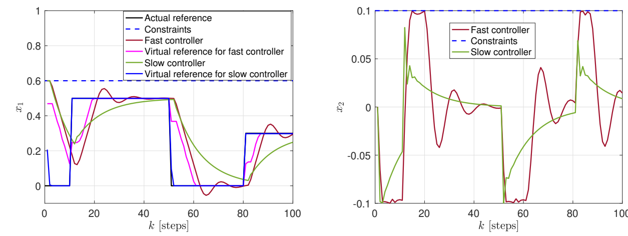
VII. Conclusion

This paper proposes a switched RG approach for rapid, non-oscillatory convergence to a given reference signal while satisfying constraints. We demonstrate robust switching, recursive feasibility, and convergence of the virtual reference to the reference alongside other key properties.



(a) The state trajectory of x_1. (b) The state trajectory of x_2.

Fig. 2: The system state trajectory using switched RG.



(a) The state trajectory of x_1. (b) The state trajectory of x_2.

Fig. 3: The system state trajectory using a single controller.

References

[1] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control: theory, computation, and design*, vol. 2. Nob Hill Pub. Madison, WI, 2017.

[2] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Trans. Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2016.

[3] R. de Castro and S. Feng, "Synthesis of recursive control barrier functions," *Journal of Dynamic Systems, Measurement, and Control*, vol. 145, no. 3, p. 031001, 2023.

[4] S. Feng, R. de Castro, and I. Ebrahimi, "Fast charging of batteries using cascade-control-barrier functions," in *2023 American Control Conference (ACC)*, pp. 2481–2486, IEEE, 2023.

[5] A. Bemporad, "Reference governor for constrained nonlinear systems," *IEEE Trans. Automatic Control*, vol. 43, no. 3, pp. 415–419, 1998.

[6] E. Garone, S. Di Cairano, and I. Kolmanovsky, "Reference and command governors for systems with constraints: A survey theory and applications," *Automatica*, vol. 75, pp. 306–328, 2017.

[7] D. Famularo, G. Franze, A. Furfaro, and M. Mattei, "A hybrid real-time command governor supervisory scheme for constrained control systems," *IEEE Trans. Control Sys. Tech.*, vol. 23, no. 3.

[8] I. Kolmanovsky and J. Sun, "A multi-mode switching-based command tracking in network controlled systems with pointwise-in-time constraints and disturbance inputs," in *2006 6th World Congress Intelligent Control and Automation*, vol. 1, pp. 199–204, IEEE, 2006.

[9] F. Tedesco and A. Casavola, "Load/frequency control in the presence of renewable energy systems: a reference-offset governor approach," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 12548–12553, 2020.

[10] Y. Liu, X. Liu, B. Helian, Z. Chen, and B. Yao, "Hybrid reference governor-based adaptive robust control of a linear motor driven system," in *2021 IEEE Int. Conf. Mechatronics*, pp. 1–7, 2021.

[11] A. Sassella, V. Breschi, S. Formentin, and S. M. Savaresi, "A data-driven switching control approach for braking systems with constraints," *Nonlinear Analysis: Hybrid Systems*, vol. 46, 2022.

[12] V. Breschi and S. Formentin, "Direct data-driven design of switching controllers," *Int. Journal of Robust and Nonlinear Control*, vol. 30, no. 15, pp. 6042–6072, 2020.

[13] H. K. Khalil, *Nonlinear Control*. Pearson, 2015.

[14] P. Ojaghi, B. Altın, and R. G. Sanfelice, "A model predictive control framework for asymptotic stabilization of discretized hybrid dynamical systems," in *58th IEEE Conf. Decision and Control*, 2019.

[15] M. Herceg, M. Kvasnica, C. N. Jones, and M. Morari, "Multi-parametric toolbox 3.0," in *Europ. Control Conf.*, pp. 502–510, 2013.