Finite Data-Rate Feedback Stabilization of Continuous-Time Switched Linear Systems with Unknown Switching Signal

Guillaume O. Berger and Raphaël M. Jungers

Abstract— In this paper, we study the problem of stabilizing switched linear systems when only limited information about the state and the mode of the system is available, which occurs in many applications involving networked switched systems (such as cyber-physical systems, IoT, etc.). First, we show that switched linear systems with arbitrary switching, i.e., with no constraint on the switching signal, are in general not stabilizable with a finite data rate. Then, drawing on this result, we restrict our attention to systems satisfying a fairly mild slow-switching assumption, in the sense that the switching signal has an average dwell time bounded away from zero. We show that under this assumption, switched linear systems that are stabilizable in the classical sense remain stabilizable with a finite data rate. A practical coder–controller that stabilizes the system is presented and its applicability is demonstrated on numerical examples.

I. INTRODUCTION

This paper studies two important and challenging features of modern control systems: data-rate constraints and switching. Many modern control systems (such as IoT, networked systems, etc.) involve spatially distributed components that communicate through a shared, digital communication network, that can carry only a finite amount of information per unit of time. This limitation on the information flow can have large negative effects on the performance of the control loop. This has motivated a considerable amount of research to study control problems subject to data-rate constraints, as surveyed in [5], [11], [21]. On the other hand, many systems encountered in practice involve switching between different operation modes; e.g., due to the interaction of physical processes and digital devices (as in cyber-physical systems), external influences (e.g., human in the loop), discontinuous dynamics (e.g., physical processes with impact), the nature of the controller (e.g., logic-dynamic controllers), etc. Control problems involving switching have also attracted a lot of attention from the control community in recent years; see, e.g., the surveys [7], [13], [16], and the references therein.

This paper focuses on the problem of limited data-rate stabilization of continuous-time Switched Linear Systems (SLSs). These are systems described by a finite set of linear modes, among which the system can switch in time (see Figure 1 for a representation). As paradigmatic examples of hybrid and cyber-physical systems, SLSs naturally appear in many engineering applications, or as abstractions of more complex dynamical systems [6], [9], [14].
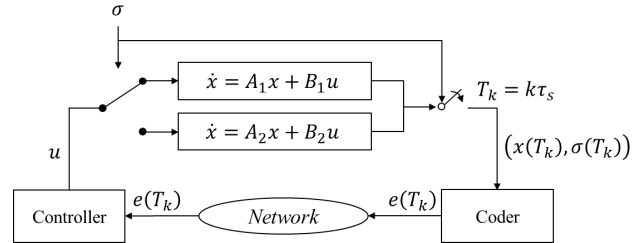
Fig. 1. Control of switched linear systems over limited, digital communication networks.

Although control of switched systems and control with data-rate constraints have been two active areas of research for some time now, the study of control problems involving switching and data-rate constraints simultaneously seems to have not received much attention so far. (Some work has been devoted to the stabilization of Markov jump linear systems with data-rate constraints [10], [12], [18], [20]. However, the information structure considered in these references implies that the mode of the system is always known to the controller, so that the problems of switching and state estimation with limited information can be treated separately.) Combining these two aspects in a unified framework is however essential if we want to address control problems encountered in a wide range of applications involving networked switched systems, which generally imply that the controller has limited information on both the state *and* the mode of the system.

When both state observation and switching signal observation are subject to data-rate constraints, the problems of state estimation and switching are intrinsically coupled. For instance, state encoding strategies must take into account the fact that unobserved switching may occur during the sampling interval. In particular, we will see that switched systems with unconstrained switching signal have in general an *infinite stabilization entropy*, meaning that they are not stabilizable with any finite data rate. On the other hand, it is a standard technique in stability and stabilizability analysis of switched systems to impose *slow-switching conditions*—generally described by a dwell time and/or an average dwell time (ADT) [4], [7]—on the switching signal to reduce its expressiveness. Recently, these techniques were used in the context of limited data-rate control of SLSs [8], [17], [19].

Our work draws upon these references, especially [8], for the formulation of the problem of interest, namely the stabilization of SLSs under data-rate constraints and subject to slow-switching assumptions. However, we consider different objectives regarding the design of a control strategy with finite data rate: while in [8], [17], [19], the ADT is used as

a design parameter to ensure stabilization of the system with data-rate constraints, our goal here is to study the limited data-rate stabilization of SLSs with *arbitrary* ADT.

The contribution of this paper is twofold. First, we show that SLSs with *no constraint* on the switching signal, are in general not stabilizable with a finite data rate. We present an example of a SLS that is feedback stabilizable for any switching signal in the absence of data-rate constraints, but cannot be stabilized with a finite data rate. This motivates the introduction of slow-switching assumptions in order to make the problem of limited data-rate stabilization of SLSs tractable, as otherwise the coder cannot transmit information fast enough to the controller to achieve stabilization.

Secondly, we show that under a fairly mild slow-switching assumption on the switching signal, SLSs that are stabilizable in the classical sense remain stabilizable with a finite data rate. More precisely, we show that any stabilizable (without data-rate constraints) SLS with ADT bounded away from zero can be stabilized by a coder–controller with finite data rate. We present a sufficient upper bound on the data rate depending on the system and the ADT, and we describe the implementation of a coder–controller that stabilizes the system. We stress out that in our analysis (unlike [8], [17], [19]) the ADT is fixed, so that the controller has no influence on the value of the ADT.

The paper is organized as follows. The problem of interest, including the definition of SLSs, the basic assumptions on the system and the concept of coder–controller, is formulated in Section II. Our main results are stated in Section III. Then, in Section IV, we describe the implementation of a coder–controller that stabilizes the system. Finally, in Section V, we illustrate the usage of the coder–controller with a numerical example.

Notation. For vectors, $\|\cdot\|$ denotes the Euclidean 2-norm, and for matrices it denotes the associated matrix norm (i.e., $\|M\| = $ largest singular value of M). $B(\xi, r)$ is the closed ball centered at $\xi \in \mathbb{R}^d$ with radius $r \geq 0$. If $f : A \to B$, and $A' \subseteq A$, then $f|_{A'}$ denotes the restriction of f to the domain A'. A function $g : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is of class-$\mathcal{K}$ if it is strictly increasing, continuous and $g(0) = 0$.

II. PROBLEM FORMULATION

A. Switched linear systems

Consider a continuous-time *Switched Linear System* (SLS) with affine control input:

$$\dot{x}(t) = A_{\sigma(t)}x(t) + B_{\sigma(t)}u(t), \quad x(0) \in K, \quad t \geq 0, \quad (1)$$

where $\sigma(t) \in \Sigma := \{1, \ldots, N\}$ and $u(t) \in \mathbb{R}^c$, $A_i \in \mathbb{R}^{d \times d}$ and $B_i \in \mathbb{R}^{d \times c}$ for all $i \in \Sigma$, and $K \subseteq \mathbb{R}^d$ is a compact set with $0 \in \text{int}(K)$. The function $\sigma : \mathbb{R}_{\geq 0} \to \Sigma$ is called the *switching signal* (or *s.s.* for short) and is assumed to be piecewise constant and right-continuous.

The discontinuity points of σ are called the "switching times" or simply "switches". For $t \geq s \geq 0$, we let $N_\sigma(t, s)$ be the number of switches of σ in the interval $[s, t)$.

As we will see in Subsection III-B, SLSs under arbitrary switching are in general not tractable for the problem of

stabilization with limited data rate. Therefore, in our analysis, we make the assumption that the system is not switching too fast, in the following sense:

Assumption 1: There is $\tau_a > 0$, called the *Average Dwell Time* (ADT), and a constant $N_0 \geq 0$ such that

$$N_\sigma(t, s) \leq N_0 + \frac{t - s}{\tau_a} \qquad \forall t \geq s \geq 0.$$

The parameter N_0 is fixed but not known by the controller a priori.

The concept of ADT, introduced in [4], has become a standard assumption in the study of stability and stabilizability of switched and hybrid systems [7], [14]. It has also received attention in the context of control of switched systems with limited information [8], [17], [19], [20].

Our goal is to stabilize (1) under data-rate constraints. Clearly, a necessary condition is that the system is stabilizable in the absence of data-rate constraints. Hence, we make the following assumption on the stabilizability of (1):

Assumption 2: There is a *feedback law* $\varphi : \mathbb{R}^d \times \Sigma \to \mathbb{R}^c$, positively homogeneous in the 1st argument and piecewise continuous, and constants $D \geq 0$ and $\mu_1, \mu_2 > 0$ such that (i) $\mu_1/\tau_a < \mu_2$, and (ii) for every s.s. σ, the solution of the closed-loop system $\dot{x}(t) = A_{\sigma(t)}x(t) + B_{\sigma(t)}\varphi(x(t), \sigma(t))$ satisfies

$$\|x(t)\| \leq D\|x(0)\| e^{\mu_1 N_\sigma(t,0) - \mu_2 t} \qquad \forall t \geq 0. \quad (2)$$

We assume that the feedback law $\varphi(\cdot, \cdot)$ and the parameters D, μ_1, μ_2 are fixed and known by the controller.

Assumption 2 implies that the closed-loop system (1) with feedback control input $u(t) = \varphi(x(t), \sigma(t))$ is asymptotically stable, for every s.s. satisfying Assumption 1. The existence of a feedback control law satisfying (2) can be ensured for instance if the system admits a *multiple Control Lyapunov Function* (CLF) [7], [9]. An interesting situation, on which we will come back in Subsection III-C, is when the system admits a *common* CLF; in this case, (2) is satisfiable with $\mu_1 = 0$, so that Assumption 2 holds for any $\tau_a > 0$.

B. Coder–controller

We investigate the problem of stabilizing system (1) when direct observation of the system is not possible. Information about the mode and the state of the system will thus be delivered by a *coder* connected to a *controller* via a digital channel that can carry only a finite amount of information per unit of time. The situation is depicted in Figure 1. At periodic sampling times $T_k := k\tau_s$, $k = 0, 1, 2, 3, \ldots$, the coder observes the current state and mode of the system, and sends one discrete-valued symbol $e(T_k)$, selected from a finite coding alphabet $\mathcal{E}_k$, to the controller. Neglecting transmission errors and delay, at time T_k the controller has the symbols $e(0), \ldots, e(k)$ available and it generates a control input $u(\cdot)$ for the coming epoch $[T_k, T_{k+1})$.

More precisely, let $\tau_s > 0$ be a sampling period and $(\mathcal{E}_k)_{k \in \mathbb{N}}$ a sequence of coding alphabets. The symbol sent by the coder at time $T_k := k\tau_s$ is defined by

$$e(T_k) = \gamma_k(x(T_0), \ldots, x(T_k); \sigma(T_0), \ldots, \sigma(T_k)), \quad (3)$$

where $\gamma_k : (\mathbb{R}^d)^k \times \Sigma^k \to \mathcal{E}_k$ is the coder function at time T_k and $x(\cdot)$ is the state of the system. Assuming the channel is noiseless and without delay, at time T_k the controller has the symbols $e(T_0), \ldots, e(T_k)$ available and it generates the input function $u(\cdot)$ defined on the interval $[T_k, T_{k+1})$ by:

$$u(\cdot)|_{[T_k, T_{k+1})} = \zeta_k(e(T_0), \ldots, e(T_k)), \qquad (4)$$

where $\zeta_k : \mathcal{E}_0 \times \ldots \times \mathcal{E}_k \to (\mathbb{R}^c)^{[T_k, T_{k+1})}$ is the controller function at time T_k, and it is assumed that $u(\cdot)|_{[T_k, T_{k+1})}$ is integrable (i.e., L^1). Let $\gamma = (\gamma_k)_{k \in \mathbb{N}}$ and $\zeta = (\zeta_k)_{k \in \mathbb{N}}$. The pair (γ, ζ) is called a *coder–controller*.

At each time T_k, the symbol $e(T_k)$ is transmitted via a limited communication channel. Using binary representation of the symbols, the *averaged communication data rate* (or *data rate* for short) [in bits per unit of time] of the coder–controller is given by

$$R(\gamma, \zeta) = \limsup_{k \to \infty} \frac{1}{k\tau_s} \sum_{j=0}^{k-1} \log_2 |\mathcal{E}_j|.$$

In modern applications involving networked systems, the communication capacity of the network is generally important. However, there are usually many resources competing for the same bandwidth (as for instance in IoT applications). Therefore, it is essential to have coding–controlling strategies with an averaged communication data rate as small as possible. The question of determining the smallest data rate that is needed to achieve a given control task is also intriguing from the theoretical point of view.

III. MAIN RESULTS

A. Finite data-rate stabilization

The control objective studied in this paper is the stabilization of system (1), defined in Subsection II-A, under data-rate constraints. More precisely, we want to show the existence of a coder–controller, as described in Subsection II-B, with finite data rate, that stabilizes the system:

Theorem 1: Consider system (1) and let Assumptions 1–2 hold. There is a coder–controller (γ, ζ) with $R(\gamma, \zeta) < \infty$ that satisfies the following properties:[1] there is $\lambda > 0$ and a class-$\mathcal{K}$ function $g(\cdot)$ such that every trajectory $x(\cdot)$ of (1) with control input $u(\cdot)$ defined by (3)–(4) satisfies

$$\|x(t)\| \le g(\|x(0)\|) e^{-\lambda t} \qquad \forall t \ge 0.$$

We will provide a constructive proof of Theorem 1. More precisely, in Section IV, we describe the implementation of a finite data-rate coder–controller achieving stabilization. A precise upper bound on the data rate will be derived in due course of the description of the coder–controller; see (7) in Section IV. As for the decay rate λ, it will be obtained in the proof that the proposed coder–controller stabilizes the system; see Subsection IV-D in the extended version of this work [1]. The function $g(\cdot)$, however, will not be explicitly characterized but its existence will be demonstrated. As a class-$\mathcal{K}$ function, $g(\cdot)$ satisfies $g(r) \to 0$ when $r \to 0$.

---

[1] A coder–controller that satisfies these properties will be said to *stabilize* the system.

However, as it will be clear from the proof of its existence, $g(r)$ is not Lipschitz continuous at $r = 0$. This lack of regularity is not due to potential sub-optimality of the proposed coder–controller, or to the switching nature of the system, but is intrinsic to *any* finite-data-rate stabilization scheme for linear systems (including LTI systems); see for instance [3, Proposition 2.2].

B. Necessity of the average dwell time

In this subsection, we would like to stress out the importance of Assumption 1 in Theorem 1. Therefore, we show with a simple example that SLSs under *arbitrary* switching (i.e., with ADT equal to zero) are in general not stabilizable with a finite data rate:

Example 1: Consider system (1) with $d = 1$, $\Sigma = \{1, 2\}$, and matrices $A_1 = A_2 = 0$, $B_1 = -1$ and $B_2 = 1$. This system is somehow the most basic affine-controlled SLS, and it clearly satisfies Assumption 2 with $\mu_1 = 0$: for instance, take $\varphi(x(t), \sigma(t)) = -B_{\sigma(t)} x(t)$.

Proposition 2: System (1), with A_Σ and B_Σ as in Example 1 and under arbitrary switching, is not stabilizable by a coder–controller with finite data rate.

Proof: Assume the contrary and let (γ, ζ) be a coder–controller that stabilizes the system. For some $T > 0$ fixed, let $\mathcal{U}_T$ be the set of all distinct input functions $u(\cdot)$ that can be generated by the coder–controller, i.e., by (3)–(4), on the interval $[0, T)$. Since $R(\gamma, \zeta) < \infty$, it follows that $\mathcal{U}_T$ is finite, and by (4), $\mathcal{U}_T$ contains only L^1 functions.

Now, for each $n \in \mathbb{N}_{>0}$, let $\sigma_n(\cdot)$ be the s.s. that oscillates between mode 1 and mode 2 with frequency $2/n$: that is, $\sigma_n(t) = 1$ if $t \in [0, 1/n) + 2\mathbb{N}/n$, and $\sigma_n(t) = 2$ if $t \in [1/n, 2/n) + 2\mathbb{N}/n$. Then, by using an adaptation of the proof of the Riemann–Lebesgue lemma [15, Corollary 14.5], one can show that for any L^1 function $f : [0, T) \to \mathbb{R}$, it holds that $\int_0^T B_{\sigma_n(t)} f(t) \, \mathrm{d}t \to 0$ when $n \to \infty$. Since $\mathcal{U}_T$ is finite, this implies that for any $\varepsilon > 0$, there is $n \in \mathbb{N}_{>0}$ such that $|\int_0^T B_{\sigma_n(t)} u(t) \, \mathrm{d}t| < \varepsilon$ for all $u \in \mathcal{U}_T$.

Thus, for every $u \in \mathcal{U}_T$, $|x(T)| > |x(0)| - \varepsilon$ where $x(\cdot)$ is a trajectory of (1) with s.s. σ_n and with input u. Since T and ε are arbitrary and K contains at least one point $x(0) \ne 0$ (since it has nonempty interior), this is a contradiction with the hypothesis that (γ, ζ) stabilizes the system. ∎

Summarizing, the above example shows that for the problem of limited data-rate stabilization of SLSs to be tractable, the switching signal cannot switch too rapidly (or at least not too rapidly during a too long period), as otherwise the uncertainty on the mode of the system, and especially on B_σ, between two transmission times (T_k, T_{k+1}) will be so large that the the system cannot be stabilized with a finite set of inputs. This motivates the introduction of Assumption 1 which reduces the set of admissible switching signals by restricting the number of switches in bounded intervals.

C. Comparison with other works

Our work is strongly connected with [8], [17], [19], where the problem of limited data-rate stabilization of SLSs with constraints on the ADT is also considered. The objectives

regarding the design of a control strategy with finite data rate are however different. In [8], [17], [19], the sampling frequency and the data rate of the coder–controller are fixed, and the objective is to obtain a sufficient lower bound on the ADT of the switching signal to ensure stabilization of the system. In our work, however, we seek to obtain a coder–controller, with suitable sampling frequency and data rate, that stabilizes the system for any ADT such that Assumption 2 holds. In particular, if Assumption 2 holds for every $\tau_a > 0$, then for any given value of the ADT, we describe a coder–controller that stabilizes the system. By contrast, in [8], [17], [19], even if the system admits a common CLF (which implies that Assumption 2 holds for every $\tau_a > 0$; see Subsection II-A), the lower bound on the ADT does not converge to zero, even if the sampling frequency and the data rate tend to infinity. The conservativeness of the lower bound on the ADT proposed in [8], [17], [19] is mainly due to the fact that its derivation is based on the decrease of a multiple Lyapunov-like function for the sampled system. In particular, the definition of this multiple Lyapunov-like function implies that each change of mode causes a nonzero "jump" in the value of the function; this results in nonzero lower bounds on the ADT, even if the system admits a common quadratic CLF; see, e.g., [8, Eq. (38)]. By contrast, our approach relies on the convergence of a reference trajectory (by using Assumption 2 and a suitable sampling frequency) and the guarantee that the true state of the system does not stray too far from the reference trajectory (thanks to a suitable data rate); see Section IV below. Last but not least, another difference of our framework with [8], [17], [19] is that we do not impose any condition on the *absolute* dwell time of the system.

IV. Description of the Coder–Controller

In this section, we describe the implementation of a coder–controller that stabilizes system (1) under Assumptions 1–2. The section is organized as follows. First, we discuss the selection of the parameters of the coder–controller, which depend on the system and the quantities appearing in Assumptions 1–2. Then, we present the implementations of the coder and the controller. Finally, we show that the proposed coder–controller satisfies the assertions of Theorem 1.

A. Parameters definition

Let $\nu = \frac{1}{2} \max_{i \in \Sigma} \lambda_{\max}(A_i + A_i^\top)$, and let
$$\Delta_1 = \max_{i,j \in \Sigma} \|A_i - A_j\|, \quad \Delta_2 = \max_{i,j \in \Sigma} \|B_i - B_j\|.$$
Also, define
$$L = \max \{ \|\varphi(\xi, i)\| : i \in \Sigma, \ \xi \in \mathbb{R}^d, \ \|\xi\| = 1 \}.$$
Pick $\tau_s > 0$, $\alpha > 0$ and $n \in \mathbb{N}$ such that
$$De^{-\mu_2 n \tau_s} + e^{\nu n \tau_s} \alpha + \varepsilon(n, \tau_s) < e^{-\mu_1 n \tau_s / \tau_a} \qquad (5)$$
where $\varepsilon(n, \tau_s) = e^{\nu n \tau_s} \tau_s \frac{n \tau_s}{\tau_a} D(\Delta_1 + \Delta_2 L).$ [2]

[2] A strategy for choosing τ_s, α, n could be: first, choose $T_s = n \tau_s$ large enough so that $De^{(\mu_1/\tau_a - \mu_2) T_s} < 1$. Then, for this T_s, choose α, n such that $e^{\nu T_s} \alpha$ and $\varepsilon(n, T_s/n)$ are small enough for (5) to be satisfied.

We will need the following lemma:

Lemma 3: Let $\alpha > 0$. There is an m-point quantizer $Q : \mathbb{R}^d \to \mathcal{Q} \subseteq B(0, 1)$ satisfying (i) $\|\xi - Q(\xi)\| \leq \alpha$ if $\|\xi\| \leq 1$, (ii) $Q(\xi) = 0$ if $\|\xi\| \leq \alpha/d^{1/2}$, and (iii)
$$m = |\mathcal{Q}| \leq \hat{m}_\alpha := \left(2 \left[\!\left[\frac{d^{1/2}}{2\alpha} \right]\!\right] + 1 \right)^d. \qquad (6)$$
where $[\![\cdot]\!]$ is the *rounding* (to the nearest integer) operator.

Proof: See the extended version of this paper [1]. ∎

We will show that there is a coder–controller (γ, ζ), with sampling period τ_s, that stabilizes the system and operates at data rate
$$R(\gamma, \zeta) = \frac{1}{\tau_s} \left[\frac{1}{n} \log_2 \hat{m}_\alpha + \frac{1}{n} \log_2(n+1) + \log_2 |\Sigma| \right] \qquad (7)$$
where $\hat{m}_\alpha$ is as in (6).

B. Coder implementation

Let τ_s, α and n be as above, and let $Q(\cdot)$ be the quantizer associated to α as in Lemma 3. Also fix $r_0 > 0$ such that $K \subseteq B(0, r_0)$.[3] The implementation of the coder is described in Figure 2.

The implementation deserves the following explanations. At every time $kn\tau_s$, $k \in \mathbb{N}_{>0}$, the coder computes the value of N_k^{sw} which is defined as the smallest integer in $\{0, \dots, n\}$ such that β_k, defined by
$$\beta_k = e^{\mu_1 N_k^{\mathrm{sw}}} \psi + \bar{\alpha} + e^{\mu_1 N_k^{\mathrm{sw}}} N_k^{\mathrm{sw}} \bar{\varepsilon} \qquad (8)$$
where $\psi = De^{-\mu_2 n \tau_s}$, $\bar{\alpha} = e^{\nu n \tau_s} \alpha$, $\bar{\varepsilon} = e^{\nu n \tau_s} \tau_s D(\Delta_1 + \Delta_2 L)$, satisfies $\|x(kn\tau_s)\| \leq \beta_k r_{k-1}$. We will see in Subsection IV-D that such an N_k^{sw} always exists. Using this β_k, the coder updates the value of r_k according to $r_k = \beta_k r_{k-1}$. If $k = 0$, simply use $N_0^{\mathrm{sw}} = 0$ and r_0.

Using the above quantities, at time $kn\tau_s$, $k \in \mathbb{N}$, the coder sends a symbol that encodes the following information: (i) an approximation η_k of the current state $x(kn\tau_s)$ scaled by $1/r_k$, using the quantizer $Q(\cdot)$, (ii) the current mode of the system, $\sigma(kn\tau_s)$, and (iii) the value of N_k^{sw}. Since η_k can take at most $\hat{m}_\alpha$ different values, $\sigma(kn\tau_s)$ at most $|\Sigma|$ values, and N_k^{sw} at most $n+1$ different values, it holds that a coding alphabet $\mathcal{E}_{kn}$ of size $\log_2 \hat{m}_\alpha + \log_2(n+1) + \log_2|\Sigma|$ is sufficient to encode the symbol $e(kn\tau_s)$. After this, at times $(kn+j)\tau_s$, $j \in \{1, \dots, n-1\}$, the coder observes the current mode of the system and sends a symbol that encodes this mode. For this, a coding alphabet $\mathcal{E}_{kn+j}$ of size $\log_2|\Sigma|$ is sufficient. Hence, it follows that the averaged communication data rate of the coder is equal to (7).

C. Controller implementation

Let τ_s, α and n be as in Subsection IV-A, and $Q(\cdot)$ and $r_0 > 0$ be as in Subsection IV-B. The implementation of the controller is described in Figure 3. See also Figure 4, where the different quantities appearing in the implementation of the controller are represented.

[3] The assumption that the initial state of the system lays in a compact set known from the coder–controller is made for convenience and simplicity of the description of the coder–controller. This assumption can be removed by using a "zooming-out" procedure as in [8].

Initialization: Let $N_0^{\mathrm{sw}} = 0$, and let r_0 be as in Subsection IV-B.

Loop: at time $kn\tau_s$ for $k = 0, 1, 2, \ldots$
- Observe $x(kn\tau_s)$ and $\sigma(kn\tau_s)$.
- **If** $k > 0$: Let N_k^{sw} be the smallest integer in $\{0, \ldots, n\}$ such that β_k defined by (8) with this N_k^{sw} satisfies $\|x(kn\tau_s)\| \le \beta_k r_{k-1}$. Let $r_k = \beta_k r_{k-1}$.
- Let $\eta_k = Q(x(kn\tau_s)/r_k)$.
- Send a symbol $e(kn\tau_s)$ to the controller that encodes the triple $(\eta_k, \sigma(kn\tau_s), N_k^{\mathrm{sw}})$.
- **Loop:** at time $(kn + j)\tau_s$ for $j = 1, 2, \ldots, n-1$
  - Observe $\sigma((kn + j)\tau_s)$.
  - Send a symbol $e((kn + j)\tau_s)$ to the controller that encodes $\sigma((kn + j)\tau_s)$.

Fig. 2.   Coder implementation.

Initialization: Let $N_0^{\mathrm{sw}} = 0$, and let r_0 be as in Subsection IV-B.

Loop: at time $kn\tau_s$ for $k = 0, 1, 2, \ldots$
- Receive symbol $e(kn\tau_s)$ and decode $(\eta_k, \sigma(kn\tau_s), N_k^{\mathrm{sw}})$.
- **If** $k > 0$: let β_k be defined as in (8) with N_k^{sw} obtained from the symbol, and let $r_k = \beta_k r_{k-1}$.
- Let $\xi_{kn} = r_k \eta_k$.
- **For** $t \in [kn\tau_s, (kn + 1)\tau_s)$: apply the input

$$u(t) = \varphi(\hat{x}_{kn}(t), \sigma(kn\tau_s))$$

where $\hat{x}_{kn}(\cdot)$ is the solution of the auxiliary system (9) with the boundary condition $\hat{x}_{kn}(kn\tau_s) = \xi_{kn}$ and with $i = \sigma(kn\tau_s)$.
- **Loop:** at time $(kn + j)\tau_s$ for $j = 1, 2, \ldots, n-1$.
  - Receive symbol $e((kn + j)\tau_s)$ and decode $\sigma((kn + j)\tau_s)$.
  - Let $\xi_{kn+j} = \hat{x}_{kn+j-1}((kn + j)\tau_s)$.
  - **For** $t \in [(kn + j)\tau_s, (kn + j + 1)\tau_s)$: apply the input

$$u(t) = \varphi(\hat{x}_{kn+j}(t), \sigma((kn + j)\tau_s))$$

where $\hat{x}_{kn+j}(\cdot)$ is the solution of the auxiliary system (9) with the boundary condition $\hat{x}_{kn+j}(kn\tau_s) = \xi_{kn+j}$ and with $i = \sigma((kn + j)\tau_s)$.

Fig. 3.   Controller implementation.

The implementation deserves the following explanations. At each time $kn\tau_s$, $k \in \mathbb{N}$, the controller receives the symbol $e(kn\tau_s)$ that encodes η_k, $\sigma(kn\tau_s)$ and N_k^{sw}. Based on this, it is able to compute β_k and r_k, and thus to compute ξ_{kn} which is an approximation of the current state $x(kn\tau_s)$. Then, on the interval $[kn\tau_s, (kn + 1)\tau_s)$, the controller simulates the following auxiliary system:

$$\dot{\hat{x}}(t) = A_i \hat{x}(t) + B_i \varphi(\hat{x}(t), i), \qquad (9)$$

with the boundary condition $\hat{x}(kn\tau_s) = \xi_{kn}$ and with $i = \sigma(kn\tau_s)$. Using the simulated trajectory, denoted by $\hat{x}_{kn}(\cdot)$, the controller applies on the system the control input defined by $u(t) = \varphi(\hat{x}_{kn}(t), i)$.

The same procedure is repeated for each $j \in \{1, \ldots, n-$
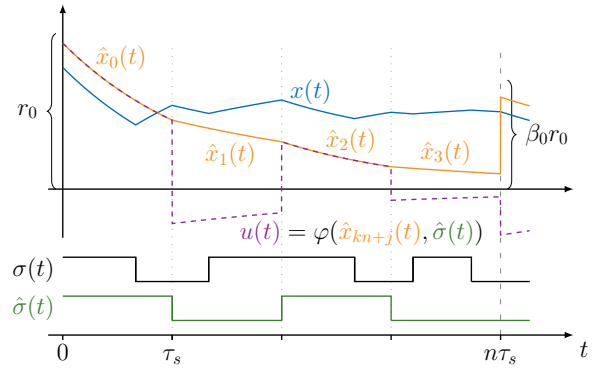

Fig. 4.   The different quantities involved in the implementation of the controller. For this example, the coder–controller is applied on the system $\dot{x}(t) = B_{\sigma(t)} u(t)$, where $B_1 = -1$ and $B_2 = 1$, and with $\varphi(x, 1) = x$ and $\varphi(x, 2) = -x/2$. The switching signal $\sigma(\cdot)$ is represented in black, and the signal $\hat{\sigma}(\cdot)$ obtained from the sampled measurements $\sigma((kn+j)\tau_s)$ (decoded from the symbols $e((kn + j)\tau_s)$) is represented in green.

$1\}$: ξ_{kn+j} is defined as $\hat{x}_{kn+j-1}((kn + j)\tau_s)$. Then, on the interval $[(kn + j)\tau_s, (kn + j + 1)\tau_s)$, the controller simulates the auxiliary system (9) with the boundary condition $\hat{x}((kn + j)\tau_s) = \xi_{kn+j}$ and with $i = \sigma((kn + j)\tau_s)$, decoded from the symbol $e((kn + j)\tau_s)$; and it applies on the system the control input defined by $u(t) = \varphi(\hat{x}_{kn+j}(t), i)$ where $\hat{x}_{kn+j}(\cdot)$ is the simulated trajectory.

D. Proof of the correctness of the coder–controller

See the extended version of this paper [1].

V. Numerical Experiments

Consider the SLS (1) with matrices $A_1 = \begin{bmatrix} 0.1 & -1.0 \\ 1.5 & 0.1 \end{bmatrix}$, $A_2 = \begin{bmatrix} -0.5 & 2.0 \\ -1.5 & 0.0 \end{bmatrix}$, $B_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, $B_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$. This system is stabilizable, in the sense of Assumption 2, via the feedback law $u(t) = K_{\sigma(t)} x(t)$ with $K_1 = [\, -0.43 \; -0.43 \,]$ and $K_2 = [\, -0.38 \; -0.52 \,]$, and with $D = 1$, $\mu_1 = 0$ and $\mu_2 = 0.15$.

First, we have simulated the system with ADT $\tau_a = 1.0$ s. We have used the values $\tau_s = 0.008$, $\alpha = 0.05$ and $n = 100$ for the parameters of the coder–controller, which satisfy (5). With these values of the parameters, the average data rate of the coder–controller is of 145 bits/s. A sample execution of the coder–controller, applied on the system with this ADT, is represented in Figure 5-(top). We observe that the state of the system converges to zero, as predicted.

Then, we have simulated the system with a smaller ADT, namely $\tau_a = 0.25$ s. We have used the values $\tau_s = 0.002$, $\alpha = 0.05$ and $n = 400$ for the parameters of the coder–controller, which satisfy (5). The average data rate of the coder–controller is of 523 bits/s. A sample execution of the coder–controller, applied on the system with this ADT, is represented in Figure 5-(bottom). Again, we observe that the sampled trajectory converges to zero, as predicted.

VI. Conclusions

In this paper, we have first shown that continuous-time switched linear systems with arbitrary switching have in general an *infinite stabilization entropy*, meaning that they
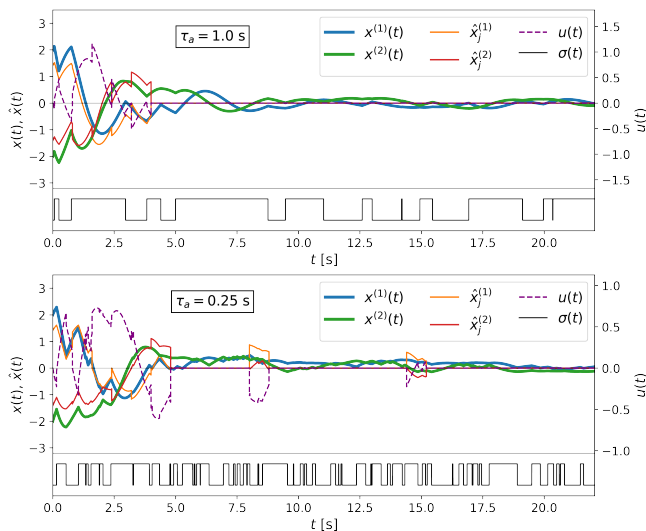
Fig. 5. Evolution of $x(t)$ and $u(t)$ for a sample execution of the coder–controller applied on the system presented in Section V, with different values for the ADT. The black curve below the plot represents the switching signal. The orange and red curves represent the trajectories $\hat{x}(t)$ simulated by the controller (see (9)) to define the input $u(t)$.

cannot be stabilized with any finite data rate. This motivated the introduction of a fairly mild slow-switching assumption on the system. This assumption requires that the switching signal has an average dwell time bounded away from zero. Under this assumption, switching linear systems that are stabilizable in the absence of data-rate constraints are stabilizable by a coder–controller with finite data rate. We have described the implementation of such a coder–controller and demonstrated its applicability on a numerical example. In future works, the question of potential improvement of the data rate of the coder–controller will be investigated. As a few examples, one could introduce the use of Lyapunov functions, as in [8], [17], [19]; refine the analysis of the propagation of reachable sets during sampling intervals, by using tools from multilinear algebra, as in [2]; consider additional assumptions on the system.

VII. Acknowledgments

References

[1] Guillaume O Berger and Raphaël M Jungers. Finite data-rate feedback stabilization of continuous-time switched linear systems with unknown switching signal. *arXiv:2009.04715*, 2020.

[2] Guillaume O Berger and Raphaël M Jungers. Worst-case topological entropy and minimal data rate for state observation of switched linear systems. In *Proceedings of the 23rd International Conference on Hybrid Systems: Computation and Control*, pages 1–11. ACM, 2020.

[3] Fritz Colonius. Minimal bit rates and entropy for exponential stabilization. *SIAM Journal on Control and Optimization*, 50(5):2988–3010, 2012.

[4] João P Hespanha and A Stephen Morse. Stability of switched systems with average dwell-time. In *Proceedings of the 38th IEEE conference on decision and control (Cat. No. 99CH36304)*, volume 3, pages 2655–2660. IEEE, 1999.

[5] João P Hespanha, Payam Naghshtabrizi, and Yonggang Xu. A survey of recent results in networked control systems. *Proceedings of the IEEE*, 95(1):138–162, 2007.

[6] Raphaël M Jungers. *The joint spectral radius: theory and applications*, volume 385 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag Berlin Heidelberg, 2009.

[7] Daniel Liberzon. *Switching in systems and control*. Systems & Control: Foundations & Applications. Birkhäuser, Boston, MA, 2003.

[8] Daniel Liberzon. Finite data-rate feedback stabilization of switched and hybrid linear systems. *Automatica*, 50(2):409–420, 2014.

[9] Hai Lin and Panos J Antsaklis. Stability and stabilizability of switched linear systems: a survey of recent results. *IEEE Transactions on Automatic Control*, 54(2):308–322, 2009.

[10] Qiang Ling and Hai Lin. Necessary and sufficient bit rate conditions to stabilize quantized Markov jump linear systems. In *Proceedings of the 2010 American Control Conference*, pages 236–240. IEEE, 2010.

[11] Alexey S Matveev and Andrey V Savkin. *Estimation and control over communication networks*. Control Engineering. Birkhäuser Basel, 2009.

[12] Girish N Nair, Subhrakanti Dey, and Robin J Evans. Infimum data rates for stabilising Markov jump linear systems. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, volume 2, pages 1176–1181. IEEE, 2003.

[13] Robert Shorten, Fabian Wirth, Oliver Mason, Kai Wulff, and Christopher King. Stability criteria for switched and hybrid systems. *SIAM Review*, 49(4):545–592, 2007.

[14] Zhendong Sun and Shuzhi Sam Ge. *Stability theory of switched dynamical systems*. Communications and Control Engineering. Springer-Verlag London, 2011.

[15] Gerald Teschl. *Topics in real and functional analysis*. Graduate studies in mathematics. American Mathematical Society. To appear.

[16] Arjan J Van Der Schaft and Johannes Maria Schumacher. *An introduction to hybrid dynamical systems*, volume 251 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag London, 2000.

[17] Masashi Wakaiki and Yutaka Yamamoto. Output feedback stabilization of switched linear systems with limited information. In *53rd IEEE Conference on Decision and Control*, pages 3892–3897. IEEE, 2014.

[18] Nan Xiao, Lihua Xie, and Minyue Fu. Stabilization of Markov jump linear systems using quantized state feedback. *Automatica*, 46(10):1696–1702, 2010.

[19] Guosong Yang and Daniel Liberzon. Feedback stabilization of switched linear systems with unknown disturbances under data-rate constraints. *IEEE Transactions on Automatic Control*, 63(7):2107–2122, 2017.

[20] Chun Zhang, Kan Chen, and Geir E Dullerud. Stabilization of Markovian jump linear systems with limited information — a convex approach. In *2009 American Control Conference*, pages 4013–4019. IEEE, 2009.

[21] Jiang Zhong-Ping and Liu Teng-Fei. Quantized nonlinear control — a survey. *Acta Automatica Sinica*, 39(11):1820–1830, 2013.