2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Safety Controller Synthesis for Switched Systems using Multi-scale Symbolic Models

Javier Camara, Antoine Girard, and Gregor Gössler

Abstract— **We propose a technique for the synthesis of safety controllers for switched systems using multi-scale abstractions. To this end we build on a recent notion of multi-scale discrete abstractions for incrementally stable switched systems. These abstractions are defined on a sequence of embedded lattices approximating the state-space, the finer ones being used only in a restricted area where fast switching is needed. This makes it possible to deal with fast switching while keeping the number of states in the abstraction at a reasonable level. We present a synthesis algorithm that exploits the specificities of multi-scale abstractions. The abstractions are computed on the fly during controller synthesis. The finest scales of the abstraction are effectively explored only when fast switching is needed, that is when the system approaches the unsafe set. We provide experimental results that show drastic improvements of the complexity of controller synthesis using multi-scale abstractions instead of uniform abstractions.**

I. Introduction

Symbolic approaches to control of hybrid systems based on the use discrete abstractions have become quite popular (see [9] and the references therein). The main advantage of these approaches is that they offer the possibility to leverage controller synthesis techniques developed in the area of discrete-event systems [4]. A recent trend in symbolic control is to use discrete abstractions that are related to the original system by some approximate equivalence relationship such as approximate bisimulation [6]. It has been shown that such abstractions are computable for several classes of control systems including incrementally stable nonlinear systems [8] and switched systems [7]. These approaches are based on sampling of time and space where the sampling parameters must satisfy some relation in order to obtain abstractions of a prescribed precision. Particularly, the faster the time sampling, the finer the lattice approximating the state-space has to be, resulting in abstractions with a large number of states.

In [3], we introduced a notion of multi-scale discrete abstraction that allows us to deal with fast time sampling while keeping the number of abstract states at a reasonable level. Following the self-triggered control paradigm [11], [2], we assume that the controller has to decide the control input and the duration during which it will be applied. Then, it is natural to consider abstractions where transitions have various durations. For transitions of longer duration, it is sufficient to consider abstract states on a coarse lattice. For transitions of shorter duration, it becomes necessary to use finer lattices. These finer lattices are effectively used only on a restricted area of the state-space where fast time-sampling is needed. The concept of approximately bisimilar multi-scale abstractions has also been explored in [10] where the multi-scale feature is used for accommodating locally the precision of the abstraction while the time sampling period remains constant. On the contrary, the approach presented in [3] seeks for a uniform precision but varying time sampling periods. In both works, the multi-scale abstractions were used to synthesize suboptimal reachability controllers.

In this paper, we propose to use these multi-scale abstractions for the synthesis of safety controllers for switched systems. We present a synthesis algorithm that exploits the specificities of multi-scale abstractions. The abstractions are computed on the fly during controller synthesis and the dynamics at the finest scales are explored only when necessary. We provide experimental results that show drastic improvements of the complexity of controller synthesis using multi-scale abstractions instead of the uniform abstractions defined in [7].

II. Preliminaries

A. Incrementally stable switched systems

Definition 2.1: A switched system is a quadruple $\Sigma = (\mathbb{R}^n, P, \mathcal{P}, F)$, where $\mathbb{R}^n$ is the state space; $P = \{1, \ldots, m\}$ is the finite set of modes; $\mathcal{P}$ is the set of piecewise constant functions from $\mathbb{R}^+$ to P, continuous from the right and with a finite number of discontinuities on every bounded interval of $\mathbb{R}^+$; $F = \{f_1, \ldots, f_m\}$ is a collection of smooth vector fields indexed by P.

A *switching signal* of Σ is a function $\mathbf{p} \in \mathcal{P}$. A piecewise $\mathcal{C}^1$ function $\mathbf{x} : \mathbb{R}^+ \to \mathbb{R}^n$ is said to be a *trajectory* of Σ if it is continuous and there exists a switching signal $\mathbf{p} \in \mathcal{P}$ such that, at each $t \in \mathbb{R}^+$ where the function $\mathbf{p}$ is continuous, $\mathbf{x}$ is continuously differentiable and satisfies:

$$\dot{\mathbf{x}}(t) = f_{\mathbf{p}(t)}(\mathbf{x}(t)).$$

We will denote the point reached at time $t \in \mathbb{R}^+$ from the initial condition x under the switching signal $\mathbf{p}$ by $\mathbf{x}(t, x, \mathbf{p})$ or by $\mathbf{x}(t, x, p)$ if $\mathbf{p}$ is constantly equal to $p \in P$.

The results presented in this paper apply to switched systems satisfying the incremental stability property (i.e. δ-GUAS [1], [7]). Essentially, a switched system is incrementally stable if all trajectories associated with the same switch-

This work was supported by the Agence Nationale de la Recherche (VEDECY project - ANR 2009 SEGI 015 01).

J. Camara is with Department of Informatics Engineering, University of Coimbra, Portugal jcmoreno@dei.uc.pt

A. Girard is with Laboratory Jean Kuntzmann, University of Grenoble, B.P. 53, 38041 Grenoble, France Antoine.Girard@imag.fr

G. Gössler is with INRIA Grenoble – Rhône-Alpes, 38334 Saint Ismier, France Gregor.Goessler@inria.fr

ing signal converge asymptotically to the same reference trajectory independently of their initial condition. Incremental stability of a switched system can be characterized using Lyapunov functions:

Definition 2.2: A smooth function $V : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^+$ is a common δ-GUAS Lyapunov function for Σ if there exist $\mathcal{K}_\infty$ functions[1] $\underline{\alpha}$, $\overline{\alpha}$ and $\kappa > 0$ such that for all $x, y \in \mathbb{R}^n$, for all $p \in P$:

$$\underline{\alpha}(\|x-y\|) \leq V(x,y) \leq \overline{\alpha}(\|x-y\|); \quad (1)$$

$$\frac{\partial V}{\partial x}(x,y) f_p(x) + \frac{\partial V}{\partial y}(x,y) f_p(y) \leq -\kappa V(x,y). \quad (2)$$

As in [7], we will make the supplementary assumption on the δ-GUAS Lyapunov function that there exists a $\mathcal{K}_\infty$ function γ such that

$$\forall x, y, z \in \mathbb{R}^n,\ |V(x,y) - V(x,z)| \leq \gamma(\|y-z\|). \quad (3)$$

This assumption was shown to be not restrictive provided V is smooth and we are interested in the dynamics of Σ on a compact subset of $\mathbb{R}^n$, which is often the case in practice. In [7], it was proved that under the existence of common δ-GUAS Lyapunov function V satisfying equation (3), it is possible to compute discrete abstractions that approximate the dynamics of Σ at any desired level of accuracy.

B. Approximate bisimulation

In this section, we present the notion of approximate equivalence which will relate a switched system to the discrete systems that we construct. We start by introducing transition systems which allow us to model switched and discrete systems in a common mathematical framework.

Definition 2.3: A transition system is a tuple $T = (Q, L, \longrightarrow, O, H, I)$ consisting of a set of states Q; a set of labels or actions L; a transition relation $\longrightarrow \subseteq Q \times L \times Q$; an output set O; an output function $H : Q \to O$; a set of initial states $I \subseteq Q$. T is said to be *metric* if the output set O is equipped with a metric d, *discrete* if Q and L are finite or countable sets.

The transition $(q, l, q') \in \longrightarrow$ will be denoted $q \xrightarrow{l} q'$, or alternatively $q' \in \text{succ}_l(q)$; this means that the system can evolve from state q to state q' under the action l. An action $l \in L$ belongs to the set of *enabled actions* at state q, denoted $\text{enab}(q)$, if $\text{succ}_l(q) \neq \emptyset$. The transition system is said to be *deterministic* if for all $q \in Q$ and $l \in \text{enab}(q)$, $\text{succ}_l(q)$ has only one element. A *trajectory* of the transition system is a finite sequence of transitions $\sigma = q_0 \xrightarrow{l_0} q_1 \xrightarrow{l_1} q_2 \xrightarrow{l_2} \dots \xrightarrow{l_{N-1}} q_N$. It is *initialized* if $q_0 \in I$. A state $q \in Q$ is *reachable* if there exists an initialized trajectory reaching q. The *observed behavior* associated to a trajectory is the sequence of outputs $o_0 o_1 o_2 \dots o_N$ where $o_i = H(q_i)$, for all $i \in \{0, \dots, N\}$.

Transition systems can describe the dynamics of switched systems. Given a switched system $\Sigma = (\mathbb{R}^n, P, \mathcal{P}, F)$, we define the transition system $T(\Sigma) = (Q, L, \longrightarrow, O, H, I)$, where the set of states is $Q = \mathbb{R}^n$; the set of labels is $L = P \times \mathbb{R}^+$; the transition relation is given by

$$x \xrightarrow{p,\tau} x' \text{ iff } \mathbf{x}(\tau, x, p) = x',$$

i.e. the switched system Σ goes from state x to state x' by applying the constant mode p for a duration τ; the set of outputs is $O = \mathbb{R}^n$; the observation map H is the identity map over $\mathbb{R}^n$; the set of initial states is $I = \mathbb{R}^n$. $T(\Sigma)$ is deterministic and metric when the set of outputs $O = \mathbb{R}^n$ is equipped with the metric $d(x, x') = \|x - x'\|$. Note that the state space of $T(\Sigma)$ is uncountable. In the following, we will compute discrete abstractions of $T(\Sigma)$ in the sense of approximate bisimulation, a system approximation relationship introduced in [6].

Definition 2.4: Let $T_i = (Q_i, L, \xrightarrow[i]{}, O, H_i, I_i)$, with $i = 1, 2$ be metric transition systems with the same sets of labels L and outputs O equipped with the metric d. Let $\varepsilon \geq 0$ be a given precision. A relation $R \subseteq Q_1 \times Q_2$ is said to be an ε-approximate bisimulation relation between T_1 and T_2 if for all $(q_1, q_2) \in R$:

- $d(H_1(q_1), H_2(q_2)) \leq \varepsilon$;
- $\forall q_1 \xrightarrow[1]{l} q_1'$, $\exists q_2 \xrightarrow[2]{l} q_2'$, such that $(q_1', q_2') \in R$;
- $\forall q_2 \xrightarrow[2]{l} q_2'$, $\exists q_1 \xrightarrow[1]{l} q_1'$, such that $(q_1', q_2') \in R$.

The transition systems T_1 and T_2 are said to be approximately bisimilar with precision ε, denoted $T_1 \sim_\varepsilon T_2$, if:

- $\forall q_1 \in I_1$, $\exists q_2 \in I_2$, such that $(q_1, q_2) \in R$;
- $\forall q_2 \in I_2$, $\exists q_1 \in I_1$, such that $(q_1, q_2) \in R$.

If T_1 is a system we want to control and T_2 is a simpler system that we want to use for controller synthesis, then T_2 is called an *approximately bisimilar abstraction* of T_1.

III. Multi-scale abstractions for switched systems

Let Σ be a switched system and let us assume that the switching in Σ is determined by a time-triggered controller of time-period $\tau > 0$. Then, the dynamics of Σ can be described by the transition system $T_\tau(\Sigma)$ obtained from $T(\Sigma)$ by selecting the transitions that describe trajectories of duration τ. In [7], an approach to compute approximately bisimilar abstractions of $T_\tau(\Sigma)$ was presented, based on a quantization of the state-space $\mathbb{R}^n$ which is approximated by the lattice:

$$[\mathbb{R}^n]_\eta = \left\{ q \in \mathbb{R}^n \;\middle|\; q[i] = k_i \frac{2\eta}{\sqrt{n}},\ k_i \in \mathbb{Z},\ i = 1, ..., n \right\}$$

where $q[i]$ is the i-th coordinate of q and $\eta > 0$ is a state space discretization parameter. The resulting abstraction $T_{\tau,\eta}(\Sigma)$ is discrete, its set of states and its set of actions are respectively countable and finite. It is shown in [7] that under the existence of a common δ-GUAS Lyapunov function and equation (3), $T_\tau(\Sigma)$ and $T_{\tau,\eta}(\Sigma)$ are approximately bisimilar. Given the time sampling parameter τ, to obtain a prescribed precision ε, it is sufficient to choose η such that

$$\eta \leq \min\left\{ \gamma^{-1}\left((1 - e^{-\kappa\tau}) \underline{\alpha}(\varepsilon) \right), \overline{\alpha}^{-1}\left(\underline{\alpha}(\varepsilon) \right) \right\} \quad (4)$$

[1] A continuous function $\gamma : \mathbb{R}^+ \to \mathbb{R}^+$ is said to belong to class $\mathcal{K}_\infty$ if it is strictly increasing, $\gamma(0) = 0$ and $\gamma(r) \to \infty$ when $r \to \infty$.

Particularly, it should be noted that given $\tau > 0$ and $\varepsilon > 0$, it is always possible to choose $\eta > 0$ such that equation (4) holds. This essentially means that approximately bisimilar discrete abstractions of arbitrary precision can be computed for $T_\tau(\Sigma)$. However, the smaller τ or ε, the smaller η must be to satisfy equation (4). In practice, for a small time sampling parameter τ, the ratio ε/η can be very large and discrete abstractions with an acceptable precision may have a very large number of states (see e.g. [7]).

There are number of applications where the switching has to be fast though this fast switching is generally necessary only on a restricted part of the state space. For instance, for safety controllers, fast switching is needed only when approaching the unsafe set. In order to enable fast switching while dealing with abstractions with a reasonable number of states, one may consider discrete abstractions enabling transitions of different durations. For transitions of long duration, it is sufficient to consider abstract states on a coarse lattice to meet the desired precision ε. As we consider transitions of shorter durations, it becomes necessary to use finer lattices for the abstract state-space. These finer lattices are effectively used only on a restricted area of the state space, where the fast switching is necessary. This allows us to keep the number of states in the abstraction at a reasonable level. This results naturally in a notion of multi-scale discrete abstraction introduced in [3].

For that purpose, we change the control paradigm and use self-triggered controllers [11], [2], where the controller not only determines the mode of the switched system but also the duration during which the mode remains active. We assume that the controller can choose from a finite set of durations $\Theta_\tau^N = \{2^{-s}\tau \mid s = 0, \ldots, N\}$ that consists of dyadic fractions of a time sampling parameter $\tau > 0$ up to some scale parameter $N \in \mathbb{N}$. The dynamics of the switched system is then naturally described by the transition system $T_\tau^N(\Sigma) = (Q_1, L, \xrightarrow[1]{}, O, H_1, I_1)$ where the set of states is $Q_1 = \mathbb{R}^n$; the set of labels is $L = P \times \Theta_\tau^N$; the transition relation is given by

$$x \xrightarrow[1]{p,2^{-s}\tau} x' \text{ iff } \mathbf{x}(2^{-s}\tau, x, p) = x';$$

the set of outputs is $O = \mathbb{R}^n$; the observation map H_1 is the identity map over $\mathbb{R}^n$; the set of initial states is $I_1 = \mathbb{R}^n$.

The computation of a discrete abstraction of $T_\tau^N(\Sigma)$ can then be done using the following approach. We approximate the set of states $Q_1 = \mathbb{R}^n$ by a sequence of embedded lattices: for $s = 0, \ldots, N$, let $Q_2^s = [\mathbb{R}^n]_{2^{-s}\eta}$, i.e.

$$Q_2^s = \left\{ q \in \mathbb{R}^n \;\middle|\; q[i] = k_i \frac{2^{-s+1}\eta}{\sqrt{n}},\ k_i \in \mathbb{Z},\ i = 1, ..., n \right\}$$

where $\eta > 0$ is a state space discretization parameter. Let us remark that we have $Q_2^0 \subseteq Q_2^1 \subseteq \cdots \subseteq Q_2^N$. By simple geometrical considerations, we can check that for all $x \in \mathbb{R}^n$ and $s = 0, \ldots, N$, there exists $q \in Q_2^s$ such that $\|x - q\| \leq 2^{-s}\eta$. Then, we can define the abstraction of $T_\tau^N(\Sigma)$ as the transition system $T_{\tau,\eta}^N(\Sigma) = (Q_2, L, \xrightarrow[2]{}, O, H_2, I_2)$, where the set of states is $Q_2 = Q_2^N$; the set of actions remains $L = P \times \Theta_\tau^N$; the transition relation is given by

$$q \xrightarrow[2]{p,2^{-s}\tau} q' \text{ iff } q' = \arg\min_{r \in Q_2^s} (\|\mathbf{x}(2^{-s}\tau, q, p) - r\|).$$

If the minimizer $r \in Q_2^s$ is not unique, then one can choose arbitrarily one of them. The set of outputs remains $O = \mathbb{R}^n$; the observation map H_2 is the natural inclusion map from Q_2^N to $\mathbb{R}^n$, i.e. $H_2(q) = q$; the set of initial states is $I_2 = Q_2^0$.

It is important to remark that all the transitions of duration $2^{-s}\tau$ end in states belonging to Q_2^s. This means that the states on the finer lattices are only accessible by transitions of shorter duration. Note that the transition system $T_{\tau,\eta}^N(\Sigma)$ is discrete since its sets of states and actions are respectively countable and finite. Also, if we only consider transitions of duration τ, the dynamics of $T_{\tau,\eta}^N(\Sigma)$ coincides with that of the uniform abstraction $T_{\tau,\eta}(\Sigma)$ defined in [7]. Both transition systems $T_\tau^N(\Sigma)$ and $T_{\tau,\eta}^N(\Sigma)$ are deterministic. It was proved in [3] that under the existence of a common δ-GUAS Lyapunov function and equation (3), these transition systems are approximately bisimilar:

Theorem 3.1: Consider a switched system Σ, time and state space sampling parameters $\tau, \eta > 0$, scale parameter $N \in \mathbb{N}$, and a desired precision $\varepsilon > 0$. Let us assume that there exists a common δ-GUAS Lyapunov function V for Σ such that equation (3) holds. If

$$\eta \leq \min\left\{ \min_{s=0\ldots N} \left[2^s \gamma^{-1}\left((1 - e^{-\kappa 2^{-s}\tau}) \underline{\alpha}(\varepsilon) \right) \right], \overline{\alpha}^{-1}\left(\underline{\alpha}(\varepsilon)\right) \right\} \tag{5}$$

then $T_\tau^N(\Sigma) \sim_\varepsilon T_{\tau,\eta}^N(\Sigma)$.

It is interesting to note that given a time sampling parameter $\tau > 0$ and a scale parameter $N \in \mathbb{N}$, for all desired precisions $\varepsilon > 0$, there exists $\eta > 0$ such that equation (4) holds. This essentially means that approximately bisimilar multi-scale abstractions of arbitrary precision can be computed for $T_\tau^N(\Sigma)$.

IV. Controller Synthesis using Multi-scale Abstractions

We illustrate the use of multi-scale abstractions for synthesizing safety controllers. This problem was considered in [5] based on the use of uniform discrete abstractions. We extend the synthesis algorithm to multi-scale abstractions that are computed on-the-fly, so as to provide a scalable trade-off between precision and cost.

A. Problem formulation

Let us consider a system $T = (Q, L, \longrightarrow, O, H, I)$ and a safety specification $Q_S \subseteq Q$ (which can easily be obtained from a subset $O_S \subseteq O$ of safe outputs). For simplicity, we assume that T is deterministic. This is satisfied by the transition systems $T_\tau^N(\Sigma)$ and $T_{\tau,\eta}^N(\Sigma)$ defined in the previous section.

Definition 4.1: A state q of T is *controllable* with respect to a safety specifications Q_S if $q \in Q_S$ and there exists an infinite sequence of transitions of T starting in q and remaining in Q_S. We denote the set of controllable states by $\text{cont}(Q_S)$.

Definition 4.2: A *controller* for T is a map $\mathcal{S} : Q \to 2^L$ such that for all $q \in Q$, $\mathcal{S}(q) \subseteq \text{enab}(q)$. The system T controlled by $\mathcal{S}$ is the system $T/\mathcal{S} = (Q, L, \xrightarrow[\mathcal{S}]{}, O, H, I)$ where the transition relation is given by

$$q \xrightarrow[\mathcal{S}]{l} q' \iff \left[(l \in \mathcal{S}(q)) \wedge (q \xrightarrow{l} q')\right].$$

The *domain* of $\mathcal{S}$ is the set $\text{dom}(\mathcal{S}) = \{q \in Q \mid \mathcal{S}(q) \neq \emptyset\}$. A controller $\mathcal{S}$ is a safety controller if for all $q \in \text{dom}(\mathcal{S})$:

1) $q \in Q_S$ (safety);
2) $\forall l \in \mathcal{S}(q)$, $\text{succ}_l(q) \in \text{dom}(\mathcal{S})$ (deadend freedom).

It is easy to show that for any safety controller $\mathcal{S}$, we have $\text{dom}(\mathcal{S}) \subseteq \text{cont}(Q_S)$. Given the set of controllable states $\text{cont}(Q_S)$, we can define a safety controller $\mathcal{S}^*$ as follows: for all $q \notin \text{cont}(Q_S)$, $\mathcal{S}^*(q) = \emptyset$ and for all $q \in \text{cont}(Q_S)$,

$$\mathcal{S}^*(q) = \{l \in \text{enab}(q) \mid \text{succ}_l(q) \in \text{cont}(Q_S)\}.$$

In that case we have $\text{dom}(\mathcal{S}) = \text{cont}(Q_S)$. This safety controller is maximal in the sense that any other safety controller $\mathcal{S}$ satisfies $\mathcal{S}(q) \subseteq \mathcal{S}^*(q)$, for all $q \in Q$. Let us remark that the set $\text{cont}(Q_S)$ and thus $\mathcal{S}^*$ are computable for our discrete abstractions. However, the larger the number of states, the more expensive the computation. For that reason, we want to exploit multi-scale abstractions to propose a more efficient algorithm for the synthesis of safety controllers.

Let us consider that the set of labels of T is $L = P \times \Theta_\tau^N$ as defined in the previous section. The lazy safety synthesis problem consists in controlling a system so as to keep any trajectory starting from some initial state in I within the safe subset of states, while applying in each state a transition of the longest possible duration for which safety can be guaranteed. For that purpose we define priority relations on the set of labels giving priority to transitions of longer duration: for $l, l' \in L$ with $l = (p, \delta)$, $l' = (p', \delta')$, $l \preceq l'$ iff $\delta \leq \delta'$, $l \prec l'$ iff $\delta < \delta'$ and $l \cong l'$ iff $\delta = \delta'$. Given a subset of labels $L' \subseteq L$, we define

$$\max_{\preceq}(L') = \{l' \in L' \mid \forall l \in L', l \preceq l'\}.$$

Definition 4.3: A maximal lazy safety controller for T and Q_S is a controller $\mathcal{S}$ such that all controllable states in I are in $\text{dom}(\mathcal{S})$, and for all states $q \in \text{dom}(\mathcal{S})$, q is reachable in $T/\mathcal{S}$ and the following conditions hold:

1) $q \in Q_S$ (safety);
2) $\forall l \in \mathcal{S}(q)$, $\text{succ}_l(q) \in \text{dom}(\mathcal{S})$ (deadend freedom);
3) if $l \in \mathcal{S}(q)$, then for any $l \prec l'$, $\text{succ}_{l'}(q) \notin \text{cont}(Q_S)$ (laziness);
4) if $l \in \mathcal{S}(q)$, then for any $l \cong l'$, $l' \in \mathcal{S}(q)$ iff $\text{succ}_{l'}(q) \in \text{cont}(Q_S)$ (maximality).

It is clear from conditions 1) and 2) that $\mathcal{S}$ is a safety controller. The controller $\mathcal{S}$ represents a trade-off between maximal permissiveness and efficiency, in the sense that it contains the same initial states as the maximal safety controller; on the other hand, in each state, the enabled transitions are those of maximal duration for which controllability is preserved.

Theorem 4.4: There exists a unique maximal lazy safety controller.

Proof: We start with existence. Let $\bar{\mathcal{S}}^*$ be the controller defined from the maximal (non-lazy) safety controller $\mathcal{S}^*$ as follows: for all $q \in Q$, $\bar{\mathcal{S}}^*(q) = \max_{\preceq} \mathcal{S}^*(q)$. It is easy to check that $\text{dom}(\mathcal{S}^*) = \text{dom}(\bar{\mathcal{S}}^*)$ and that for all $q \in \text{dom}(\bar{\mathcal{S}}^*)$, conditions 1) to 4) of Definition 4.3 hold for $\bar{\mathcal{S}}^*$. Now let $\mathcal{S}$ be the controller defined from $\bar{\mathcal{S}}^*$ by $\mathcal{S}(q) = \bar{\mathcal{S}}^*(q)$ if q is reachable in $T/\bar{\mathcal{S}}^*$ and $\mathcal{S}(q) = \emptyset$ otherwise. It is clear that the reachable states in $T/\bar{\mathcal{S}}^*$ and $T/\mathcal{S}$ are the same. Hence, for all $q \in \text{dom}(\mathcal{S})$, q is reachable in $T/\mathcal{S}$. Moreover, conditions 1) to 4) of Definition 4.3 hold for $\mathcal{S}$. Let $q \in I$ be controllable, then $q \in \text{dom}(\mathcal{S}^*) = \text{dom}(\bar{\mathcal{S}}^*)$. Since any initial state is reachable, we have $\mathcal{S}(q) = \bar{\mathcal{S}}^*(q) \neq \emptyset$ and $q \in \text{dom}(\mathcal{S})$.

We now prove uniqueness. Let $\mathcal{S}_1$ and $\mathcal{S}_2$ be two maximal lazy safety controllers and assume that there exists $q \in Q$ such that $\mathcal{S}_1(q) \neq \mathcal{S}_2(q)$. If both $\mathcal{S}_1(q)$ and $\mathcal{S}_2(q)$ are not empty, we can assume without loss of generality that there exists $l_1 \in \mathcal{S}_1(q)$ such that $l_1 \notin \mathcal{S}_2(q)$. Then, let $l_2 \in \mathcal{S}_2(q)$. If $l_1 \prec l_2$ then condition 3) does not hold for $\mathcal{S}_1$ since $\text{succ}_{l_2}(q) \in \text{dom}(\mathcal{S}_2) \subseteq \text{cont}(Q_S)$. If $l_2 \prec l_1$ then condition 3) does not hold for $\mathcal{S}_2$ since $\text{succ}_{l_1}(q) \in \text{dom}(\mathcal{S}_1) \subseteq \text{cont}(Q_S)$. If $l_1 \cong l_2$, then condition 4) does not hold for $\mathcal{S}_2$ since $\text{succ}_{l_1}(q) \in \text{dom}(\mathcal{S}_1) \subseteq \text{cont}(Q_S)$. In all the cases one of the controllers is not a maximal lazy safety controller.

If one of $\mathcal{S}_1(q)$ and $\mathcal{S}_2(q)$ is empty, we can assume without loss a generality that $\mathcal{S}_1(q) \neq \emptyset$ and $\mathcal{S}_2(q) = \emptyset$. $q \in \text{dom}(\mathcal{S}_1) \subseteq \text{cont}(Q_S)$ therefore q cannot be in I otherwise we would have $\mathcal{S}_2(q) \neq \emptyset$. Since $q \in \text{dom}(\mathcal{S}_1)$, q is reachable in $T/\mathcal{S}_1$. Let us consider the initialized trajectory of $T/\mathcal{S}_1$, $q_0 \xrightarrow{l_0} q_1 \xrightarrow{l_1} \dots \xrightarrow{l_{N-1}} q_N = q$. $q_0 \in \text{dom}(\mathcal{S}_1)$ is a controllable initial states and therefore $\mathcal{S}_2(q_0) \neq \emptyset$. Then, there exists $i \in \{0, \dots, N-1\}$ such that $\mathcal{S}_2(q_i) \neq \emptyset$ and $l_i \notin \mathcal{S}_2(q_i)$ (otherwise we would have $\mathcal{S}_2(q) \neq \emptyset$). Therefore, there exists $q_i \in Q$, such that $\mathcal{S}_1(q_i) \neq \mathcal{S}_2(q_i)$ and both $\mathcal{S}_1(q_i)$ and $\mathcal{S}_2(q_i)$ are not empty. We have already proved that in this case one of the controllers is not a maximal lazy safety controller. ■

B. Discrete controller synthesis for multi-scale abstractions

Algorithm 1 computes the safety controller for given T, and Q_S. It works as follows. The **for** loop in line 5 iterates over all states q to be (re)visited. Initially, this is the set of initial states. The **repeat** loop in line 8 iterates over the decreasing durations $i = \tau, ..., 2^{-N}\tau$ in order to explore durations of longer durations first (line 9). The **for** loop in line 10 iterates over all modes p and computes the successor q' under transition $a = (p, 2^{-i}\tau)$. If q' is safe (line 13) then the transition is added to the controller. If in addition, q' has not been visited yet (line 16), then q' is added to the set of states to be visited. After termination of the **repeat – until** loop of lines 8 – 23, if all modes and durations have been explored but none leads to a safe state, then q is removed from the set of safe states Q_S, the states whose only successor is q are scheduled to be revisited, and the transitions from and to q are removed from the controller. Finally the obtained controller is returned.

Algorithm 1 Safety synthesis.

Input: System $T = (Q, L, \to, O, H, I)$, safe state space Q_S, priority order $\preceq \subseteq L \times L$.
Output: controller $\mathcal{S} : Q \to 2^L$.
Auxiliary variables: explored states X, states $todo$ to be (re)visited, map $explored : X \to 2^L$, $maximal \subseteq L$, Boolean $found\_succ$.
Invariant: $todo \subseteq X \subseteq Q_S \subseteq Q$.

1: $(todo, X, \to_{\mathcal{S}}) := (I, I, \emptyset)$;
2: **for** $q \in I$ **do**
3: $\quad explored(q) := \emptyset$
4: **end for**
5: **for** $q \in todo$ **do**
6: $\quad todo := todo \setminus \{q\}$;
7: $\quad found\_succ := false$;
8: $\quad$ **repeat**
9: $\quad\quad maximal := \max_{\preceq} \big(L \setminus explored(q)\big)$;
10: $\quad\quad$ **for** $a \in maximal$ **do**
11: $\quad\quad\quad explored(q) := explored(q) \cup \{a\}$;
12: $\quad\quad\quad q' := succ_a(q); \quad \{q \xrightarrow{a} q'\}$
13: $\quad\quad\quad$ **if** $q' \in Q_S$ **then**
14: $\quad\quad\quad\quad \to_{\mathcal{S}} := \to_{\mathcal{S}} \cup \{(q, a, q')\}$;
15: $\quad\quad\quad\quad found\_succ := true$;
16: $\quad\quad\quad\quad$ **if** $q' \notin X$ **then**
17: $\quad\quad\quad\quad\quad X := X \cup \{q'\}$;
18: $\quad\quad\quad\quad\quad todo := todo \cup \{q'\}$;
19: $\quad\quad\quad\quad\quad explored(q') := \emptyset$
20: $\quad\quad\quad\quad$ **end if**
21: $\quad\quad\quad$ **end if** {ignore a if successor is unsafe}
22: $\quad\quad$ **end for**
23: $\quad$ **until** $found\_succ \vee explored(q) = L$
24: $\quad$ **if** $(explored(q) = L) \wedge (\{a \in L \mid \exists q' : q \xrightarrow{a}_{\mathcal{S}} q'\} = \emptyset)$ **then** {revisit possibly unsafe state q}
25: $\quad\quad X := X \setminus \{q\}$;
26: $\quad\quad Q_S := Q_S \setminus \{q\}$;
27: $\quad\quad todo := todo \cup \{q' \in X \mid \forall (a, q'') : (q' \xrightarrow{a}_{\mathcal{S}} q'' \Rightarrow q'' = q)\}$; {revisit predecessors of q whose only successor is q}
28: $\quad\quad \to_{\mathcal{S}} := \to_{\mathcal{S}} \setminus \{(q_1, a, q_2) \mid a \in L \wedge (q_1 = q \vee q_2 = q)\}$
29: $\quad$ **end if**
30: **end for**
31: **return** $\{(q, \ell) \mid \exists q' : q \xrightarrow{\ell}_{\mathcal{S}} q')\}$

Algorithm 1 terminates: the outer loop iterates over $todo$ to which each state q from the finite set Q_S is added at most once when q is added to X (line 18), and at most once for each successor that is removed from Q_S (line 27); the set Q_S is decreasing. The inner **repeat** loop iterates in the worst case over the finite set of transitions issued from q.

Let us remark that the multi-scale abstraction is computed on the fly during the synthesis algorithm. Therefore, the dynamics at the finer scales is only explored when necessary. This allows us to synthesize safety controllers at a reduced computational cost as shown in the following section.

V. Experimental Results

For illustration purpose, we apply our approach to a boost DC-DC converter. It is a switched system with modes, the two dimensional dynamics associated with both modes are affine of the form $\dot{x}(t) = A_p x(t) + b$ for $p = 1, 2$ (see [7] for numerical values). It can be shown that it is incrementally stable and thus approximately bisimilar discrete abstractions can be computed. We consider the problem of keeping the state of the system in a desired region of operation given by the safe set $O_S = [1.15, 1.55] \times [5.45, 5.85]$.

In the following, we use approximately bisimilar abstractions to synthesize safety switching controllers. We set the desired precision of abstractions to 0.1. For the sake of comparison, we choose to work both with uniform and multi-scale abstractions. The uniform abstractions $T_{\tau_i,\eta_i}(\Sigma)$ are computed according to [7] for time sampling parameters $\tau_1 = 1$ and $\tau_2 = 0.5$. The state-space sampling parameters are chosen according to equation (4), that is $\eta_1 = 5 \times 10^{-4}\sqrt{2}$ and $\eta_2 = 2.5 \times 10^{-4}\sqrt{2}$, respectively. We also use multi-scale abstractions $T^N_{\tau,\eta}$ for parameters $\tau = 2$, $\eta = 10^{-3}\sqrt{2}$ and $N \in \{1, 2\}$ chosen according to Theorem 3.1. This corresponds to transitions of possible duration $\Theta^1_\tau = \{2, 1\}$ and $\Theta^2_\tau = \{2, 1, 0.5\}$. Hence, the controllers synthesized using $T^1_{\tau,\eta}$ and $T^2_{\tau,\eta}$ are to be compared with those of T_{τ_1,η_1} and T_{τ_2,η_2}, respectively.

Figure 1 depicts the controllers for T_{τ_1,η_1} and T_{τ_2,η_2}, as well as the controllers computed by Algorithm 1 for $T^1_{\tau,\eta}$ and $T^2_{\tau,\eta}$. Table I details the experimental results obtained for the synthesis of the aforementioned set of controllers. Looking at the results, it is worth emphasizing that in general, there is a remarkable reduction in the overall time used to compute the controller using multi-scale abstractions with respect to the use of uniform ones (up to a 84% improvement between T_{τ_2,η_2} and $T^2_{\tau,\eta}$). However, this reduction in computation time is obtained for the finest level of resolution (on the contrary, the improvement is close to a 20% when we compare the use of abstractions T_{τ_1,η_1} and $T^1_{\tau,\eta}$). This is due to the fact that the size of uniform abstractions grows exponentially with higher resolutions, whereas the process that we use with multi-scale abstractions bounds this growth by refining the abstraction only in some specific regions of the state-space (we have observed a reduction in the size of abstractions of 38% and 74% for two and three resolution levels, respectively). Interestingly, the aforementioned reduction in computation time and abstraction size does not affect the performance of the multi-scale controllers, which yield a ratio of controllable initial states[2] over the safety specification comparable to that of their uniform counterparts.

Furthermore, the resulting multi-scale controllers have a lower switching frequency, since they apply shorter transition durations only when they are necessary (the duration of about 30% of transitions in the controllers obtained both from $T^1_{\tau,\eta}$ and $T^2_{\tau,\eta}$ is 2 seconds). Let us remark that synthesizing a

[2]The ratio of controllable initial states for a a controller $\mathcal{S} : Q \to 2^L$ and a system $T = (Q, L, \to, O, H, I)$ is computed as $|\{q \in I | \mathcal{S}(q) \neq \emptyset\}|/|I|$.

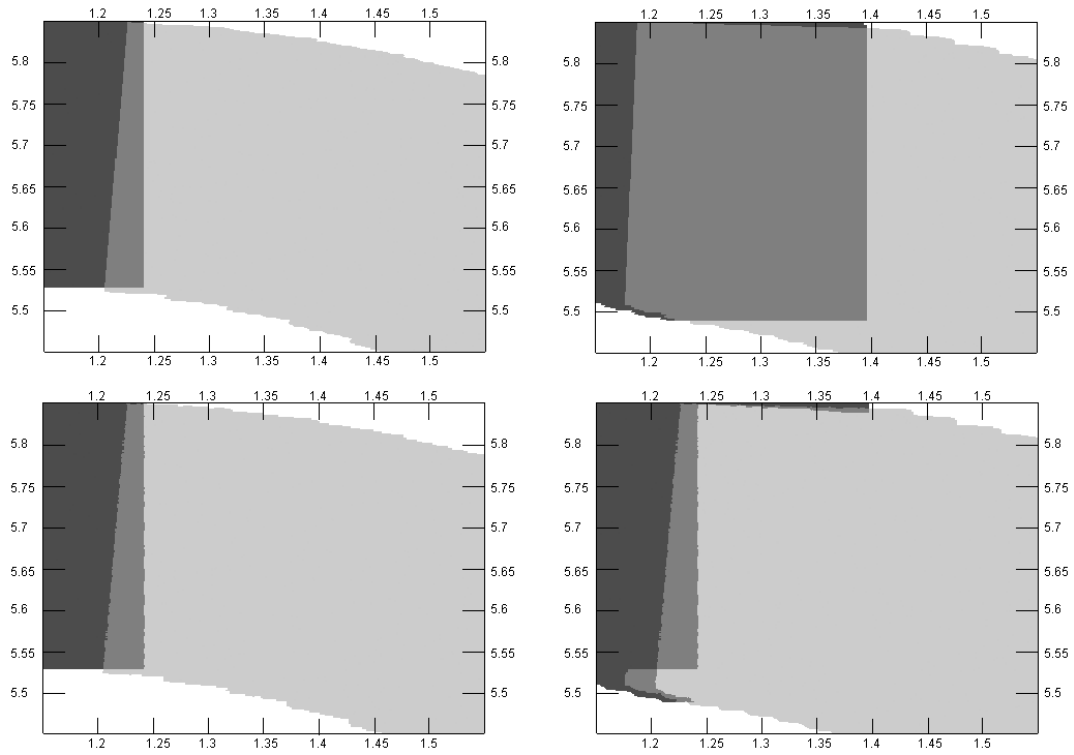

Fig. 1. Top: safety controllers for uniform abstractions $T_{\tau_1,\eta_1}(\Sigma)$ (left) and $T_{\tau_2,\eta_2}(\Sigma)$ (right). Bottom: safety controllers for multi-scale abstractions $T^1_{\tau,\eta}(\Sigma)$ (left) and $T^2_{\tau,\eta}(\Sigma)$ (right) computed using Algorithm 1. (dark gray: mode 1, light gray: mode 2, medium gray: modes 1 and 2).

	Uniform abstractions $T_{\tau,\eta}(\Sigma)$		Multi-scale abstractions $T^N_{\tau,\eta}(\Sigma)$	
	$\tau = 1s$ $\eta = 5 \times 10^{-4}\sqrt{2}$	$\tau = 0.5s$ $\eta = 2.5 \times 10^{-4}\sqrt{2}$	$N = 1, \tau = 2s$ $\eta = 10^{-3}\sqrt{2}$	$N = 2, \tau = 2s$ $\eta = 10^{-3}\sqrt{2}$
controllability ratio	85%	94%	85%	94%
computation time	29s	253s	23s (-20%)	40s (-84%)
abstraction size [10^3 states]	160	641	100 (-38%)	168 (-74%)
transition durations	1s: 100%	0.5s: 100%	2s: 32% / 1s: 68%	2s: 29% / 1s: 67% / 0.5s: 4%

TABLE I

COMPARISON OF EXPERIMENTAL RESULTS FOR UNIFORM AND MULTI-SCALE ABSTRACTIONS FOR THE BOOST DC-DC CONVERTER.

controller for the safety specification used in this problem based on an uniform abstraction with $\tau = 2s$ leads to an empty controller.

VI. CONCLUSION

In this paper, we have proposed the use of multi-scale, approximately bisimilar discrete abstractions for the computation of controllers, applying them to the specific case of safety problems. In particular, our experimental results have shown that we can achieve a remarkable reduction in the computation time of such controllers in comparison with the use of uniform abstractions, while preserving similar levels of performance. Future work will deal with the application of multi-scale abstractions to other kinds of control problems.

REFERENCES

[1] D. Angeli. A Lyapunov approach to incremental stability properties. *IEEE Trans. on Automatic Control*, 47(3):410–421, March 2002.

[2] A. Anta and P. Tabuada. To sample or not to sample: Self-triggered control for nonlinear systems. *IEEE Trans. on Automatic Control*, 55(9):2030–2042, 2010.

[3] J. Camara, A. Girard, and G. Goessler. Synthesis of switching controllers using approximately bisimilar multiscale abstractions. In *Hybrid Systems: Computation and Control*, 2011.

[4] C. G. Cassandras and S. Lafortune. *Introduction to Discrete Event Systems - Second Edition*. Springer, 2007.

[5] A. Girard. Synthesis using approximately bisimilar abstractions: state-feedback controllers for safety specifications. In *Hybrid Systems: Computation and Control*, 2010.

[6] A. Girard and G. Pappas. Approximation metrics for discrete and continuous systems. *IEEE Trans. on Automatic Control*, 52(5):782–798, 2007.

[7] A. Girard, G. Pola, and P. Tabuada. Approximately bisimilar symbolic models for incrementally stable switched systems. *IEEE Trans. on Automatic Control*, 55(1):116–126, 2010.

[8] G. Pola, A. Girard, and P. Tabuada. Approximately bisimilar symbolic models for nonlinear control systems. *Automatica*, 44(10):2508–2516, 2008.

[9] P. Tabuada. *Verification and Control of Hybrid Systems - A Symbolic Approach*. Springer, 2009.

[10] Y. Tazaki and J. Imura. Approximately bisimilar discrete abstractions of nonlinear systems using variable-resolution quantizers. In *American Control Conference*, pages 1015–1020, 2010.

[11] M. Velasco, J. Fuertes, and P. Marti. The self triggered task model for real-time control systems. In *24th IEEE Real-Time Systems Symposium*, pages 67–70, 2003.