

CONTROLLER SYNTHESIS FOR SAFETY AND REACHABILITY VIA APPROXIMATE BISIMULATION

ANTOINE GIRARD

ABSTRACT. In this paper, we consider the problem of controller design using approximately bisimilar abstractions with an emphasis on safety and reachability specifications. We propose abstraction-based approaches to solve both classes of problems. We start by synthesizing a controller for an approximately bisimilar abstraction. Then, using a concretization procedure, we obtain a controller for our initial system that is proved “correct by design”. We provide guarantees of performance by giving estimates of the distance of the synthesized controller to the maximal (i.e the most permissive) safety controller or to the time-optimal reachability controller. Finally, we use the presented techniques combined with discrete approximately bisimilar abstractions of switched systems developed recently, for switching controller synthesis.

1. INTRODUCTION

The use of discrete abstractions has become a standard approach to hybrid systems design [RO98, MR99, HvS01, TP06, KB06, Rei09]. The benefit of this approach is double. Firstly, by abstracting the continuous dynamics, controller synthesis problems can be efficiently solved using techniques developed in the areas of supervisory control of discrete-event systems [RW87] or algorithmic game theory [AVW03]. Secondly, if the behaviors of the original system and of the discrete abstraction are formally related by an inclusion or equivalence relationship, the synthesized controller is known to be correct by design and thus the need of formal verification is reduced. Abstraction, using traditional systems behavioral relationships, relies on inclusion or equality of observed behaviors. One of the most common notions is that of bisimulation equivalence [Mil89]. However, for systems observed over metric spaces, requiring strict equality of observed behaviors is often too strong. Indeed, the class of continuous or hybrid systems admitting bisimilar discrete abstractions is quite restricted [AHLP00, Tab09]. In [GP07], a notion of approximate bisimulation, which only asks for closeness of observed behaviors, was introduced. This relaxation made it possible to extend the class of systems for which discrete abstractions can be computed [PGT08, GPT10].

This paper deals with the synthesis of controllers using approximately bisimilar abstractions with an emphasis on safety and reachability problems. Safety problems consist in synthesizing a controller that restricts the behaviors of a system so that its outputs remain in some specified safe set. One is usually interested in designing a controller that is as permissive as possible since this makes it possible, using modular approaches, to ensure, a posteriori, secondary control objectives (see e.g. [RW87]). Reachability problems consist in synthesizing a controller that steers the observations of the system to some target region while keeping them in a given safe set along the way. In addition, in order to choose among the possible controllers, we try to minimize the time to reach the target. Hence, we consider a time-optimal control problem. We propose abstraction-based approaches to solve both classes of problems. We start by synthesizing a controller for an approximately bisimilar abstraction

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

of our concrete system. Then, using a concretization procedure that is problem-specific, we obtain a controller for our concrete system that is proved “correct by design”. For safety problems, we provide estimates of the distance between the synthesized controller and the maximal (i.e the most permissive) safety controller. For reachability problems, we provide estimates of the distance between the performances of the synthesized controller and of the time-optimal controller. As an illustration, we use these techniques in combination with the discrete approximately bisimilar abstractions of switched systems developed in [GPT10], for switching controller synthesis. Preliminary versions of these results appeared in the conference papers [Gir10a, Gir10b]. The presentation has been improved and new results on estimates of the distance to maximal or optimal controllers have been added. Deeper numerical experiments have also been provided.

Controller synthesis using approximately (bi)similar abstractions has also been considered in [TI08, MT10]. In [TI08], the authors use approximately bisimilar abstractions to design a suboptimal controller for a fixed bounded horizon optimal control problem. In this paper, we consider time-optimal control for reachability specifications, thus the time horizon is variable. Our work is more closely related to [MT10] where time-optimal control is considered as well. We shall discuss further in the paper the main differences with our approach. Regarding safety specifications, this is the first paper proposing a specific approach using approximately bisimilar abstractions.

2. PRELIMINARIES

We start by introducing the class of transition systems which serves as a common abstract modeling framework for discrete, continuous or hybrid systems (see e.g. [AHL00, Tab09]).

Definition 2.1. A *transition system* is a tuple $T = (Q, L, \delta, O, H)$ consisting of a set of states Q ; a set of actions L ; a transition relation $\delta \subseteq Q \times L \times Q$; a set of observations O ; an output function $H : Q \rightarrow O$. T is said to be *discrete* if Q and L are finite or countable sets, *metric* if the set of observations O is equipped with a metric d .

The transition $(q, l, q') \in \delta$ will be denoted $q' \in \delta(q, l)$; this means that the system can evolve from state q to state q' under the action l . Given a subset of actions $L' \subseteq L$, we denote $\delta(q, L') = \bigcup_{l \in L'} \delta(q, l)$. An action $l \in L$ belongs to the set of *enabled actions* at state q , denoted $\text{Enab}(q)$, if $\delta(q, l) \neq \emptyset$. If $\text{Enab}(q) = \emptyset$, then q is said to be a *blocking* state; otherwise it is said to be *non-blocking*. If all states are non-blocking, we say that the transition system T is non-blocking. The transition system is said to be *non-deterministic* if there exists $q \in Q$ and $l \in \text{Enab}(q)$ such that $\delta(q, l)$ has several elements. A *trajectory* of the transition system is a finite sequence of states and actions $(q_0, l_0), (q_1, l_1), \dots, (q_{N-1}, l_{N-1}), q_N$ where $q_{i+1} \in \delta(q_i, l_i)$ for all $i \in \{0, \dots, N-1\}$. $N \in \mathbb{N}$ is referred to as the *length* of the trajectory. The *observed behavior* associated to trajectory is the finite sequence of observations $o_0 o_1 o_2 \dots o_N$ where $o_i = H(q_i)$, for all $i \in \{0, \dots, N\}$.

This paper deals with controller synthesis for transition systems; we shall consider only static (i.e. without memory) state-feedback controllers. However, we will just use the term controller for brevity.

Definition 2.2. A *controller* for transition system T is a map $\mathcal{S} : Q \rightarrow 2^L$. It is *well-defined* if $\mathcal{S}(q) \subseteq \text{Enab}(q)$, for all $q \in Q$. The dynamics of the controlled system is described by the transition system $T_{\mathcal{S}} = (Q, L, \delta_{\mathcal{S}}, O, H)$ where the transition relation is given by $q' \in \delta_{\mathcal{S}}(q, l)$ if and only if $l \in \mathcal{S}(q)$ and $q' \in \delta(q, l)$.

Given a subset of states $Q' \subseteq Q$, we will denote $\mathcal{S}(Q') = \bigcup_{q \in Q'} \mathcal{S}(q)$. Let us remark that a state q of $T_{\mathcal{S}}$ is non-blocking if and only if $\mathcal{S}(q) \neq \emptyset$. A controller essentially executes as follows. The state

q of T is measured, an action $l \in \mathcal{S}(q)$ is selected and actuated. Then, the system takes a transition $q' \in \delta(q, l)$, this is always possible if \mathcal{S} is well-defined.

In this paper, we consider approximate equivalence relationships for transition systems defined by approximate bisimulation relations introduced in [GP07].

Definition 2.3. Let $T_i = (Q_i, L, \delta_i, O, H_i)$, $i = 1, 2$, be two metric transition systems with the same sets of actions L and observations 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 $(q_1, q_2) \in R$:

- $d(H_1(q_1), H_2(q_2)) \leq \varepsilon$;
- $\forall l \in \text{Enab}_1(q_1), \forall q'_1 \in \delta_1(q_1, l), \exists q'_2 \in \delta_2(q_2, l)$, such that $(q'_1, q'_2) \in R$;
- $\forall l \in \text{Enab}_2(q_2), \forall q'_2 \in \delta_2(q_2, l), \exists q'_1 \in \delta_1(q_1, l)$, 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:

- $\forall q_1 \in Q_1, \exists q_2 \in Q_2$, such that $(q_1, q_2) \in R$;
- $\forall q_2 \in Q_2, \exists q_1 \in Q_1$, 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 .

We will denote for $q_1 \in Q_1$, $R(q_1) = \{q_2 \in Q_2 \mid (q_1, q_2) \in R\}$ and for $Q'_1 \subseteq Q_1$, $R(Q'_1) = \bigcup_{q_1 \in Q'_1} R(q_1)$; for $q_2 \in Q_2$, $R^{-1}(q_2) = \{q_1 \in Q_1 \mid (q_1, q_2) \in R\}$ and for $Q'_2 \subseteq Q_2$, $R^{-1}(Q'_2) = \bigcup_{q_2 \in Q'_2} R^{-1}(q_2)$.

Remark 2.4. We assume that systems T_1 and T_2 have the same sets of actions and that matching transitions in the second and third items of the previous definition share the same input. These conditions can actually be relaxed using the notion of alternating approximate bisimulation relation [Tab09]. The results presented in this paper can be easily extended to that setting.

The problem of computing approximately bisimilar discrete abstractions has been considered for nonlinear control systems [PGT08] and switched systems [GPT10]. A controller designed for an abstraction can be used to synthesize a controller for the concrete system via a natural *concretization* procedure, described in [Tab09, Gir10a], which essentially renders the two controlled systems approximately bisimilar. This is the approach used in [PGT08, GPT10, MT10]. However, the controller for the concrete system obtained via this concretization procedure has several drawbacks. Firstly, it is generally a dynamic state-feedback controller (i.e. the controller has a memory) when it is known that for some control specifications such as safety [RW87] or reachability [Ber00], it is sufficient to consider static state-feedback controllers. Secondly, the implementation of this controller requires the encoding of the dynamics of the abstraction which may result in a higher implementation cost. Thirdly, if the abstraction is deterministic, then there is essentially no more feedback: at each step the controller selects the control action and its internal state independently from the state of the concrete system. This may cause some robustness issues in case of unmodeled disturbances or fault occurrences. In the following, we present specific controller concretization procedure for safety and reachability specifications which do not suffer from the previous drawbacks. These techniques are readily applicable using the discrete abstractions mentioned above.

3. CONTROLLER SYNTHESIS FOR SAFETY SPECIFICATIONS

3.1. Problem Formulation. Let $T = (Q, L, \delta, O, H)$ be a transition system, let $O_s \subseteq O$ be a set of outputs associated with safe states. We consider the synthesis problem that consists in determining a controller that keeps the output of the system inside the specified safe set O_s .

Definition 3.1. A controller \mathcal{S} for T is a *safety controller for specification O_s* if, for all non-blocking states q_0 of the controlled system $T_{\mathcal{S}}$ (i.e. $\mathcal{S}(q_0) \neq \emptyset$), for all trajectories of $T_{\mathcal{S}}$ starting from q_0 , $(q_0, l_0), (q_1, l_1), \dots, (q_{N-1}, l_{N-1}), q_N$; for all $i \in \{0, \dots, N\}$, $H(q_i) \in O_s$ and q_N is a non-blocking state of $T_{\mathcal{S}}$ (i.e. $\mathcal{S}(q_N) \neq \emptyset$).

The condition that all trajectories end in a non-blocking state ensures that, starting from a non-blocking state, the controlled system can evolve indefinitely while keeping its output in the safe set O_s . It is easy to verify by induction that an equivalent characterization of safety controllers is given as follows:

Lemma 3.2. A controller \mathcal{S} for T is a safety controller for specification O_s if and only if for all non-blocking states q of the controlled system $T_{\mathcal{S}}$ (i.e. $\mathcal{S}(q) \neq \emptyset$); $H(q) \in O_s$ and for all $q' \in \delta(q, \mathcal{S}(q))$, q' is a non-blocking state of $T_{\mathcal{S}}$ (i.e. $\mathcal{S}(q') \neq \emptyset$).

There are in general several controllers that solve the safety problem. We are usually interested in synthesizing a controller that enables as many actions as possible. This notion of permissivity can be formalized by defining a partial order on controllers.

Definition 3.3. Let \mathcal{S}_1 and \mathcal{S}_2 be two controllers for transition system T , \mathcal{S}_1 is *more permissive* than \mathcal{S}_2 , denoted $\mathcal{S}_2 \preceq \mathcal{S}_1$, if for all $q \in Q$, $\mathcal{S}_2(q) \subseteq \mathcal{S}_1(q)$. The controller \mathcal{S}^* for T is the *maximal safety controller for specification O_s* , if \mathcal{S}^* is a safety controller for specification O_s , and for all safety controllers \mathcal{S} for specification O_s , $\mathcal{S} \preceq \mathcal{S}^*$.

It is well known that the maximal safety controller exists, is unique and can be computed using a fixed point algorithm (see e.g. [RW87, Tab09]). This algorithm is guaranteed to terminate in a finite number of steps provided $H^{-1}(O_s) \subseteq Q$ is a finite set which is often the case for discrete systems. For other transition systems, there is no guarantee that the algorithm will terminate. In this case, a synthesis approach based on approximately bisimilar abstractions can help to compute effectively a safety controller with, in addition, an estimation of the distance to maximality.

3.2. Abstraction-based Controller Synthesis. Let $T_i = (Q_i, L, \delta_i, O, H_i)$, $i = 1, 2$, be metric transition systems such that $T_1 \sim_{\varepsilon} T_2$. Let T_1 be the system that we want to control and T_2 be an approximately bisimilar abstraction of T_1 . We present an approach to safety controller synthesis for specification O_s .

Definition 3.4. Let $O' \subseteq O$ and $\varphi \geq 0$. The φ -contraction of O' is the subset of O defined as follows

$$C_{\varphi}(O') = \{o' \in O' \mid \forall o \in O, d(o, o') \leq \varphi \implies o \in O'\}.$$

The φ -expansion of O' is the subset of O defined as follows

$$E_{\varphi}(O') = \{o \in O \mid \exists o' \in O', d(o, o') \leq \varphi\}.$$

By straightforward applications of the previous definitions, we have:

Lemma 3.5. Let $O' \subseteq O$ and $\varepsilon \geq 0$, then $C_{2\varepsilon}(O') \subseteq C_{\varepsilon}(C_{\varepsilon}(O'))$, $E_{\varepsilon}(E_{\varepsilon}(O')) \subseteq E_{2\varepsilon}(O')$ and $O' \subseteq C_{\varepsilon}(E_{\varepsilon}(O'))$.

We start by synthesizing a safety controller for the abstraction T_2 and the specification $C_\varepsilon(O_s)$. This controller is denoted $\mathcal{S}_{2,C_\varepsilon}$. We shall not discuss further the synthesis of this controller which can be done, if T_2 is discrete, using a fixed point algorithm. The second step of our approach allows us to design a safety controller for system T_1 and specification O_s , obtained from the controller $\mathcal{S}_{2,C_\varepsilon}$ using the following concretization procedure:

Theorem 3.6. *Let $T_1 \sim_\varepsilon T_2$, let $R \subseteq Q_1 \times Q_2$ denote the ε -approximate bisimulation relation between T_1 and T_2 . Let $\mathcal{S}_{2,C_\varepsilon}$ be a safety controller for T_2 and specification $C_\varepsilon(O_s)$. Let us define \mathcal{S}_1 , the controller for T_1 given by*

$$(3.1) \quad \forall q_1 \in Q_1, \mathcal{S}_1(q_1) = \mathcal{S}_{2,C_\varepsilon}(R(q_1)).$$

Then, \mathcal{S}_1 is well-defined and is a safety controller for specification O_s .

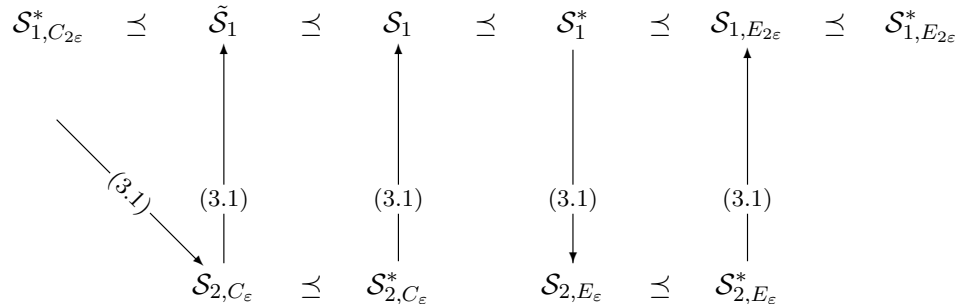
Proof. First, let us show that the controller \mathcal{S}_1 is well-defined. Let $q_1 \in Q_1$, let $l \in \mathcal{S}_1(q_1)$ then, from (3.1), there exists $q_2 \in Q_2$ such that $(q_1, q_2) \in R$ and $l \in \mathcal{S}_{2,C_\varepsilon}(q_2)$. $\mathcal{S}_{2,C_\varepsilon}$ is well-defined, then $l \in \text{Enab}_2(q_2)$, i.e. there exists $q'_2 \in \delta_2(q_2, l)$. By Definition 2.3, it follows that there exists $q'_1 \in \delta_1(q_1, l)$ (such that $(q'_1, q'_2) \in R$), which implies that $l \in \text{Enab}_1(q_1)$. Thus, for all $q_1 \in Q_1$, $\mathcal{S}_1(q_1) \subseteq \text{Enab}_1(q_1)$; \mathcal{S}_1 is well-defined. Let us now prove that \mathcal{S}_1 is a safety controller for the specification O_s . Let $q_1 \in Q_1$ such that $\mathcal{S}_1(q_1) \neq \emptyset$, let $l \in \mathcal{S}_1(q_1)$, by (3.1) there exists $q_2 \in Q_2$ such that $(q_1, q_2) \in R$ and $l \in \mathcal{S}_{2,C_\varepsilon}(q_2)$. Since $\mathcal{S}_{2,C_\varepsilon}$ is a safety controller for specification $C_\varepsilon(O_s)$ and $\mathcal{S}_{2,C_\varepsilon}(q_2) \neq \emptyset$, we have from Lemma 3.2, that $H_2(q_2) \in C_\varepsilon(O_s)$. By Definition 2.3, $d(H_1(q_1), H_2(q_2)) \leq \varepsilon$ and therefore $H_1(q_1) \in O_s$. Now, let $q'_1 \in \delta_1(q_1, l)$, by Definition 2.3 there exists $q'_2 \in \delta_2(q_2, l)$ such that $(q'_1, q'_2) \in R$. Since $\mathcal{S}_{2,C_\varepsilon}$ is a safety controller for specification $C_\varepsilon(O_s)$ and $l \in \mathcal{S}_{2,C_\varepsilon}(q_2)$, we have from Lemma 3.2, that $\mathcal{S}_{2,C_\varepsilon}(q'_2) \neq \emptyset$. Finally, (3.1) implies that $\mathcal{S}_{2,C_\varepsilon}(q'_2) \subseteq \mathcal{S}_1(q'_1)$ and therefore $\mathcal{S}_1(q'_1) \neq \emptyset$. From Lemma 3.2, \mathcal{S}_1 is a safety controller for specification O_s . \square

If we use the maximal safety controller for T_2 , it is desirable to have an estimate of the distance between the controller given by the concretization equation (3.1) and the maximal safety controller for T_1 . This is given by the following result:

Theorem 3.7. *Let $\mathcal{S}_{2,C_\varepsilon}^*$ and $\mathcal{S}_{2,E_\varepsilon}^*$ be the maximal safety controllers for T_2 and specifications $C_\varepsilon(O_s)$ and $E_\varepsilon(O_s)$ respectively. Let \mathcal{S}_1 and $\mathcal{S}_{1,E_{2\varepsilon}}$ be the controllers for T_1 obtained by the concretization equation (3.1) from $\mathcal{S}_{2,C_\varepsilon}^*$ and $\mathcal{S}_{2,E_\varepsilon}^*$ respectively. Let \mathcal{S}_1^* , $\mathcal{S}_{1,C_{2\varepsilon}}^*$ and $\mathcal{S}_{1,E_{2\varepsilon}}^*$ be the maximal safety controllers for T_1 and specifications O_s , $C_{2\varepsilon}(O_s)$ and $E_{2\varepsilon}(O_s)$ respectively. Then,*

$$\mathcal{S}_{1,C_{2\varepsilon}}^* \preceq \mathcal{S}_1 \preceq \mathcal{S}_1^* \preceq \mathcal{S}_{1,E_{2\varepsilon}} \preceq \mathcal{S}_{1,E_{2\varepsilon}}^*.$$

Proof. The proof relies on the introduction of several auxiliary controllers. The relations between these controllers are presented in the following sketch, where arrows correspond to application of the concretization equation (3.1):



Let us go into the details of the proof. From Theorem 3.6, \mathcal{S}_1 is a safety controller for specification O_s , then $\mathcal{S}_1 \preceq \mathcal{S}_1^*$. Let us prove that $\mathcal{S}_{1,C_{2\varepsilon}}^* \preceq \mathcal{S}_1$. Since $\mathcal{S}_{1,C_{2\varepsilon}}^*$ is a safety controller for specification $C_{2\varepsilon}(O_s)$ and from Lemma 3.5, $C_{2\varepsilon}(O_s) \subseteq C_\varepsilon(C_\varepsilon(O_s))$, it is clear that $\mathcal{S}_{1,C_{2\varepsilon}}^*$ is a safety controller for specification $C_\varepsilon(C_\varepsilon(O_s))$. Now, let us define $\mathcal{S}_{2,C_\varepsilon}$, the controller for T_2 such that for $q_2 \in Q_2$, $\mathcal{S}_{2,C_\varepsilon}(q_2) = \mathcal{S}_{1,C_{2\varepsilon}}^*(R^{-1}(q_2))$. The symmetry of approximate bisimulation allows us to reverse the role of T_1 and T_2 in Theorem 3.6. This gives that $\mathcal{S}_{2,C_\varepsilon}$ is a safety controller for T_2 and specification $C_\varepsilon(O_s)$ which yields $\mathcal{S}_{2,C_\varepsilon} \preceq \mathcal{S}_{2,C_\varepsilon}^*$. We now define $\tilde{\mathcal{S}}_1$, the controller for T_1 such that for $q_1 \in Q_1$, $\tilde{\mathcal{S}}_1(q_1) = \mathcal{S}_{2,C_\varepsilon}(R(q_1))$. Then, $\mathcal{S}_{2,C_\varepsilon} \preceq \mathcal{S}_{2,C_\varepsilon}^*$ gives $\tilde{\mathcal{S}}_1 \preceq \mathcal{S}_1$. Finally, we remark that for all $q_1 \in Q_1$, $\tilde{\mathcal{S}}_1(q_1) = \mathcal{S}_{1,C_{2\varepsilon}}^*(R^{-1}(R(q_1)))$ which leads to $\mathcal{S}_{1,C_{2\varepsilon}}^* \preceq \tilde{\mathcal{S}}_1$. Let us show that $\mathcal{S}_1^* \preceq \mathcal{S}_{1,E_{2\varepsilon}}$. Since \mathcal{S}_1^* is a safety controller for O_s and since from Lemma 3.5, $O_s \subseteq C_\varepsilon(E_\varepsilon(O_s))$, it is clear that \mathcal{S}_1^* is a safety controller for specification $C_\varepsilon(E_\varepsilon(O_s))$. Now let us define $\mathcal{S}_{2,E_\varepsilon}$, the controller for T_2 such that for $q_2 \in Q_2$, $\mathcal{S}_{2,E_\varepsilon}(q_2) = \mathcal{S}_1^*(R^{-1}(q_2))$. By reversing the role of T_1 and T_2 in Theorem 3.6, we obtain that $\mathcal{S}_{2,E_\varepsilon}$ is a safety controller for T_2 and specification $E_\varepsilon(O_s)$. Then, $\mathcal{S}_{2,E_\varepsilon} \preceq \mathcal{S}_{2,E_\varepsilon}^*$. Then, for all $q_1 \in Q_1$, $\mathcal{S}_1^*(q_1) \subseteq \mathcal{S}_1^*(R^{-1}(R(q_1))) = \mathcal{S}_{2,E_\varepsilon}(R(q_1)) \subseteq \mathcal{S}_{2,E_\varepsilon}^*(R(q_1)) = \mathcal{S}_{1,E_{2\varepsilon}}(q_1)$. Hence, $\mathcal{S}_1^* \preceq \mathcal{S}_{1,E_{2\varepsilon}}$. Finally, since $\mathcal{S}_{2,E_\varepsilon}^*$ is a safety controller for T_2 and specification $E_\varepsilon(O_s)$ and since by Lemma 3.5, $E_\varepsilon(O_s) \subseteq C_\varepsilon(E_\varepsilon(E_\varepsilon(O_s))) \subseteq C_\varepsilon(E_{2\varepsilon}(O_s))$, it follows that $\mathcal{S}_{2,E_\varepsilon}^*$ is a safety controller for T_2 and specification $C_\varepsilon(E_{2\varepsilon}(O_s))$. Then, from Theorem 3.6, $\mathcal{S}_{1,E_{2\varepsilon}}$ is a safety controller for T_1 and specification $E_{2\varepsilon}(O_s)$ which yields $\mathcal{S}_{1,E_{2\varepsilon}} \preceq \mathcal{S}_{1,E_{2\varepsilon}}^*$. \square

By computing the controllers \mathcal{S}_1 and $\mathcal{S}_{1,E_{2\varepsilon}}$ one is able to give a certified upper-bound on the distance between the controller \mathcal{S}_1 we will use to control T_1 and the maximal safety controller \mathcal{S}_1^* . Moreover, if the safety problem is somehow robust, in the sense that $\mathcal{S}_{1,C_{2\varepsilon}}^*$ and $\mathcal{S}_{1,E_{2\varepsilon}}^*$ approach \mathcal{S}_1^* as ε approaches 0 (i.e. slightly different specifications result in only slightly different maximal controllers); then \mathcal{S}_1 and $\mathcal{S}_{1,E_{2\varepsilon}}$ also approach \mathcal{S}_1^* as ε gets smaller and \mathcal{S}_1^* can be approximated arbitrarily close.

4. CONTROLLER SYNTHESIS FOR REACHABILITY SPECIFICATIONS

4.1. Problem Formulation. Let $T = (Q, L, \delta, O, H)$ be a transition system, let $O_s \subseteq O$ be a set of outputs associated with safe states, let $O_t \subseteq O_s$ be a set of outputs associated with target states. We consider the synthesis problem that consists in determining a controller steering the output of the system to O_t while keeping the output in O_s along the way. In addition, in order to choose among the possible controllers, we try to minimize the time to reach the target. Thus, we consider an optimal control problem. In this section, we assume for simplicity, that T is non-blocking; it would actually be sufficient to assume that all the states of T associated to observations in O_s are non-blocking.

Definition 4.1. Let \mathcal{S} be a controller for T such that for all $q \in Q$, $\mathcal{S}(q) \neq \emptyset$. The *entry time* of $T_{\mathcal{S}}$ from $q_0 \in Q$ for *reachability specification* (O_s, O_t) is the smallest $N \in \mathbb{N}$ such that for all trajectories of the controlled system $T_{\mathcal{S}}$, of length N and starting from q_0 , $(q_0, l_0), (q_1, l_1), \dots, (q_{N-1}, l_{N-1}), q_N$, there exists $K \in \{0, \dots, N\}$ such that for all $k \in \{0, \dots, K\}$, $H(q_k) \in O_s$ and $H(q_K) \in O_t$. The entry time is denoted by $J(T_{\mathcal{S}}, O_s, O_t, q_0)$. If such a $N \in \mathbb{N}$ does not exist, then we define $J(T_{\mathcal{S}}, O_s, O_t, q_0) = +\infty$.

The condition that $\mathcal{S}(q) \neq \emptyset$, for all $q \in Q$, ensures that the controlled system $T_{\mathcal{S}}$ is non-blocking. The states from which the system is guaranteed to reach O_t without leaving O_s are the states with finite entry-time. The following result is quite standard (see e.g. [Ber00]) and is therefore stated without proof:

Lemma 4.2. *The entry time of T_S for reachability specification (O_s, O_t) satisfies:*

- For all $q \in Q \setminus H^{-1}(O_s)$, $J(T_S, O_s, O_t, q) = +\infty$,
- For all $q \in H^{-1}(O_t)$, $J(T_S, O_s, O_t, q) = 0$,
- For all $q \in H^{-1}(O_s) \setminus H^{-1}(O_t)$,

$$(4.1) \quad J(T_S, O_s, O_t, q) = 1 + \max_{q' \in \delta(q, S(q))} J(T_S, O_s, O_t, q').$$

We can now define the notion of time-optimal controller:

Definition 4.3. We say that a controller \mathcal{S}^* for T is *time-optimal for reachability specification* (O_s, O_t) if for all controllers \mathcal{S} , for all $q \in Q$, $J(T_{\mathcal{S}^*}, O_s, O_t, q) \leq J(T_S, O_s, O_t, q)$. The *time-optimal value function for reachability specification* (O_s, O_t) is defined as $J^*(T, O_s, O_t, q) = J(T_{\mathcal{S}^*}, O_s, O_t, q)$.

Solving the time-optimal control problem consists in synthesizing a time-optimal controller. It is well known that a time-optimal controller exists (but may be not unique) and can be computed using dynamic programming [Ber00, Tab09]. The dynamic programming algorithm is guaranteed to terminate in a finite number of steps provided $H^{-1}(O_s) \subseteq Q$ is a finite set which is often the case for discrete systems. Here again, for other systems, there is no guarantee that the algorithm will terminate and an abstraction-based approach may help to compute a sub-optimal controller with an estimation of the distance to optimality.

4.2. Abstraction-based Controller Synthesis. Let $T_i = (Q_i, L, \delta_i, O, H_i)$, $i = 1, 2$, be metric transition systems such that $T_1 \sim_\varepsilon T_2$. Let T_1 be the system that we want to control and T_2 be an approximately bisimilar abstraction of T_1 . We present an approach to controller synthesis for reachability specifications. We first synthesize a controller $\mathcal{S}_{2, C_\varepsilon}$ for the abstraction T_2 and the reachability specification given by the contracted safe set $C_\varepsilon(O_s)$ and target set $C_\varepsilon(O_t)$. If T_2 is discrete, this can be done using dynamic programming. Then, we design a controller for T_1 and reachability specification (O_s, O_t) using the following concretization procedure:

Theorem 4.4. *Let $T_1 \sim_\varepsilon T_2$, let $R \subseteq Q_1 \times Q_2$ denote the ε -approximate bisimulation relation between T_1 and T_2 . Let $\mathcal{S}_{2, C_\varepsilon}$ be a controller for T_2 , let us define \mathcal{S}_1 , the controller for T_1 given by¹*

$$(4.2) \quad \forall q_1 \in Q_1, \mathcal{S}_1(q_1) = \mathcal{S}_{2, C_\varepsilon} \left(\arg \min_{q_2 \in R(q_1)} J(T_2, \mathcal{S}_{2, C_\varepsilon}, C_\varepsilon(O_s), C_\varepsilon(O_t), q_2) \right)$$

where $q_2 \in R(q_1)$ stands for $(q_1, q_2) \in R$. Then, \mathcal{S}_1 is well-defined and for all $q_1 \in Q_1$:

$$(4.3) \quad J(T_1, \mathcal{S}_1, O_s, O_t, q_1) \leq \min_{q_2 \in R(q_1)} J(T_2, \mathcal{S}_{2, C_\varepsilon}, C_\varepsilon(O_s), C_\varepsilon(O_t), q_2).$$

Proof. The fact that \mathcal{S}_1 is well-defined can be shown similarly to the proof of Theorem 3.6. Let us prove (4.3), we denote for all $q_1 \in Q_1$, $\tilde{J}(q_1) = \min_{q_2 \in R(q_1)} J(T_2, \mathcal{S}_{2, C_\varepsilon}, C_\varepsilon(O_s), C_\varepsilon(O_t), q_2)$. If $\tilde{J}(q_1) = 0$, then it means that there exists $q_2 \in R(q_1)$ such that $J(T_2, \mathcal{S}_{2, C_\varepsilon}, C_\varepsilon(O_s), C_\varepsilon(O_t), q_2) = 0$. This implies that $H_2(q_2) \in C_\varepsilon(O_t)$. Since $(q_1, q_2) \in R$, Definition 2.3 gives that $d(H_1(q_1), H_2(q_2)) \leq \varepsilon$. Hence $H_1(q_1) \in O_t$ and $J(T_1, \mathcal{S}_1, O_s, O_t, q_1) = 0$. Thus, if $\tilde{J}(q_1) = 0$, (4.3) holds. We now proceed by induction, let us assume that there exists $k \in \mathbb{N}$ such that for all $q_1 \in Q_1$ such that $\tilde{J}(q_1) \leq k$, (4.3) holds. Let $q_1 \in Q_1$ such that $\tilde{J}(q_1) = k + 1$, let $q_2 \in Q_2$ be given by $q_2 = \arg \min_{p_2 \in R(q_1)} J(T_2, \mathcal{S}_{2, C_\varepsilon}, C_\varepsilon(O_s), C_\varepsilon(O_t), p_2)$, then, $J(T_2, \mathcal{S}_{2, C_\varepsilon}, C_\varepsilon(O_s), C_\varepsilon(O_t), q_2) = k + 1$. This implies $H_2(q_2) \in C_\varepsilon(O_s) \setminus C_\varepsilon(O_t)$. Since $(q_1, q_2) \in R$, $d(H_1(q_1), H_2(q_2)) \leq \varepsilon$ and $H_1(q_1) \in O_s$.

¹If there are several states $q_2 \in R(q_1)$ minimizing $J(T_2, \mathcal{S}_{2, C_\varepsilon}, C_\varepsilon(O_s), C_\varepsilon(O_t), q_2)$ then we just pick one of them.

If $H_1(q_1) \in O_t$ then $J(T_1, \mathcal{S}_1, O_s, O_t, q_1) = 0$ and the induction step is completed. Hence, let us assume $H_1(q_1) \notin O_t$. By (4.2), $\mathcal{S}_1(q_1) = \mathcal{S}_{2, C_\varepsilon}(q_2)$. Let $l \in \mathcal{S}_1(q_1)$, let $q'_1 \in \delta_1(q_1, l)$, since $(q_1, q_2) \in R$, there exists, by Definition 2.3, $q'_2 \in \delta_2(q_2, l)$ such that $(q'_1, q'_2) \in R$. It follows that $\tilde{J}(q'_1) \leq J(T_2, \mathcal{S}_{2, C_\varepsilon}, C_\varepsilon(O_s), C_\varepsilon(O_t), q'_2)$. Since $H_2(q_2) \in C_\varepsilon(O_s) \setminus C_\varepsilon(O_t)$, we have by (4.1), $J(T_2, \mathcal{S}_{2, C_\varepsilon}, C_\varepsilon(O_s), C_\varepsilon(O_t), q'_2) \leq J(T_2, \mathcal{S}_{2, C_\varepsilon}, C_\varepsilon(O_s), C_\varepsilon(O_t), q_2) - 1 = k$. Therefore, $\tilde{J}(q'_1) \leq k$. Then, by the induction hypothesis we get that $J(T_1, \mathcal{S}_1, O_s, O_t, q'_1) \leq \tilde{J}(q'_1) \leq k$. Since this holds for all $l \in \mathcal{S}_1(q_1)$ and all $q'_1 \in \delta_1(q_1, l)$, and since $H_1(q_1) \in O_s \setminus O_t$, we have by (4.1) that $J(T_1, \mathcal{S}_1, O_s, O_t, q_1) \leq k + 1$ which completes the induction step. Thus, we have proved by induction that for all $q_1 \in Q_1$ such that $\tilde{J}(q_1) \in \mathbb{N}$, (4.3) holds. If $\tilde{J}(q_1) = +\infty$, then (4.3) clearly holds as well. \square

The previous theorem gives us a way by equation (4.2) to concretize a controller for abstraction T_2 into a controller for T_1 . Equation (4.3) provides guarantees on the performance of this controller. Particularly, let us remark that the states of T_1, \mathcal{S}_1 from which the control objective is achieved (i.e. the states with finite entry-time) are those related through the approximate bisimulation relation R to states of $T_2, \mathcal{S}_{2, C_\varepsilon}$ with finite entry-time. In addition, if $\mathcal{S}_{2, C_\varepsilon}$ is the time-optimal controller for T_2 and reachability specification $(C_\varepsilon(O_s), C_\varepsilon(O_t))$, the following result gives estimates of the distance to optimality for the controller \mathcal{S}_1 .

Theorem 4.5. *Let $\mathcal{S}_{2, C_\varepsilon}^*$, $\mathcal{S}_{2, E_\varepsilon}^*$ be time-optimal controllers for T_2 and specification $(C_\varepsilon(O_s), C_\varepsilon(O_t))$ and $(E_\varepsilon(O_s), E_\varepsilon(O_t))$ respectively. Let \mathcal{S}_1 be the controller for T_1 obtained from $\mathcal{S}_{2, C_\varepsilon}^*$ by the concretization equation (4.2). Let $\mathcal{S}_{1, E_{2\varepsilon}}$ be the controller for T_1 obtained from $\mathcal{S}_{2, E_\varepsilon}^*$ by*

$$(4.4) \quad \forall q_1 \in Q_1, \mathcal{S}_{1, E_{2\varepsilon}}(q_1) = \mathcal{S}_{2, E_\varepsilon}^* \left(\arg \min_{q_2 \in R(q_1)} J^*(T_2, E_\varepsilon(O_s), E_\varepsilon(O_t), q_2) \right).$$

Then, for all $q_1 \in Q_1$,

$$\begin{aligned} J^*(T_1, E_{2\varepsilon}(O_s), E_{2\varepsilon}(O_t), q_1) &\leq J(T_1, \mathcal{S}_{1, E_{2\varepsilon}}, E_{2\varepsilon}(O_s), E_{2\varepsilon}(O_t), q_1) \leq \\ &J^*(T_1, O_s, O_t, q_1) \leq \\ &J(T_1, \mathcal{S}_1, O_s, O_t, q_1) \leq J^*(T_1, C_{2\varepsilon}(O_s), C_{2\varepsilon}(O_t), q_1). \end{aligned}$$

Proof. The proof essentially follows the same line as the proof of Theorem 3.7. The first and third inequalities are direct consequences of the definition of time-optimal value function. Let us prove the fourth inequality. Let $\mathcal{S}_{1, C_{2\varepsilon}}^*$ be the time-optimal controller for T_1 and reachability specification $(C_{2\varepsilon}(O_s), C_{2\varepsilon}(O_t))$. From Lemma 3.5, we have $C_{2\varepsilon}(O_s) \subseteq C_\varepsilon(C_\varepsilon(O_s))$ and $C_{2\varepsilon}(O_t) \subseteq C_\varepsilon(C_\varepsilon(O_t))$. Then, for all $q_1 \in Q_1$, $J^*(T_1, C_{2\varepsilon}(O_s), C_{2\varepsilon}(O_t), q_1) \geq J(T_1, \mathcal{S}_{1, C_{2\varepsilon}}^*, C_\varepsilon(C_\varepsilon(O_s)), C_\varepsilon(C_\varepsilon(O_t)), q_1)$. Now let us define the controller for T_2 , $\mathcal{S}_{2, C_\varepsilon}$, given, for all $q_2 \in Q_2$, by

$$\mathcal{S}_{2, C_\varepsilon}(q_2) = \mathcal{S}_{1, C_{2\varepsilon}}^* \left(\arg \min_{q_1 \in R^{-1}(q_2)} J(T_1, \mathcal{S}_{1, C_{2\varepsilon}}^*, C_\varepsilon(C_\varepsilon(O_s)), C_\varepsilon(C_\varepsilon(O_t)), q_1) \right).$$

By reversing the role of T_1 and T_2 in Theorem 4.4, it follows that for all $q_2 \in Q_2$,

$$\begin{aligned} J(T_2, \mathcal{S}_{2, C_\varepsilon}, C_\varepsilon(O_s), C_\varepsilon(O_t), q_2) &\leq \min_{q_1 \in R^{-1}(q_2)} J(T_1, \mathcal{S}_{1, C_{2\varepsilon}}^*, C_\varepsilon(C_\varepsilon(O_s)), C_\varepsilon(C_\varepsilon(O_t)), q_1) \\ &\leq \min_{q_1 \in R^{-1}(q_2)} J^*(T_1, C_{2\varepsilon}(O_s), C_{2\varepsilon}(O_t), q_1). \end{aligned}$$

Moreover, for all $q_2 \in Q_2$, $J(T_2, \mathcal{S}_{2, C_\varepsilon}^*, C_\varepsilon(O_s), C_\varepsilon(O_t), q_2) \leq J(T_2, \mathcal{S}_{2, C_\varepsilon}, C_\varepsilon(O_s), C_\varepsilon(O_t), q_2)$ which gives together with Theorem 4.4, for all $q_1 \in Q_1$,

$$\begin{aligned} J(T_1, \mathcal{S}_1, O_s, O_t, q_1) &\leq \min_{q_2 \in R(q_1)} J(T_2, \mathcal{S}_{2, C_\varepsilon}^*, C_\varepsilon(O_s), C_\varepsilon(O_t), q_2) \\ &\leq \min_{q_2 \in R(q_1)} \min_{p_1 \in R^{-1}(q_2)} J^*(T_1, C_{2\varepsilon}(O_s), C_{2\varepsilon}(O_t), p_1) \leq J^*(T_1, C_{2\varepsilon}(O_s), C_{2\varepsilon}(O_t), q_1). \end{aligned}$$

Let us now prove the second inequality. From Lemma 3.5, $O_s \subseteq C_\varepsilon(E_\varepsilon(O_s))$ and $O_t \subseteq C_\varepsilon(E_\varepsilon(O_t))$. Then, for all $q_1 \in Q_1$, $J^*(T_1, O_s, O_t, q_1) \geq J(T_1, \mathcal{S}_1^*, C_\varepsilon(E_\varepsilon(O_s)), C_\varepsilon(E_\varepsilon(O_t)), q_1)$. Now let us define $\mathcal{S}_{2, E_\varepsilon}$, the controller for T_2 given for all $q_2 \in Q_2$ by

$$\mathcal{S}_{2, E_\varepsilon}(q_2) = \mathcal{S}_1^* \left(\arg \min_{q_1 \in R^{-1}(q_2)} J(T_1, \mathcal{S}_1^*, C_\varepsilon(E_\varepsilon(O_s)), C_\varepsilon(E_\varepsilon(O_t)), q_1) \right).$$

By reversing the role of T_1 and T_2 in Theorem 4.4, it follows that for all $q_2 \in Q_2$,

$$\begin{aligned} J(T_2, \mathcal{S}_{2, E_\varepsilon}, E_\varepsilon(O_s), E_\varepsilon(O_t), q_2) &\leq \min_{q_1 \in R^{-1}(q_2)} J(T_1, \mathcal{S}_1^*, C_\varepsilon(E_\varepsilon(O_s)), C_\varepsilon(E_\varepsilon(O_t)), q_1) \\ &\leq \min_{q_1 \in R^{-1}(q_2)} J^*(T_1, O_s, O_t, q_1). \end{aligned}$$

Using the fact that from Lemma 3.5, $E_\varepsilon(O_s) \subseteq C_\varepsilon(E_{2\varepsilon}(O_s))$ and $E_\varepsilon(O_t) \subseteq C_\varepsilon(E_{2\varepsilon}(O_t))$, we can show similar to Theorem 4.4 that for all $q_1 \in Q_1$

$$\begin{aligned} J(T_1, \mathcal{S}_{1, E_{2\varepsilon}}, E_{2\varepsilon}(O_s), E_{2\varepsilon}(O_t), q_1) &\leq \min_{q_2 \in R(q_1)} J^*(T_2, E_\varepsilon(O_s), E_\varepsilon(O_t), q_2) \\ &\leq \min_{q_2 \in R(q_1)} J(T_2, \mathcal{S}_{2, E_\varepsilon}, E_\varepsilon(O_s), E_\varepsilon(O_t), q_2) \\ &\leq \min_{q_2 \in R(q_1)} \min_{p_1 \in R^{-1}(q_2)} J^*(T_1, O_s, O_t, p_1) \leq J^*(T_1, O_s, O_t, q_1). \end{aligned}$$

□

By computing the controllers \mathcal{S}_1 and $\mathcal{S}_{1, E_{2\varepsilon}}$ one is able to give a certified upper-bound on the distance to optimality of the controller \mathcal{S}_1 we will use to control T_1 . Moreover, if the reachability problem is robust (i.e. the time-optimal value function depends continuously on the specification); then $J^*(T_1, O_s, O_t, q_1)$ can be approximated arbitrarily close. We would like to highlight the differences of our approach with [MT10] where time-optimal reachability controllers are synthesized using discrete abstractions related by alternating simulations. These are weaker assumptions than those considered in the present work. In [MT10], an approach to compute guaranteed upper and lower bounds of the value function is given. However, contrarily to our approach there is no clue that these lower and upper bounds can approach the true time-optimal value function arbitrarily close. Also, in [MT10], the controllers are refined via the natural concretization procedure that suffers from the drawbacks described in Section 2 whereas our approach does not.

5. APPLICATION TO SWITCHING CONTROLLER DESIGN

In this section, we present an effective approach to switching controller design based on the synthesis approaches introduced in this paper in combination with the approximately bisimilar discrete abstractions of switched systems developed in [GPT10].

Definition 5.1. A *switched system* is a triple $\Sigma = (\mathbb{R}^n, P, F)$, where \mathbb{R}^n is the state space; $P = \{1, \dots, m\}$ is the finite set of modes; $F = \{f_1, \dots, f_m\}$ is a collection of vector fields indexed by P .

Given a switched system $\Sigma = (\mathbb{R}^n, P, \mathcal{P}, F)$ and a parameter $\tau > 0$, we define a transition system $T_\tau(\Sigma)$ that describes trajectories of duration τ of Σ . This can be seen as a time sampling process. This is natural when the switching in Σ is determined by a time-triggered controller with period τ . Formally, $T_\tau(\Sigma) = (Q_1, L, \delta_1, O, H_1)$ where the set of states is $Q_1 = \mathbb{R}^n$; the set of actions is $L = P$; the transition relation is given by $x' \in \delta_1(x, p)$ if and only if the solution of the differential equation $\dot{\mathbf{x}}(t) = f_p(\mathbf{x}(t))$ with $\mathbf{x}(0) = x$ satisfies $\mathbf{x}(\tau) = x'$; 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 observations $O = \mathbb{R}^n$ is equipped with the metric $d(x, x') = \|x - x'\|$ where $\|\cdot\|$ is the usual Euclidean norm. The existence of approximately bisimilar discrete abstractions of $T_\tau(\Sigma)$ is related to the notion of incremental stability [Ang02]. Under this assumption, it is possible to compute approximately bisimilar discrete abstractions of arbitrary precision for $T_\tau(\Sigma)$ based on a gridding of the state-space. Moreover, the approximate bisimulation relation can be fully characterized by a Lyapunov function proving incremental stability of switched system Σ (see [GPT10] for details).

Controller synthesis for switched systems with safety or reachability specifications can be tackled by direct application of fixed-point computation or dynamic programming using guaranteed over-approximations [ABD⁺00] or convergent approximations [MT00] of reachable sets. In the first case, the synthesized controllers are “correct by design” but there is no guarantee that the synthesis algorithm will terminate. In the second case, we can only prove that the synthesized controllers are “correct in the limit” in the sense that correct controllers can be approximated arbitrarily close. The approach described in this paper does not have these problems but only applies to incrementally stable systems. 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 [GPT10] for numerical values). It can be shown that it is incrementally stable and thus approximately bisimilar discrete abstractions can be computed.

We first consider the problem of regulating the state of the DC-DC converter around a desired nominal state. This can be done for instance by synthesizing a controller that keeps the state of the switched system in a set centered around the nominal state. This is a safety specification. In the following, we consider the specification given by the set $O_s = [1.1, 1.6] \times [5.4, 5.9]$. We use a time sampling parameter $\tau = 1$ and choose to work with a discrete abstraction that is approximately bisimilar to $T_\tau(\Sigma)$ with precision $\varepsilon = 0.05$. We compute a safety controller for the switched system $T_\tau(\Sigma)$ by the approach described in Section 3.2. The discrete abstraction has a finite number of states inside $H_2^{-1}(C_\varepsilon(O_s))$ (actually 169744). The fixed point algorithm for the synthesis of the maximal safety controller for the abstraction and specification $C_\varepsilon(O_s)$ terminates in 2 iterations. The resulting safety controller \mathcal{S}_1 for the switched system $T_\tau(\Sigma)$ and the specification O_s is shown on the left part of Figure 1 where we have represented a trajectory of the system where the switching is controlled using a lazy implementation of the controller \mathcal{S}_1 : when the controller has the choice between mode 1 and 2, it keeps the current mode active. We can check that the specification is effectively met. We also compute the upper-bound of the maximal safety controller \mathcal{S}_1^* for switched system $T_\tau(\Sigma)$ and specification O_s , given by Theorem 3.7. The abstraction has 383161 states inside $H_2^{-1}(E_\varepsilon(O_s))$ and the fixed point algorithm for computing the maximal safety controller terminates in 4 iterations. The resulting controller $\mathcal{S}_{1, E_{2\varepsilon}}$, shown on the right side of Figure 1, is an upper-bound of the maximal safety controller \mathcal{S}_1^* for $T_\tau(\Sigma)$ and specification O_s .

We now consider the problem of steering in minimal time the state of the DC-DC converter in the desired region of operation while respecting some safety constraints. This is a time-optimal control problem. We consider the specification given by the safe set $O_s = [0.65, 1.65] \times [4.95, 5.95]$ and the target set $O_t = [1.1, 1.6] \times [5.4, 5.9]$. This time, we use a time sampling parameter $\tau = 0.5$ and

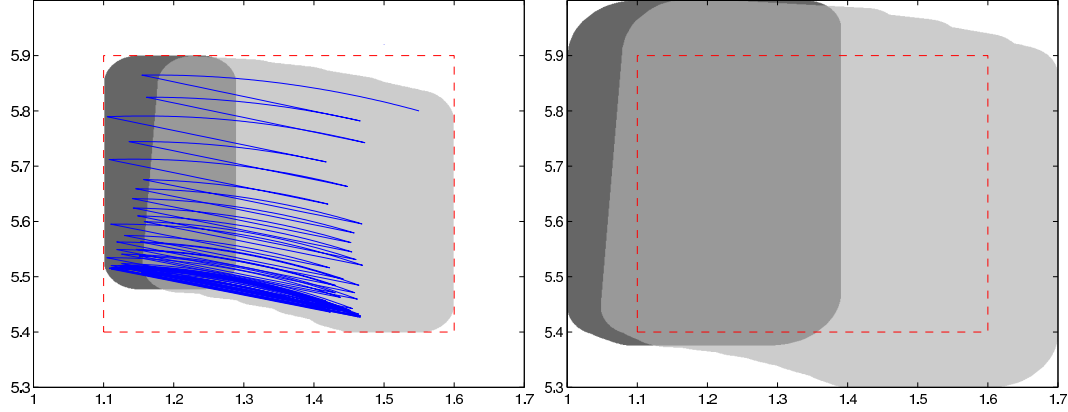


FIGURE 1. Safety controller \mathcal{S}_1 for the switched system $T_\tau(\Sigma)$ and specification O_s with controlled trajectory (left); Safety controller $\mathcal{S}_{1,E_{2\epsilon}}$ for the switched system $T_\tau(\Sigma)$ and specification $E_{2\epsilon}(O_s)$ (right); dark gray: mode 1, light gray: mode 2, medium gray: both modes are acceptable, white: no action is allowed. The maximal safety controller \mathcal{S}_1^* for $T_\tau(\Sigma)$ and specification O_s satisfies $\mathcal{S}_1 \preceq \mathcal{S}_1^* \preceq \mathcal{S}_{1,E_{2\epsilon}}$.

choose to work with a discrete abstraction that is approximately bisimilar to $T_\tau(\Sigma)$ with precision $\varepsilon = 0.1$. We compute a suboptimal reachability controller for the switched system $T_\tau(\Sigma)$ by the approach described in Section 4.2. The discrete abstraction has a finite number of states inside $H_2^{-1}(C_\varepsilon(O_s))$ (actually 674041). The dynamic programming algorithm for the synthesis of the time-optimal controller for the abstraction and reachability specification $(C_\varepsilon(O_s), C_\varepsilon(O_t))$ terminates in 94 iterations. The resulting suboptimal controller \mathcal{S}_1 for the switched system $T_\tau(\Sigma)$ for the reachability specification (O_s, O_t) is shown on Figure 2 where we have also represented trajectories of the system where the switching is controlled using the synthesized controller. We can check that the specification is effectively met. The entry time associated to \mathcal{S}_1 , $J(T_\tau(\Sigma)_{\mathcal{S}_1}, O_s, O_t, q_1)$ shown on the left part of Figure 3, gives an upper-bound of the time-optimal value function. We also compute the lower-bound of the time-optimal value function given by Theorem 4.5. The abstraction has 1520289 states inside $H_2^{-1}(E_\varepsilon(O_s))$ and the fixed point algorithm for computing the maximal safety controller terminates in 66 iterations. The resulting controller $\mathcal{S}_{1,E_{2\epsilon}}$ with entry time $J(T_\tau(\Sigma)_{\mathcal{S}_{1,E_{2\epsilon}}}, E_{2\epsilon}(O_s), E_{2\epsilon}(O_t), q_1)$ provides a lower-bound of the time-optimal value function $J^*(T_\tau(\Sigma), O_s, O_t, q_1)$. This lower-bound is shown on the right side of Figure 3.

6. CONCLUSION

In this paper, we proposed a methodology, based on the use of approximately bisimilar discrete abstractions, for effective computation of controllers for safety and reachability specifications. We provided guarantees of performances of the resulting controllers by giving estimates of the distance of the synthesized controller to the maximal (i.e the most permissive) safety controller or to the time-optimal reachability controller. We showed the effectiveness of our approach by synthesizing controllers for a switched system. Let us remark that the techniques presented in the paper are independent of the type of abstractions considered as long as these are approximately bisimilar. Future work will deal with the development of similar approaches to handle different optimal control problems and richer specifications given e.g. in temporal logic.

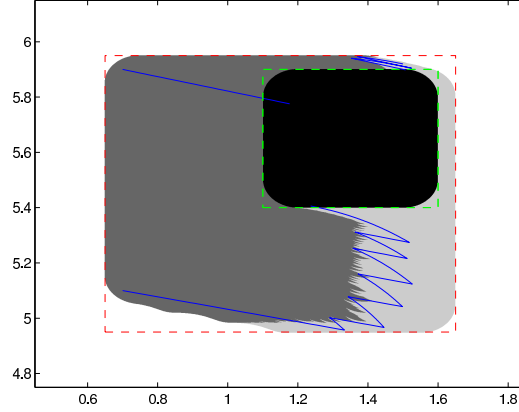


FIGURE 2. Suboptimal controller \mathcal{S}_1 for the switched system $T_\tau(\Sigma)$ and reachability specification (O_s, O_t) and trajectories of the controlled switched system.

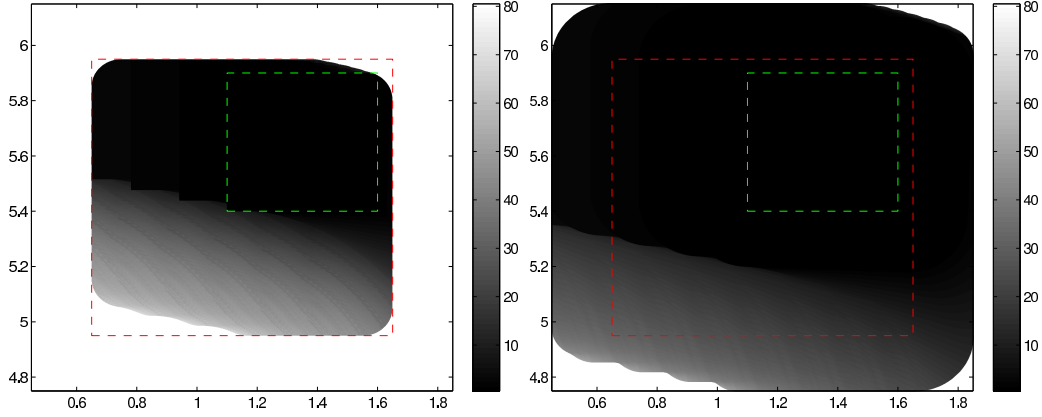


FIGURE 3. Entry-time $J(T_\tau(\Sigma)_{\mathcal{S}_1}, O_s, O_t, q_1)$ for the controller \mathcal{S}_1 shown in Figure 2 (left); Entry-time $(T_\tau(\Sigma)_{\mathcal{S}_1, E_{2\varepsilon}}, E_{2\varepsilon}(O_s), E_{2\varepsilon}(O_t), q_1)$ (right); The time-optimal value function for the switched system $T_\tau(\Sigma)$ and reachability specification (O_s, O_t) satisfies $J(T_\tau(\Sigma)_{\mathcal{S}_1, E_{2\varepsilon}}, E_{2\varepsilon}(O_s), E_{2\varepsilon}(O_t), q_1) \leq J^*(T_\tau(\Sigma), O_s, O_t, q_1) \leq J(T_\tau(\Sigma)_{\mathcal{S}_1}, O_s, O_t, q_1)$.

Acknowledgments. The author would like to thank Gunther Reißig for his valuable comments on an earlier version of this paper.

REFERENCES

- [ABD⁺00] E. Asarin, O. Bournez, T. Dang, O. Maler, and A. Pnueli. Effective synthesis of switching controllers for linear systems. *Proc. IEEE*, 88(7):1011–1025, 2000.
- [AHLP00] R. Alur, T. Henzinger, G. Lafferriere, and G. J. Pappas. Discrete abstractions of hybrid systems. *Proc. of the IEEE*, 88(2):971–984, 2000.
- [Ang02] D. Angeli. A Lyapunov approach to incremental stability properties. *IEEE Trans. on Automatic Control*, 47(3):410–421, March 2002.
- [AVW03] A. Arnold, A. Vincent, and I. Walukiewicz. Games for synthesis of controllers with partial observation. *Th. Comp. Sc.*, 28(1):7–34, 2003.
- [Ber00] D. P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 2000.

- [Gir10a] A. Girard. Synthesis using approximately bisimilar abstractions: state-feedback controllers for safety specifications. In *Hybrid Systems: Computation and Control*, pages 111–120, 2010.
- [Gir10b] Antoine Girard. Synthesis using approximately bisimilar abstractions: Time-optimal control problems. In *IEEE Conf. Decision and Control*, 2010.
- [GP07] A. Girard and G. J. Pappas. Approximation metrics for discrete and continuous systems. *IEEE Trans. on Automatic Control*, 52(5):782–798, 2007.
- [GPT10] 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.
- [HvS01] L. C. G. J. M. Habets and J. H. van Schuppen. Control of piecewise-linear hybrid systems on simplices and rectangles. In *Hybrid Systems: Computation and Control*, volume 2034 of *LNCS*, pages 261–274. Springer, 2001.
- [KB06] M. Kloetzer and C. Belta. A fully automated framework for control of linear systems from ltl specifications. In *Hybrid Systems: Computation and Control*, volume 3927 of *LNCS*, pages 333–347. Springer, 2006.
- [Mil89] R. Milner. *Communication and Concurrency*. Prentice Hall, 1989.
- [MR99] T. Moor and J. Raisch. Supervisory control of hybrid systems within a behavioral framework. *Systems and Control Letters*, 38(3):157–166, 1999.
- [MT00] I. Mitchell and C. Tomlin. Level set methods for computation in hybrid systems. In *Hybrid Systems: Computation and Control*, volume 1790 of *LNCS*, pages 310–323. Springer, 2000.
- [MT10] M. Mazo Jr. and P. Tabuada. Approximate time-optimal control via approximate alternating simulations. In *American Control Conference*, pages 10201–10206, 2010.
- [PGT08] G. Pola, A. Girard, and P. Tabuada. Approximately bisimilar symbolic models for nonlinear control systems. *Automatica*, 44(10):2508–2516, 2008.
- [Rei09] G. Reißig. Computation of discrete abstractions of arbitrary memory span for nonlinear sampled systems. In *Hybrid Systems: Computation and Control*, volume 5469 of *LNCS*, pages 306–320. Springer, 2009.
- [RO98] J. Raisch and S. O’Young. Discrete approximation and supervisory control of continuous systems. *IEEE Trans. on Automatic Control*, 43(4):569–573, 1998.
- [RW87] P. J. Ramadge and W. M. Wonham. Modular feedback logic for discrete event systems. *SIAM J. on Con. and Opt.*, 25(5):1202–1218, 1987.
- [Tab09] P. Tabuada. *Verification and Control of Hybrid Systems - A Symbolic Approach*. Springer, 2009.
- [TI08] Y. Tazaki and J. I. Imura. Finite abstractions of discrete-time linear systems and its application to optimal control. In *17th IFAC World Congress*, pages 10201–10206, 2008.
- [TP06] P. Tabuada and G. J. Pappas. Linear time logic control of discrete-time linear systems. *IEEE Trans. on Automatic Control*, 51(12):1862–1877, 2006.

LABORATOIRE JEAN KUNTZMANN, UNIVERSITÉ DE GRENOBLE, B.P. 53, 38041 GRENOBLE, FRANCE

E-mail address: antoine.girard@imag.fr