Logarithmic Quantization based Symbolic Abstractions for Nonlinear Control Systems

Wei Ren, and Dimos V. Dimarogonas

***Abstract*— This paper studies symbolic abstractions for nonlinear control systems using logarithmic quantization. With a logarithmic quantizer, we approximate the state and input sets, and then construct a novel discrete abstraction for nonlinear control systems. A feedback refinement relation between the constructed discrete abstraction and the original system is established. Using the constructed discrete abstraction, the safety controller synthesis problem is studied. With the discrete abstraction and the abstract specification, the existence of a safety controller is investigated, and the algorithm is proposed to compute the abstract controller. Finally, a numerical example is given to illustrate the obtained results.**

I. Introduction

The use of discrete abstractions [1], [2] has gradually become a standard approach for the design of hybrid systems due to the following two main advantages. First, thanks to discrete abstractions of continuous dynamics, one can deal with controller synthesis problems efficiently via techniques developed in the fields of supervisory control [3] or algorithmic game theory [4]. Second, with an inclusion or equivalence relationship between the original system and the discrete abstraction, the synthesized controller is guaranteed to be correct by design, and thus formal verification is either not needed or can be reduced [5]. To construct the discrete abstraction, the key is to find an equivalence relation on the state space of dynamic systems. Such an equivalence relation leads to a new system, which is on the quotient space and shares the interested properties with the original system.

In the literature on the construction of the discrete abstraction, the most commonly-used approach is based on (alternating) (bi-)simulation relations and their approximate variants in [6], [7]. The simulation relation and related concepts capture equivalences of dynamic systems in an exact or approximate setting. However, this type of relations results in the requirement of exact information of the original system to obtain the refined controller, and a huge computational complexity for the abstract controller due to its abstraction refinement. As a result, the feedback refinement relation was proposed in [8], and provides an alternative to connect the discrete abstraction and the original system. With a feedback refinement relation, the abstract controller can be connected to the original system via a static quantizer [9]. Some salient results can be found; see [10], [11].

This work was supported by the H2020 ERC Starting Grant BUCOPH-SYS, the EU H2020 Co4Robots Project, the Swedish Foundation for Strategic Research (SSF), the Swedish Research Council (VR) and the Knut och Alice Wallenberg Foundation (KAW).

W. Ren and D. Dimarogonas are with Division of Decision and Control Systems, EECS, KTH Royal Institute of Technology, SE-10044, Stockholm, Sweden. Email: weire@kth.se, dimos@kth.se.

On the other hand, due to time-invariant quantization regions and the resulting simple structures [12], a static quantizer is applied in the construction of discrete abstractions [6], [7]. The uniform quantizer, which is a static quantizer with uniform time or space partitions [13], is commonly used in approximations of both the state and input sets. Since the uniform quantization partitions the state set with equal distance, a huge computational complexity may be needed to compute the discrete abstraction [6], [7]. To reduce the computational complexity, a coarse quantizer [14]–[16] can be instead applied such that the state or input space can be partitioned with different distance, and this is the main motivation of this paper. Using the coarse quantizers like logarithmic quantizer and hysteresis quantizer, the approximate bisimulation is not valid any more, and thus the feedback refinement relation is applied.

In this paper, we study the discrete abstraction and controller synthesis of nonlinear control systems via logarithmic quantization. Using the logarithmic quantizer of [14], which is a coarse quantizer, the state and input sets are approximated, and then a novel discrete abstraction is constructed. According to the constructed discrete abstraction, the safety controller synthesis is studied via abstract specification, which is obtained via the logarithmic quantizer. A numerical example is given to demonstrate the obtained results. The main contributions of this paper are three-fold. To begin with, logarithmic quantization based discrete abstraction is first proposed, which provides an alternative approach to approximate the state and input sets. In addition, since the logarithmic quantization is coarser than the uniform quantization, the computational complexity of the obtained discrete abstraction is reduced greatly. Second, using the feedback refinement relation proposed in [9], abstract specification is constructed via logarithmic quantization, and further used in controller synthesis. Third, using the obtained abstraction and the abstract specification, the safety controller synthesis is investigated for the original system, and an algorithm is proposed to construct the safety controller.

II. Nonlinear Control Systems

A. Notations

We denote $\mathbb{R} := (-\infty, +\infty)$; $\mathbb{R}_0^+ := [0, +\infty)$; $\mathbb{R}^+ := (0, +\infty)$; $\mathbb{N} := \{0, 1, \ldots\}$; $\mathbb{N}^+ := \{1, 2, \ldots\}$. $\|\cdot\|$ represents the infinite vector norm. Given $a, b \in \mathbb{R} \cup \{\pm\infty\}$ with $a \leq b$, we denote by $[a, b]$ a closed interval. Given $a, b \in (\mathbb{R} \cup \{\pm\infty\})^n$, we define the relations $<, >, \leq, \geq$ on a, b component-wise. Given $x \in \mathbb{R}^n$, x_i denotes the i-th element of x, $|x_i|$ denotes the absolute value of x_i,

and $|x| = (|x_i|, \ldots, |x_n|)$. A cell $[\![a, b]\!]$ is the closed set $\{x \in \mathbb{R}^n | a_i \leq x_i \leq b_i\}$. Given two sets $A, B \subset \mathbb{R}^n$ with $A \subseteq B$, denote by $\mathrm{Id}_A : A \hookrightarrow B$ the natural inclusion map from $a \in A$ to $\mathrm{Id}(a) = a \in B$. A relation $\mathcal{R} \subset A \times B$ with the map $\mathcal{R} : A \to 2^B$ defined by $b \in \mathcal{R}(a)$ if and only if $(a, b) \in \mathcal{R}$. $\mathcal{R}^{-1}$ denotes the inverse relation of $\mathcal{R}$, i.e. $\mathcal{R}^{-1} := \{(b, a) \in B \times A : (a, b) \in \mathcal{R}\}$. Given a set A, $A^{[0,t)}$ denotes the set of all the signals, which take values in A and are defined on intervals of the form $[0, t)$; $A^{\infty} = \bigcup_{t \in \mathbb{N}} A^{[0,t)}$.

B. Nonlinear Control Systems

The class of nonlinear control systems considered in this paper is introduced in the following definition.

Definition 1 ([6]): A *control system* Σ is a quadruple $\Sigma = (\mathbb{R}^n, U, \mathcal{U}, f)$, where,

- $\mathbb{R}^n$ is the state set;
- $U \subseteq \mathbb{R}^m$ is the input set;
- $\mathcal{U}$ is a subset of all piecewise continuous functions from the interval $(a, b) \subset \mathbb{R}$ to U, with $a < 0, b > 0$;
- $f : \mathbb{R}^n \times U \to \mathbb{R}^n$ is a continuous map satisfying the following Lipschitz assumption: there exists a constant $L \in \mathbb{R}^+$ such that for all $x, y \in \mathbb{R}^n$ and all $u \in U$, we have $\|f(x, u) - f(y, u)\| \leq L\|x - y\|$.

A curve $\xi : (a, b) \to \mathbb{R}^n$ is said to be a *trajectory* of Σ, if there exists $u \in U$ such that $\dot{\xi}(t) = f(\xi(t), u(t))$ for almost all $t \in (a, b)$. Different from the trajectory defined above over the open domain, we refer to the trajectory $\mathbf{x} : [0, \tau] \to \mathbb{R}^n$ defined on a closed domain $[0, \tau]$ with $\tau \in \mathbb{R}^+$ such that $\mathbf{x} = \xi|_{[0,\tau]}$. Denote by $\mathbf{x}(t, x, u)$ the point reached at time $t \in (a, b)$ under the input u from the initial condition x. Such a point is uniquely determined, since the assumptions on f ensure the existence and uniqueness of the trajectory.

III. Feedback Refinement Relation

In this section, we introduce the notion of feedback refinement relation upon which the following results rely. To begin with, the class of transition systems is introduced.

Definition 2 ([5]): A *transition system* is a sextuple $T = (X, X^0, U, \Delta, Y, H)$, comprising of: (i) a set of states X; (ii) a set of initial states X^0; (iii) a set of inputs U; (iv) a transition relation $\Delta : X \times U \times X$; (v) an output set Y; (vi) an output map $H : X \to Y$.

The transition $(x, u, x') \in \Delta$ is denoted by $x' \in \Delta(x, u)$, which means that the system can evolve from x to x' under the input u. An input $u \in U$ is said to belong to *the set of enabled inputs* at $x \in X$, denoted by $\mathrm{enab}(x)$, if $\Delta(x, u) \neq \varnothing$. If $\mathrm{enab}(x) = \varnothing$, then $x \in X$ is said to be *blocking*; otherwise, it is said to be *non-blocking*. If all the states are non-blocking, the system T is called to be *non-blocking*.

Similar to approximate simulation relations and their variants in [5], [6], a feedback refinement relation between two transition systems T_1 and T_2 is introduce as follows.

Definition 3 ([9]): Let $T_i = (X_i, X_i^0, U_i, \Delta_i, Y_i, H_i)$ be two transition systems with $i \in \{1, 2\}$, and assume that $U_2 \subseteq U_1$. A relation $\mathcal{F} \subseteq X_1 \times X_2$ is a *feedback refinement relation* from T_1 to T_2, if for all $(x_1, x_2) \in \mathcal{F}$, (i) $U_2(x_2) \subseteq U_1(x_1)$; (ii) $u \in U_2(x_2), x_1' = \Delta_1(x_1, u) \Rightarrow \mathcal{F}(x_1') \subseteq \Delta_2(x_2, u)$, where $U_i(x) := \{u \in U_i : \Delta_i(x, u) \neq \varnothing\}$. Denote $T_1 \preceq_{\mathcal{F}} T_2$ if $\mathcal{F}$ is a feedback refinement relation from T_1 to T_2.

IV. Symbolic Model

In this section, we work with the time-discretization of the control system Σ. Assume the sampling period is $\tau > 0$, which is a design parameter. We define the time-discretization of the control system Σ as the transition system $T_\tau(\Sigma) := (X_1, X_1^0, U_1, \Delta_1, Y_1, H_1)$, where,

- the set of states is $X_1 := \mathbb{R}^n$;
- the set of initial states is $X_1^0 := \mathbb{R}^n$;
- the set of inputs is $U_1 := \{u \in \mathcal{U} | \mathbf{x}(t, x, u)$ is defined for all $x \in \mathbb{R}^n\}$;
- the transition relation is given as follows: for $x \in X_1$ and $u \in U_1$, $x' = \Delta_1(x, u)$ if and only if $x' = \mathbf{x}(\tau, x, u)$;
- the set of outputs is $Y_1 := \mathbb{R}^n$;
- the output map is $H : X_1 \hookrightarrow X_1$.

A. Logarithmic Quantization based Approximation

To construct a discrete abstraction of a control system, the state and input sets need to be approximated first. To reduce the computational complexity, the following logarithmic quantizer is applied, which provides an alternative for the approximation of the state and input sets.

Definition 4 ([14]–[16]): A quantizer is called a *logarithmic quantizer*, if it has the following form

$$Q(z) := \begin{cases} z_i, & (1+\eta)^{-1} z_i < z \leq (1-\eta)^{-1} z_i; \\ 0, & 0 \leq z \leq (1+\eta)^{-1} d; \\ -Q(-z), & z < 0, \end{cases} \tag{1}$$

where $z_i = \rho^{(1-i)} d$, $\rho = \frac{1-\eta}{1+\eta}$, $\eta \in (0, 1)$, $d > 0$, and $i \in \mathbb{N}^+$.

In Definition 4, the parameter $\rho \in (0, 1)$ is called the quantization density and $z_{\min} := (1+\eta)^{-1} d$ is the size of the deadzone. For a quantized measurement $z_i > 0$, the quantization region is $\hat{z}_i := ((1+\eta)^{-1} z_i, (1-\eta)^{-1} z_i]$. The quantization error $z - Q(z)$ can be written as (see [17])

$$z - Q(z) := \Lambda(z) z, \quad \Lambda(z) \in [-\eta, \eta]. \tag{2}$$

Using the logarithmic quantizer (1), the state set $\mathbb{R}^n$ is approximated by the sequence of embedded lattices $[\mathbb{R}^n]_\eta$:

$$[\mathbb{R}^n]_\eta := \Big\{ q \in \mathbb{R}^n : q_i = \pm \frac{\rho^{(1-k_i)} d}{\sqrt{n}}, k_i \in \mathbb{N}^+, \\ i \in \{1, \ldots, n\} \Big\} \cup \{0\},$$

where, $\rho = (1+\eta)^{-1}(1-\eta)$, $\eta \in (0, 1)$ is treated as a state space parameter, and $d > 0$ is a fixed constant. We associate a quantizer $Q_\eta : \mathbb{R}^n \to [\mathbb{R}^n]_\eta$ such that $Q_\eta(x) = Q(x)$ if and only if for $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$ and $i \in \{1, \ldots, n\}$,

$$(\sqrt{n}(1+\eta))^{-1} |q_i| \leq |x_i| \leq (\sqrt{n}(1-\eta))^{-1} |q_i|,$$

or

$$-(\sqrt{n}(1+\eta))^{-1} d \leq x_i \leq (\sqrt{n}(1+\eta))^{-1} d.$$

As a result, from (2) and simple geometrical considerations, $\|x-Q_\eta(x)\| \leq \Lambda(x)\|x\|$ holds for all $x \in \mathbb{R}^n$, where $\Lambda(x) \in [-\eta, \eta]$. With the quantizer Q_η, the state set is partitioned as

$$\hat{X} := \bigcup_{q \in [\mathbb{R}^n]_\eta} \hat{q},$$

where $\hat{q}$ is the quantization region corresponding to the quantized measurement $q \in [\mathbb{R}^n]_\eta$.

In the following, the approximation of the input set U_1 of $T_\tau(\Sigma)$ is presented; see also [6] for a similar mechanism. We approximate U_1 by means of the set:

$$U_2 := \bigcup_{q \in [\mathbb{R}^n]_\eta} U_2(\hat{q}), \tag{3}$$

where $U_2(\hat{q})$ captures the set of inputs that can be applied at the symbolic state $\hat{q} \in \hat{X}$. $U_2(\hat{q})$ is defined based on the reachable sets. Starting from a state $q \in [\mathbb{R}^n]_\eta$ (thus $q \in X_1$), the set of reachable states of $T_\tau(\Sigma)$ is obtained below.

$$\mathfrak{R}(\tau, q) := \{x' \in X_1 : \mathbf{x}(\tau, q, u) = x', u \in U_1\},$$

which is well-defined from the definition of the input set U_1.

The reachable set $\mathfrak{R}(\tau, q)$ is approximated as follows. Given any $\mu \in \mathbb{R}^+$, consider the following set

$$\mathcal{Z}_\mu(\tau, q) := \{y \in [\mathbb{R}^n]_\mu : \exists z \in \mathfrak{R}(\tau, q) \text{ s.t. } y = Q_\mu(z)\}.$$

Here, μ is a design parameter, whose choice is not related to η. Define the function $\phi : \mathcal{Z}_\mu(\tau, q) \to U_1$, which means that, for any $y \in \mathcal{Z}_\mu(\tau, q)$, there exists an input $u_1 = \phi(y) \in U_1$ such that $y = Q_\mu(\mathbf{x}(\tau, q, u_1))$. Note that the function ϕ is not unique. Thus, the set $U_2(\hat{q})$ in (3) can be defined by $U_2(\hat{q}) := \phi(\mathcal{Z}_\mu(\tau, q))$. Since the set $U_2(\hat{q})$ is the image through the map ϕ of a countable set, we have that $U_2(\hat{q})$ is countable, which implies that U_2 as defined in (3) is countable. As a result, the set U_2 approximates the set U_1 in the following way: given any $q \in [\mathbb{R}^n]_\eta$, for any $u_1 \in U_1$, there exists $u_2 \in U_2(\hat{q})$ such that $Q_\mu(\mathbf{x}(\tau, q, u_1)) = Q_\mu(\mathbf{x}(\tau, q, u_2))$. That is, $\mathbf{x}(\tau, q, u_1)$ and $\mathbf{x}(\tau, q, u_2)$ are in the same quantization region.

In contrast to the uniform quantization of the state and input sets as in [5], [6], the logarithmic partition proposed here significantly reduces the computation complexity of the developed abstraction; see Section VI.

B. Symbolic Abstraction

With the partitions of the state and input sets, the symbolic abstraction of the system $T_\tau(\Sigma)$ is described in this subsection. The developed symbolic abstraction is a transition system $T_{\tau,\eta,\mu}(\Sigma) = (X_2, X_2^0, U_2, \Delta_2, Y_2, H_2)$, where,

- the set of states is $X_2 = \hat{X}$;
- the set of initial states is $X_2^0 = \hat{X}$;
- the set of inputs is $U_2 = \bigcup_{q \in [\mathbb{R}^n]_\eta} U_2(q)$;
- the transition relation is given as follows: for $\hat{q}_1, \hat{q}_2 \in X_2$ and $u \in U_2$, $\hat{q}_2 \in \Delta_2(\hat{q}_1, u)$ if and only if

$$\hat{q}_2 \cap \left(\mathbf{x}(\tau, q_1, u) + [\![-\theta e^{L\tau}\bar{q}_1, \theta e^{L\tau}\bar{q}_1]\!]\right) \neq \varnothing, \tag{4}$$

  where $\theta := \eta(1-\eta)^{-1}$, $\bar{q}_1 := |q_1| + E_{q_1}$, $E_{q_1} \in \mathbb{R}^n$ is a vector whose the components are 1 if the corresponding components of q_1 are 0; and zero otherwise, and $L > 0$ is the Lipschitz constant of the function f;
- the set of outputs is $Y_2 = \mathbb{R}^n$;
- the output map is $H_2 : X_2 \hookrightarrow X_2$.

In the construction of the abstraction $T_{\tau,\eta,\mu}(\Sigma)$, the technique applied in (4) is similar to those in [9], [11], [18], where the overapproximation of successors of states is applied. $\theta e^{L\tau}\bar{q}_1$ in (4) plays the same role as the growth bound in [9]. Since the logarithmic quantizer is implemented here, the components of X_2 are the quantization region related to the quantized measurements. Hence, the developed abstraction extends those in previous works [6], [9], and provides an alternative for the abstraction construction. On the other hand, due to the logarithmic quantizer, the quantization errors are not bounded. Hence, the abstraction $T_{\tau,\eta,\mu}(\Sigma)$ and the system $T_\tau(\Sigma)$ do not satisfy the approximate bisimulation relation; see [6]. To deal with this issue, a feedback refinement relation is applied to connect $T_\tau(\Sigma)$ with $T_{\tau,\eta,\mu}(\Sigma)$.

Theorem 1: Consider the control system Σ with the time and state space sampling parameters $\tau, \eta, \mu \in \mathbb{R}^+$. Let the map $\mathcal{F} : X_1 \to X_2$ be given by $\mathcal{F}(x) = \hat{q}$ if and only if $x \in \hat{q}$. Then $T_\tau(\Sigma) \preceq_{\mathcal{F}} T_{\tau,\eta,\mu}(\Sigma)$.

Proof: Following from the definitions of $T_\tau(\Sigma)$ and $T_{\tau,\eta,\mu}(\Sigma)$, one has that $U_2 \subseteq U_1$. Let $(x_1, \hat{q}_1) \in \mathcal{F}$ with $x_1 \in X_1$ and $\hat{q}_1 \in X_2$, and we have that $x_1 \in \hat{q}_1$. For each $u \in U_2(\hat{q}_1)$, we obtain that $u \in U_2(\hat{q}_1) \subseteq U_2 \subseteq U_1$. In addition, $\Delta_2(\hat{q}_1, u) \neq \varnothing$ holds from the definition of $U_2(\hat{q}_1)$. If $\Delta_1(x_1, u) = \varnothing$, then we have that $u \notin U_1$, which is a contradiction. As a result, $\Delta_1(x_1, u) \neq \varnothing$ and $u \in U_1(x_1)$. We thus conclude that $U_2(\hat{q}_1) \subseteq U_1(x_1)$.

Given $\hat{q}_1, \hat{q}_2 \in X_2$ and $u \in U_2(\hat{q}_1)$, define $x_2 := \Delta_1(x_1, u)$, and thus $x_1 \in \hat{q}_1$ holds from $(x_1, \hat{q}_1) \in \mathcal{F}$, combining which with (1) yields that $\|x_1 - q_1\| \leq \theta\|q_1\|$. If $\Delta_1(x_1, u) \cap \hat{q}_2 \neq \varnothing$, there exists $x_2 := \mathbf{x}(\tau, x_1, u) \in X_1$ such that $x_2 \in \hat{q}_2$ holds from (4). From the Lipschitz property of the function f, one has $\|\mathbf{x}(\tau, x_1, u) - \mathbf{x}(\tau, q_1, u)\| \leq e^{L\tau}\|x_1 - q_1\| \leq \theta e^{L\tau}\|q_1\|$, which implies that $\hat{q}_2 \cap (\mathbf{x}(\tau, q_1, u) + [\![-\theta e^{L\tau}\bar{q}_1, \theta e^{L\tau}\bar{q}_1]\!]) \neq \varnothing$. Hence, $\hat{q}_2 \in X_2$ holds from the construction of the abstraction $T_{\tau,\eta,\mu}(\Sigma)$, which in turn completes the proof. ■

In the proof of Theorem 1, $\Delta_1(x_1, u) \cap \hat{q}_2 \neq \varnothing$ holds due to the unbounded state set studied in this paper. If the state set is bounded as in practical systems, we can impose an additional requirement such that $\Delta_2(\hat{q}_1, u) = \varnothing$ if $\hat{q}_1$ does not belong to the state set. Therefore, the feedback refinement relation is still valid in this case; see also [9], [11]. Since the logarithmic quantization is coarse and may lead to large approximation error, we can reduce the approximation error by applying logarithmic quantization to the components of X_2, thereby leading to the improvement of the approximation accuracy. In such setting, the state and input sets are not rediscretized, and the obtained abstraction is refined.

V. Controller Synthesis

With the feedback refinement relation established in Section IV, the next step is to study controller synthesis for the

system $T_\tau(\Sigma)$ via its abstraction $T_{\tau,\eta,\mu}(\Sigma)$. To begin with, we recall the definition of the abstract specification from [9].

Definition 5: Given a transition system $T := (X, X^0, U, \Delta, Y, H)$ and a set $Z \in \mathbb{R}^n$, any subset $\mathcal{S} \subseteq Z^\infty$ is called a *specification* on Z. The system T is said to *satisfy a specification* $\mathcal{S}$ on $U \times Y$ if from certain time instant, the trajectory of T always belongs to $\mathcal{S}$.

Definition 6: Given two transition systems $T_i := (X_i, X_i^0, U_i, \Delta_i, Y_i, H_i)$, $i \in \{1, 2\}$. Let $\mathcal{F} \subseteq X_1 \times X_2$ be a relation and $\mathcal{S}_1$ be a specification on $U_1 \times X_1$. A specification $\mathcal{S}_2$ on $U_2 \times X_2$ is called an *abstract specification* associated with $T_1, T_2, \mathcal{S}_1$ and $\mathcal{F}$, if $(u, x_1) \in \mathcal{S}_1$ holds for all $(u, x_2) \in \mathcal{S}_2$ and all $x_1 \in X_1$ with $(x_1, x_2) \in \mathcal{F}$.

If $T_1 \preceq_{\mathcal{F}} T_2$ and $\mathcal{S}_2$ is an abstract specification associated with $T_1, T_2, \mathcal{S}_1$ and $\mathcal{F}$, then we write $(T_1, \mathcal{S}_1) \preceq_{\mathcal{F}} (T_2, \mathcal{S}_2)$ for the sake of simplicity. In the following, we assume that $\mathcal{F}$ is a feedback refinement relation. For the control system $T_\tau(\Sigma)$, assume that the desired specification is given by $\mathcal{S} := \bar{U}_1 \times \bar{X}_1 \subseteq U_1 \times X_1$ with $\bar{U}_1 = \bigcup_{x \in \bar{X}_1} \mathrm{enab}(x)$. Define

$$\bar{X}_2 := \left\{\hat{q} \in X_2 : (x, \hat{q}) \in \mathcal{F}, x \in \bar{X}_1, \hat{q} \subseteq \bar{X}_1\right\},$$
$$\bar{U}_2 := \bigcup_{q \in [\mathbb{R}^n]_\eta, \hat{q} \in \bar{X}_2} U_2(\hat{q}), \quad Q_\eta(\mathcal{S}) := \bar{U}_2 \times \bar{X}_2.$$

As a result, $Q_\eta(\mathcal{S}) \subseteq U_2 \times X_2$.

Proposition 1: Assume that $T_\tau(\Sigma) \preceq_{\mathcal{F}} T_{\tau,\eta,\mu}(\Sigma)$. If $\mathcal{S} \subseteq U_1 \times X_1$ is a specification for the control system $T_\tau(\Sigma)$, then $Q_\eta(\mathcal{S})$ is a abstract specification for $T_{\tau,\eta,\mu}(\Sigma)$.

Proof: For any $(u, \hat{q}_1) \in Q_\eta(\mathcal{S})$, we have that $u \in \bar{U}_2 \subseteq \bar{U}_1$ and $\hat{q}_1 \in \bar{X}_2 \subseteq X_2$. Since $T_\tau(\Sigma) \preceq_{\mathcal{F}} T_{\tau,\eta,\mu}(\Sigma)$, there exists $x_1 \in X_1$ such that $(x_1, \hat{q}_1) \in \mathcal{F}$, which implies that $x_1 \in \hat{q}_1$. Thus, we obtain from the definition of $\bar{X}_2$ that $x_1 \in \bar{X}_2 \subseteq \bar{X}_1$, which in turn gives that $(u, x_1) \in \mathcal{S}$.

Given a $u \in U_2(\hat{q}_1)$, define $x_2 := \Delta_1(x_1, u)$ and $\hat{q}_2 := \Delta_2(\hat{q}_1, u) \in \bar{X}_2 \subseteq X_2$. $(x_2, \hat{q}_2) \in \mathcal{F}$ holds from $T_\tau(\Sigma) \preceq_{\mathcal{F}} T_{\tau,\eta,\mu}(\Sigma)$, which thus implies that $x_2 \in \hat{q}_2$. Hence, $x_2 \in \bar{X}_1 \subseteq X_1$ holds from the definition of $\bar{X}_2$, which indicates that $(u, x_2) \in \mathcal{S}$. By iteration, we deduce that $(u, x_2) \in \mathcal{S}$ for all $(u, \hat{q}_2) \in Q_\eta(\mathcal{S})$ and all $(x_1, \hat{q}_1) \in \mathcal{F}$. ■

In the following, we recall the definition of the controller for the control system $T = (X, X^0, U, \Delta, Y, H)$ from [5].

Definition 7: Given a transition system $T = (X, X^0, U, \Delta, Y, H)$, a *controller* is a map $\mathbb{C} : X \to 2^U$, and is *well-defined* if $\mathbb{C}(x) \subseteq \mathrm{enab}(x)$ for all $x \in X$. The *controlled system* is denoted by the transition system $T_c = (X, X^0, U, \Delta_c, Y, H)$ with the transition relation given by $x' \in \Delta_c(x, u)$ if and only if $u \in \mathbb{C}(x)$ and $x' \in \Delta(x, u)$.

According to Proposition 1 and Theorem VI.3 in [9], the following result is direct, and the proof is omitted here.

Proposition 2: If $(T_\tau(\Sigma), \mathcal{S}) \preceq_{\mathcal{F}} (T_{\tau,\eta,\mu}(\Sigma), Q_\eta(\mathcal{S}))$ and $\mathbb{C}_1 : X_2 \to 2^{U_2}$ is a controller for $(T_{\tau,\eta,\mu}(\Sigma), Q_\eta(\mathcal{S}))$, then the map $\mathbb{C} : X_1 \to 2^{U_1}$, defined as $\mathbb{C}(x) := \mathbb{C}_1(\mathcal{F}(x))$, is a controller for $(T_\tau(\Sigma), \mathcal{S})$.

A. Safety Controller Synthesis

Let $\mathcal{O}_s \subseteq Y$ be a output set associated with safe states. In this subsection, we consider the safety synthesis problem, which is to determine a controller to keep the system output inside the specified safe set $\mathcal{O}_s$.

Definition 8 (see [19]): Let $\mathcal{O}_s \subseteq Y$ be a set of safe outputs. A controller $\mathbb{C}$ is a *safety controller* for $T_c = (X, X^0, U, \Delta_c, Y, H)$ with the specification $\mathcal{O}_s$, if for all $x \in \mathrm{dom}(\mathbb{C})$, (i) $H(x) \in \mathcal{O}_s$; (ii) $\forall u \in \mathbb{C}(x)$, $\Delta_c(x, u) \subseteq \mathrm{dom}(\mathbb{C})$, where $\mathrm{dom}(\mathbb{C}) := \{x \in X : \mathbb{C}(x) \neq \varnothing\}$.

Lemma 1 (see [5]): Given a transition system T with the specification $\mathcal{O}_s$, a controller $\mathbb{C}$ is a safety controller if and only if for all the non-blocking states of the controlled system T_c, $H(x) \in \mathcal{O}_s$ and $x' \in \Delta(x, \mathbb{C}(x))$ is non-blocking.

We are now in the position to design a safety controller for the control system $T_\tau(\Sigma)$ with the specification $\mathcal{O}_s$.

Theorem 2: Assume that $T_\tau(\Sigma) \preceq_{\mathcal{F}} T_{\tau,\eta,\mu}(\Sigma)$. If $\mathbb{C}_1 : X_2 \to 2^{U_2}$ is a safety controller for $T_{\tau,\eta,\mu}(\Sigma)$ with the specification $Q_\eta(\mathcal{O}_s)$, let $\mathbb{C} : X_2 \to 2^{U_1}$ be given by

$$\mathbb{C}(x) := \mathbb{C}_1(\mathcal{F}(x)), \quad \forall x \in X_1, \tag{5}$$

then the map $\mathbb{C} : X_1 \to 2^{U_1}$ is well-defined, and is a safety controller for $T_\tau(\Sigma)$ with the specification $\mathcal{O}_s$.

Proof: First, we prove that the controller $\mathbb{C}$ is well-defined. Let $x_1 \in X_1$, and $u \in \mathbb{C}(x_1)$. It follows from (5) that there exists $\hat{q}_1 \in X_2$ such that $(x_1, \hat{q}_1) \in \mathcal{F}$ and $u \in \mathbb{C}_1(\hat{q}_1)$. Since $\mathbb{C}_1$ is well-defined, we have that $u \in \mathrm{enab}(\hat{q}_1)$. That is, there exists $\hat{q}_2 \in \Delta_2(\hat{q}_1, u)$. It follows from the feedback refinement relation that there exists $x_2 \in \Delta_1(x_1, u)$ such that $(x_2, \hat{q}_2) \in \mathcal{F}$, which implies that $u \in \mathrm{enab}(x_1)$. Thus, for all $x_1 \in X_1$, $\mathbb{C}(x_1) \subseteq \mathrm{enab}(x_1)$. Thus, $\mathbb{C}$ is well-defined.

Next, we prove that $\mathbb{C}$ is a safety controller for the specification $\mathcal{O}_s$. Let $x_1 \in X_1$ such that $\mathbb{C}(x_1) \neq \varnothing$, and let $u \in \mathbb{C}(x_1)$. By (5), there exists $\hat{q}_1 \in X_2$ such that $(x_1, \hat{q}_1) \in \mathcal{F}$ and $u \in \mathbb{C}_1(\hat{q}_1)$. Since $\mathbb{C}_1$ is a safety controller for the specification $Q_\eta(\mathcal{O}_s)$ and $\mathbb{C}_1(x_1) \neq \varnothing$, we have from Lemma 1 that $\hat{q}_1 \in Q_\eta(\mathcal{O}_s)$. It follows from Proposition 1 that $Q_\eta(\mathcal{O}_s)$ is an abstraction of the specification $\mathcal{O}_s$. Therefore, we obtain from $(x_1, \hat{q}_1) \in \mathcal{F}$ that $x_1 \in \mathcal{O}_s$.

Let $x_2 := \Delta_1(x_1, u)$. We have from $T_\tau(\Sigma) \preceq_{\mathcal{F}} T_{\tau,\eta,\mu}(\Sigma)$ that there exists $\hat{q}_2 \in \Delta_2(\hat{q}_1, u)$ such that $(x_2, \hat{q}_2) \in \mathcal{F}$. Since $\mathbb{C}_1$ is a safety controller for the specification $Q_\eta(\mathcal{O}_s)$ and $u \in \mathbb{C}_1(\hat{q}_1)$, we have, from Lemma 1, that $\mathbb{C}_1(\hat{q}_2) \neq \varnothing$. Finally, (5) implies that $\mathbb{C}_1(\hat{q}_2) \subseteq \mathbb{C}(x_2)$, and therefore $\mathbb{C}(X_2) \neq \varnothing$. As a result, we conclude that $\mathbb{C}$ is a safety controller for the specification $\mathcal{O}_s$, which completes the proof. ■

In Theorem 2, the abstract specification $Q_\eta(\mathcal{O}_s)$ is applied in synthesizing the safety controller, which is different from the results in [5], where the contraction and expansion of $\mathcal{O}_s$ are used. Following Definition 7, the controller for $T_{\tau,\eta,\mu}(\Sigma)$ can be written as a transition system $\mathbb{C}_1 = (X_c, U_2, G)$ with the state set $X_c \subseteq X_2$, the output set U_2 and the transition relation $G \subseteq X_c \times U_2$. In practice, we are interested in a bounded set of states X_2, which implies that the state set X_1 is also bounded; see [11]. Next, we focus on how to obtain the controller on X_2 via the abstract specification.

To this end, assume that $\hat{\mathcal{O}}_s := \{\hat{q} \in X_2 | (x, \hat{q}) \in \mathcal{F}, x \in X_1, \hat{q} \in \mathcal{O}_s\}$ is a under-approximation of $\mathcal{O}_s \in Y$. From Proposition 1, it is easy to verify that $\hat{\mathcal{O}}_s$ is a abstract

Algorithm 1 Safe Controller Design

Input: $\Upsilon, \Upsilon_1 \subseteq X_2, \mathbb{C}_1$
Output: $(X_c, \mathbb{C}_1)$ with $X_c = \Upsilon$

Explore($\Upsilon \backslash \Upsilon_1$)
$W = \text{ConPre}(\Upsilon) \cap \Upsilon$
$\mathbb{C}_1 = \mathbb{C}_1 \cup \{(W, U_2, \varnothing)\}$
$\Upsilon_1 = \Upsilon_1 \cup W$
if $\Upsilon = \Upsilon_1$ **then**
    return $(\Upsilon, \mathbb{C}_1)$
else
    for $\Upsilon \neq \Upsilon_1$ **do**
        $\Upsilon = \Upsilon_1$
        $\Upsilon_1 = \varnothing$
        Explore(Υ)
        $W = \text{ConPre}(\Upsilon) \cap \Upsilon$
        $\mathbb{C}_1 = \mathbb{C}_1 \cup \{(W, U_2, \varnothing)\}$
        $\Upsilon_1 = W$

Algorithm 2 Explore

Input: $\Upsilon \subseteq X_2$
Output: transition relation G

for $\hat{q} \in \Upsilon$, $u \in U_2$ **do**
    if $\Delta_2(\hat{q}, u)$ is not defined **then**
        compute $\Delta_2(\hat{q}, u)$
        if $\Delta_2(\hat{q}, u) \subseteq \Upsilon$ **then**
            map $\hat{q}$ to u

specification of $\mathcal{O}_s$. Define the following sets (see [20])

$$W_0 := \hat{\mathcal{O}}_s, \quad W_{i+1} := \text{ConPre}(W_i) \cap \hat{\mathcal{O}}_s, \quad i \in \mathbb{N}^+. \quad (6)$$

In (6), the function $\text{ConPre} : 2^{X_2} \to 2^{X_2}$ is the controllable predecessor operator [20], and defined as: for a set $\Upsilon \subseteq X_2$,

$$\text{ConPre}(\Upsilon) := \{\hat{q} \in X_2 | \exists u \in U_2 \text{ such that } \Delta_2(\hat{q}, u) \subseteq \Upsilon\}.$$

The iteration in (6) ends when $W_i = W_{i+1}$. Since the sequence $\{W_i\}$ is monotone over a finite domain, the convergence of such a sequence is guaranteed in finite time; see [20]. Assume that the number of the iterations is $M \in \mathbb{N}^+$.

According to (6), define the state set $X_c := W_M$. Following the definition of the abstraction $T_{\tau,\eta,\mu}(\Sigma)$, the transition $G : X_c \to U_2$ is defined as: for all $q \in X_c$ and $u \in U_2$, $u = G(\hat{q}) \Leftrightarrow \Delta_2(\hat{q}, u) \subseteq X_c$. As a result, we have that $\mathbb{C}_1 = (X_c, U_2, G)$, which is a safety controller for the abstraction $T_{\tau,\eta,\mu}(\Sigma)$ with the specification $\hat{\mathcal{O}}_s$.

Based on the above analysis, the synthesis algorithm is summarized in Algorithm 1, which is terminated after M iterations. In Algorithm 1, Υ and Υ_1 are initialized as $\hat{\mathcal{O}}_s$ and $\varnothing$, respectively. Algorithm 2 is to develop the maps from $\hat{q} \in \Upsilon$ to $u \in U_2$ such that the transition G is obtained. Because of the monotonic nature of the iterative computation of safe sets, the set Υ is a subset of $\hat{\mathcal{O}}_s$. Based on (6) and Algorithm 1, the controller $\mathbb{C}_1$ is obtained iteratively, which further leads to the controller for the system $T_\tau(\Sigma)$.

Theorem 3: Assume that $T_\tau(\Sigma) \preceq_{\mathcal{F}} T_{\tau,\eta,\mu}(\Sigma)$. The controller $\mathbb{C}_1$ obtained via Algorithm 1 is a safety controller for $T_{\tau,\eta,\mu}(\Sigma)$ with the specification $\hat{\mathcal{O}}_s$. Furthermore, the controller $\mathbb{C}(x) := \mathbb{C}_1(\mathcal{F}(x))$ is a safety controller for $T_\tau(\Sigma)$ with the specification $\mathcal{O}_s$.

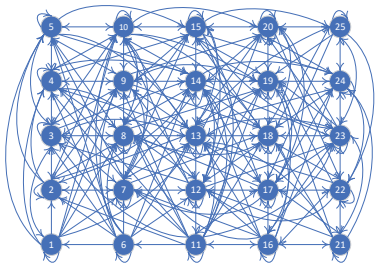

Fig. 1. Symbolic model $T_{0.2,0.2}(\Sigma)$ for the control system Σ. The abstract state (z_i, z_j) in $T_{0.2,0.2}(\Sigma)$ with $i, j \in \{-2, -1, 0, 1, 2\}$ corresponds to the state $5(i+2)+j+3$ in this figure.

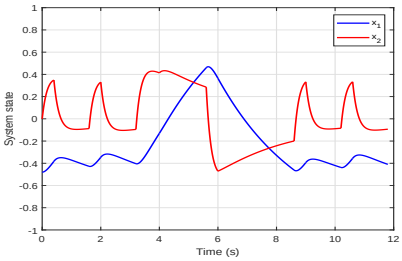

Fig. 2. Trajectory of the control system Σ with initial condition $(-0.48, 0)$ and control strategy synthesized on $T_{0.2,0.2}(\Sigma)$.

VI. Illustrative Example

As a simple mechanical control system studied in the literature [6], the pendulum is described as

$$\Sigma : \dot{x}_1 = x_2, \quad \dot{x}_2 = -gl^{-1}\sin(x_1) - km^{-1}x_2 + u,$$

where x_1 and x_2 are respectively the angular position and velocity of the point mass, u is the torque which can be treated as the control variable. In addition, $g = 9.8$ is the gravity acceleration, $l = 5$ is the length of the rod, $m = 0.5$ is the mass, and $k = 3$ is the coefficient of friction. Assume that the control input u is piecewise-constant and bounded in the set $U = [-2.5, 2.5]$. For simplicity the state set is bounded in the set $X = [-1, 1] \times [-1, 1]$.

To construct the abstraction, the applied quantizer is

$$Q(z) := \begin{cases} \dfrac{(1+\eta)^{k+1}a}{(1-\eta)^k}, & \dfrac{(1+\eta)^k a}{(1-\eta)^k} < z \leq \dfrac{(1+\eta)^{k+1}a}{(1-\eta)^{k+1}}; \\ 0, & 0 \leq z \leq a; \\ -Q(-z), & z < 0. \end{cases}$$

Let $\eta = 0.2$ and $a = 0.4$, and thus there are 25 quantization regions for the logarithmic quantizer. Comparing with the uniform quantizer applied in [6], the quantization regions for the logarithmic quantizer are not the same with equivalent size. By adjusting the parameters η and a, we can change the precision of the logarithmic quantizer, whereas the precision of the uniform quantizer depends on the precision of the approximate bisimulation; see also [6], [7].

Let $\tau = 0.2$ and $\mu = 2 \times 10^{-3}$. In addition, the Lipschitz constant for Σ is 6. The symbolic model

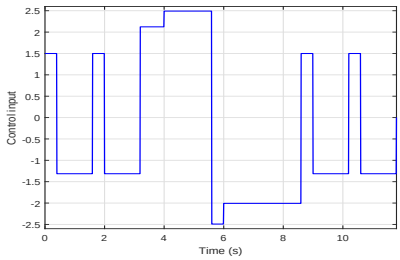

Fig. 3. Control strategy synthesized on $T_{0.2,0.2,2\times10^{-3}}(\Sigma)$.

$T_{0.2,0.2,2\times10^{-3}}(\Sigma) = (X_2, X_2^0, U_2, \Delta_2, Y_2)$ is given by: (i) X_2 is the union of the quantization regions partitioned via the logarithmic quantizer Q; (ii) $X_2^0 = X_2$; (iii) $U_2 = \bigcup_{q\in[\mathbb{R}^n]_{2\times10^{-3}}} U_2(q)$; (iv) the transition relation Δ_2 is depicted in Fig. 1; (v) $Y_2 = X_2$. The transition system $T_{0.2,0.2,2\times10^{-3}}(\Sigma)$ is shown in Fig. 1, where the transition relation is obtained via (5) and the numerical integration of the trajectories of Σ. Comparing with the uniform quantization based abstraction in [6], there are more (loop) transitions in $T_{0.2,0.2,2\times10^{-3}}(\Sigma)$ emanating from the abstract states, which implies that some complexity issues can be avoided; see [8].

In the following, the controller synthesis is illustrated via the symbolic model $T_{0.2,0.2,2\times10^{-3}}(\Sigma)$. Assume that the objective is to design a controller to enforce an alternation between two different periodic motions, which are respectively denoted as $\mathcal{S}_1$ and $\mathcal{S}_2$. The periodic motion $\mathcal{S}_1$ requires the state of Σ to cycle between $(-0.48, 0)$ and $(0, 0)$, whereas the periodic motion $\mathcal{S}_2$ requires the state to cycle between $(-0.48, 0)$ and $(0.48, 0)$. Thus, the control aim is to design a controller such that the system Σ satisfies a specification $\mathcal{S}$ requiring the execution of the sequence of periodic motions $\mathcal{S}_1, \mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_1, \mathcal{S}_1$.

A control strategy for periodic motions $\mathcal{S}_1$ and $\mathcal{S}_2$ can be obtained by performing a search on $T_{0.2,0.2,2\times10^{-3}}(\Sigma)$ using standard methods in supervisory control [3]. A possible solution for $\mathcal{S}_1$ is given by $(-0.48, 0) \overset{1.4991}{\longrightarrow} (0, 0) \overset{-1.2127}{\longrightarrow} (-0.48, 0)$, and a solution for $\mathcal{S}_2$ is given by $(-0.48, 0) \overset{2.1230}{\longrightarrow} (0, 0.48) \overset{2.4914}{\longrightarrow} (0.48, 0) \overset{-2.4914}{\longrightarrow} (0, -0.48) \overset{-2.0074}{\longrightarrow} (-0.48, 0)$. Based on such two solutions, a control strategy for $\mathcal{S}$ is derived by combining the trajectories associated with the motions $\mathcal{S}_1, \mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_1$ and $\mathcal{S}_1$. As a result, we have the following transitions: $(-0.48, 0) \overset{1.4991}{\longrightarrow} (0, 0) \overset{-1.2127}{\longrightarrow} (-0.48, 0) \overset{1.4991}{\longrightarrow} (0, 0) \overset{-1.2127}{\longrightarrow} (-0.48, 0) \overset{2.1230}{\longrightarrow} (0, 0.48) \overset{2.4914}{\longrightarrow} (0.48, 0) \overset{-2.4914}{\longrightarrow} (0, -0.48) \overset{-2.0074}{\longrightarrow} (-0.48, 0) \overset{1.4991}{\longrightarrow} (0, 0) \overset{-1.2127}{\longrightarrow} (-0.48, 0)$. Note that some transitions are not obtained by one sampling period. This means that the abstract state may stay the same after certain transitions, which results from the symbolic abstraction via the logarithmic quantization; see also the loop transitions in Fig. 1. The control strategy is presented in Fig. 3. With such control strategy, the evolution of the system state is shown in Fig. 2. The completion time of the specification $\mathcal{S}$ is 11.8s, whereas the completion time in [6] is 24s, which implies that the computation time is reduced significantly.

VII. Conclusion

In this paper, we applied logarithmic quantization to construct the symbolic abstraction for nonlinear control systems. Based on the constructed discrete abstraction, the controller synthesis problem was studied via abstract specification, and a novel algorithm was proposed to compute the safety controller. Finally, a numerical example was provided to illustrate the obtained results. Future researches will be directed to symbolic abstractions via dynamic quantizers.

References

[1] R. Milner, *Communication and Concurrency*. Prentice Hall, 1989.

[2] P. Tabuada and G. J. Pappas, "Linear time logic control of discrete-time linear systems," *IEEE Transactions on Automatic Control*, vol. 51, no. 12, 2006.

[3] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete event processes," *SIAM Journal on Control and Optimization*, vol. 25, no. 1, pp. 206–230, 1987.

[4] ——, "Modular feedback logic for discrete event systems," *SIAM Journal on Control and Optimization*, vol. 25, no. 5, pp. 1202–1218, 1987.

[5] A. Girard, "Controller synthesis for safety and reachability via approximate bisimulation," *Automatica*, vol. 48, no. 5, pp. 947–953, 2012.

[6] G. Pola, A. Girard, and P. Tabuada, "Approximately bisimilar symbolic models for nonlinear control systems," *Automatica*, vol. 44, no. 10, pp. 2508–2516, 2008.

[7] A. Girard, G. Pola, and P. Tabuada, "Approximately bisimilar symbolic models for incrementally stable switched systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 1, pp. 116–126, 2010.

[8] G. Reissig and M. Rungger, "Feedback refinement relations for symbolic controller synthesis." in *Proceedings of IEEE Conference on Decision and Control*, 2014, pp. 88–94.

[9] G. Reissig, A. Weber, and M. Rungger, "Feedback refinement relations for the synthesis of symbolic controllers," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1781–1796, 2017.

[10] M. Khaled, M. Rungger, and M. Zamani, "Symbolic models of networked control systems: A feedback refinement relation approach," in *Annual Allerton Conference on Communication, Control, and Computin*. IEEE, 2016, pp. 187–193.

[11] P.-J. Meyer, A. Girard, and E. Witrant, "Compositional abstraction and safety synthesis using overlapping symbolic models," *IEEE Transactions on Automatic Control*, vol. 63, no. 6, pp. 1835–1841, 2018.

[12] W. Ren and J. Xiong, "Quantized feedback stabilization of nonlinear systems with external disturbance," *IEEE Transactions on Automatic Control*, vol. 63, no. 9, pp. 3167–3172, 2018.

[13] D. F. Delchamps, "Stabilizing a linear system with quantized state feedback," *IEEE transactions on automatic control*, vol. 35, no. 8, pp. 916–924, 1990.

[14] M. Fu and L. Xie, "The sector bound approach to quantized feedback control," *IEEE Transactions on Automatic control*, vol. 50, no. 11, pp. 1698–1711, 2005.

[15] D. F. Coutinho, M. Fu, and C. E. de Souza, "Input and output quantized feedback linear systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 3, pp. 761–766, 2010.

[16] N. Elia and S. K. Mitter, "Stabilization of linear systems with limited information," *IEEE transactions on Automatic Control*, vol. 46, no. 9, pp. 1384–1400, 2001.

[17] M. Fu and L. Xie, "Finite-level quantized feedback control for linear systems," *IEEE Transactions on Automatic Control*, vol. 54, no. 5, pp. 1165–1170, 2009.

[18] G. Reißig, "Computing abstractions of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 11, pp. 2583–2598, 2011.

[19] A. Girard, "Low-complexity quantized switching controllers using approximate bisimulation," *Nonlinear Analysis: Hybrid Systems*, vol. 10, pp. 34–44, 2013.

[20] O. Maler, A. Pnueli, and J. Sifakis, "On the synthesis of discrete controllers for timed systems," in *Annual Symposium on Theoretical Aspects of Computer Science*. Springer, 1995, pp. 229–242.