Symbolic Supervisory Control of Periodic Event-Triggered Control Systems ⋆

Wei Ren *, **Dimos V. Dimarogonas** *

* *Division of Decision and Control Systems, EECS, KTH Royal Institute of Technology, SE-10044, Stockholm, Sweden. (emails: weire@kth.se, dimos@kth.se).*

Abstract: This paper studies supervisory control of periodic event-triggered control (PETC) systems based on the construction of symbolic abstractions. To this end, we first construct symbolic abstractions for PETC systems, and establish feedback refinement relation from the PETC system to its symbolic models. Here, the constructed symbolic models are represented by the form of discrete event systems (DESs), including extended finite state machines, finite state machines, and classic DESs. With the constructed symbolic models, we study the supervisory control of PETC systems to achieve the desired specification. Since the constructed symbolic models are nondeterministic, we first transfer the symbolic models into deterministic versions, and then verify the existence of the supervisor. Finally, the obtained results are illustrated via a numerical example.

Keywords: Discrete event systems, periodic event-triggered control systems, symbolic model, supervisory control.

1. INTRODUCTION

As a standard approach for the design of hybrid systems, discrete abstractions have gained considerable attention in the past few decades; see Milner (1989); Tabuada (2009). With discrete abstractions of continuous dynamics, controller synthesis problems can be studied efficiently by using techniques developed in supervisory control (Ramadge and Wonham (1987b)) and algorithmic game theory (Ramadge and Wonham (1987a)). Since an inclusion or equivalence relationship is ensured between the original system and its discrete abstraction, the synthesized controller is correct-by-design, and thus formal verification is either not needed or can be reduced; see Tabuada (2009).

Since not all the dynamical systems possess symbolic abstractions, a fundamental problem is to identify more general classes of continuous dynamical systems admitting symbolic abstractions. Along this direction, different types of dynamical systems have been studied in the literature, such as switched control systems in Girard et al. (2010), time-delay control systems in Pola et al. (2010), and stochastic systems in Zamani et al. (2014). The equivalence relations between dynamical systems and symbolic abstractions can be classified into two types: the (approximate) (bi)simulation relation and its variants, which lead to equivalences of dynamic systems in an exact or approximate setting (Girard and Pappas (2007); Tabuada (2009)), and the feedback refinement relation (Reissig et al. (2017)), which is based on the principle of "accepting more inputs and generating fewer outputs" and can be applied to deal with state information and refinement complexity issues appearing in control synthesis. Based on these equivalence relations, symbolic abstractions can be measured quantitatively using metric transition systems.

⋆ This work was partially supported by the H2020 ERC Starting Grant BUCOPHSYS, 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).

As a special class of dynamical systems, event-triggered control (ETC) systems have been extensively investigated due to many applications in filed of networked control systems (Heemels et al. (2012)) and multi-agent systems (Dimarogonas et al. (2012)). However, to the best of our knowledge, there are few works (Kolarijani and Mazo (2018); Fu and Mazo (2018)) on symbolic abstractions of ETC systems. Therefore, the topic of this paper is on a class of event-triggered control systems, namely, periodic event-triggered control (PETC) systems. Our first contribution is to propose a novel symbolic abstraction for PETC systems. To this end, the applied construction approach is based on embedded lattices, and the symbolic model is constructed as a (extended) discrete event system (DES), such as extended finite state machines (EFSMs) (Zhao et al. (2012); Teixeira et al. (2014)) and finite state machines (FSMs) (Cassandras and Lafortune (2009)). The applied approach is different from the one implemented in Kolarijani and Mazo (2018); Fu and Mazo (2018), where the symbolic abstraction is constructed as a timing model by transforming the event-triggered mechanism into a time-triggered mechanism.

Since the symbolic abstractions are of (extended) DES forms, our second contribution is to apply supervisory control theory for DESs to deal with the controller synthesis for the desired specification, which transfers the supervisory control theory from qualitative systems like DESs into quantitative systems described by discrete-time control systems; see Pola et al. (2017); Cassandras and Lafortune (2009). Since the constructed symbolic abstractions are not deterministic, we first transfer the symbolic abstractions into deterministic forms by introducing finite unobservable events to label the nondeterministic transitions; similar techniques can be found in Heymann and Lin (1998); Hopcroft (2008). We further verify the existence of the supervisor for the desired specification, and then refine the supervisor into the periodic event-triggered mechanism for the original system. Finally, a numerical example is given to show the obtained results.

The remainder of this paper is as follows. In Section 2, definitions and preliminaries for PETC systems are introduced. Symbolic abstractions are constructed as DESs in Section 3. Symbolic supervisory control for PETC systems is studied in Section 4. A numerical example is given in Section 5. Finally, conclusion and future works are presented in Section 6.

Notation. $\mathbb{R} := (-\infty, +\infty)$; $\mathbb{R}_0^+ := [0, +\infty)$; $\mathbb{R}^+ := (0, +\infty)$; $\mathbb{N} := \{0, 1, \ldots\}$; $\mathbb{N}^+ := \{1, 2, \ldots\}$. $\mathbb{R}^n$ is the n-dimensional Euclidean space. E denotes the vector with appropriate dimension and all the components equal to 1. 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 componentwise; and a cell $[\![a, b]\!]$ is the closed set $\{x \in \mathbb{R}^n : a_i \leq x_i \leq b_i\}$. Given $x \in \mathbb{R}^n$, denote by $\|x\|$ the infinite norm of x. Given two sets $A, B \subset \mathbb{R}^n$, $B \backslash A := \{x : x \in B, x \notin A\}$; $|A|$ denotes the cardinality of A; a relation $\mathscr{R} \subset A \times B$ is the map $\mathscr{R} : A \to 2^B$ defined as $b \in \mathscr{R}(a)$ if and only if $(a, b) \in \mathscr{R}$. The inverse relation of $\mathscr{R}$ is denoted by $\mathscr{R}^{-1} := \{(b, a) \in B \times A : (a, b) \in \mathscr{R}\}$.

2. PERIODIC EVENT-TRIGGERED CONTROL SYSTEMS

2.1 Periodic Event-Triggered Control Systems

Consider the following linear system

$$\dot{x}(t) = Ax(t) + Bu(t), \quad \forall t \in \mathbb{R}^+, \tag{1}$$

where $A \in \mathbb{R}^{n \times n}$ and $B \in \mathbb{R}^{n \times m}$ are constant matrices, $x \in X \subset \mathbb{R}^n$ is the system state and $u \in U \subset \mathbb{R}^m$ is the control input. For the system (1), the controller for (1) is given by

$$u(t) = K\bar{x}(t_k), \quad t \in [t_k, t_{k+1}), \tag{2}$$

where $\bar{x}(t_k) \in \mathbb{R}^n$ is the most recently transmitted state measurement to the controller at the transmission time instant $t_k > 0$. The initial value of $\bar{x}$ can be given *a priori*, and the update of $\bar{x}$ is given by

$$\bar{x}(t_k) = \begin{cases} x(t_k), & \text{if } \|x(t_k) - \bar{x}(t_k)\| \geq \delta \|x(t_k)\|, \\ \bar{x}(t_{k-1}), & \text{if } \|x(t_k) - \bar{x}(t_k)\| < \delta \|x(t_k)\|, \end{cases} \tag{3}$$

where $\delta \in \mathbb{R}^+$ is constant parameter to be designed, and $\bar{x}$ is the latest transmitted state measurement. Therefore, the next transmission time instant for the system (1)-(2) is determined by the following event-triggered mechanism (ETM):

$$t_{k+1} := \min\{t > t_k : \|x(t) - \bar{x}(t_k)\| \geq \delta \|x(t)\|\}. \tag{4}$$

This event-triggered condition $\mathbb{C}(x, \bar{x}) := \|x - \bar{x}\| - \delta \|x(t)\| \geq 0$ is based on the state error, and is used to determine whether or not the new state measurement is transmitted to the controller.

In this paper, we focus on PETC systems, which refer to the class of ETC systems with periodic ETMs. Hence, the following assumption is made.

Assumption 1. The applied ETM is periodic with the period $\tau_e = N\tau$, where $N \in \mathbb{N}$ and $\tau > 0$ is the sampling period.

From Assumption 1, $t_{k+1} - t_k \geq \tau_e$, which implies that Zeno phenomena are excluded. Assumption 1 implies that the event-triggering time sequence satisfies $\{t_k : k \in \mathbb{N}\} \subset \{i\tau_e : i \in \mathbb{N}\}$. For PETC systems, the determination of the event-triggering time instants depends on both the event-triggered condition $\mathbb{C} : X \times X \to \mathbb{R}$ and the periodic condition $(t_{k+1} - t_k)/\tau_e \in \mathbb{N}^+$, which leads to the periodic ETM (PETM):

$$t_{k+1} = \min\{t = t_k + j\tau_e : \mathbb{C}(x(t), \bar{x}(t_k)) \geq 0, j \in \mathbb{N}\}. \tag{5}$$

In particular, if $N = 1$, then $\tau_e = \tau$ and the PETM (5) is verified at each sampling time as in Heemels et al. (2012).

Denote by Σ the system (1)-(2). Similarly to Tabuada (2009), a curve $\xi : (a, b) \to X$ is said to be a *trajectory* of Σ, if there exists a control input $\mathbf{u} \in \mathscr{U}$, where $\mathscr{U}$ is a subset of all piecewise continuous functions of time from $(a, b) \subset \mathbb{R}$ to U with $a < 0 < b$ and which depends on the PETM (5), such that

$$\dot{\xi}(t) = A\xi(t) + B\mathbf{u}(t_k), \quad \forall t \in (a, b) \cap [t_k, t_{k+1}). \tag{6}$$

In addition, we can define the trajectory $\mathbf{x} : [0, \tau] \to X$ on a closed interval $[0, \tau]$ with $\tau \in \mathbb{R}^+$ such that $\mathbf{x} = \xi|_{[0,\tau]}$. Denote by $\mathbf{x}(t, x, \mathbf{u})$ the point reached at time $t \in (a, b)$ under the input $\mathbf{u} \in \mathscr{U}$ and the PETM (5) from the initial condition $x \in X_0 \subset X$.

2.2 Feedback Refinement Relation

In this subsection, we recall the notion of feedback refinement relation for two transition systems.

Definition 1. (Girard and Pappas (2007)). A *transition system* is a sextuple $T = (X, X^0, U, \Delta, Y, H)$, consisting of: (i) a set of states X; (ii) a set of initial states $X^0 \subseteq X$; (iii) a set of inputs U; (iv) a transition relation $\Delta \subseteq X \times U \times X$; (v) a set of outputs Y; (vi) an output function $H : X \to Y$. T is said to be *metric* if the output set Y is equipped with a metric $\mathbf{d} : Y \times Y \to \mathbb{R}_0^+$, and *symbolic* if the sets X and U are finite or countable.

The transition $(x, u, x') \in \Delta$ is denoted $x' \in \Delta(x, u)$, which means that the system can evolve from a state x to a state x' under the input u. An input $u \in U$ belongs to *the set of enabled inputs* at the state x, denoted by $\text{enab}(x)$, if $\Delta(x, u) \neq \varnothing$. If $\text{enab}(x) = \varnothing$, then x is said to be *blocking*, otherwise, it is said to be *non-blocking*. The transition system T is said to be *deterministic*, if for all $x \in X$ and all $u \in \text{enab}(x)$, $\Delta(x, u)$ has exactly one element; otherwise, T is said to be *nondeterministic*.

Definition 2. (Reissig et al. (2017)). Let T_1 and T_2 be two transition systems with $T_i = (X_i, X_i^0, U_i, \Delta_i, Y_i, H_i)$ for $i \in \{1, 2\}$, and assume that $U_2 \subseteq U_1$. A relation $\mathscr{F} \subseteq X_1 \times X_2$ is a *feedback refinement relation* from T_1 to T_2, if for all $(x_1, x_2) \in \mathscr{F}$, (i) $U_2(x_2) \subseteq U_1(x_1)$; (ii) $u \in U_2(x_2) \Rightarrow \mathscr{F}(\Delta_1(x_1, u)) \subseteq \Delta_2(x_2, u)$, where $U_i(x) := \{u \in U_i : u \in \text{enab}(x)\}$ and $i \in \{1, 2\}$. Denote by $T_1 \preceq_{\mathscr{F}} T_2$ if $\mathscr{F} \subseteq X_1 \times X_2$ is a feedback refinement relation from T_1 to T_2.

3. DISCRETE EVENT SYSTEM AS SYMBOLIC MODEL

In this section, we construct the symbolic abstractions of PETC systems as the forms of discrete event systems, and establish the feedback refinement relation from the PETC system to its symbolic model.

3.1 Time Discretization

To develop the symbolic abstraction, we work with the time-discretization of the PETC system Σ, which is presented below. Assume here that the sampling period is $\tau > 0$, which is a parameter to be designed. The time discretization of the PETC system Σ is represented as the transition system $T_\tau(\Sigma) := (\mathfrak{X}_1, \mathfrak{X}_1^0, U_1, \Delta_1, Y_1, H_1)$, where

- the state set is $\mathfrak{X}_1 := X \times U \times \mathbb{N}$ with $\mathbb{N} := \{0, \ldots, N-1\}$;
- the set of initial states is $\mathfrak{X}_1^0 := X \times U \times \{0\}$;
- the input set is $U_1 := U$;

- the transition relation is given as follows: for any $(x,u,l) \in \mathfrak{X}_1$ and $u \in U_1$, $(x',u',l') = \Delta_1(x,u,l)$ if and only if $x' = \mathbf{x}(\tau,x,u)$ and
  - $u' = u$ and $l' = l+1$for $l < N-1$;
  - $u' = u$ and $l' = 0$ for $l = N-1$ and $\mathbb{C}_1(x,\bar{x}) < 0$;
  - $u' = Kx'$ and $l' = 0$ for $l = N-1$ and $\mathbb{C}_1(x,\bar{x}) \geq 0$,

  where $\mathbb{C}_1(x,\bar{x}) = \mathbb{C}(x,\bar{x})$ given in (5);
- the output set is $Y_1 := \mathbb{R}^n$;
- the output map is $H_1 : \mathfrak{X}_1 \to \mathbb{R}^n$ with $H_1((x,u,l)) = x$.

In the transition system $T_\tau(\Sigma)$, the state is augmented to include the original state $x \in X$, the control input $u \in U$ and the auxiliary variable $l \in \mathbb{N}$. Here, the variable l is to show whether it is time to verify the PETM (5). If $\tau_e = \tau$, then $N = 1$ and the variable l can be removed such that $T_\tau(\Sigma)$ is reduced to a simpler form.

3.2 Partition of State and Input Sets

We first approximate the set X. The set X is approximated by the sequence of finite embedded lattices $[X]_\eta$, where $[X]_\eta := \{q \in X : q_i = k_i\eta, k_i \in \mathbb{Z}, i \in \{1,\ldots,n\}\}$, where $\eta \in \mathbb{R}^+$ is called the state space sampling parameter. We further associate a quantizer $Q_\eta : X \to [X]_\eta$ such that $Q_\eta(x) = q$ if and only if $q_i - \eta/2 \leq x_i \leq q_i + \eta/2$, for $x = (x_1,\ldots,x_n) \in \mathbb{R}^n$ and $i \in \{1,\ldots,n\}$. Hence, for all $x \in X$, there exists a $q \in [X]_\eta$ such that $x \in [\![q - 0.5\eta E, q + 0.5\eta E]\!]$, and $\|x - Q_\eta(x)\| \leq \eta/2$ holds based on geometrical considerations. As a result, we can define the partition $\hat{X} := \{\hat{q} = [\![q - 0.5\eta E, q + 0.5\eta E]\!] : q \in [X]_\eta\}$.

Similarly, we can approximate the set U with the sequence of finite embedded lattices $[U]_\mu$, where $[U]_\mu := \{v \in U : q_i = k_i\mu, k_i \in \mathbb{Z}, i \in \{1,\ldots,m\}\}$, where $\mu \in \mathbb{R}^+$ is called the input space sampling parameter. A quantizer $Q_\mu : U \to [U]_\mu$ is associated to the approximation of the input set. Here, the parameters η and μ determine the approximation precision. Given the desired approximation precision, the choices of η and μ are limited, which will be studied further in Subsection 3.3.

With the approximation of the sets X and U, we can obtain the approximation of the state set $\mathfrak{X}_1$ in $T_\tau(\Sigma)$. Even though the input set U is approximated via the sequence of finite embedded lattices $[U]_\mu$, the PETM (5) leads to the constrained choice of the control input and the limitation of the transitions.

3.3 Symbolic Models based on DESs

With the partitions of the state and input sets, the symbolic abstractions of the PETC system $T_\tau(\Sigma)$ are constructed in the DES forms. We start by developing the symbolic abstraction of $T_\tau(\Sigma)$ as an EFSM. An EFSM is a conventional FSM with a set of state variables, whose updates are associated with the transitions; see Teixeira et al. (2014); Cassandras and Lafortune (2009). From $T_\tau(\Sigma)$, the state variable is defined as $l \in \mathbb{N}$. That is, the set of state variables is $V := \{l\}$ with the domain $\mathrm{dom}(V) = \mathbb{N}$. Based on the set of state variables V, we define the set of Boolean formulas over $V \times V$, which is denoted by Π_{v}. The transition occurs only when the Boolean formula $p \subset V \times V$ is true; otherwise, the transition is forbidden. Here, $p(l,l')$ is true in the following two cases.

$$p(l,l') = \texttt{true} \text{ if } \begin{cases} l < N-1, \ l' = l+1, \\ l = N-1, \ l' = 0. \end{cases} \tag{7}$$

Note that the Boolean formula $p \subset V \times V$ shows the update of the state variable $l \in \mathbb{N}$.

Now, we are ready to define the symbolic abstraction of $T_\tau(\Sigma)$, which is given by $\mathbf{EFSM}(\Sigma) = (\mathfrak{X}_2, \mathfrak{X}_2^0, V, \mathbb{E}_2, \Delta_2)$, where,

- the state set is $\mathfrak{X}_2 := \hat{X} \times [U]_\mu$;
- the set of initial states is $\mathfrak{X}_2^0 := \mathfrak{X}_2$;
- the set of state variables is $V := \{l\}$ with the domain $\mathrm{dom}(V) = \mathbb{N}$;
- the event set is $\mathbb{E}_2 := \{\pi, \texttt{TG}, \texttt{NTG}\}$, where π means the silent transition (i.e., $l < N-1$); $\texttt{TG}$ means the triggering event (i.e., $\mathbb{C}(q,\bar{q}) \geq 0$) and $\texttt{NTG}$ means the non-triggering event (i.e., $\mathbb{C}(q,\bar{q}) < 0$);
- the transition relation $\Delta_2 \subseteq \mathfrak{X}_2 \times \mathbb{E}_2 \times \Pi_{\mathrm{v}} \times \mathfrak{X}_2$ is given as follows: for any $(\hat{q},v) \in \mathfrak{X}_2$, $\sigma \in \mathbb{E}_2$ and $p \in \Pi_{\mathrm{v}}$, $(\hat{q}',v') \in \Delta_2((\hat{q},v),\sigma,p)$ if and only if $\hat{q}' \cap [\![\mathbf{x}(\tau,q,v) - \eta\|e^{A\tau}\|E/2, \mathbf{x}(\tau,q,v) + \eta\|e^{A\tau}\|E/2]\!] \neq \varnothing$, and
  (1) for $p(l,l') = \texttt{true}$ with $l' = l+1 \leq N-1$,
  $$(\hat{q},v) \xrightarrow{\pi} (\hat{q}',v') \text{ with } v' = v; \tag{8}$$
  (2) for $p(l,l') = \texttt{true}$ with $l = N-1$ and $l' = 0$,
  $$(\hat{q},v) \xrightarrow{\texttt{TG}} (\hat{q}',v') \text{ with } v' = Q_\mu(Kq), \tag{9}$$
  $$(\hat{q},v) \xrightarrow{\texttt{NTG}} (\hat{q}',v') \text{ with } v' = v. \tag{10}$$

If the outputs need to be studied, we define the output set as $Y_2 = \mathbb{R}^n$ and the output function as $H_2((\hat{q},v)) = q$. If the set X is bounded, we define $\Delta_2((\hat{q},v),\sigma,p) = \varnothing$ directly for $q \in \mathbb{R}^n \setminus X$.

Theorem 1. Consider the PETC system Σ with the time and state space sampling parameters $\tau, \eta \in \mathbb{R}^+$. Let the map $\mathscr{F} : \mathfrak{X}_1 \to \mathfrak{X}_2$ be given by $\mathscr{F}((x,u,l)) = (\hat{q},v)$ if and only if $x \in \hat{q}$. Then $T_\tau(\Sigma) \preceq_{\mathscr{F}} \mathbf{EFSM}(\Sigma)$.

In $\mathbf{EFSM}(\Sigma)$, the set $\mathfrak{X}_2$ is based on the partition $\hat{X}$. If the set $\mathfrak{X}_2$ is derived by $[X]_\eta \times [U]_\mu$, then the next abstract state is defined as follows

$$\{q' \in \mathfrak{X}_2 : \|\mathbf{x}(\tau,q,v) - q'\| \leq \eta/2 + e^{\lambda_{\max}(A)\tau}\varpi\}, \tag{11}$$

and v' is the same as in (8)-(10), where $\varpi \in \mathbb{R}^+$ is the desired precision such that $\eta < 2\varpi$ and $\lambda_{\max}(A)$ is the maximal singular value of A. In this case, the state set $\mathfrak{X}_2$ and the transition relation is redefined, and thus we obtain an alternative symbolic model $\mathbf{EFSM}_1(\Sigma)$.

Proposition 2. Consider the PETC system Σ with the time and state space sampling parameters $\tau, \eta \in \mathbb{R}^+$. Given a desired precision $\varpi \in \mathbb{R}^+$, if the map $\mathscr{F} : \mathfrak{X}_1 \to \mathfrak{X}_2$ is given by

$$\mathscr{F}((x,u,l)) = \{(q,v) \in \mathfrak{X}_2 : \|x - q\| \leq \varpi\}, \tag{12}$$

then $T_\tau(\Sigma) \preceq_{\mathscr{F}} \mathbf{EFSM}_1(\Sigma)$.

The approximation precision is not studied in Theorem 1 and depends on the choice of $\eta \in \mathbb{R}^+$, whereas it is studied in Proposition 2 and constraints the choice of $\eta \in \mathbb{R}^+$. Note that since $\hat{X}$ is partition of X in $\mathbf{EFSM}(\Sigma)$, the feedback refinement relation obtained in Theorem 1 is a special case of the alternating simulation relation as in (Tabuada (2009)) with the equivalence of the control inputs for $T_\tau(\Sigma)$ and $\mathbf{EFSM}(\Sigma)$.

From the perspective of the unfolding interpretation (Teixeira et al. (2014)), the abstraction $\mathbf{EFSM}(\Sigma)$ can be further transformed into an FSM given by $\mathbf{FSM}(\Sigma) = (\mathfrak{X}_3, \mathfrak{X}_3^0, \mathbb{E}_3, \Delta_3)$,

- the set of states is $\mathfrak{X}_3 := [X]_\eta \times [U]_\mu \times \mathbb{N}$;
- the set of initial states is $\mathfrak{X}_3^0 := [X]_\eta \times [U]_\mu \times \{0\}$;
- the set of events is $\mathbb{E}_3 := \{\pi, \texttt{TG}, \texttt{NTG}\}$;
- the transition relation $\Delta_3 \subseteq \mathfrak{X}_3 \times \mathbb{E}_3 \times \mathfrak{X}_3$ is as follows: for $(q,v,l) \in \mathfrak{X}_3$, $\sigma \in \mathbb{E}_3$ and $l,l' \in \mathbb{N}$, $(q,v,l) \xrightarrow{\sigma} (q',v',l')$

Algorithm 1 Determinization of $\mathbf{DES}(\Sigma)$

Input: $\mathbf{DES}(\Sigma)$
Output: A deterministic discrete event system $\overline{\mathbf{DES}}(\Sigma)$
Initial: $\bar{\mathfrak{X}}_4 = \mathfrak{X}_4$ and $\mathbb{E}' = \varnothing$

1: **for** $q \in \mathfrak{X}_4$ and $\sigma \in \mathbb{E}_4$ **do**
2:     **if** $|\Delta_4(q,\sigma)| > 1$ **then**
3:         $\bar{\mathfrak{X}}_4 = \mathfrak{X}_4 \cup \{\boldsymbol{q}\}$ with the auxiliary state $\boldsymbol{q}$
4:         $\bar{\Delta}_4(q,\sigma) = \{\boldsymbol{q}\}$
5:         $\bar{\Delta}_4(\boldsymbol{q},\varepsilon) = \Delta_4(q,\sigma)$
6: **for** $q \in \bar{\mathfrak{X}}_4$ **do**
7:     **if** $\bar{\Delta}_4(q,\varepsilon) = \{q_1,\dots,q_m\}$ **then**
8:         Introduce the auxiliary events: $\tau_1,\dots,\tau_m$
9:         $\bar{\Delta}_4(q,\tau_1) = \{q_1\},\dots,\bar{\Delta}_4(q,\tau_m) = \{q_m\}$
10:         $\mathbb{E}' = \mathbb{E}' \cup \{\tau_1,\dots,\tau_m\}$
11: $\bar{\mathbb{E}}_4 = \mathbb{E}_4 \cup \mathbb{E}'$
12: **return** $\overline{\mathbf{DES}}(\Sigma) = (\bar{\mathfrak{X}}_4, \mathfrak{X}_4^0, \bar{\mathbb{E}}_4, \bar{\Delta}_4)$

if and only if $(q,v) \overset{\sigma:p}{\to} (q',v')$ such that $p(l,l') = \texttt{true}$, where $(q,v) \overset{\sigma:p}{\to} (q',v')$ denotes the existence of a transition from state (q,v) to state (q',v') with q' satisfying (9), the event $\sigma \in \mathbb{E}_3$ and the update $p \in \Pi_{\mathrm{v}}$.

A special case of Assumption 1 is that $N = 1$, which implies that the state variable l is not needed. In this case, the symbolic abstraction of $T_\tau(\Sigma)$ can be presented directly as a discrete event system $\mathbf{DES}(\Sigma) = (\mathfrak{X}_4, \mathfrak{X}_4^0, \mathbb{E}_4, \Delta_4)$, where

- the state set is $\mathfrak{X}_4 := [X]_\eta \times [U]_\mu$;
- the set of initial states is $\mathfrak{X}_4^0 := [X]_\eta \times [U]_\mu$;
- the event set is $\mathbb{E}_4 := \{\texttt{TG}, \texttt{NTG}\}$;
- the transition relation $\Delta_4 \subseteq \mathfrak{X}_4 \times \mathbb{E}_4 \times \mathfrak{X}_4$ is a special case of Δ_3 in $\mathbf{FSM}(\Sigma)$ without the state variable l.

4. SYMBOLIC SUPERVISORY CONTROL

Since symbolic abstractions of PETC systems are constructed in the DES form, we now apply the supervisory control theory of DES to achieve the desired specifications. We start by studying $\mathbf{DES}(\Sigma)$ (i.e., the case $N = 1$), and then extend the obtained results to $\mathbf{EFSM}(\Sigma)$.

4.1 Determinization of Symbolic Abstractions

For a DES $G = (X, X_0, \mathbb{E}, \Delta)$, the generated language is denoted by $\mathfrak{L}(G)$. The event set $\mathbb{E}$ is partitioned as the controllable event set $\mathbb{E}_c$ and the uncontrollable event set $\mathbb{E}_{uc}$ in terms of controllability; the observable event set $\mathbb{E}_o$ and the unobservable event set $\mathbb{E}_{uo}$ in terms of observability. That is, $\mathbb{E} = \mathbb{E}_c \cup \mathbb{E}_{uc} = \mathbb{E}_o \cup \mathbb{E}_{uo}$. Here, $\mathbb{E}_{uc} = \{\pi\}$ and $\mathbb{E}_{uo} = \varnothing$ for both $\mathbf{EFSM}(\Sigma)$ and $\mathbf{FSM}(\Sigma)$, whereas $\mathbb{E}_{uc} = \mathbb{E}_{uo} = \varnothing$ for $\mathbf{DES}(\Sigma)$. That is, all the events in $\mathbf{DES}(\Sigma)$ are controllable and observable, whereas the event π in $\mathbf{EFSM}(\Sigma)$ and $\mathbf{FSM}(\Sigma)$ is uncontrollable.

Definition 3. (Wonham and Cai (2018)). A language $K \subseteq \mathfrak{L}(G)$ is: *prefix-closed* if it equals to its prefix closure $\bar{K}$; *controllable* if for all $s \in \bar{K}$ and $\sigma \in \mathbb{E}_{uc}$, $s\sigma \in \mathfrak{L}(G)$ implies $s\sigma \in \bar{K}$; *observable* if $s_2\sigma \in \bar{K}$ holds for all $s_1, s_2 \in \bar{K}$ with $\mathscr{P}(s_1) = \mathscr{P}(s_2)$ and $\sigma \in \mathbb{E}$ with $s_1\sigma \in \bar{K}$ and $s_2\sigma \in \mathfrak{L}(G)$, where $\mathscr{P} : \mathbb{E}^* \to \mathbb{E}_o^*$ is the projection map to delete the unobservable events and $\mathbb{E}^*$ is the set of all finite strings over $\mathbb{E}$, including the empty string ε.

Definition 4. (Cassandras and Lafortune (2009)). A DES $G = (X, X_0, \mathbb{E}, \Delta)$ is, *deterministic* if there cannot be two transitions with the same event out of a state; *nondeterministic* if there can be more than one transition with the same event out of a state.

Obviously, $\mathbf{DES}(\Sigma)$ is nondeterministic, but the language generated by $\mathbf{DES}(\Sigma)$ is both controllable and observable. We first provide a procedure to construct the determinization of $\mathbf{DES}(\Sigma)$, which is given in Algorithm 1. Algorithm 1 consists of two steps: the first step (lines 1-5) is to extend $\mathbf{DES}(\Sigma)$ to a nondeterministic automaton with ε-transitions (which refer to the transitions on the empty string; see Cassandras and Lafortune (2009)), and the second step (lines 6-10) is to replace all the ε labels by the labels from $\mathbb{E}'$. In Algorithm 1, the auxiliary event set $\mathbb{E}' = \{\tau_1, \tau_2, \dots\}$ is introduced (line 10), and thus the extended event set is given as $\bar{\mathbb{E}}_4 = \mathbb{E}_4 \cup \mathbb{E}'$ (line 11). Hence, we obtain a novel transition system $\overline{\mathbf{DES}}(\Sigma)$.

Lemma 3. The system $\overline{\mathbf{DES}}(\Sigma)$ generated by Algorithm 1 is deterministic and $\mathbf{DES}(\Sigma) = \overline{\mathbf{DES}}(\Sigma) \backslash \mathbb{E}'$, where $\overline{\mathbf{DES}}(\Sigma) \backslash \mathbb{E}'$ is the DES after removing the auxiliary event set $\mathbb{E}'$.

Lemma 3 implies that the nondeterministic DES is lifted to a deterministic DES such that the projection of this deterministic DES equals to the original nondeterministic DES. Note that the auxiliary event set $\mathbb{E}'$ is unobservable in $\overline{\mathbf{DES}}(\Sigma)$. Similarly, we can obtain the determinization of $\mathbf{EFSM}(\Sigma)$ and $\mathbf{FSM}(\Sigma)$ using a similar technique. For instance, to obtain $\overline{\mathbf{EFSM}}(\Sigma)$, we can set $\bar{\Delta}_2(q,\sigma,p) = \{\boldsymbol{q}\}$ (line 4), $\bar{\Delta}_2(\boldsymbol{q},\varepsilon,p) = \Delta_2(q,\sigma,p)$ (line 5), and $\bar{\Delta}_2(q,\tau_1,p) = \{q_1\}, \dots, \bar{\Delta}_2(q,\tau_m,p) = \{q_m\}$ (line 9).

4.2 Existence of Symbolic Supervisors

Assume that the desired specification is a given language. Therefore, for a DES G, a supervisor is a map $\mathscr{S} : \mathscr{P}(\mathfrak{L}(G)) \to 2^{\mathbb{E}_c}$ to disable some controllable events such that the language of the supervised system, denoted by $\mathfrak{L}(\mathscr{S}, G)$, satisfies the desired specification. In addition, the supervisor needs to be safe. That is, the forbidden states, which refer to the states that the system is not allowed to visit, should be avoided. For the system $\mathbf{DES}(\Sigma)$, the forbidden state set is assumed to be $\mathfrak{X}_{\mathrm{f}}$, and then the safe state set is $\mathfrak{X}_4 \backslash \mathfrak{X}_{\mathrm{f}}$. Since the auxiliary states and events are introduced in $\overline{\mathbf{DES}}(\Sigma)$, the forbidden state set for $\overline{\mathbf{DES}}(\Sigma)$ is defined by $\bar{\mathfrak{X}}_{\mathrm{f}} := \mathfrak{X}_{\mathrm{f}} \cup \{q \in \bar{\mathfrak{X}}_4 \backslash \mathfrak{X}_4 : \bar{\Delta}_4(q,s) \in \mathfrak{X}_{\mathrm{f}}, s \in \bar{\mathbb{E}}_4^*\}$, which includes $\mathfrak{X}_{\mathrm{f}}$ and all the auxiliary states starting from which the reachable states are in $\mathfrak{X}_{\mathrm{f}}$. Furthermore, the legal language for $\overline{\mathbf{DES}}(\Sigma)$ is defined as $\mathfrak{S} = \{s \in \mathfrak{L}(\overline{\mathbf{DES}}(\Sigma)) : \bar{\Delta}_4(q_0, r) \notin \bar{\mathfrak{X}}_{\mathrm{f}} \text{ for all } r \text{ with } |r| \leq |s|\}$, which is the set of all the strings in $\mathfrak{L}(\overline{\mathbf{DES}}(\Sigma))$ that only visit the safe states in $\overline{\mathbf{DES}}(\Sigma)$.

Proposition 4. Consider the PETC system Σ with $\tau_e = \tau$ and its symbolic model $\mathbf{DES}(\Sigma)$. A supervisor $\mathscr{S}$ for $\mathbf{DES}(\Sigma)$ exists such that $\mathfrak{L}(\mathscr{S}, \mathbf{DES}(\Sigma))$ satisfies the desired specification if and only if the legal language $\mathfrak{S}$ is controllable and observable with respect to $\mathfrak{L}(\overline{\mathbf{DES}}(\Sigma))$.

Proposition 4 provides the verification for the existence of the supervisor for $\mathbf{DES}(\Sigma)$ using its determinization $\overline{\mathbf{DES}}(\Sigma)$. To be specific, from Zhao et al. (2012) and Wonham and Cai (2018), the controllability and observability of $\mathfrak{S}$ imply the existence of the supervisor for $\overline{\mathbf{DES}}(\Sigma)$, which is further applied in Proposition 4 to show the existence of the supervisor for $\mathbf{DES}(\Sigma)$. Note that the projection of the supervisor for $\overline{\mathbf{DES}}(\Sigma)$ is the supervisor for $\mathbf{DES}(\Sigma)$.

Next, we extend the obtained results to $\mathbf{EFSM}(\Sigma)$. Assume that the determinization of $\mathbf{EFSM}(\Sigma)$ is $\overline{\mathbf{EFSM}}(\Sigma)$, and that the desired specification is given as a deterministic EFSM S, whose language is denoted by $\mathfrak{L}(S) \subseteq \mathbb{E}_2^*$.

Definition 5. (Teixeira et al. (2014)). For an EFSM $T = (X, X_0, V, \mathbb{E}, \Delta)$, it is *state-deterministic* if $|X_0| \leq 1$, and $\Delta(x, \sigma, p) = \Delta(x, \sigma, p')$ for all $x \in X$, $\sigma \in \mathbb{E}$, and $p, p' \in \Pi_v$; *V-deterministic* if $v_1 = v_2$ for $x' = \Delta(x, \sigma, p(v, v_1)) = \Delta(x, \sigma, p(v, v_2))$ with $p \in \Pi_v$. An update $p \in \Pi_v$ is *total*, if for all $v \in \mathrm{dom}(V)$, there exists $v' \in \mathrm{dom}(V)$ such that $p(v, v') = \mathtt{true}$. An EFSM is *total* if all its updates are total.

$\overline{\mathbf{EFSM}}(\Sigma)$ is state-deterministic, whereas $\mathbf{EFSM}(\Sigma)$ is not. The V-deterministic and total properties of $\overline{\mathbf{EFSM}}(\Sigma)$ and $\mathbf{EFSM}(\Sigma)$ come from the construction of $\mathbf{EFSM}(\Sigma)$ directly.

Definition 6. (Cassandras and Lafortune (2009)). Given two EFSMs $T_1 = (X_1, X_1^0, V_1, \mathbb{E}_1, \Delta_1)$ and $T_2 = (X_2, X_2^0, V_2, \mathbb{E}_2, \Delta_2)$, the *synchronous composition* of T_1 and T_2 is $T_1 \| T_2 = (X_1 \times X_2, X_1^0 \times X_2^0, V_1 \cup V_2, \mathbb{E}_1 \cup \mathbb{E}_2, \Delta)$, where,

- $(x_1', x_2') = \Delta(x_1, x_2, \sigma, p_1, p_2)$ if $\sigma \in \mathbb{E}_1 \cap \mathbb{E}_2$, $x_1' = \Delta_1(x_1, \sigma, p_1)$, and $x_2' = \Delta_2(x_2, \sigma, p_2)$;
- $(x_1', x_2) = \Delta(x_1, x_2, \sigma, p_1)$ if $\sigma \in \mathbb{E}_1 \backslash \mathbb{E}_2$ and $x_1' = \Delta_1(x_1, \sigma, p_1)$;
- $(x_1, x_2') = \Delta(x_1, x_2, \sigma, p_2)$ if $\sigma \in \mathbb{E}_2 \backslash \mathbb{E}_1$ and $x_2' = \Delta_2(x_2, \sigma, p_2)$.

Definition 7. (Teixeira et al. (2014)). Given two deterministic EFSMs $T_1 = (X_1, X_1^0, V, \mathbb{E}, \Delta_1)$ and $T_2 = (X_2, X_2^0, V, \mathbb{E}, \Delta_2)$, T_1 is *V-controllable* with respect to T_2, if for all $\sigma \in \mathbb{E}_u$, all $v, v' \in \mathrm{dom}(V)$, and $x \in X_1, q \in X_2$ under the same finite string in $\mathbb{E}^*$, $q' = \Delta_2(q, \sigma, p(v, v')) \in X_2$ implies $x' = \Delta_1(x, \sigma, p(v, v')) \in X_1$.

The synchronous composition of $\overline{\mathbf{EFSM}}(\Sigma)$ and S is given by $\overline{\mathbf{EFSM}}(\Sigma) \| S$. If the language $\mathfrak{L}(S)$ of the specification S is controllable and observable, then there exists a supervisor to ensure the satisfaction of the specification; see Zhao et al. (2012). If the language $\mathfrak{L}(S)$ is not controllable, then our aim is to find the largest controllable sublanguage of $\mathfrak{L}(S)$. For this case, define the set $\mathscr{C} := \{\mathscr{K} \subseteq \overline{\mathbf{EFSM}}(\Sigma) \| S : \mathscr{K}$ is V-controllable with respect to $\overline{\mathbf{EFSM}}(\Sigma)\}$, which contains all the V-controllable subsystems of $\overline{\mathbf{EFSM}}(\Sigma) \| S$. The largest subsystem in $\mathscr{C}$ is called the supremal $\overline{\mathbf{EFSM}}$ and denoted by $\sup \mathscr{C}(S, \overline{\mathbf{EFSM}}(\Sigma))$, which represents the most permissive behavior implemented in $\overline{\mathbf{EFSM}}(\Sigma)$ such that S is satisfied. In terms of languages, we use the notation $\sup \mathscr{C}_{\mathfrak{L}}(\mathfrak{L}(S), \mathfrak{L}(\overline{\mathbf{EFSM}}(\Sigma)))$ to denote the largest sublanguage of $\mathfrak{L}(S)$ achieved by controlling the behavior $\mathfrak{L}(\overline{\mathbf{EFSM}}(\Sigma))$.

Proposition 5. Consider the PETC system Σ and its symbolic model $\mathbf{EFSM}(\Sigma)$ with the determinization $\overline{\mathbf{EFSM}}(\Sigma)$. If the desired specification S is state-deterministic, then we have $\mathfrak{L}(\sup \mathscr{C}(S, \overline{\mathbf{EFSM}}(\Sigma))) = \sup \mathscr{C}_{\mathfrak{L}}(\mathfrak{L}(\overline{\mathbf{EFSM}}(\Sigma) \| S), \mathfrak{L}(\overline{\mathbf{EFSM}}(\Sigma)))$.

Proposition 5 implies the equivalence between the language of the largest subsystem $\sup \mathscr{C}(S, \overline{\mathbf{EFSM}}(\Sigma))$ in $\mathscr{C}$ and the largest sublanguage of $\mathfrak{L}(\overline{\mathbf{EFSM}}(\Sigma) \| S)$ achieved by controlling the plant behavior $\mathfrak{L}(\overline{\mathbf{EFSM}}(\Sigma))$. That is, the largest controllable sublanguage of $\mathfrak{L}(S)$ can be found and achieved by controlling $\mathfrak{L}(\overline{\mathbf{EFSM}}(\Sigma))$. Furthermore, the projection of the supervisor for $\overline{\mathbf{EFSM}}(\Sigma)$ is the supervisor for $\mathbf{EFSM}(\Sigma)$.

4.3 From Supervisors to PETMs

The final step is to derive the PETM for the system $T_\tau(\Sigma)$ from the supervisor (i.e., the controller) for the symbolic abstractions. According to Theorem 1 and Proposition 2, if there exists a controller such that the symbolic abstraction $\mathbf{EFSM}(\Sigma)$ (or $\mathbf{DES}(\Sigma)$) satisfies the desired specification, then this supervisor can be applied with the map $\mathscr{F}$ to derive a controller ensuring the satisfaction of the same specification for $T_\tau(\Sigma)$.

Now, the key is how to determine the parameter $\delta \in \mathbb{R}^+$ in the PETM for $T_\tau(\Sigma)$ via the supervisor for $\mathbf{EFSM}(\Sigma)$ (or $\mathbf{DES}(\Sigma)$). First, assume that the supervisor for $\mathbf{EFSM}(\Sigma)$ (or $\mathbf{DES}(\Sigma)$) exists, and thus we can obtain from the supervisor that there exists a string $s \in \mathbb{E}_2^*$ (or $s \in \mathbb{E}_4^*$) such that the specification is satisfied. From this string, we have all the events that occur at the sampling times, and thus achieve the information on the triggering event $\mathtt{TG}$. Second, we can computer the relation between the measurement error $q - \bar{q}$ and the abstract state q at the occurrence time instances of the events $\mathtt{TG}$ and $\mathtt{NTG}$. For instance, we obtain $\|q - \bar{q}\| / \|q\|$ at the occurrence time instances of the event $\mathtt{TG}$. Third, based on the relation between $q - \bar{q}$ and q at the sampling times and the occurrences of the events $\mathtt{TG}$, we can determine the parameter $\delta \in \mathbb{R}^+$ and construct a piecewise PETM. Note that the generated PETM may not possess a uniform form. That is, the parameter δ is not necessarily the same along the timeline, but is piecewise constant. Therefore, a PETM can be generated from the string obtained by the supervisor.

5. NUMERICAL EXAMPLE

Consider the following second-order dynamics of an agent

$$\Sigma : \dot{x}_1 = x_2, \quad \dot{x}_2 = -2x_1 + 3x_2 + u, \tag{13}$$

and the state-feedback controller (2) is given by $u = x_1 - 4x_2$ with the following PETM:

$$\begin{aligned} t_{k+1} = \min\{ & t = t_k + j\tau_e : j \in \mathbb{N}, \\ & \mathbb{C}(x(t), \bar{x}(t)) = \|x(t) - \bar{x}(t)\| - \delta \|x(t)\| \geq 0\}. \end{aligned} \tag{14}$$

Assume that the state space is $X = [-0.5, 1.5] \times [-1.5, 3.5]$, the input space is $U = [-14.5, 7.5]$. Our aim is to design a supervisor (i.e., the controller) such that the agent starting from $(0, 3)$ reaches the origin as close as possible by avoiding an obstacle, which is the cyan region in Fig. 1.

To this end, we first construct the symbolic model for the time discretization of Σ. Let $\tau = 0.15, \varpi = 0.4, \eta = 0.1, \mu = 0.2$ and $\tau_e = 2\tau$, which implies that $\mathbb{N} = \{0, 1\}$. The resulting symbolic model is given by $\mathbf{EFSM}(\Sigma) = (\mathfrak{X}, \mathfrak{X}_0, V, \mathbb{E}, \Delta)$ with (i) $\mathfrak{X} = [X]_{0.1} \times [U]_{0.2}$; (ii) $\mathfrak{X}_0 = \mathfrak{X}$; (iii) $V = \{l\}$ with $\mathrm{dom}(V) = \mathbb{N}$; (iv) $\mathbb{E} := \{\pi, \mathtt{TG}, \mathtt{NTG}\}$; and (v) the transition relation Δ given as follows: $((q, v), \sigma, p, (q', v')) \in \mathfrak{X} \times \mathbb{E} \times \Pi_v \times \mathfrak{X}$ if and only if $p(l, l') = \mathtt{true}$, $\|\mathbf{x}(\tau, q, v) - q'\| \leq 0.05 + 0.4e^{0.3}$ and

$$\Delta((q, v), \pi, p) = \Delta((q, v), \mathtt{NTG}, p) = (q', v),$$
$$\Delta((q, v), \mathtt{TG}, p) = (q', Q_\mu(Kq)).$$

It takes 268.647 seconds (Intel Core i7, 1.9GHz) resulting in 1071 abstract states, 110 abstract inputs and 354456 transitions. By Proposition 2, we can establish the feedback refinement relation from $T_{0.15}(\Sigma)$ to $\mathbf{EFSM}(\Sigma)$.

Second, we transform $\mathbf{EFSM}(\Sigma)$ into its deterministic version by adding the auxiliary states and events, and obtain $\overline{\mathbf{EFSM}}(\Sigma) = (\bar{\mathfrak{X}}, \mathfrak{X}_0, \bar{\mathbb{E}}, \bar{\Delta})$. Since we focus on the path from the starting point $(0, 3)$ to the target region given by the orange

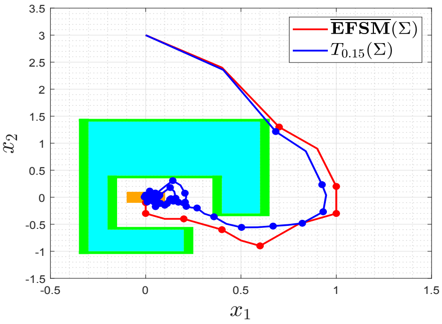

Fig. 1. Sate trajectories under the PETM (15) and the refined PETM (16). The dots denote the event triggering. The cyan region is the original obstacle; the union of the cyan and green regions is the obstacle for the symbolic abstraction.

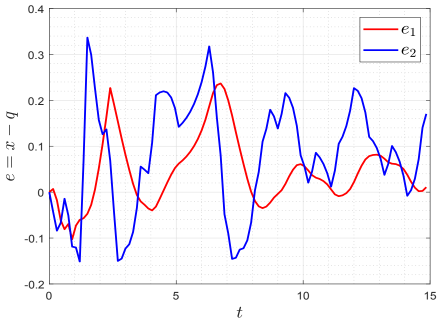

Fig. 2. The error between the abstract state and the original state, which shows that the desired precision is achieved.

region in Fig. 1, we can extract an EFSM with the initial state fixed at $(0,3)$ and the marked state set including those in the orange region, which is denoted by $\mathbf{EFSM_r}(\Sigma)$, such that the numbers of the auxiliary states and events are reduced. For $\overline{\mathbf{EFSM_r}}(\Sigma)$, we can derive the following admissible string $s = \{\mathtt{NTG}, \pi, \mathtt{TG}, \pi, \mathtt{TG}, \pi, \mathtt{TG}, \pi, \mathtt{NTG}, \pi, \mathtt{TG}, \pi, \mathtt{TG}, \pi, \mathtt{TG}, \pi, \mathtt{TG}, \pi, \mathtt{TG}, \pi, \ldots\}$ such that the agent moves to a neighbour of the origin; see the red line in Fig. 1. The generated string can be reformulated as the following PETM:

$$t_{k+1} = \min\{t = t_k + j\tau_e : j \in \mathbb{N}, \quad \|q(t) - \bar{q}(t_k)\| \geq 0.4\|q(t)\|\}. \tag{15}$$

Here, we take $\delta = 0.4$. Therefore, the controller for the system (13)-(14) is $u_1(t) = Q_\mu(Q_\eta(x_1(t_k)) - 4Q_\eta(x_2(t_k)))$ with $t \in [t_k, t_{k+1})$, and the PETM refined from (15) is given by

$$t_{k+1} = \min\{t = t_k + j\tau_e : j \in \mathbb{N}, \quad \|\mathscr{F}_X(x(t)) - \mathscr{F}_X(\bar{x}(t_k))\| \geq 0.4\|\mathscr{F}_X(x(t))\|\}, \tag{16}$$

where $\mathscr{F}_X$ is the projection of the relation $\mathscr{F}$ on the state set X. Under the PETM (16), the state trajectory of the time discretization $T_{0.15}(\Sigma)$ are given in Fig. 1. Observe from Figs. 1-2 that the desired specification is satisfied for both the symbolic model $\overline{\mathbf{EFSM_r}}(\Sigma)$ and the system $T_{0.15}(\Sigma)$.

6. CONCLUSIONS

In this paper, symbolic supervisory control was studied for periodic event-triggered control systems. We first constructed a symbolic model for periodic event-triggered control systems in the form of a discrete event system. Based on the constructed symbolic model, we further studied the supervisory control for periodic event-triggered control systems. Finally, a numerical example was given to illustrate the obtained results. Future work will focus on symbolic supervisory control for distributed event-triggered control systems.

REFERENCES

Cassandras, C.G. and Lafortune, S. (2009). *Introduction to Discrete Event Systems*. Springer Science & Business Media.

Dimarogonas, D.V., Frazzoli, E., and Johansson, K.H. (2012). Distributed event-triggered control for multi-agent systems. *IEEE Transactions on Automatic Control*, 57(5), 1291–1297.

Fu, A. and Mazo, M. (2018). Traffic models of periodic event-triggered control systems. *IEEE Transactions on Automatic Control*.

Girard, A. and Pappas, G.J. (2007). Approximation metrics for discrete and continuous systems. *IEEE Transactions on Automatic Control*, 5(52), 782–798.

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

Heemels, W.H., Donkers, M., and Teel, A.R. (2012). Periodic event-triggered control for linear systems. *IEEE Transactions on automatic control*, 58(4), 847–861.

Heymann, M. and Lin, F. (1998). Discrete-event control of nondeterministic systems. *IEEE Transactions on Automatic Control*, 43(1), 3–17.

Hopcroft, J.E. (2008). *Introduction to Automata Theory, Languages, and Computation*. Pearson Education India.

Kolarijani, A.S. and Mazo, M. (2018). Formal traffic characterization of LTI event-triggered control systems. *IEEE Transactions on Control of Network Systems*, 5(1), 274–283.

Milner, R. (1989). *Communication and Concurrency*. Prentice Hall.

Pola, G., Pepe, P., Di Benedetto, M.D., and Tabuada, P. (2010). Symbolic models for nonlinear time-delay systems using approximate bisimulations. *Systems & Control Letters*, 59(6), 365–373.

Pola, G., Pepe, P., and Di Benedetto, M.D. (2017). Decentralized supervisory control of networks of nonlinear control systems. *IEEE Transactions on Automatic Control*, 63(9), 2803–2817.

Ramadge, P.J. and Wonham, W.M. (1987a). Modular feedback logic for discrete event systems. *SIAM Journal on Control and Optimization*, 25(5), 1202–1218.

Ramadge, P.J. and Wonham, W.M. (1987b). Supervisory control of a class of discrete event processes. *SIAM Journal on Control and Optimization*, 25(1), 206–230.

Reissig, G., Weber, A., and Rungger, M. (2017). Feedback refinement relations for the synthesis of symbolic controllers. *IEEE Transactions on Automatic Control*, 62(4), 1781–1796.

Tabuada, P. (2009). *Verification and Control of Hybrid Systems: A Symbolic Approach*. Springer Science & Business Media.

Teixeira, M., Malik, R., Cury, J.E., and de Queiroz, M.H. (2014). Supervisory control of DES with extended finite-state machines and variable abstraction. *IEEE Transactions on Automatic Control*, 60(1), 118–129.

Wonham, W.M. and Cai, K. (2018). *Supervisory Control of Discrete-Event Systems*. Springer.

Zamani, M., Esfahani, P.M., Majumdar, R., Abate, A., and Lygeros, J. (2014). Symbolic control of stochastic systems via approximately bisimilar finite abstractions. *IEEE Transactions on Automatic Control*, 59(12), 3135–3150.

Zhao, J., Chen, Y.L., Chen, Z., Lin, F., Wang, C., and Zhang, H. (2012). Modeling and control of discrete event systems using finite state machines with variables and their applications in power grids. *Systems & Control Letters*, 61(1), 212–222.