Decentralized supervisory control of continuous timed discrete event systems

Fei WANG, Fu-jiang JIN, Ji-liang LUO

(College of Information Science and Engineering, Huaqiao University, Xiamen, Fujian, 361021, P. R. China)

Abstract—In this paper, we presented the decentralized supervisory control problem of discrete event system with continuous-time variable. By presenting the definition of coobservability for the timed specification, a necessary and sufficient condition for the existence of decentralized supervisors is obtained. Finally, a numerical example is given.

Keywords—Discrete event systems; Decentralized supervisors; Coobservability.

1 Introduction

Discrete event systems (DES) are systems which states are driven by events. Ramadge and Wonham [1], [2] have studied the logical supervisory control theory instantaneously. For distributed systems, such as communication systems and manufacturing systems, decentralized supervisors are more suitable than a centralized one. Decentralized supervisory control has been initiated in [3-7]. Recently, some issues of decentralized supervisory control, such as the state feedback problem [8-9], general architecture [10], reliability [11-12] and synthesis problem with communication [13-15], have been studied.

In the range of control problem, real time is more suitable than instantaneity. Supervisory control of discrete timed DES (DTDES) has been studied by Brandin and Wonham, i.e. [16] has considered the time feature of [1] and introduced the event tick to represent ‘tick of global clock’. Obviously, the advantage of event tick incorporated into logical DES lies in preserving logical feature, while the disadvantage lies that can cause state explosion. To avoid state explosion, discrete timed DES has been extended by other researchers, i.e. [17] and [18] have introduced dense event and solved supervisory control problem on the base of state space of timed discrete event systems. [19] has eliminated event time and incorporated time information in the state; [20] has constrained the time information of [19] to be in the eligible time bounds; [21] has extended the model of [20] and considered robust supervisory control problem of uncertain DTDES; [22] has extended full observation [16] to partial observation; and [23] has solved robust supervisory control for partially observed DTDES, which is an extension of [24].

In the manufacture cells and logical cells, it always takes some time to operate and handle. In general, the service time of the operation is continuous variable in an interval. Under the circumstances, continuous time and timed control are considered as a new dimension of timed-DES in [25-26]. By using the model, the state explosion can be reduced in discrete-time and continuous-time models. In this paper, we have developed the model of [25-26] and introduced the synthesis problem of decentralized supervisory control for timed-DES. To solve the synthesis problem, a necessary and sufficient condition for the existence of decentralized supervisors has been presented.

2 Supervisory control of DES

In the model of [1-2], the plant to be controlled is modelled by an automaton $G = (Q, \Sigma, \delta, q_0)$, where $Q$ is a countable state space, $\Sigma$ is a finite event set, $\delta$ is a partial function from $Q \times \Sigma$ to $Q$, and $q_0 \in Q$ is an initial state. Let $\Sigma^*$ denote the set of all finite strings on $\Sigma$ including the empty string $\epsilon$. $\delta$ can be generalized by $\delta: Q \times \Sigma^* \rightarrow Q$. The language generated by the DES $G$ is defined by $L(G) = \{s \in \Sigma^* | \delta(q_0, s)\}$ and means the set of all possible event sequences. Let $K \subseteq \Sigma^*$ be a language. We denote the set of all prefixes of traces in $K$ by $\text{pre}(K)$. $K$ is (prefix-)closed if $K = \text{pre}(K)$.

The event set $\Sigma$ is divided into an uncontrollable event set $\Sigma_u$ and a controllable event set $\Sigma_c$. A language $K$ is controllable if $\text{pre}(K)\Sigma_u \cap L(G) \subseteq \text{pre}(K)$. A control input is an event subset $\gamma$ satisfying $\Sigma_c \subseteq \Sigma$. The set of control inputs is denoted by $\Gamma$. A supervisor is an enable map $f : L(G) \rightarrow \Gamma$. Formally, the
language generated by the supervised system $f / G$, denoted by $L(f / G)$ [1-2], is defined as follows.

- $\epsilon \in L(f / G)$, where $\epsilon$ is the empty string.
- $(\forall s \in L(f / G)): (\sigma \epsilon \Sigma) \sigma \epsilon L(f / I G) \epsilon \sigma \epsilon L(G) \wedge \sigma \epsilon f(s)$.

For a nonempty and closed languages $K$, there exists a supervisor $s$ for $G$ such that $L(f / G) = K$ if and only if $K$ is controllable.

Let $\Sigma_{o}$ be the set of observable events. An observable function is a natural projection $P : \Sigma \rightarrow \Sigma_{0}$ which is satisfied with $P(\epsilon) = \epsilon$, $P(\sigma) = \begin{cases} \sigma & \text{if } \sigma \in \Sigma_{o}, \\ \epsilon & \text{otherwise} \end{cases}$ for all $\sigma \in \Sigma$. A partial supervisor which enables and disables any of the controllable events through its observation of the sequence of events as it is generated by $G$ is a map $f : P(L(G)) \rightarrow \Gamma$. A language $K$ is said to be observable if $P(s) = \sigma$, $s \epsilon L(G)$ and $s \epsilon K$ imply $s \epsilon K$ for any $s, s \epsilon K$ and $s \epsilon \Sigma_{o}$. For a nonempty and closed languages $K$, there exists a partial supervisors $f$ for $G$ such that $L(f / G) = K$ if and only if $K$ is controllable and observable.

3 Supervisory control of continuous timed-DES

Continuous timed-DES is modelled by an automaton $G = (Q, \Sigma_{e}, \delta, q_{0})$ in [26], where $\Sigma_{e}$ is the set of timed events, and $\delta : Q \times \Sigma_{e} \rightarrow Q$ is a partial function, e.g. $(\sigma, t_{o})$ is fired in $\tau$ time if logical event $\sigma$ is enable. Let $\Sigma_{t}$ be the set of all finite timed strings of elements in $\Sigma_{e}$, including the empty string $\epsilon$. The function $\delta$ can be generalized to $\delta : Q \times \Sigma_{t} \rightarrow Q$ in the natural way. The timed event $\Sigma_{t}$ can be divided into controllable event set $\Sigma_{c}$ and uncontrollable event set $\Sigma_{u}$. To describe the relations between logical event and timed event, we make the following assumption in [26]:

1) For any timed controllable event $(\sigma, t_{o})$, logical event $\sigma$ is controllable.
2) For any timed uncontrollable event $(\sigma, t_{o})$, logical event $\sigma$ is uncontrollable.

From the assumption, we have $(\sigma, t_{o}) \in \Sigma_{c} \Leftrightarrow \exists \sigma \in \Sigma_{c}$ and $(\sigma, t_{o}) \in \Sigma_{u} \Leftrightarrow \sigma \in \Sigma_{u}$.

Let $TL(G)$ be the timed language generated by $G$, that is $TL(G) = \{ s | \delta(q, s), s \epsilon \Sigma_{t} \}$, where $s$ is a timed string. The traces of $TL(G)$ is defined as $L(G) = tr(TL(G))$, where $tr(\cdot)$ is the function of logical trace sets for timed language.

Definition 1: A timed language $K$ is said to be (prefix-)closed [26] if $pre(K)$.

Let $\tau_{K}(\sigma) = [t_{o}| t_{u}] \sigma i \sigma, t_{u} \in pre(K)$ be the set of service time of $\sigma$ following the string $s$ under the restriction of $K$.

Definition 2: A timed language $K$ is said to be $G_{c}$-controllable [26] if the following conditions are satisfied.

- trace-control: $pre(K) \Sigma_{c} \cap L(G) \subseteq pre(K)$
- time-control: For any $s \epsilon pre(K)$ and $s \epsilon \Sigma_{u}$, $T_{s}(L(G)) \subseteq T_{K}(s)$ holds.

A supervisor $se$ is defined as an ordered pair $se = (f, I)$, where $f(L(G)) = \Gamma$ is a logical supervisor which is the set of enable logical events and $L(L(G)) \epsilon \Sigma \epsilon \Omega$ is a timed supervisor which is the enable time-interval of logical events such that $\Sigma_{c} \subseteq f(s)$ and $t \epsilon \Omega(s, \sigma)$ for any $s \epsilon TL(G)$ and $\sigma \epsilon \Sigma_{u}$, where $\Omega = ([R_{1}, R_{2}] R_{1} \epsilon R_{2}, R_{1} \epsilon R_{2}^{*})$ is the set of time intervals. The closed-loop system under $se$ is denoted by $sc / G_{c}$. Definition 3: The timed language $TL(sc / G_{c})$ generated by $sc / G_{c}$ is defined as follows:

- $\epsilon \epsilon TL(sc / G_{c})$, where $\epsilon$ is the empty timed string.
- $s(\delta, \sigma) \epsilon TL(sc / G_{c}) \Leftrightarrow s(\delta, \sigma) \epsilon TL(G)$.
- $sc \epsilon TL(sc / G_{c})$ for any $\sigma \epsilon TL(G)$ and $\sigma \epsilon \Sigma_{u}$.

Theorem 1: If $K$ is a closed timed language, there exists a supervisor $sc = (f, I)$ such that $TL(sc / G_{c}) = K$ if and only if $K$ is $G_{c}$-controllable [26].

Let $P_{i}$: $\Sigma_{t} \rightarrow \Sigma_{o}$ be the observed function, where $\Sigma_{t}$ is the timed events set and $\Sigma_{o}$ is the observable timed event set which consists of observable event and its service time. For the projection $P_{i}$, we suppose $P_{i}(\epsilon) = \epsilon$, $P_{i}(\epsilon, i) = \begin{cases} (\sigma, i) & \text{if } (\sigma, i) \epsilon \Sigma_{e}, \\ \epsilon & \text{otherwise} \end{cases}$ and $P_{i}(s(\sigma, i)) = P_{i}(s)P_{i}(\sigma, i)$. A partial supervisor is defined as an ordered pair $sc = (f, I)$, where $f(L(G)) = \Gamma$ and $I : P_{i}(L(G)) \epsilon \Sigma \epsilon \Omega$, such that $\Sigma_{c} \subseteq f(L(G))$ and $s \epsilon L(G)$ and $\sigma \epsilon \Sigma_{u}$.

Definition 4: For timed languages $K$, $K$ is said to be observable [26] if $s(\sigma, i) \epsilon pre(K)$, $s(\sigma, i) \epsilon TL(G)$ and $P_{i}(s) = P_{i}(s)$ imply $s(\sigma, i) \epsilon pre(K)$ for any $s, s \epsilon pre(K)$ and $(\sigma, i) \epsilon \Sigma_{c}$.

Theorem 2: For closed timed languages $K$, there exists a partial supervisor $sc = (f, I)$ such that $TL(sc / G_{c}) = K$ if and only if $K$ is $G_{c}$-controllable and observable [26] (controllable and observable).

Theorem 3: For closed timed languages $K$, there exists decentralized local supervisors $(sc_{i})_{i \epsilon b}$ such that $TL(sc_{i} / G_{i}) = K$ if and only if $K$ is $G_{c}$-controllable and observable where $sc_{i}$ is the global extension of $sc_{i}$.

Proof (Only if) It is assumed that there exists decentralized local supervisors $sc_{i}$ and $sc_{j}$ such that $TL(sc_{i} / G_{i}) = K$.


(trace-controllable) Since K is closed, we have \( \text{pre}(tr(K))=tr(K) \). For any \( s_1 \in tr(K) \) and \( \sigma \in \Sigma_u \) such that \( s_1 \in L(G) \), there exists \( s \in K \) and \( t \in T_{L(G)}(\sigma) \) such that \( s_1 \in tr(K) \) and \( t(\sigma,t) \in TL(G) \). It follows from \( TL\left( \left\{ \sigma \right\}_{t},G \right) = K \) that \( s \in TL\left( \left\{ \sigma \right\}_{t},G \right) \).

If \( (\sigma,t) \in \Sigma_{u1} \wedge \Sigma_{u2} \), we have \( f_1(P_0(s)) \wedge f_2(P_2(s)) \) and \( t \in I_1(P_1(s),\sigma) \wedge I_2(P_2(s),\sigma) \). So, \( (\sigma,t) \in \Sigma_{c1} \wedge \Sigma_{c2} \) holds, and then \( (\sigma,t) \in \Sigma_{c1} \wedge \Sigma_{c2} \) holds. If \( (\sigma,t) \in \Sigma_{u1} \wedge \Sigma_{u2} \), we have \( (\sigma,t) \in \Sigma_{c2} \). By the above proof and the definition of \( \Sigma_{u1} \wedge \Sigma_{u2} \), \( (\sigma,t) \in \Sigma_{u1} \wedge \Sigma_{u2} \), similarly, we have \( (\sigma,t) \in \Sigma_{c2} \). By the definition of \( TL\left( \left\{ \sigma \right\}_{t},G \right) \), we have \( s(\sigma,t) \in TL\left( \left\{ \sigma \right\}_{t},G \right) \).

(time-controllable) For any \( s \in K \) and \( \sigma \in \Sigma_u \), we need to show \( T_{L(G)}(\sigma) \subseteq T_K(\sigma) \).

If \( T_{L(G)}(\sigma) = \emptyset \), we have \( T_{L(G)}(\sigma) \subseteq T_K(\sigma) \).

If \( T_{L(G)}(\sigma) \neq \emptyset \), we have \( s(\sigma,t) \in TL(G) \) for any \( t \in T_{L(G)}(\sigma) \). From the above proof and \( \sigma \in \Sigma_u \), we have \( (\sigma,t) \in \Sigma_{c1} \). It follows from \( s \in K \) that \( s(\sigma,t) \in TL\left( \left\{ \sigma \right\}_{t},G \right) \) and \( K \). Therefore, \( T_{L(G)}(\sigma) \subseteq T_K(\sigma) \) holds.

(coobservable timed languages) To show timed language K is coobservable, we need to consider the following cases for any \( s_1, s_2, \in K \). Let \( K \in TL\left( \left\{ \sigma \right\}_{t},G \right) \).

Case 1 If there exists \( (\sigma,t) \in \Sigma_{c1} \wedge \Sigma_{c2} \) such that \( P_c(s_2) = P_c(t), s_2(\sigma,t) \in K \) and \( t(\sigma,t) \in TL(G) \), we need to show \( t(\sigma,t) \in TL(G) \). By the formula \( (\sigma,t) \in \Sigma_{c1} \wedge \Sigma_{c2} \), we have \( (\sigma,t) \in \Sigma_{c2} \). From \( P_c(s_2) = P_c(t) \), we have \( (\sigma,t) \in \Sigma_{c2} \). Since \( t(\sigma,t) \in TL(G) \), we have \( t \in TL\left( \left\{ \sigma \right\}_{t},G \right) \) and \( K \).

Case 2 If there exists \( (\sigma,t) \in \Sigma_{c1} \wedge \Sigma_{c2} \) such that \( P_c(s_2) = P_c(t), s_2(\sigma,t) \in K \) and \( t(\sigma,t) \in TL(G) \), we need to show \( t(\sigma,t) \in TL(G) \). By the formula \( (\sigma,t) \in \Sigma_{c1} \wedge \Sigma_{c2} \), we have \( t(\sigma,t) \in TL(G) \). Since \( t(\sigma,t) \in TL(G) \), we have \( t \in TL\left( \left\{ \sigma \right\}_{t},G \right) \) and \( K \).

Case 3 If there exists \( (\sigma,t) \in \Sigma_{c1} \wedge \Sigma_{c2} \) such that \( P_c(s_2) = P_c(t), P_c(s_2) = P_c(t), s_2(\sigma,t) \in K \) and \( t(\sigma,t) \in TL(G) \), we need to show \( t(\sigma,t) \in K \). By the formula \( (\sigma,t) \in \Sigma_{c1} \wedge \Sigma_{c2} \), we have \( t(\sigma,t) \in TL(G) \). Since \( t(\sigma,t) \in TL(G) \), we have \( t \in TL\left( \left\{ \sigma \right\}_{t},G \right) \) and \( K \).

So, K is coobservable from the definition of coobservability.

(If) Assuming that K is closed, G-controllable and coobservable timed language, we need to prove there exists decentralized supervisors \( \{\sigma(\sigma(t))\}_{t} \) such that \( TL\left( \left\{ \sigma \right\}_{t},G \right) = K \). For any \( s \in TL(G) \), we construct supervisors as follows.

\[
\begin{align*}
\tilde{f}(P_c(s)) = \Sigma_u \cup \{ \sigma \in \Sigma_u \mid \exists s' \in K, P_c(s) = P_c(s') \} \\
I_t(P_c(s),\sigma) = I_{t,s} \cup \{ t_{s} \mid \sigma \in \Sigma_u \cup \{ s' \mid s' \in K, P_c(s) = P_c(s') \} \\
K = K(\sigma / s')
\end{align*}

Using the length of \( tr(s) \), we can show \( TL\left( \left\{ \sigma \right\}_{t},G \right) = K \) by the methods of mathematics induction. It is obviously that \( e \in TL\left( \left\{ \sigma \right\}_{t},G \right) \) \( \wedge K \). Assuming that \( s \in TL\left( \left\{ \sigma \right\}_{t},G \right) \wedge K \), we need to prove \( s(\sigma,t) \in TL\left( \left\{ \sigma \right\}_{t},G \right) \) for any \( (\sigma,t) \in \Sigma_t \).

(We) prove \( TL\left( \left\{ \sigma \right\}_{t},G \right) \subseteq K \). Take \( s(\sigma,t) \in TL\left( \left\{ \sigma \right\}_{t},G \right) \). It is obvious that \( s(\sigma,t) \in TL(G) \) from \( TL\left( \left\{ \sigma \right\}_{t},G \right) \subseteq TL(G) \). So, \( tr(\sigma) \in L(G) \) and \( t \in T_{L(G)}(\sigma) \) hold. If \( (\sigma,t) \in \Sigma_u \),
we have $tr(s)\sigma e tr(K)$ and $t_1\epsilon T_k(\sigma s)$ from the trace-controllability and time-controllability of $K$. Therefore, $s(\sigma t_0)\epsilon K$ holds. If $\sigma e C_{i}$, we have $(\sigma t_0)\epsilon s_c(P_n(s))$ from $\sigma e TL\left(\frac{s_c}{s_c} \cup G_i\right)$ for any $i\epsilon I_n$. Since $\Sigma c\epsilon \Sigma c_1 \cup \Sigma c_2$, we consider the following three cases.

Case 1 If $(\sigma t_0)\epsilon \Sigma c_1 \cup \Sigma c_2$, there exists $s_1K$ such that $P_{t_1}(s_1)=P_{t_1}(s)$ and $i\epsilon T_k(\sigma s_1)$ by the formula $(\sigma t_0)\epsilon s_c_1(P_n(s_1))$. So, $s_1(\sigma t_0)\epsilon K$ holds. Since $s(\sigma t_0)\epsilon TL(G_i)$, we have $s(\sigma t_0)\epsilon K$ from the definition of coobservability.

Case 2 If $(\sigma t_0)\epsilon \Sigma c_2 \cup \Sigma c_1$, similarly, we have $s(\sigma t_0)\epsilon K$ from the proof of case 1.

Case 3 If $(\sigma t_0)\epsilon \Sigma c_1 \cup \Sigma c_2$, there exists $s_1P_{t_1}(s_1)=P_{t_1}(s)$ and $i\epsilon T_k(\sigma s_1)$ by the formula $(\sigma t_0)\epsilon s_c_1(P_n(s))$. Since $s(\sigma t_0)\epsilon K$ and $s(\sigma t_0)\epsilon K$ by the definition of coobservability.

So, $TL\left(\frac{s_c}{s_c} \cup G_i\right)\subseteq K$ holds.

(We prove $K \subseteq TL\left(\frac{s_c}{s_c} \cup G_i\right)$). Take $s(\sigma t_0)\epsilon K$. If $(\sigma t_0)\epsilon \Sigma a_1 \cup \Sigma a_2$, it is obvious that $\sigma e f_1(P_{t_1}(s))$ and $i\epsilon I_1(P_{t_1}(s)) \cap I_2(P_{t_2}(s))$, and then $(\sigma t_0)\epsilon s_c_1(P_n(s)) \cup s_c_2(P_n(s))$. If $(\sigma t_0)\epsilon \Sigma a_1 \cup \Sigma a_2$, we have $(\sigma t_0)\epsilon s_c_1(P_n(s)) \cup s_c_2(P_n(s))$ by the above proof and definition of $s_c_2$. If $(\sigma t_0)\epsilon \Sigma a_1 \cup \Sigma a_2$, we have $(\sigma t_0)\epsilon s_c_1(P_n(s)) \cup s_c_2(P_n(s))$. If $(\sigma t_0)\epsilon \Sigma a_1 \cup \Sigma a_2$, we have $(\sigma t_0)\epsilon s_c_1(P_n(s)) \cup s_c_2(P_n(s))$ by the definitions of $s_c_1$ and $s_c_2$. By the formulas we have $TL\left(\frac{s_c}{s_c} \cup G_i\right)$ and $s(\sigma t_0)\epsilon TL(G_i)$. We consider the following three cases.

Case 1 If $(\sigma t_0)\epsilon \Sigma a_1 \cup \Sigma a_2$, we have $(\sigma t_0)\epsilon \Sigma a_1 \cup (\Sigma a_2 \cup (\Sigma a \cap \Sigma a_2))$. It is obvious $(\sigma t_0)\epsilon s_c_1(P_n(s))$. Since $s(\sigma t_0)\epsilon K$, there exists $s_1K$ such that $s_1(\sigma t_0)\epsilon K$ and $P_{t_1}(s)\epsilon P_{t_1}(s)$ by the definition of coobservability. So, $t_1\epsilon T_k(\sigma s_1)$, $i\epsilon I_1(P_{t_1}(s))$, and $P_{t_1}(s)\epsilon P_{t_1}(s)$ hold. By the construction of $s_c_1$, we have $(\sigma t_0)\epsilon s_c_1(P_n(s))$ and $i\epsilon I_1(P_{t_1}(s))$. So, $(\sigma t_0)\epsilon s_c_1(P_n(s))$, and then $(\sigma t_0)\epsilon s_c_1(P_n(s))$. By the formulas $s(\sigma t_0)\epsilon TL\left(\frac{s_c}{s_c} \cup G_i\right)$ and $s(\sigma t_0)\epsilon K \subseteq TL(G_i)$, we have $s(\sigma t_0)\epsilon TL\left(\frac{s_c}{s_c} \cup G_i\right)$ from the proof of case 1.

Case 2 If $(\sigma t_0)\epsilon \Sigma a_1 \cup \Sigma a_2$, there exists $s_1, s_2K$ such that $P_{t_1}(s)\epsilon P_{t_1}(s)$, $P_{t_2}(s)\epsilon P_{t_2}(s)$, $s_1(\sigma t_0)\epsilon K$ and $s_2(\sigma t_0)\epsilon K$ by the assumption of $s(\sigma t_0)\epsilon K$. Since $(\sigma t_0)\epsilon K$, we have $(\sigma t_0)\epsilon \Sigma a_1 \cup \Sigma a_2$. By the construction of $s_c_1$ and $s_c_2$, we have $(\sigma t_0)\epsilon s_c_1(P_n(s))$, $(\sigma t_0)\epsilon s_c_2(P_n(s))$, $i\epsilon I_1(P_{t_1}(s))$, and $(\sigma t_0)\epsilon I_2(P_{t_2}(s))$. So, $(\sigma t_0)\epsilon s_c_1(P_n(s)) \cup s_c_2(P_n(s))$, and then $(\sigma t_0)\epsilon s_c_1(P_n(s)) \cup s_c_2(P_n(s))$. By the definition of $TL\left(\frac{s_c}{s_c} \cup G_i\right)$, we have $s(\sigma t_0)\epsilon TL\left(\frac{s_c}{s_c} \cup G_i\right)$.

Therefore, $K \subseteq TL\left(\frac{s_c}{s_c} \cup G_i\right)$ holds.

**Example 1:** A continuous Timed-DES $G_i$, considered is shown in Figure 1, where timed event set $\Sigma c_1 = \{a, [3, 8], \beta, [2, 7], \gamma, [3, 5], \mu, [0, 5]\}$ and controllable timed event set $\Sigma c_2 = \{a, \beta, \gamma\}$.

![Fig 1: Timed-DES $G_i$](image)

**Example 2:** A continuous Timed Language $K$

For timed system $G_i$, we have $TL(G_i) = pre([([a, 5, 8]) + ([\gamma, [0, 7]]) \cup (a, [3, 5]) + ([\beta, [2, 7]) + ([\mu, [0, 5]]) + (\beta, [3, 4]))]$ by Figure 1. Let $K = pre([([\gamma, [2, 5]) \cup (a, [3, 5]) + (\mu, [0, 5]) + (\beta, [3, 4]])]$ be the timed specification shown in Figure 2. By the definition of closeness, $G_i$-controllability and coobservability, we can show $K$ is closed, GT-controllable and co-observable. So, there exists decentralized supervisors $\{sc_1, sc_2\}$ such that

![Fig 2: Timed Language $K$](image)
constructed followed the above proof.

By the observed functions, we have the local systems $G_{1}$ and $G_{2}$ shown in Figure 3 and Figure 4. For all $s \in TL(G_{i})$, we can construct decentralized local supervisors $\{s_{c_{1}},s_{c_{2}}\}$ as follows.

\[
\text{sc}_{1};
\]

If $P_{t}(s) = [(\gamma, [3, 5])(\alpha, [3, 5])) + ([\beta, [3, 4]][\beta, [3, 4]])], f_{1}(P_{t}(s)) = \{ \alpha, \beta, \gamma \}, I_{1}(P_{t}(s), \alpha) = [3, 5, 3.5), I_{1}(P_{t}(s), \beta) = [3, 4) and I_{1}(P_{t}(s), \gamma) = [2, 5).

\[
\text{Fig 3: Observed Timed-DES } G_{1}
\]

If $P_{t}(s) = [(\gamma, [3, 5])(\alpha, [3, 5])) + ([\beta, [3, 4]][\beta, [3, 4]])], f_{1}(P_{t}(s)) = \{ \alpha, \beta, \gamma \}, I_{1}(P_{t}(s), \alpha) = [3, 3.5), I_{1}(P_{t}(s), \beta) = [3, 4) and I_{1}(P_{t}(s), \gamma) = [2, 5).

\[
\text{Fig 4: Observed Timed-DES } G_{2}
\]

If $P_{t}(s) = [(\gamma, [3, 5])(\alpha, [3, 5])) + ([\beta, [3, 4]][\beta, [3, 4]])], f_{1}(P_{t}(s)) = \{ \alpha \} and I_{1}(P_{t}(s), \alpha) = [3, 3.5).

If $P_{t}(s) = [(\gamma, [3, 5])(\alpha, [3, 5])) + ([\beta, [3, 4]][\beta, [3, 4]])], f_{1}(P_{t}(s)) = \{ \alpha, \beta, \gamma \}, I_{1}(P_{t}(s), \alpha) = [3, 3.5), I_{1}(P_{t}(s), \beta) = [3, 4) and I_{1}(P_{t}(s), \gamma) = [2, 5).

If $P_{t}(s) = TL(G_{1}) + \text{pre}(([\beta, [3, 4]](\alpha, [3, 3.5]) + ([\beta, [3, 4]](\beta, [3, 4]))], s_{c_{1}}(P_{t}(s)) = \emptyset.

\[
\text{sc}_{2};
\]

If $P_{t}(s) = [(\beta, [3, 4])(\mu, [0, 5])(\beta, [3, 4])]$, $f_{2}(P_{t}(s)) = \{ \alpha, \mu \}, I_{2}(P_{t}(s), \alpha) = [3, 3.5) and I_{2}(P_{t}(s), \mu) = [0, 5).

If $P_{t}(s) = [(\beta, [3, 4])(\mu, [0, 5])(\beta, [3, 4])]$, $f_{2}(P_{t}(s)) = \{ \beta \} and I_{2}(P_{t}(s), \beta) = [3, 4).

If $P_{t}(s) = TL(G_{2}) + \text{pre}(([\beta, [3, 4]])(\mu, [0, 5])(\beta, [3, 4]))], s_{c_{2}}(P_{t}(s)) = \emptyset.

Remark 1: If the service time of any event is discrete, definition 5 and theorem 3 hold.

Remark 2: If the service time of any event is 0, definition 5 is the coobservable language of [7] and theorem 3 can be got from [7].

5 Conclusions

In this paper, decentralized supervisory control of discrete event system with continuous service time is considered. With the continuous time feature incorporated into DES, timed-DES with continuous-time variable is suitable to model. To solve synthesis problem of distributed systems, decentralized supervisory control for timed-DES is considered.

(Continued on P.81)
The main work of realizing configuration is to establish level control objects and make animating display scenes. Controlled objects include inleting water flow, exporting water flow and the numerical object of the boiler level. When animation connection is established, the basic graphic elements and animation component library are called in the user window to construct configuration diagram. Graphic objects and data objects defined by the state are set in the state of the corresponding attribute and animation connection is defined. Having finished the design of the developing system, you can switch to run mode to carry on the real-time monitoring to the control system and test configuration.

6 Conclusions

This paper has introduced the composition and running of EFPT process control system based on ControlLogix5550 PLC control, the mathematical model establishing of controlled object and the parameter tuning of PID. The use of configuration software extends the communication function. Through experimental testing, the control curve’s overshoot is small and the transition time is short, so the control effect is quite ideal. This device being reliable and intuitive is suitable for scientific research and teaching, and has important application value in the actual industrial production.

References


(From P.61)