Fault Tolerance in Asynchronously Operated Machines Allowing Temporary Violation of Normal Specification

Jung–Min Yang and Seong Woo Kwak

***Abstract*—A novel scheme of fault tolerance for input/state asynchronous sequential machines is presented in this paper. The machine may undergo unauthorized state transitions caused by adversarial inputs. The considered faults have the feature of intermittency in their influences so that the adverse effect persists for some finite time after initial occurrence. Since strong fault tolerance is impossible, we define the notion of weak fault tolerance and propose a fault tolerant control law for which the closed-loop system recovers the nominal input/state behavior with a bounded delay. The existence condition and design procedure for a controller are addressed based on corrective control theory for asynchronous sequential machines. An illustrative example is provided for demonstrating the proposed fault tolerant control scheme.**

***Index Terms*—asynchronous sequential machines, corrective control, state feedback, intermittent faults, fault tolerance.**

I. Introduction

In this paper, we study the problem of fault tolerance for asynchronous sequential machines. Governed by no global synchronizing clock, asynchronous machines offer several advantages over synchronous machines such as fast response, low power consumption, less emission of electro-magnetic noise, etc [1]. On the other hand, asynchronous machines are more difficult to design than synchronous machines since we need to realize handshaking between the components of asynchronous machines in order to perform the necessary synchronization, communication, and sequencing of operations. Thus, a majority of the previous studies on asynchronous machines were mainly about their design principles and related problems [1]–[3].

Our study takes an alternative view on asynchronous machines: a control theoretic technique to compensate the behavior of an existent asynchronous machine without resort to re-design of its inner logic. This scheme, *corrective control theory* as it is often called, can be said to lay a novel foundation on asynchronous machines, because it proves, both theoretically and experimentally, that the asynchronous machine is also controllable in a similar way to continuous-time systems in the framework of feedback control. Notice that the controlled behavior of an asynchronous machine must be interpreted in the aspect of stable-state operations. In corrective control, the original definition of state transitions of a machine is intact, as the inner logic of the machine remains the same throughout the control algorithm. Instead, the controller accelerates the feedback trajectory to the limit—ideally, in zero time—in asynchronous mechanism so that any control procedure is completed instantaneously. Thus, the closed-loop system can show the desired behavior while the interaction between the system and the controller is unnoticeable.

For the past decade, corrective control has been successfully applied to the problem of eliminating various deficiencies in the operation of asynchronous sequential machines. [4]–[7] address the model matching problem for asynchronous machines with critical races. [8] presents dynamic feedback controllers for input/output asynchronous machines in which the information on the output values of the machine is unavailable to the controller. In [9] and [10], state feedback controllers are used to eliminate the effects of infinite cycles on asynchronous sequential machines. [11]–[14] develop corrective controllers for diagnosing and tolerating transient faults that cause a violation of state transition characteristics of asynchronous machines. [15] and [16] apply the foregoing theoretic results on asynchronous digital systems implemented in FPGA. In [17], a corrective controller is presented to realize model matching with the constraint that some external input characters are uncontrollable. A similar study with the application to error counters can be found in [18]. [19] and [20] present fault tolerant corrective control schemes for tolerating permanent faults occurring to input/output machines. Finally, [21] addresses identification and corrective control of asynchronous machines with unspecified transition parts based on an adaptive control law.

The objective of this paper is to propose a fault tolerant corrective controller for input/state asynchronous sequential machines subject to intermittent faults. Intermittent faults are defined as violation of the system's normal behavior occurring repeatedly or at intervals [22]. In this study, we stipulate that the occurrence of an intermittent fault makes the machine go through an unauthorized state transition. Thus, if not recovered immediately, further change of the external input would cause the mismatch of input/state behaviors. In this respect, the outcome of the intermittent faults is the same as that of the transient faults studied in [11]–[14]. But, unlike the case of transient faults, instantaneous counteracting to the original state of the machine is impossible due to the feature of the intermittent faults. If the intermittent fault does not vanish, the machine would be expelled from the original state again even if it could reach the state by a fault tolerant

This work was supported by the National Research Foundation of Korea grant funded by the Korea government (MSIP) (No. NRF–2011–0027705), and in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. NRF–2010–0007271).

J.–M. Yang is with the School of Electronics Engineering, Kyungpook National University, Daegu, 702-701, Republic of Korea. E-mail: jmyang@ee.knu.ac.kr.

S. W. Kwak is with the Department of Electronic Engineering, Keimyung University, Daegu, 704-701, Republic of Korea. E-mail: ksw@kmu.ac.kr (corresponding author).

ISBN: 978-988-19252-5-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

control scheme.

Since strong fault tolerance is infeasible in our setting, we propose the notion of weak fault tolerance, namely the closed-loop system recovers the normal input/state behavior within a bounded delay after fault occurrence. Note that [23], [24] present a similar subject, namely the system recovers from any fault within a bounded delay. But, as their studies are based on supervisory control of discrete-event systems [25], they cannot be applied to asynchronous machines. We propose a novel structure of corrective controllers that realize bounded-delay fault tolerance. To this end, analyses on stable reachability of the controlled machine are conducted to characterize the fault tolerance capability of the machine. Until accomplishing perfect input/state matching, the closed-loop system must endure some interval in which the normal specification of the machine's behavior is violated. An illustrative example is provided for demonstrating the proposed fault tolerant control scheme.

II. Notation and Basics

We represent an input/state asynchronous machine Σ as the following finite state machine:

$$\Sigma = (A, X, x_0, f),$$

where A is the input set, X is the state set, $x_0 \in X$ is the initial state, and $f : X \times A \to X$ is the state transition function defined as a partial function onto $X \times A$. Σ operates according to the recursion

$$x_{k+1} = f(x_k, u_k),\ k = 0, 1, 2, \ldots,$$

where $u_0 u_1 u_2 \ldots$ is the input sequence and $x_0 x_1 x_2 \ldots$ is the state sequence generated by Σ starting from the initial state x_0. The step counter k advances by one upon a change of input or state.

A sate–input pair $(x, u) \in X \times A$ is valid if f is defined at (x, u). (x, u) is a stable combination if it is a fixed point of f, i.e., if $f(x, u) = x$. Σ stays at (x, u) indefinitely as long as the external input remains unchanged. (x, u) is a transient combination if $f(x, u) \neq x$. Consider a case where Σ is at a stable combination (x, u') when the external input switches to u for which (x, u) is a transient combination. This change gives rise to a chain of transient transitions

$$\begin{aligned} x_1 &= f(x, u), \\ x_2 &= f(x_1, u), \\ &\vdots \end{aligned}$$

In asynchronous mechanism, Σ passes through the transient states $x_1, x_2, \ldots$ very quickly (in zero time, ideally). Assuming that no infinite cycles exist in the behavior of Σ, Σ reaches a stable state $x_k = f(x_k, u)$. x_k is termed the next stable state of (x, u) [5]. Since the transient states underlying in the chain of transitions are unnoticeable, we properly exclude all transient states and express only the *stable transition* from a stable state to its next stable state. The stable recursion function

$$s : X \times A \to X$$

embodies all the stable transitions of Σ. For a valid pair (x, u), $s(x, u)$ is defined as the next stable state of (x, u). s is often extended from input characters to sequences recursively: for $x \in X$ and $u_1 u_2 \cdots u_k \in A^+$, where A^+ is the set of all non-empty strings of characters of A,

$$s(x, u_1 u_2 \cdots u_k) := s(s(x, u_1), u_2 \cdots u_k).$$

To include the set of intermittent faults, we partition the input set A into two mutually exclusive subsets

$$A := A_N \dot{\cup} A_T,$$

where A_N is the normal input set and A_T is the set of intermittent fault inputs. Suppose that Σ has been staying at a stable combination $(x, u) \in X \times A_N$, when an intermittent fault $w \in A_T$ defined at the state x infiltrates into Σ. Σ then undergoes an unauthorized state transition, namely its state is switched without any change of the external input. As the result of the fault occurrence, Σ will move to the deviated state $s(x, w)$. For immediate fault tolerance, Σ must be driven to return to the original state x instantaneously after the fault occurrence, i.e., before further change of the external input. However, such an objective is impossible to achieve in this case, because the intermittent fault imposes a restraint that its adverse effect lasts for some finite time after initial occurrence. Thus even if Σ could return to the original state x right after the fault occurrence, it would experience another unauthorized transition to $s(x, w)$.

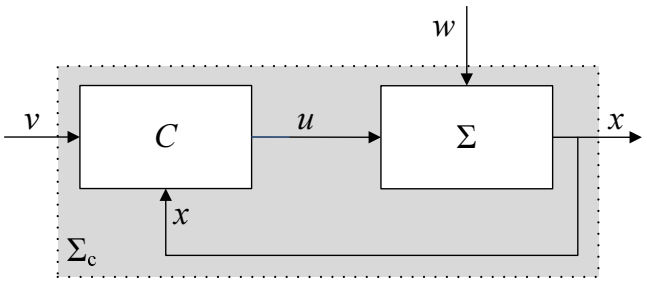

Fig. 1. Corrective control system.

Fig. 1 is the structure of the corrective control system for an input/state asynchronous machine Σ, adapted from the former researches [5], [11]. C is the corrective controller that has the form of an input/output asynchronous sequential machine. $v \in A_N$ is the external input, $u \in A_N$ is the control input generated by C, and $x \in X$ is the state (or output) of Σ delivered to C as the state feedback. We denote by Σ_c the closed-loop system represented by the diagram. $w \in A_T$ is the intermittent fault occurring to Σ. Complying with the feature of adversarial entities, we suppose that w is neither observable nor disable by the controller C. Upon occurrence, w overrides the present value of u. Hence, the input to Σ is determined as one of u and w whose value changes at the last.

When dealing with asynchronous sequential machines, we have to be careful to avoid any operation that may result in an unpredictable response of the machine. In particular, we must avoid changing the input character while a machine is in transition transitions. If, on the contrary, the input character of an asynchronous machine is changed while the machine undergoes a chain of transitions, asynchrony and the rapid speed at which transient transitions occur prevent an exact characterization of the machine's state at the instant at which the input character changes. Since we do not know

ISBN: 978-988-19252-5-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

the exact state at which the input change occurs, switching the input while a machine is in transitions may result in an unpredictable outcome.

To avoid such potential uncertainty, a common operating policy for asynchronous machines is to preserve the principle of fundamental mode operation [26], namely to prohibit changes to the input while a machine is in transition. Under this operating policy, input changes are allowed only when the machine is in a stable state. Then, the state of the machine at which the input change occurs is well defined, and, as a result, so is the effect of the input change on the response of the machine. Adopting the former studies on the system configuration [12], [21], we write the following condition for the closed-loop system Σ_c of Fig. 1 to guarantee fundamental mode operation.

Condition 1. *The closed-loop system Σ_c of Fig. 1 operates in fundamental mode when all the following conditions are valid:*

i) *Σ stays at a stable state while C undergoes transient transitions.*
ii) *C stays at a stable state while Σ undergoes transient transitions.*
iii) *The external input v and the intermittent fault w change only when Σ and C are both in stable states, and only one at a time.*

Parts i) and ii) of Condition 1 must be implemented during the design of the controller C; part iii) on the other hand, is a restriction on the operation of the closed-loop system Σ_c. We should expect that w, an independent adversarial entity, enters into Σ only when both Σ and C are at stable states. Nonetheless, as transitions of asynchronous machines occur very quickly, this assumption does not impose a burdensome requirement on the system.

III. Stable Reachability

The existence condition for a corrective controller depends on the stable reachability of the machine Σ. Given $\Sigma = (A, X, x_0, f)$, let $X := \{x_1, \ldots, x_n\}$.

Definition 1. *Given $\Sigma = (A, X, x_0, f)$ with $A = A_N \dot\cup A_T$, a state $x_j \in X$ is said to be stably reachable from $x_i \in X$ if there exists a string of nominal input characters $t = v_1 v_2 \cdots v_k \in A_N^+$ such that $s(x_i, t) = x_j$ [5].*

To quantify the stable reachability, we introduce a numerical matrix, called the skeleton matrix [5], as follows.

Definition 2. *Given $\Sigma = (A, X, x_0, f)$ with $A = A_N \dot\cup A_T$, the one-step skeleton matrix $K^1(\Sigma)$ is an $n \times n$ matrix of zeros and ones with the entries*

$$K^1_{i,j}(\Sigma) = \begin{cases} 1 & \exists v \in A_N \text{ s.t. } x_j = s(x_i, v) \\ 0 & \text{else} \end{cases}$$
$$\forall i, j \in \{1, \ldots, n\}.$$

The skeleton matrix $K(\Sigma)$ is an $n \times n$ matrix of zeros and ones with the entries

$$K_{i,j}(\Sigma) = \begin{cases} 1 & \exists t \in A_N^+ \text{ s.t. } x_j = s(x_i, t) \text{ and } 1 \leqslant |t| \leqslant n-1 \\ 0 & \text{else} \end{cases}$$
$$\forall i, j \in \{1, \ldots, n\},$$

where A_N^+ is the set that includes all nonempty strings of characters of A_N and $|t|$ denotes the length of the string t.

$K^1_{i,j}(\Sigma) = 1$ implies that x_j can be reachable from x_i by a unit input character in A_N, whereas $K_{i,j}(\Sigma) = 1$ represents that x_j can be reachable from x_i by an input sequence, i.e., by multiple steps of stable transitions. Stable reachability is a crucial requirement for guaranteeing a corrective controller for asynchronous machines [5]. In elucidating stable reachability, it is sufficient to consider input strings whose length is less than or equal to $n-1$, where n is the cardinality of the state set X [5].

For a state $x_i \in X$, define $R^1(x_i)$ and $R(x_i)$ as the sets of states that are one-step stably reachable and stably reachable from x_i, respectively. We can express $R^1(x_i)$ and $R(x_i)$ using $K^1(\Sigma)$ and $K(\Sigma)$ as follows.

$$R^1(x_i) = \{x_j \in X | K^1_{i,j}(\Sigma) = 1\}$$
$$R(x_i) = \{x_j \in X | K_{i,j}(\Sigma) = 1\}.$$

Clearly, $R^1(x_i) \subseteq R(x_i)$ for any state x_i.

Lemma 1. *For a state $x_i \in X$ of $\Sigma = (A, X, x_0, f)$ with the state set $X = \{x_1, \ldots, x_n\}$,*

$$R(x_j) \subseteq R(x_i), \ \forall x_j \in R(x_i).$$

Proof: If $x_k \in R(x_j)$, there exists an input string $t_1 \in A_N^+$ such that $s(x_j, t_1) = x_k$. Since $x_j \in R(x_i)$, there exists an input string $t_2 \in A_N^+$ such that $s(x_i, t_2) = x_j$. Then, $s(x_i, t_2 t_1) = x_k$ where $t_2 t_1$ is the concatenation of t_2 an t_1. By definition, x_k is stably reachable from x_i and we have $x_k \in R(x_i)$. □

Lemma 1 implies that the set of stably reachable states from the present state decreases as Σ proceeds normal state transitions. In other words, the ranges of states that can be reached by Σ cannot be enlarged by state transitions. This property will be used in designing fault tolerant controllers later in this paper.

In a similar manner to the one-step skeleton matrix $K^1(\Sigma)$ with respect to the normal input set A_N, we define the skeleton matrix with respect to the adversarial input set A_T as follows.

Definition 3. *Given $\Sigma = (A, X, x_0, f)$ with $A = A_N \dot\cup A_T$, the adversarial skeleton matrix $K^d(\Sigma)$ is an $n \times n$ matrix of zeros and ones with the entries*

$$K^d_{i,j}(\Sigma) = \begin{cases} 1 & \exists w \in A_T \text{ s.t. } x_j = s(x_i, w) \text{ and } i \neq j \\ 0 & \text{else} \end{cases}$$
$$\forall i, j \in \{1, \ldots, n\}.$$

$K^d_{i,j}(\Sigma) = 1$ means that Σ may undergo an unauthorized state transition from x_i to x_j by an (unidentified) intermittent fault. $i \neq j$ in the above definition implies that we exclude latent adversarial transitions, namely the case where Σ maintains the present state despite the occurrence of an intermittent fault. Since no violation of normal behaviors is manifested in this case, fault tolerant control is not needed. Also, note that we consider only the unit adversarial input characters in Definition 3. Since the scheme of fault tolerance will be initiated upon detecting the fault occurrence, we do not have to consider further instances of intermittent faults.

ISBN: 978-988-19252-5-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

IV. Controller Design

Referring to Fig. 1, C has the following form of an input/output asynchronous machine:

$$C := (X \times A_N, A_N, \Xi, \xi_0, \phi, \eta),$$

where $X \times A_N$ is the input set, A_N is the output set, Ξ is the state set, $\xi_0 \in \Xi$ is the initial state,

$$\phi : \Xi \times X \times A_N \to \Xi$$

is the state transition function, and

$$\eta : \Xi \to A_N$$

is the output function (assuming C is a Moore machine [26]). Since the information on the intermittent fault w is unavailable, C must conduct fault diagnosis and tolerance by observing the change of the state feedback x.

C begins with the initial state ξ_0 in the first. Suppose that Σ reaches a state x_i for which there exists at least a state x_j with $K^d_{i,j}(\Sigma) = 1$, i.e., an intermittent fault can occur to Σ at x_i. C then transfers to the *transition state* ξ_t. Since the intermittent fault may happen only when Σ is at a stable state, C prepares possible occurrences of faults at the state ξ_t. To this end, set the recursion function ϕ as

$$\begin{aligned}
\phi(\xi_0, x, v) &= \xi_t \ \forall (x, v) \in \{x_i\} \times U(x_i) \\
\phi(\xi_t, x, v) &= \xi_t \ \forall (x, v) \in \{x_i\} \times U(x_i) \\
\phi(\xi_0, x, v) &= \xi_0 \ \forall (x, v) \in X \times A_N \backslash \{x_i\} \times U(x_i)
\end{aligned}$$

where $U(x_i) \subset A_N$ denotes the set of nominal input characters that make a stable combination with x_i. Since no actual control is executed at either ξ_0 or ξ_t, C relays the present external input v to the control input channel u:

$$\begin{aligned}
\eta(\xi_0) &= v, \\
\eta(\xi_t) &= v.
\end{aligned}$$

Suppose further that while the external input remains unchanged, the state feedback is observed to switch from x_i to x_j. We deduce that an intermittent fault w has happened to Σ, causing the unauthorized transition to $x_j = s(x_i, w)$. Since immediate return to the original state x_i is infeasible, Σ must endure temporary violation of the normal input/state specification by staying at the state x_j. C waits for the subsequent change of the external input and determines the next control input according to the incoming external input and stable reachability of the states. The following definition of ϕ makes Σ remain at the deviated state x_j after the fault occurrence.

$$\phi(\xi_t, x_j, v) = \xi_t \ \forall x_j \text{ s.t. } K^d_{i,j}(\Sigma) = 1,$$

where v means that the present external input remains fixed.

Recall that $R^1(x)$ and $R(x)$ are the sets of the states that are stably reachable from a state x in one or more steps. The control behavior after the fault occurrence depends on the relation between $R^1(x_i)$ and $R(x_j)$. First, consider the case of

$$R^1(x_i) \subseteq R(x_j). \tag{1}$$

This means that every state that is stably reachable from x_i in one-step is also stably reachable from x_j. Hence, we can find a feedback trajectory from the deviated state x_j to any state that would be reached in response to the new external input. Suppose that the external input changes to v' after Σ reaches x_j by fault. Were it not for the occurrence of the intermittent fault, Σ would stay at the stable state x_i and in response to the new external input v', would undergo the normal stable transition to $s(x_i, v') := x_k$. Since $x_k \in R^1(x_i)$, $x_k \in R(x_j)$ and $K_{j,k}(\Sigma) = 1$ by (1), and there exists a normal input string $t_{j,k} = v_1 \cdots v_m \in A_N^+$ such that $s(x_j, t_{j,k}) = x_k$. Using $t_{j,k}$, we can design a corrective controller module that takes Σ from x_j to x_k upon receiving the external input v' [5], [8].

Since $|t_{j,k}| = m$, the controller C needs m auxiliary states, termed $\xi_1, \ldots, \xi_m \in \Xi$. The correction procedure from x_j to x_k is realized by the following recursive assignment of ϕ and η.

$$\begin{aligned}
\phi(\xi_t, x_j, v') &= \xi_1 \ \forall v' \in A \text{ s.t. } s(x_i, v') = x_k \\
\eta(\xi_h) &= v_h \\
\phi(\xi_h, x^{h-1}, v') &= \xi_h \\
\phi(\xi_h, x^h, v') &= \xi_{h+1} \\
h &= 1, \ldots, m-1,
\end{aligned} \tag{2}$$

where

$$\begin{aligned}
x^h &= s(x^{h-1}, v_h), \\
x^0 &:= x_j, \ h = 1, \ldots, m-1,
\end{aligned}$$

denote the intermediate stable states Σ passes through the feedback trajectory characterized by $t_{j,k}$.

At ξ_m, finally, Σ reaches the state x_k, achieving the normal input/state specification with one step delay, i.e., after the external input changes once. C then returns to the initial state ξ_0. To this end, we set

$$\begin{aligned}
\eta(\xi_m) &= v_m \\
\phi(\xi_m, x, v') &= \xi_m \ \forall x \neq x_k \\
\phi(\xi_m, x_k, v') &= \xi_0.
\end{aligned} \tag{3}$$

Provided that condition (1) is held true, we can design a feedback trajectory for any other external input that will enter into the system right after Σ reaches the deviated state x_j.

Now suppose that condition (1) is not valid, i.e.,

$$R^1(x_i) \not\subseteq R(x_j). \tag{4}$$

Then, there exists a nonempty state set

$$D(x_i, x_j) := R^1(x_i) \backslash R(x_j)$$

whose members are stably reachable from x_i but not from x_j. Continuing to use the foregoing notation, let v' be the changed external input and let $x_k = s(x_i, v')$ be the state that Σ is supposed to move to in the normal behavior. If $x_k \in R(x_j)$, C can steer Σ toward x_k using the correction procedure that is designed above. On the other hand, if $x_k \notin R(x_j)$, that is, if $x_k \in D(x_i, x_j)$, no corrective control mechanism exists for driving Σ from x_j to x_k. In this case, Σ must endure violation of the normal specification one step more by staying at the present state x_j. Note from Lemma 1 that transferring to another state does not enhance the stable stability of Σ.

ISBN: 978-988-19252-5-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

To signify that the external input changes, C moves to the second transition state ξ'_t in response to v', as follows.

$$\begin{aligned}\phi(\xi_t, x_j, v') &= \xi'_t \ \forall v' \in D(x_i, x_j)\\ \eta(\xi'_t) &= v_s\end{aligned}$$

where $v_s \in U(x_j)$ is an input character that makes a stable combination with x_j. Receiving v_s, Σ maintains the present state x_j.

The behavior of C at ξ'_t is similar to the case of the transition state ξ_t. Since Σ is supposed to stay at x_k, the next input character will make a valid pair with x_k and Σ would move to a state of $R^1(x_k)$ if it had the normal behavior. If $R^1(x_k) \subseteq R(x_j)$, for every incoming input character, there exists a corrective controller module that takes Σ to the desired state. In particular, let $v'' \in A_N$ be the changed input character, and let $x_l := s(x_k, v'')$ be the state that Σ is supposed to reach in the normal behavior. Since $x_l \in R^1(x_k)$ and $R^1(x_k) \subseteq R(x_j)$, x_l is stably reachable from x_j ($K_{j,l}(\Sigma) = 1$) and thus there exists an input string

$$t_{j,l} := r_1 r_2 \cdots r_p \in A_N^+.$$

As $|t_{j,l}| = p$, C needs p more auxiliary states, termed $\xi'_1, \ldots, \xi'_p \in \Xi$. In a similar fashion to (2), we design the control behavior initiating from ξ'_t as follows.

$$\begin{aligned}\phi(\xi'_t, x_j, v'') &= \xi'_1 \quad \forall v'' \in A \text{ s.t. } s(x_k, v'') = x_l\\ \eta(\xi'_h) &= r_h\\ \phi(\xi'_h, z^{h-1}, v'') &= \xi'_h\\ \phi(\xi'_h, z^h, v'') &= \xi'_{h+1}\\ h &= 1, \ldots, p-1,\end{aligned} \tag{5}$$

where

$$\begin{aligned}z^h &= s(z^{h-1}, r_h),\\ z^0 &:= x_j, \ h = 1, \ldots, p-1,\end{aligned}$$

denote the intermediate stable states Σ passes through the feedback trajectory characterized by $t_{j,l}$. Also adapting the foregoing assignment (2), we complete the controller design at the final state ξ'_p:

$$\begin{aligned}\eta(\xi'_p) &= r_p\\ \phi(\xi'_p, x, v'') &= \xi'_p \ \forall x \neq x_l\\ \phi(\xi'_p, x_l, v'') &= \xi_0.\end{aligned} \tag{6}$$

On the other hand, if $R^1(x_k) \not\subseteq R(x_j)$, it is possible that recovery to the normal input/state specification is not achievable if the changed input character v'' invokes the stable transition to a state that is stably reachable from x_k but not from x_j. Then, like the former case, C must advance forward to the third transition state. Σ must again endure temporary violation of the normal behavior and will restart the correction procedure according to the next input character. In this way, the corrective controller C realizes fault tolerance against the intermittent fault with a bounded delay. As the steps of delay increase, so does the number of transition states of C.

V. Example

As an example instance, consider an input/state asynchronous machine $\Sigma = (A, X, x_0, f)$ whose state flow diagram is shown in Fig. 2. For the simplicity's sake, we set $f(x,u) = s(x,u)$ for all valid pair $(x,u) \in X \times A$. Σ has the following components:

$$\begin{aligned}A_N &= \{a, b, c, d, e\}\\ A_T &= \{w\}\\ X &= \{x_1, x_2, x_3, x_4, x_5\}\\ x_0 &= x_1.\end{aligned}$$

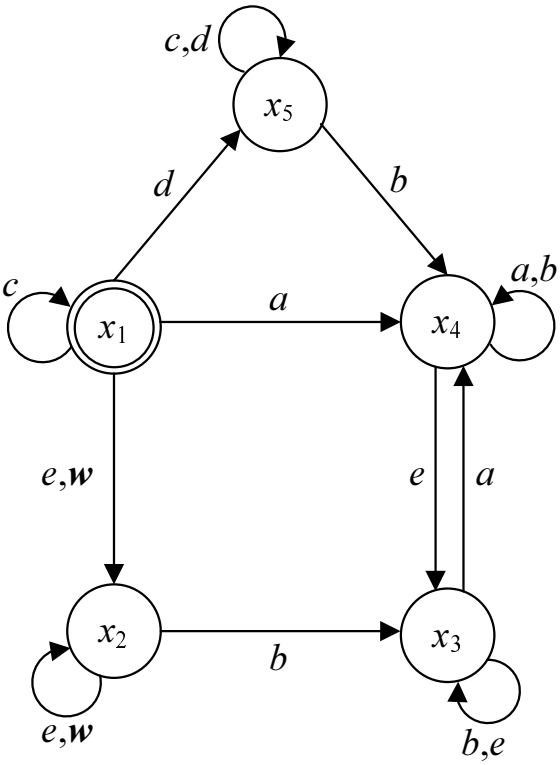

Fig. 2. State flow diagram of Σ.

Referring to Fig. 2, we see that the intermittent fault w may occur to Σ when the machine stays at the stable combination (x_1, c). Whenever w infiltrates into Σ, Σ undergoes the unauthorized state transition to $s(x_1, w) = x_2$. In order to investigate whether there exists a corrective controller that realizes fault tolerance within a bounded delay, we first derive $R^1(x_1)$ and $R(x_2)$ as

$$\begin{aligned}R^1(x_1) &= \{x_1, x_2, x_4, x_5\}\\ R(x_2) &= \{x_2, x_3, x_4\}.\end{aligned}$$

Since $R^1(x_1) \not\subseteq R(x_2)$, one-step recovery to the normal input/state behavior may not be successful depending on the next external input character. From $R^1(x_1)$ and $R(x_2)$, we induce the difference set

$$\begin{aligned}D(x_1, x_2) &= R^1(x_1) \backslash R(x_2)\\ &= \{x_1, x_5\}.\end{aligned}$$

A slight examination of Fig. 2 shows that among the input characters that make a valid pair with x_1, $\{a, e\}$ take Σ to the states that belong to $R(x_2)$ and $\{c, d\}$ to the states that belong to $D(x_1, x_2)$.

ISBN: 978-988-19252-5-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

We first select the control input sequences for the input characters $\{a, e\}$. Referring to Fig. 2, $s(x_1, a) = x_4$ and $s(x_1, e) = x_2$. For each state pair, we select the control input sequences as

$$x_2 \to x_2 \ :: t_{2,2} = \varnothing$$
$$x_2 \to x_4 \ :: t_{2,4} = ba.$$

In the above strings, $t_{2,2} = \varnothing$ implies that the corrective controller executes no control action when it receives e. Note that the present state of the machine Σ will be x_2 after undergoing the unauthorized transition. As the present state x_2 complies with the normal input/state specification with respect to e, no additional correction procedure is required for e. The controller construction using the sequence $t_{2,4}$ can be conducted based on the assignments (2) and (3). The discussion in the previous section leads us to that the controller $C = (X \times A_N, A_N, \Xi, \xi_0, \phi, \eta)$ will involve the states $\xi_0, \xi_t, \xi_1, \xi_2$ for materializing the fault tolerant control procedure up to this phase.

Secondly, consider the situation that after the occurrence of w, the external input changes to one of $\{c, d\}$. Specifically, assume that the external input changes to d. Then, in view of Fig. 2, Σ should go to $s(x_1, d) = x_5$ if it were in the normal status. As analyzed before, x_5 is not stably reachable from the present state x_2. Hence, Σ must endure the temporary violation of the normal specification, while C transfers from the transition state ξ_t to the second transition state ξ_t'. In Fig. 2, we know that

$$R(x_5) = \{x_3, x_4, x_5\}$$

in which Σ will move to x_4 in response to the input character b, and will stay at x_5 if the input character switches to c ('switch to d' is meaningless in the operation of Σ because the previous character is identical to d). If the next input character becomes c, Σ must still endure violation of the normal specification by maintaining the present state. If, on the other hand, it changes to b, Σ is supposed to move to x_4. As $x_4 \in R(x_2)$, we can design a feedback path that drives Σ from x_2 to x_4. In fact, we already elucidate that $t_{2,4} = ba$ can serve as the control input sequence that can realize the feedback trajectory. In a similar fashion to the foregoing design, C needs two more auxiliary states ξ_1' and ξ_2' for this correction procedure. The detailed design algorithm can be implemented using (5) and (6). Hence the proposed corrective controller achieves recovery to the normal input/state specification within two steps of input changes.

VI. Conclusion

A fault tolerant control methodology for asynchronous sequential machines has been presented in this paper. Since the considered adversarial entity is the intermittent fault, immediate recovery to the normal behavior is impossible. Instead, we have proposed a corrective control scheme that makes the closed-loop system recover the nominal specification of input/state behavior within a bounded delay. After a fault occurrence, the controller records the reference state that the machine must reach. According to the incoming input character, the controller provides an appropriate control input sequence for which the machine can reach the desired state. Temporary violation of the normal specification must be endured for finite steps of input changes. The investigation on the controller existence has been demonstrated in a case study.

References

[1] J. Sparsø and S. Furber, *Principles of Asynchronous Circuit Design — A Systems Perspective*, Kluwer Academic Publishers, 2001.
[2] C. J. Myers, *Asynchronous Circuit Design*, New York: John Wiley & Sons, 2001.
[3] R. F. Tinder, *Asynchronous Sequential Machine Design and Analysis*, Morgan & Claypool Publishers, 2009.
[4] T. E. Murphy, X. Geng, and J. Hammer, "Controlling races in asynchronous sequential machines," in *Proc. 15th IFAC World Congress*, Barcelona, Jul. 2002.
[5] T. E. Murphy, X. Geng, and J. Hammer, "On the control of asynchronous machines with races," *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 1073–1081, 2003.
[6] J. Peng and J. Hammer, "Input/output control of asynchronous sequential machines with races," *Int. J. Control*, vol. 83, no. 1, pp. 125–144, 2010.
[7] J. Peng and J. Hammer, "Bursts and output feedback control of nondeterministic asynchronous sequential machines," *European J. Control*, vol. 18, no. 3, pp. 286–300, 2012.
[8] X. Geng and J. Hammer, "Input/output control of asynchronous sequential machines," *IEEE Trans. Autom. Control*, vol. 50, no. 12, pp. 1956–1970, 2005.
[9] N. Venkatraman and J. Hammer, "Stable realizations of asynchronous sequential machines with infinite cycles," in *Proc. 2006 Asian Control Conf.*, Bali, Indonesia, pp. 45-51, 2006.
[10] N. Venkatraman and J. Hammer, "On the control of asynchronous sequential machines with infinite cycles," *Int. J. Control*, vol. 79, no. 7, pp. 764–785, 2006.
[11] J.–M. Yang, "Corrective control of asynchronous sequential machines in the presence of adversarial input," *IET Control Theory Appl.*, vol. 2, no. 8, pp. 706–716, 2008.
[12] J.–M. Yang and J. Hammer, "State feedback control of asynchronous sequential machines with adversarial inputs," *Int. J. Control*, vol. 81, no. 12, pp. 1910-1929, 2008.
[13] J.–M. Yang, "Corrective control of input/output asynchronous sequential machines with adversarial inputs," *IEEE Trans. Autom. Control*, vol. 55, no. 3, pp. 755–761, 2010.
[14] J.–M. Yang and J. Hammer, "Asynchronous sequential machines with adversarial intervention: the use of bursts," *Int. J. Control*, vol. 83, no. 5, pp. 956–969, 2010.
[15] J.–M. Yang and S. W. Kwak, "Realizing fault-tolerant asynchronous sequential machines using corrective control," *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 6, pp. 1457–1463, 2010.
[16] J.–M. Yang and S. W. Kwak, "Model matching for asynchronous sequential machines with adversarial inputs using state bursts," *Int. J. Control, Autom., Syst.*, vol. 8, no. 5, pp. 985–993, 2010.
[17] J.–M. Yang and S. W. Kwak, "Model matching for asynchronous sequential machines with uncontrollable inputs," *IEEE Trans. Autom. Control*, vol. 56, no. 9, pp. 2140–2145, 2011.
[18] J.–M. Yang and S. W. Kwak, "Corrective control of asynchronous machines with uncontrollable inputs: application to single-event-upset error counters," *IET Control Theory Appl.*, vol. 4, no. 11, pp. 2454–2462, 2010.
[19] J.–M. Yang, "Fault tolerance in asynchronous sequential machines using output feedback control," *IEEE Trans. Autom. Control*, vol. 57, no. 6, pp. 1604–1609, 2012.
[20] J.–M. Yang and S. W. Kwak, "Fault diagnosis and fault-tolerant control of input/output asynchronous sequential machines," *IET Control Theory Appl.*, vol. 6, no. 11, pp. 1682–1689, 2012.
[21] J.–M. Yang, T. Xing, and J. Hammer, "Adaptive control of asynchronous sequential machines with state feedback," *European J. Control*, vol. 18, no. 6, pp. 503–527, 2012.
[22] C. M. Krishina and K. G. Shin, *Real-Time Systems*, New York: McGraw-Hill, 1997.
[23] C. M. Özveren, A. S. Willsky, and P. J. Antsaklis, "Stability and stabilizability of discrete event dynamical systems," *J. ACM*, vol. 38, no. 3, pp. 730–752, 1991.
[24] Q. Wen, R. Kumar, J. Huang, and H. Liu, "A framework for fault-tolerant control of discrete event systems," *IEEE Trans. Autom. Control*, vol. 53, no. 8, pp. 1839–1849, 2003.
[25] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Norwell, MA: Kluwer, 1999.
[26] Z. Kohavi and N. K. Jha, *Switching and Finite Automata Theory*, 3rd ed. Cambridge University Press: Cambridge, UK, 2010.

ISBN: 978-988-19252-5-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)