Proceedings of the 7th Asian Control Conference,
Hong Kong, China, August 27-29, 2009

Using Bursts to Overcome The Effects of Adversarial Interventions on Asynchronous Sequential Machines

Jung-Min Yang and Jacob Hammer

Abstract— **Feedback controllers that automatically counteract the effects of adversarial interventions on the operation of asynchronous sequential machines are considered. Bursts - fast outbursts of characters generated by the controlled machine during transitions - are employed to broaden the conditions under which such controllers exist. Included are necessary and sufficient conditions for the existence of controllers as well as design considerations.**

I. INTRODUCTION

Asynchronous sequential machines, or clockless logic circuits, play important roles in many branches of engineering and science. They form building blocks of high-speed computing machines, industrial controllers, and traffic control systems; they are used in the modeling, analysis, and design of parallel computing systems; and they appear in the modeling and analysis of signaling chains in molecular biology (e.g., [5]). The present note concentrates on the development of controllers that automatically counteracts the effects of adversarial interventions and disturbances. Examples of adversarial interventions include attempts by rogue computer operators to infiltrate computing networks or the impact of viruses on biological cells.

To represent adversarial interventions we consider asynchronous machines with two inputs: a legitimate input - the *control input* (the input u of Σ in Figure 1); and a subversive input - the *adversarial input* (the input w of Σ in the figure). The objective is to develop a controller C - another asynchronous machine - that automatically counteracts the effects of commands received at the adversarial input. The closed loop machine Σ_c is shown in Figure 1.

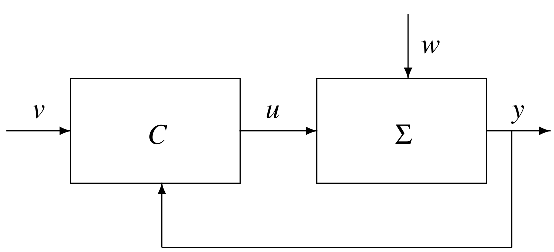

Fig. 1. The basic configuration

An adversarial command at w may cause Σ to undergo state transitions. The controller C detects the bursts created by these transitions and reacts by applying to Σ a control command sequence that takes Σ back to the state it had before the adversarial event. Being an asynchronous machine, the controller's reaction is very quick (ideally, in zero time), so users of Σ_c remain unaffected by adversarial interferences.

An asynchronous machine may be in a *stable state* - a state at which it lingers until an input change occurs, or in a *transient state* - a state through which it passes quickly (ideally, in zero time) on its way from one stable state to another. A *burst* is a fast string of output characters produced by the transient states traversed along a transition from one stable state to another. Controllers that counteract adversarial interventions without using bursts are discussed in [20, 21]. The present note extends these results by incorporating the use of bursts, thus broadening the controller's capabilities.

Fundamental mode operation is an operating policy that guarantees deterministic behavior; it allows only one variable change value at a time (e.g., [10]). Asynchronous machines are almost always operated in fundamental mode.

Fact 1. *The composite machine Σ_c of Figure 1 operates in fundamental mode if and only if all the following requirements are satisfied:*
(i) Σ is in a stable state when C undergoes transitions.
(ii) C is in a stable state when Σ undergoes transitions.
(iii) The variables v and w change value only when Σ and C are both in a stable state, and then only one at a time. □

Other studies on the control of sequential machines can be found in the literature on discrete-event systems ([15], [16]); in [5, 6, 7, 8, 9, 2, 1], where issues related to model matching are considered; in the references listed in these publications; and in may other sources. These publications do not consider specialized issues related to the function of asynchronous machines, such as the presence of stable and transient states or fundamental mode operation.

The present paper extends the framework developed in [3, 4, 12, 13, 14, 18, 19, 21, 22] and is organized as follows. Section 2 reviews notation and background. Section 3 introduces notions of detectability, which determine whether the machine Σ can be operated in a deterministic and unambiguous way after an adversarial intervention. The remaining sections derive necessary and sufficient conditions for the existence of controllers that counteract the effects of adversarial interventions. An illustrative example is included.

II. ASYNCHRONOUS SEQUENTIAL MACHINES

Given an alphabet D, denote by D^* the set of all strings of characters of D and by D^+ the set of all non-empty strings

J.-M. Yang is with the Department of Electrical Engineering, Catholic University of Daegu, 330 Kumrak, Hayang, Kyongsan, Kyongbuk, 712-702, Republic of Korea. jmyang@cu.ac.kr

J. Hammer is with the Department of Electrical and Computer Engineering, University of Florida, P.O. Box 116130, Gainesville, FL 32611, USA. hammer@mst.ufl.edu

978-89-956056-9-1/09/©2009 ACA

of characters of D. For a string $z := z_1z_2 \in D^+$ formed by the concatenation of two strings $z_1, z_2 \in D^*$, the string z_1 is called a *prefix* of z. If neither z_1 nor z_2 are empty strings, then z_1 is a *strict prefix* of z.

An input/state asynchronous sequential machine Σ with two inputs is represented by a triple $(A \times B, X, f)$, where A is the control input alphabet, B is the adversarial input alphabet, X is a set of n states, and $f : X \times A \times B \to X$ is a partial function serving as the *recursion function* of Σ. The machine operates according to the recursion

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

with an initial condition $x_0 \in X$. Here, $u_0, u_1, u_2, ...$ is the control input sequence; $w_0, w_1, w_2, ...$ is the adversarial input sequence; and $x_0, x_1, x_2, ...$ is the resulting sequence of states.

An element $(x, u, w) \in X \times A \times B$ is a *valid combination* if the function f is defined at it. A pair $(x, u) \in X \times A$ is *valid* if there is a character $w \in B$ such that (x, u, w) is a valid combination. A *stable combination* is a valid combination for which $x = f(x, u, w)$, i.e., a "fixed point" of f; the state x is then a *stable state*. By (1), an asynchronous machine rests at a stable combination until a change occurs at one of its inputs.

When (x, u, w) is not a stable combination, it initiates a chain of transitions $x_0 = x, x_1 = f(x_0, u, w), x_2 = f(x_1, u, w), \dots$ If this chain terminates, then there is an integer $i \geq 0$ such that $x_i = f(x_i, u, w)$, and x_i is the *next stable state* of x with the input (u, w). The *stable recursion function* $s : X \times A \times B \to X$ of Σ is defined by $s(x, u, w) := x'$, where x' is the next stable state of x with the input (u, w). The stable recursion function gives rise to the *stable-state machine* $\Sigma_{|s} := (A \times B, X, s)$. For a user, the observed behavior of an asynchronous machine is its stable-state machine, since transients disappear very quickly (ideally, in zero time). An input string $\alpha = \alpha_0\alpha_1...\alpha_m \in (A \times B)^+$ takes Σ to the stable state

$$s(x, \alpha) := s(s(s(x, \alpha_0), \alpha_1), ...\alpha_m). \tag{2}$$

In fundamental mode operation, only one component of α can change at each step.

Assume now that the machine Σ is at a stable combination $(x, a) \in X \times (A \times B)$, when a change occurs in one of the input characters. Let $a' \in A \times B$ be the new input pair and let $x' := s(x, a')$ be the next stable state of Σ. This change may take Σ through $i \geq 1$ transient steps $x_1 := f(x_0, a'), x_2 := f(x_1, a'), \dots, x_i := f(x_{i-1}, a') = x'$. The string

$$b(x, a) := x_1x_2...x_i$$

is then the *burst* created by (x, a'). In an asynchronous machine, a burst occurs very quickly (ideally, in zero time).

The set $\Omega \subset B$ of all adversarial input characters that may appear at the adversarial input of Σ is called the *adversarial uncertainty*; it describes all options of adversarial input characters. Our discussion centers on controllers that eliminate the effects of adversarial interventions, as follows.

Problem 2. *Control objective*: For an asynchronous machine Σ with adversarial uncertainty Ω, find necessary and sufficient conditions for the existence of a controller C for which the stable state composite machine of Figure 1 is unaffected by adversarial interventions and operates in fundamental mode. If such a controller exists, describe its design. □

III. DETECTABILITY

Recall from Fact 1 that in fundamental mode operation, the controller C must wait until the machine Σ has reached its next stable state before reacting to an adversarial intervention. Thus, it must be possible for C to determine whether Σ has reached its next stable state, despite uncertainty about the adversarial input. This determination must be based on information available to the controller; it leads to the following notion (compare to [20], where bursts were not used).

Definition 3. Let Σ be an asynchronous machine with adversarial input resting in a stable combination at a state x, when the control input character switches to u'. Let b be the burst induced by the resulting stable transition. The pair (x, u) is *detectable* if it is possible to determine from b and u' whether Σ has reached its next stable state. □

Fundamental mode operation of the composite machine of Figure 1 is possible only at detectable pairs of Σ. Detectability depends on the information we can deduce about the adversarial input, as follows. Let s be the stable recursion function of Σ; assume that Σ is in a stable combination at an initial state x with the control input character u. The set of adversarial input characters compatible with this data is

$$\omega(x, u) := \{w \in \Omega : s(x, u, w) = x\}. \tag{3}$$

When x is not an initial state, further information about the adversarial input can be derived from the burst history of Σ. The *residual adversarial uncertainty* $\upsilon(x, u)$ is the set of all adversarial input characters that are compatible with the available information. Consider a switch of the control input character to u', and let x' be the next stable state of Σ. The adversarial input character w remains constant during this transition (fundamental mode operation), which may consist of $q \geq 1$ steps: $x_1 := f(x, u', w)$, $x_2 := f(x_1, u', w)$, $\dots, x_q := f(x_{q-1}, u', w) = x'$. The burst of this transition depends on w and is given by

$$b(x, u', w) := x_1...x_q. \tag{4}$$

As the set of all possible adversarial input characters is $\upsilon(x, u)$, the set of all possible bursts is

$$B(x, u, u') := \{b(x, u', w) : w \in \upsilon(x, u)\}. \tag{5}$$

Example 4. Consider an input/state asynchronous machine Σ with $A = \{a, b, c\}$, $B = \{\alpha, \beta, \gamma\}$, $X = \{x^1, x^2, x^3\}$, and state flow diagram shown in Figure 2.

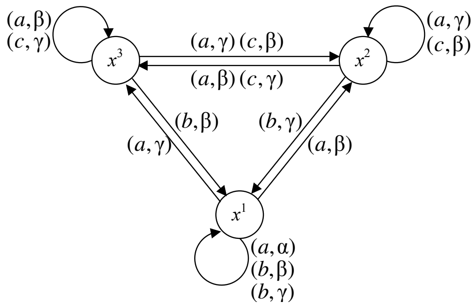

Fig. 2. State flow diagram of an example

Assume that Σ is in an initial stable combination at x^1 with control input b, and let $\Omega = B$. Then, $\omega(x^1, b) = \{\beta, \gamma\}$. Assume further the control input changes to a, starting state transitions. Observing Figure 2, we obtain $b(x^1, a, \beta) = x^2x^3$ and $b(x^1, a, \gamma) = x^3x^2$. Thus,

$$B(x^1, b, a) = \{x^2x^3, x^3x^2\}. \square \tag{6}$$

We arrive at the following characterization of detectability (compare to [13], where a related approach is used in the control of asynchronous machines with races).

Theorem 5. *Let $\Sigma = (A \times B, X, f)$ be an asynchronous machine in a stable combination at the state x, when the control input character switches from u to u'. Let $\upsilon(x,u)$ be the residual adversarial uncertainty and let $B(x,u,u')$ be the set of bursts (5). Then, the following are equivalent.*

(i) The pair (x, u') is detectable.

(ii) No member of $B(x,u,u')$ is a strict prefix of another member.

Proof: (sketch) If *(ii)* is not valid, then there are adversarial input characters $w, w' \in \upsilon(x,u)$ for which $b(x,u',w)$ is a strict prefix of $b(x,u',w')$. Letting x' (respectively, x'') be the stable state at the end of the burst $b(x,u',w)$ (respectively, $b(x,u',w')$), we have $b(x,u',w) = \cdots x'$ and $b(x,u',w') = \cdots x' \cdots x''$. Thus, at the end of $b(x,u',w)$, it is impossible to tell whether Σ is in a stable combination without knowing which of w or w' the adversarial input character was. As the latter is unknown, (x,u') is not detectable, and *(i)* implies *(ii)*. Conversely, when *(ii)* is valid, the end of a burst always signifies a stable combination, and *(ii)* implies *(i)*. $\square$

Example 6. Consider the machine Σ of Example 4. Assume that Σ is in a stable combination at (x^1, b) when the control input switches to a. We can see from (6) that no member of $B(x^1, b, a)$ is a strict prefix of another member. Hence, by Theorem 5, the pair (x^1, b) is detectable. $\square$

Similarly, fundamental mode operation requires that it be possible to determine whether Σ has reached its next stable state after a character switch at the adversarial input.

Definition 7. Assume that the asynchronous machine Σ is in a stable combination at the state x with the control input character u, when a switch of the adversarial input character occurs. The pair (x,u) is *adversarially detectable* if it is possible to determine from the control input and the burst of Σ whether Σ has reached its next stable state. The machine Σ is *adversarially detectable* if every valid pair (x,u) of Σ is adversarially detectable. $\square$

Necessary and sufficient conditions for adversarial detectability are analogous to the conditions of Theorem 5. For a state x and a control input character u, the set of all bursts that can result from a switch of the adversarial input character (recalling the adversarial uncertainty Ω) is

$$B_a(x,u) := \{b(x,u,w) : w \in \Omega\}. \tag{7}$$

Example 8. Suppose that Σ of Figure 2 is in a stable combination at the state x^1 with control input a. Then, the adversarial input character must be α, so it can switch to either β or γ. Hence,

$$B_a(x^1,a) = \{b(x,a,\alpha), b(x,a,\beta), b(x,a,\gamma)\} = \{x^1, x^2x^3, x^3x^2\}. \square \tag{8}$$

Theorem 9. *Assume that the asynchronous machine Σ with adversarial uncertainty Ω is in a stable combination at the state x with the control input character u, when the adversarial input character is switched. Let $B_a(x,u)$ be the set of bursts (7). Then, the following are equivalent.*

(i) The pair (x,u) is adversarially detectable.

(ii) No member of $B_a(x,u)$ is a strict prefix of another member. $\square$

Example 10. From Example 8, the set $B_a(x^1, a)$ has no member that is a strict prefix of another member. Hence (x^1, a) is adversarially detectable by Theorem 9. $\square$

For an adversarially detectable machine Σ, the controller C of Figure 1 operates as follows: C records the latest stable state x of Σ and its current control input value u; when C detects a burst b of Σ without a change of the control input, it compares b to members of $B_a(x,u)$. By Theorem 9*(ii)*, Σ has reached its next stable state when b becomes equal to a member of $B_a(x,u)$. At that point, C can start to counteract the adversarial transition in fundamental mode operation.

IV. Uncertainty

A. Uncertainty After an Adversarial Transition

Consider an asynchronous input/state machine $\Sigma = (A \times B, X, f)$ with adversarial uncertainty Ω resting in a stable combination at an adversarially detectable pair $(x,u) \in X \times A$. A change of the adversarial input character to w' causes Σ to move to a stable combination at the state x'. Let b be the burst generated by Σ during this transition. The set of adversarial input characters compatible with this data is

$$\upsilon(x,u,b) := \{w \in \Omega : b(x,u,w) = b\}. \tag{9}$$

This set characterizes the uncertainty about the adversarial input character immediately before controller reaction.

The reaction of the controller C of Figure 1 to an adversarial transition is a string of control input characters it applies to the machine Σ to reverse the adversarial transition. The controller applies the string one character at a time; after each character, C waits until Σ has reached its next stable state, and then applies the next character of the string. The resulting chain of transitions of Σ forms part of a single stable transition of the closed loop machine Σ_c, and is completed quickly (ideally, in zero time). The adversarial input character remains constant during this process (fundamental mode operation). After each step of this transition chain, more information may be gained about the adversarial input character, as follows.

Let $X = \{x^1, x^2, ..., x^n\}$ be the state set of Σ, and let s be its stable recursion function. Assume that Σ is at a stable combination $(x^i, u_0, w) \in X \times A \times B$ when a control input string $u = u_0u_1u_2...u_t \in A^+$ is applied, with the adversarial input fixed at w. Denote $\alpha := w|u$. Suppose that α takes Σ to a stable combination with the state x^j through the chain of stable transitions $x_0 = x^i$, $x_1 = s(x_0, u_1, w)$, $x_2 = s(x_1, u_2, w)$, ..., $x_t = s(x_{t-1}, u_t, w) = x^j$. Let $b_p(\alpha)$, $p \in \{1, 2, ..., t\}$, be the burst generated by Σ on its way from x_{p-1} to x_p. Then, using (9), the set of all adversarial input characters that are compatible with the data about step p is given by

$$\upsilon(x_{p-1}, u_p, b_p(\alpha)) = \{w \in \Omega : b(x_{p-1}, u_p, w) = b_p(\alpha)\} \quad (10)$$

The *residual uncertainty* $\upsilon_p(\alpha)$ at the end of step p is the set of adversarial input characters that are compatible with all the data collected up to the end of step p of the transition chain. This includes the bursts of steps $1, 2, ..., p$ as well as the initial information about the adversarial input character.

Recall that our objective is to counteract adversarial transitions. Thus, the controlled transition from x^i to x^j is in response to an adversarial transition from x^j to x^i. Denoting by $b_a(j,i)$ the burst of this adversarial transition, it follows from (9) that, immediately after the adversarial transition, we can infer that the adversarial input character belongs to the set

$$\upsilon_{ij}(u_0) := \upsilon(x^j, u_0, b_a(j,i)). \quad (11)$$

Combining this with (10), the residual adversarial uncertainty $\upsilon_p(\alpha)$ satisfies

$$\begin{aligned} &\upsilon_0(\alpha) := \upsilon_{ij}(u_0), \\ &\upsilon_r(\alpha) = \upsilon_{r-1}(\alpha) \cap \upsilon(x_{r-1}, u_r, b_r(\alpha)), r = 1, ..., p. \end{aligned} \quad (12)$$

This characterizes the information about the residual adversarial uncertainty available to the controller before step $p+1$ of the control input string, and proves the following.

Lemma 11. *The set $\upsilon_p(\alpha)$ of (12) forms the residual adversarial uncertainty at the end of step p of a chain of stable transitions induced by an input string $\alpha = w|u \in \Omega|A^+$ in response to an adversarial transition.* $\square$

B. The extended matrix of stable transitions

Recall that the adversarial input character remains constant during the controller's response to an adversarial transition. We investigate next the operation of an asynchronous machine Σ under conditions of constant adversarial input. Let s be the stable recursion function of Σ and let Ω be the adversarial uncertainty. A state x' of Σ is *stably reachable* from a state x in the presence of the adversarial input character w if there is a control input string $u \in A^+$ such that $x' = s(x, u, w)$. We construct a matrix that characterizes the stable reachability features of the machine Σ (compare to [20, 21]), starting with some notation.

For a string $w|u \in \Omega|A^+$, where $u = u_0u_1...u_t$, define the projection $\Pi_i^c : B|A^+ \to A$ onto the i-th control input character

$$\Pi_i^c w|u := \begin{cases} u_i \text{ if } i \le t, \\ u_t \text{ if } i > t, \end{cases}$$

$i = 0, 1, 2, ...$ Denote by $\Pi^a : B|(A^+ \cup N) \to B : (w|u) \mapsto w$ the projection onto the adversarial input character.

Let $X = \{x^1, x^2, ..., x^n\}$ be the state set of Σ. The set of all control input strings $u \in A^+$ that take Σ from a stable combination with x^i to a stable combination with x^j in the presence of the adversarial input character $w \in \Omega$ is

$$\sigma(w, x^i, x^j) := \left\{ w|u \in w|A^+ : \begin{array}{l} x^j = s(x^i, u, w) \text{ and} \\ (x^i, \Pi_0^c w|u, w) \text{ is a stable} \\ \text{combination.} \end{array} \right\}$$

Define the quantity

$$\rho_{ij} := \bigcup_{w \in \Omega} \sigma(w, x^i, x^j).$$

Assume now that a stable adversarial transition from x^j to x^i has occurred, ending with the control input character u_0. Consider now an element $\alpha \in \rho_{ij}$. For α to represent a control string that takes Σ back from x^i to x^j, its adversarial input character $w := \Pi^a\alpha$ must be a member of the set $\upsilon_{ij}(u_0)$ of (11), which consists of all adversarial input characters compatible with the adversarial transition from x^j to x^i. As the initial control input character is $u_0 = \Pi_0^c\alpha$, the set of all members of ρ_{ij} that can take Σ back from x^i to x^j under the present conditions is

$$R_{ij}^*(\Sigma, \Omega) := \{\alpha \in \rho_{ij} : \Pi^a\alpha \in \upsilon_{ij}(\Pi_0^c\alpha)\}, \quad (13)$$

$i, j = 1, 2, ..., n$. We call $R^*(\Sigma, \Omega)$ the *extended matrix of stable transitions* of Σ. The features of our construction are summarized in the following.

Lemma 12. *Let Σ be an asynchronous machine with adversarial uncertainty Ω, state set $X = \{x^1, x^2, ..., x^n\}$, and extended matrix of stable transitions $R^*(\Sigma, \Omega)$. Then, the following are equivalent for all $i, j = 1, 2, ..., n$.*

(i) The entry $R_{ij}^*(\Sigma, \Omega)$ includes a string $w|u$.

(ii) The control input string u takes Σ from a stable combination at x^i to a stable combination at x^j in the presence of the adversarial input character w; here, w is consistent with an earlier adversarial transition from x^j to x^i.

V. Detectable Feedback Paths

In this section, we examine the existence of automatic controllers that counteract adversarial transitions. Consider a machine Σ with state set $X = \{x^1, x^2, ..., x^n\}$, adversarial uncertainty Ω, stable recursion function s, and extended matrix of stable transitions $R^*(\Sigma, \Omega)$. For a string $\alpha = w|u \in R_{ij}^*(\Sigma, \Omega)$, where $u = u_0u_1...u_t$, let $x_p := s(x^i, u_0u_1...u_p, w)$ be the stable state of Σ at step p, $p \in \{0, 1, ..., t\}$; here, $x_0 := x^i$ and $x_t := x^j$. In order for the controller to operate in fundamental mode, the pair (x_p, u_{p+1}) must be detectable with respect to the residual adversarial uncertainty $\upsilon_p(\alpha)$ at all steps $p = 0,1,...,t-1$ (see section 3). Let $b_k(\alpha)$ be the burst generated by Σ during its transition from x_{k-1} to x_k, so that the string of bursts generated by Σ up to step p is

$$b_0^p(\alpha) := \{x_0, b_1(\alpha), b_2(\alpha), ..., b_p(\alpha)\}.$$

As the only information the controller receives from Σ are the bursts, the controller cannot distinguish at step p between adversarial input characters that produce equal bursts $b_0^p(\alpha)$ (given the same control input history of Σ). For all such adversarial input characters, the controller produces for Σ the same control input character at step $p+1$. This is, of course, a fundamental restriction on feedback controllers.

Before continuing, we need some notation. Denote by $\Pi^c : B|A^+ \to A^+ : (w|u) \mapsto u$ the projection onto the control input string, and, for a string $\alpha = w|u_0u_1...u_t \in B|A^+$ and an integer $p \geq 0$, denote the truncation

$$\alpha_{|p} := \begin{cases} w|u_0u_1...u_p \text{ for } p \leq t, \\ w|u_0u_1...u_t \text{ for } p > t. \end{cases}$$

Now, given a set of strings $S \subset B|A^+$, an integer $p \geq 0$, and a string $\alpha \in S$, denote by $S(\alpha, p)$ the set of all strings of S which, up to step p, have the same control input characters and produce the same string of bursts as α; namely,

$$S(\alpha, p) := \{a \in S : \Pi^c a_{|p} = \Pi^c \alpha_{|p} \text{ and } b_0^p(a) = b_0^p(\alpha)\} \quad (14)$$

Strings in $S(\alpha, p)$ may have different adversarial input characters as well as different continuations beyond step p. The following notion is critical to our discussion (compare to [17, 18, 19, 20, and 21]).

Definition 13. Let Σ be an asynchronous machine with state set $X = \{x^1, x^2, ..., x^n\}$, stable recursion function s, and initial adversarial uncertainty Ω. Assume that Σ is in a stable combination at the state x^i with the control input value u_0. A subset $S \subset R^*_{ij}(\Sigma, \Omega)$ is a *detectable feedback path* from x^i to x^j if the following conditions are satisfied for every element $\alpha \in S$ and for every integer $p \geq 0$:

(i) $\Pi_0^c S$ consists of a single element u_0;

(ii) The set $\Pi_{p+1}^c S(\alpha, p)$ consists of a single element; and

(iii) The pair $(s(x^i, \alpha_{|p}), \Pi_{p+1}^c \alpha)$ is detectable with respect to the residual uncertainty $\upsilon_p(\alpha)$ of (12).

Here, u_0 is the *initial control input character* of S. $\square$

Example 14. To derive a detectable feedback path from x^2 to x^1 for the machine Σ of Figure 2, note that $\rho_{21} = \{\gamma|ab, \beta|cab\}$. Assume that an adversarial transition occurred from x^1 to x^2 with the burst $b_a(1,2) = x^3x^2$, ending at the stable pair (x^2, a); then, $u_0 = a$. By (11), we have $\upsilon_{21}(a) = \upsilon(x^1, a, x^1x^3x^2) = \{\gamma\}$ and $R^*_{21}(\Sigma, \Omega) = \{\gamma|ab\}$ (see (13)). Clearly, $\gamma|ab$ satisfies conditions *(i)* and *(ii)* of Definition 13. As there is only one step in this path, we need to consider only $p = 0$; then, in condition *(iii)* of Definition 13, we have $(s(x^2, \gamma|a), b) = (x^2, b)$, which is detectable with respect to $\upsilon_0(\alpha) = \{\gamma\}$ ($B(x^2, a, b)$ has only one member). Hence, $S = \{\gamma|ab\}$ is a detectable feedback path from x^2 to x^1. $\square$

The presence of a feedback path is equivalent to the existence of a controller that counteracts adversarial transitions, as follows.

Theorem 15. *Let Σ be an asynchronous machine with adversarial uncertainty Ω, state set $X = \{x^1, x^2, ..., x^n\}$, and extended matrix of stable transitions $R^*(\Sigma, \Omega)$. Assume that Σ underwent a detectable stable adversarial transition from x^j to x^i in the presence of the control input character u_0. Then, the following are equivalent for all $i, j \in \{1, 2, ..., n\}$.*

(i) There is a controller $C(i, j, v)$ that takes Σ back from a stable combination with x^i to a stable combination with x^j in fundamental mode operation, where v is the external input character of Figure 1.

(ii) The entry $R^*_{ij}(\Sigma, \Omega)$ includes a detectable feedback path with initial control input character u_0.

Proof: (sketch) Note that the external input v of Figure 1 is constant during this process (fundamental mode operation). First, assume that *(i)* is valid, and let $S \subset B|A^+$ be the set of all input strings that $C(i, j, v)$ can generate to steer Σ from x^i to x^j. The initial control input character must be $u_0 := \Pi_0^c S$, the character in force during the adversarial transition from x^j to x^i. Hence, condition *(i)* of Definition 13 is valid. The fact the $C(i, j, v)$ is a causal feedback controller means that its response is determined by its past and present inputs; hence, condition *(ii)* of Definition 13 is valid. Finally, fundamental mode operation of the composite machine $\Sigma_{C(i,j,v)}$ implies that all steps are detectable (see section 3); consequently, condition *(iii)* of Definition 13 is valid. Thus, S is a detectable feedback path.

Conversely, assume that condition *(ii)* of the theorem is valid, and let $S \subset R^*_{ij}(\Sigma, \Omega)$ be a detectable feedback path with initial control input character u_0. Define a controller $C(i, j, v)$ as follows:

(a) The initial output character of $C(i, j, v)$ is u_0.

(b) Recursively, assuming that $C(i, j, v)$ was defined up to step $p \geq 0$, the $p+1$ output character of $C(i, j, v)$ is the single member of $\Pi_{p+1}^c S(\alpha, p)$ (Definition 13 *(ii)*).

Then, condition *(iii)* of Definition 13 assures detectability, guaranteeing fundamental mode operation of the closed loop machine $\Sigma_{C(i,j,v)}$ (section 3). Finally, since $S \subset R^*_{ij}(\Sigma, \Omega)$, all strings generated by $C(i, j, v)$ take Σ from a stable combination with x^i to a stable combination with x^j. $\square$

To show that the controller of Theorem 15 can be implemented with a finite state set, denote by $\#Z$ the cardinality of a set Z. The *length* of a string $\alpha = w|u \in B|A^+$ is the length of the string u, i.e., $|\alpha| := |u|$.

Definition 16. Let Σ be an asynchronous machine with state set X, adversarial uncertainty Ω, and extended matrix of stable transitions $R^*(\Sigma, \Omega)$. Denote $\kappa := (\#X)(\#\Omega)$. The *matrix of stable transitions* $R(\Sigma, \Omega)$ is obtained from $R^*(\Sigma, \Omega)$ by deleting all strings of length exceeding κ. $\square$

The following indicates that the matrix of stable transitions can replace the extended matrix of stable transitions.

Lemma 17. *Let Σ be an asynchronous machine with adver-*

sarial uncertainty Ω, state set $X = \{x^1, x^2, ..., x^n\}$, extended matrix of stable transitions $R^*(\Sigma,\Omega)$, and matrix of stable transitions $R(\Sigma,\Omega)$. Then, the following are equivalent for all $i, j \in \{1,2,...,n\}$.

(i) The entry $R^*_{ij}(\Sigma,\Omega)$ includes a detectable feedback path.

(ii) The entry $R_{ij}(\Sigma,\Omega)$ includes a detectable feedback path.

Proof: (sketch) Clearly, as $R_{ij}(\Sigma,\Omega) \subset R^*_{ij}(\Sigma,\Omega)$, *(ii)* implies *(i)*. Conversely, consider a detectable feedback path $S^* \subset R^*_{ij}(\Sigma,\Omega)$. For a string $\alpha \in S^*$, let $\upsilon_p(\alpha)$ be the residual uncertainty at a step $p \geq 0$. Then, $\upsilon_p(\alpha), p = 0,1,...$ is a monotone decreasing sequence of subsets, so $\#\upsilon_p(\alpha) \leq \#\upsilon_0 \leq \#\Omega$. As there are only n states, the number of distinct pairs $(s(x^i, \alpha_{|p}), \upsilon_p(\alpha))$ cannot exceed $\kappa = n(\#\Omega)$. Thus, removing all the repetitions from the list $\{(s(x^i, \alpha_{|k}), \upsilon_k(\alpha))\}_{k=0,1,...}$ yields a list of length not exceeding κ. Applying this process to every member $\alpha \in S^*$, we obtain a detectable feedback path included in $R_{ij}(\Sigma,\Omega)$. $\square$

By Lemma 17, we can replace $R^*(\Sigma,\Omega)$ in Theorem 15 by $R(\Sigma,\Omega)$. As $R(\Sigma,\Omega)$ includes only strings of length bounded by κ, this assures that the controller can be implemented with a finite state set.

Corollary 18. *Let Σ be an asynchronous machine with adversarial uncertainty Ω, state set $X = \{x^1, x^2, ..., x^n\}$, and matrix of stable transitions $R(\Sigma,\Omega)$. Assume that Σ underwent a detectable adversarial transition from the state x^j to the state x^i in the presence of the control input character u_0. Then, the following two statements are equivalent for all $i, j \in \{1,2,...,n\}$.*

(i) There is a controller $C(i, j, v)$ that takes Σ back from a stable combination with x^i to a stable combination with x^j in fundamental mode operation, where v is the external input character of Figure 1.

(ii) The entry $R_{ij}(\Sigma,\Omega)$ includes a detectable feedback path with initial control input character u_0. $\square$

Thus, the existence of detectable feedback paths is the critical condition for automatic counteraction of adversarial interventions. An algorithm for the derivation of detectable feedback paths is described in [22].

Example 19. For machine Σ of Figure 2, we construct a controller $C(2,1,a)$ that counteracts a detectable adversarial transition from x^1 to x^2 that occurs in the presence of the control input character a. Here, $C(2,1,a) = (X \times A, A, \Xi, \xi^0, \phi, \eta)$, where Ξ is the controller's state set, ξ^0 is its initial state, $\phi : \Xi \times X \times A \to \Xi$ is its recursion function, and $\eta : \Xi \times X \times A \to A$ is its output function. From the initial condition ξ^0, the controller $C(2,1,a)$ moves to the state ξ_1 when it detects the stable pair (x^1, a), in preparation for a possible adversarial transition. To implement this transition, set $\phi(\xi^0, x^1, a) := \xi_1$, and $\eta(\xi_1, x^1, a) := a$. Note that the controller's output value at ξ_1 is a according to (a) of the proof of Theorem 15.

When the adversarial transition from x^1 to x^2 occurs, the control input remains at a. Based on adversarial detectability, $C(2,1,a)$ reacts as soon as Σ has reached x^2 by moving to the controller state ξ_2. Using the detectable feedback path $\gamma|ab$ of Example 14, set $C(2,1,a)$ to apply the control input character b to Σ, i.e., set $\phi(\xi_1, x^2, a) := \xi_2$, and $\eta(\xi_2, x^2, a) := b$. This returns Σ to x^1. $\square$

REFERENCES

[1] G. BARRETT and S. LAFORTUNE [1998] "Bisimulation, the supervisory control problem, and strong model matching for finite state machines," Discrete Event Dynamic Systems: Theory and Application, vol. 8, no. 4, pp. 377-429.

[2] M. D. DIBENEDETTO, A. SALDANHA and A. SANGIOVANNI-VINCENTELLI [1994] "Model matching for finite state machines," Proceedings of the IEEE Conference on Decision and Control, vol. 3, 1994, pp. 3117-3124.

[3] X. J. GENG and J. HAMMER [2004] "Asynchronous sequential machines: input/output control," Proceedings of the 12th Mediterranean Conference on Control and Automation, Kusadasi, Turkey, June 2004.

[4] X. J. GENG and J. HAMMER [2005] "Input/output control of asynchronous sequential machines," IEEE Transactions on Automatic Control, vol. 50, no. 12, pp. 1956-1970.

[5] J. HAMMER [1994] "On some control problems in molecular biology," Proceedings of the IEEE conference on Decision and Control, pp. 4098-4103, December 1994.

[6] J. HAMMER [1995] "On the modeling and control of biological signal chains," Proceedings of the IEEE conference on Decision and Control, pp. 3747–3752, December 1995.

[7] J. HAMMER [1996a] "On the corrective control of sequential machines," International Journal of Control, vol. 65, no. 2, pp. 249-276.

[8] J. HAMMER [1996b] "On the control of incompletely described sequential machines," International Journal of Control, vol. 63, no. 6, pp. 1005-1028.

[9] J. HAMMER [1997] "On the control of sequential machines with disturbances," International Journal of Control, vol. 67, no. 3, pp. 307-331.

[10] Z. KOHAVI [1970] "Switching and finite automata theory," McGraw-Hill Book Company, New York.

[11] T. E. MURPHY, X. J. GENG and J. HAMMER [2002] "Controlling races in asynchronous sequential machines," Proceeding of the IFAC World Congress, Barcelona, July 2002.

[12] T. E. MURPHY, X. J. GENG and J. HAMMER [2003] "On the control of asynchronous machines with races," IEEE Transactions on Automatic Control, vol. 48, no. 6, pp. 1073-1081.

[13] J. PENG and J. HAMMER [2008] "Output feedback control of non-deterministic asynchronous sequential machines", submitted for publication.

[14] J. PENG and J. HAMMER [2008] "Bursts and output feedback control of non-deterministic asynchronous sequential machines", in preparation.

[15] P. J. G. RAMADGE and W. M. WONHAM [1987] "Supervisory control of a class of discrete event processes," SIAM Journal of Control and Optimization, vol. 25, no. 1, pp. 206-230.

[16] J. G. THISTLE and W. M. WONHAM [1994] "Control of infinite behavior of finite automata," SIAM Journal on Control and Optimization, vol. 32, no. 4, pp. 1075-1097.

[17] N. VENKATRAMAN and J. HAMMER [2006a] "Stable realizations of asynchronous sequential machines with infinite cycles," Proceedings of the 2006 Asian Control Conference, Bali, Indonesia, 2006.

[18] N. VENKATRAMAN and J. HAMMER [2006b] "Controllers for asynchronous machines with infinite cycles," Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems, Kyoto, Japan, 2006.

[19] N. VENKATRAMAN and J. HAMMER [2006c] "On the control of asynchronous sequential machines with infinite cycles," International Journal of Control, vol. 79, no. 7, pp. 764-785.

[20] J.-M. YANG and J. HAMMER [2008a] "State feedback control of asynchronous sequential machines with adversarial inputs", International J. Control, Vol. 81, No. 12, pp. 1910-1929.

[21] J.-M. YANG and J. HAMMER [2008b] "Counteracting the effects of adversarial inputs on asynchronous sequential machines", Proceedings of the IFAC World Congress, pp. 1432–1437, Seoul, Korea, July 2008.

[22] J.-M. YANG and J. HAMMER [2008c] "Asynchronous sequential machines with adversarial intervention: the use of bursts", to be submitted for publication.