Math. Control Signals Systems (1992) 5: 365–390

Mathematics of Control, Signals, and Systems

Invertibility of Discrete-Event Dynamic Systems*

Cüneyt M. Özveren† and Alan S. Willsky‡

Abstract. In this paper we consider a class of Discrete-Event Dynamic Systems (DEDS) modeled as finite-state automata in which only some of the transition events are directly observed. An invertible DEDS is one for which it is possible to reconstruct the entire event string from the observation of the output string. The dynamics of invertibility are somewhat complex, as ambiguities in unobservable events are typically resolved only at discrete intervals and, perhaps, with finite delay. A notion of resiliency or error recovery is developed for invertibility, and polynomial-time tests for invertibility and for resilient invertibility, as well as a procedure for the construction of a resilient inverter, are discussed.

Key words. Automata, Invertibility, Observability, Resiliency, Error recovery, Discrete-event dynamic systems.

1. Introduction

For Discrete-Event Dynamic Systems (DEDS) state evolution is triggered by the occurrence of discrete events. Such behavior can be found in many complex, man-made systems at some level of abstraction, such as flexible manufacturing systems and communication systems. DEDS have been studied extensively by computer scientists, and the study of DEDS was introduced into the systems and control context by Wonham, Ramadge, and others. [OW1], [RW1], [RW2], [VW]. This work assumes a finite-state model with certain state transitions that can be enabled or disabled. The control of the system is achieved by choice of control inputs that enable or disable these transitions.

The initial work (see [CDFV], [LW], [OW1], [RW1], [RW2], and [VW]) in this area dealt primarily with linguistic questions—e.g., the use of control to force the trajectory of transition events to lie in a specified language of event strings—and these results prompted a number of researchers to investigate a variety of alternate formulations and questions. In our work we have been motivated primarily by a

* Date received: September 5, 1989. Date revised: February 23, 1992. Research supported by the Air Force Office of Scientific Research under Grant AFOSR-88-0032 and by the Army Research Office under Grant DAAL03-86-K0171. This research was partially done during our stay at the Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA), Rennes, France, and the second author was also supported by IRISA during this time.

† Telecommunications and Networking, Digital Equipment Corporation, 550 King Street, Littleton, Massachusetts 01460, U.S.A.

‡ Laboratory for Information and Decision Systems, MIT, Cambridge, Massachusetts 02139, U.S.A.

desired to develop counterparts for DEDS of standard control and system-theoretic concepts, and, more specifically, by the need we perceived for the development of concepts of *error recovery* or resiliency in DEDS. For example, in [OWA] we develop a notion of stability for DEDS and investigate the design of stabilizing state feedback laws. In [OW2] we focus on the problems of observability and state reconstruction, with an associated investigation of error recovery, while in [OW3] we do the same in the context of designing stabilizing dynamic output compensators.

The focus of attention in this paper is the system-theoretic problem of *invertibility*, i.e., the problem of reconstructing the full transition event sequence given observations of certain output events. Such a problem may arise in the monitoring of a complex system or in troubleshooting a faulty system. Also, the dual of this problem, the generation of *input* sequences to *achieve* specified output behavior, is of considerable importance in characterizing the tracking and regulation capabilities of a DEDS. In addition, as we will see, an inverter for a DEDS can be quite nonresilient. In particular, much as in catastrophic error propagation in sequential decoding [PW2], some DEDS inversion problems have the undesirable property that a finite burst of observation errors can lead to an unbounded sequence of inversion errors. Our work here contributes to the development of a theory and methodology for characterizing when large-scale DEDS can exhibit such behavior and for designing compensation that provides enhanced robustness to errors.

In the next section we introduce the mathematical framework considered in this paper and summarize aspects of our previous work needed in what follows. In Section 3 we define several notions of invertibility, develop efficient tests for these notions, and describe a procedure for constructing an inverter. In Section 4 we define a notion of resilient invertibility very much in the spirit of resilient observability as formulated in [OW2], and we provide a polynomial-time test for this notion as well as a construction for a resilient observer. Finally, we conclude with a brief discussion in Section 5.

2. Background and Preliminaries

2.1. System Model

The class of systems we consider are nondeterministic finite-state automata with intermittent event observations. The basic object of interest is the triple[1]

$$G = (X, \Sigma, \Gamma), \tag{2.1}$$

where X is the finite set of states, with $n = |X|$, Σ is the finite set of possible events, and $\Gamma \subset \Sigma$ is the set of observable events. The dynamics defined on G that we consider in [OWA] are of the form

$$x[k+1] \in f(x[k], \sigma[k+1]), \tag{2.2}$$

$$\sigma[k+1] \in d(x[k]). \tag{2.3}$$

[1] In the complete model considered, for example, in [CDFV], [LW], [OW1], [OW3], and [OWA] we also include control by allowing some events to be disabled. In the present context we do not need to introduce control since we are only interested in an observation problem.

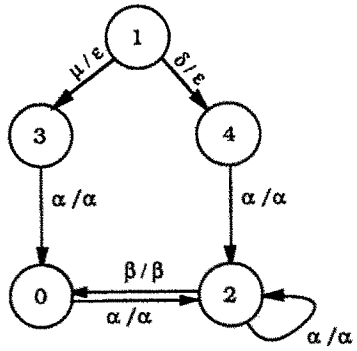

Fig. 2.1. A simple example.

Here, $x[k] \in X$ is the state after the kth event, and $\sigma[k] \in \Sigma$ is the $(k+1)$st event. The function $d: X \to 2^{\Sigma}$ is a set-valued function that specifies the set of possible events defined at each state (so that, in general, not all events are possible from each state), and the function $f: X \times \Sigma \to X$ is also set-valued, so that the state following a particular event is not necessarily known with certainty.

Our model of the output process is quite simple: whenever an event in Γ occurs, we observe it; otherwise, we see nothing. Specifically, we define the output function $h: \Sigma \to \Gamma \cup \{\varepsilon\}$, where ε is the "null transition," by

$$h(\sigma) = \begin{cases} \sigma & \text{if } \sigma \in \Gamma \\ \varepsilon & \text{otherwise.} \end{cases} \tag{2.4}$$

Then our output equation is

$$\gamma[k+1] = h(\sigma[k+1]). \tag{2.5}$$

Note that h can be thought of as a map from Σ^* to Γ^*, where Γ^* denotes the set of all strings of finite length with elements in Γ, including the empty string ε. In particular, $h(\sigma_1 \cdots \sigma_n) = h(\sigma_1) \cdots h(\sigma_n)$. The quadruple $A = (G, f, d, h)$ representing our system can also be visualized graphically as in Fig. 2.1. Here, circles denote states and events are represented by arcs. The first symbol in each arc label denotes the event, while the second symbol denotes the corresponding output. Thus, in this example, $X = \{0, 1, 2, 3, 4\}$, $\Sigma = \{\alpha, \beta, \delta, \mu\}$, and $\Gamma = \{\alpha, \beta\}$.

There are several basic notions needed in our investigation. The first is the notion of *liveness*. A system is alive if it cannot reach a point at which no event is possible. That is, A is alive if, for all $x \in X$, $d(x) \neq \varnothing$. We assume that this is the case. A second notion that we need is the *inverse* of an automaton. Specifically, we define $A^{-1} = (G, f^{-1}, d^{-1}, h)$ by reversing all the arcs in the graph of A so that

$$f^{-1}(x, \sigma) = \{y \in X | x \in f(y, \sigma)\}, \tag{2.6}$$

$$d^{-1}(x) = \{\sigma \in \Sigma | \sigma \in d(y) \text{ for some } y \in X \text{ and } x \in f(y, \sigma)\}. \tag{2.7}$$

2.2. *Stability*

In [OWA] we define a notion of stability which requires that the trajectories go through a given set E infinitely often:

Definition 2.1. Let E be a specified subset of X. A state $x \in X$ is *E-pre-stable* if there exists some integer i such that every trajectory starting from x passes through E in at most i transitions. The state $x \in X$ is *E-stable* if A is alive and every state reachable from x is E-pre-stable. The DEDS is *E-stable* if every $x \in X$ is E-stable (note that this is equivalent to every $x \in X$ being E-pre-stable).

By a *cycle* we mean a finite sequence of states $x_1, x_2, \ldots, x_k$, with $x_k = x_1$, so that there exists an event sequence s that permits the system to follow this sequence of states. Note that E-stability is equivalent to the absence of cycles that do not pass through E [OWA]. We also need the following:

Definition 2.2. The *radius* of A is the length of the longest cycle-free trajectory between any two states of A. The *E-radius* of an E-stable system A is the maximum number of transitions it takes for any trajectory to enter E.

Note that an upper bound on both the radius and the E-radius, for any E, of an E-stable system is n. We refer the reader to [OWA] for a more complete discussion of this subject and for an $O(n^2)$ test for E-stability of a DEDS. Finally, we note that in [OWA] and Definition 2.1, we require liveness in order for a system to be stable so that trajectories can be continued indefinitely. We always require liveness in this paper.

2.3. Observability and Observers

In [OW2] we term a system *observable* if the current state is known perfectly at intermittent but not necessarily fixed intervals of time. Obviously, a necessary condition for observability is that it is not possible for our DEDS to generate arbitrarily long sequences of unobservable events, i.e., events in $\bar{\Gamma}$, the complement of Γ. This is not difficult to check (see [OW2]) and thus observability is assumed.

Let us now introduce some notation that will be useful:

- Let $x \to^s y$ denote the statement that state y is reached from x via the occurrence of event sequence s. Also, let $x \to^* y$ denote that x reaches y in any number of transitions, including none. For any set $Q \subset X$ we define the *reach of Q in A* as

$$R(A, Q) = \{y \in X | \exists x \in Q \text{ such that } x \to^* y\}. \tag{2.8}$$

- Let

$$Y_0 = \{x \in X | \nexists y \in X, \sigma \in \Sigma, \text{ such that } x \in f(y, \gamma)\}, \tag{2.9}$$

$$Y_1 = \{x \in X | \exists y \in X, \gamma \in \Gamma, \text{ such that } x \in f(y, \gamma)\}, \tag{2.10}$$

$$Y = Y_0 \cup Y_1. \tag{2.11}$$

  Thus, Y is the set of states x such that either there exists an observable transition defined from some state y to x (as captured in Y_1) or x has no transitions defined to it (as captured in Y_0). Let $q = |Y|$.
- Let $L(A, x)$ denote the language generated by A, from the state $x \in X$, i.e., $L(A, x)$ is the set of all possible event trajectories of finite length that can be

generated if the system is started from the state x. Also, let $L_f(A, x)$ be the set of strings in $L(A, x)$ that have an observable event as the last event, and let $L(A) = \bigcup_{x \in X} L(A, x)$ be the set of all event trajectories that can be generated by A. Finally, let $L_f(A)$ be the set of strings in $L(A)$ that have an observable event as the last event.

Given $s \in L(A, x)$ such that $s = pr$, p is termed a *prefix* of s and we use s/p to denote the corresponding suffix r, i.e., the remaining part of s after p is taken out.

In [OW2] we present a straightforward design of an observer that produces "estimates" of the state of the system after each observation $\gamma[k] \in \Gamma$. Each such estimate is a subset of Y corresponding to the set of possible states into which A transitioned when the last observable event occurred. Mathematically, if we let a function $\hat{\mathbf{x}}$: $h(L(A)) \to 2^Y$ denote the estimate of the current state given the observed output string $t \in h(L(A))$, then

$$\hat{\mathbf{x}}(t) = \{x \in Y | \exists y \in X \text{ and } s \in L_f(A, y) \text{ such that } h(s) = t \text{ and } x \in f(y, s)\}. \quad (2.12)$$

The observer, for which the state space is a subset Z of 2^Y and the events and observable events are both Γ, is a DEDS which realizes this function. Suppose that the present observer estimate is $\hat{x}[k] \in Z$ and that the next observed event is $\gamma[k+1]$. The observer must then account for the possible occurrence of one or more unobservable events prior to $\gamma[k+1]$ and then the occurrence of $\gamma[k+1]$:

$$\hat{x}[k+1] = w(\hat{x}[k], \gamma[k+1]) \triangleq \bigcup_{x \in R(A|\bar{\Gamma}, \hat{x}[k])} f(x, \gamma[k+1]), \quad (2.13)$$

$$\gamma[k+1] \in v(\hat{x}[k]) \triangleq h\left(\bigcup_{x \in R(A|\bar{\Gamma}, \hat{x}[k])} d(x)\right). \quad (2.14)$$

The set Z is then in the reach of $\{Y\}$ using these dynamics, i.e., we start the observer in the state corresponding to a complete lack of state knowledge and let it evolve.

Our observer then is the DEDS $O = (F, w, v, i)$, where $F = (Z, \Gamma, \Gamma)$ and i is the identity output function. The observer for the example in Fig. 2.1 is illustrated in Fig. 2.2. In [OW2] we show that a system A is observable if and only if O is E-stable for all singleton sets of states E. We also show that if A is observable, then all trajectories from an obsever state pass through a singleton state in at most q^2

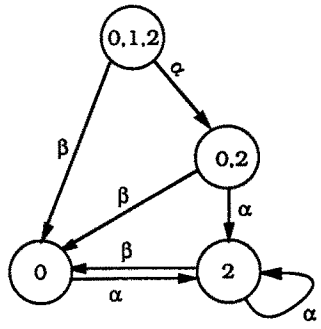

Fig. 2.2. Observer for the system in Fig. 2.1.

transitions. Since there can also be at most q singleton states, the radius of the observer is at most q^3.

In order to construct a polynomial-time test for observability, we use what we term the *pair automaton* associated with A which we define as follows: Let $P = Y \times Y$ and construct an automaton $O_P = (G_P, f_P, d_P, 1)$ with $G_P = (P, \Gamma)$ such that

$$f_P(p, \gamma) = (f'(x, \gamma) \cup f'(y, \gamma)) \times (f'(x, \gamma) \cup f'(y, \gamma)), \tag{2.15}$$

$$d_P(p) = d'(x) \cup d'(y), \tag{2.16}$$

where $p = (x, y) \in P$, and

$$f'(x, \gamma) = f(R(A|\bar{\Gamma}, x), \gamma), \tag{2.17}$$

$$d'(x) = h(d(R(A|\bar{\Gamma}, x))). \tag{2.18}$$

Note that since it is nondeterministic, O_P is certainly not an observer for A. However, if its state ever evolves deterministically to a state of the form (x, x), the automaton A must be in state x. Thus,

Proposition 2.3 [OW2]. *A is observable if and only if O_P is E_P-stable where $E_P = \{(x, x) | x \in Y\}$.*

Since $|P| = q^2$, this gives us a test for observability that has time-complexity $O(q^4)$.

2.4. Observability with a Delay

An extension of the notion of the observability is formulated and analyzed in [OW1], and we use it in what follows. In the following notion of observability with a delay, we only require perfect state knowledge a finite number of transitions into the past:

Definition 2.4. A is *observable with a delay* (WD-observable), if, for all $x \in X$, $s \in L(A, x)$ such that $|s| \geq nq^2$, there exist prefixes $p_1 \in L_f(A, x)$ of s and $p_2 \in L_f(A, x)$ of p_1 such that

- $|s/p_2| \leq nq^2$,
- $f(x, p_2)$ is single valued,
- for all $y \in X$ and $t_1 \in L_f(A, y)$, $h(t_1) = h(p_1)$ implies $f(y, t_2) = f(x, p_2)$ where t_2 is the prefix of t_1 such that $h(t_2) = h(p_2)$.

In the above definition, p_1 is the event trajectory so that given the observation $h(p_1)$, we know precisely the state of A after the occurrence of p_2. The length of the string p_1/p_2 then corresponds to the observation delay. The first condition thus bounds this delay appropriately, and the second and third conditions assure that we know the system state after the occurrence of p_2.

In constructing an algorithm for testing WD-observability, we use the following notion:

Definition 2.5. Given $x \in X$, let $L_\infty(A, x)$ denote the set of infinite length event trajectories generated from x, and let $h(L_\infty(A, x))$ be the corresponding set of output

trajectories. The pair $(x, y) \in Y \times Y$ is an *indistinguishable pair* if $h(L_\infty(A, x)) \cap h(L_\infty(A, y)) \neq \emptyset$, i.e., if there is an infinite length output sequence that could have been generated starting from either x or y.

Let I_M denote the maximal set of indistinguishable pairs; in [OW2] we present a polynomial-time algorithm to compute I_M. We also have the following characterization of WD-observability:

Proposition 2.6. *A is WD-observable if and only if O is E_W-stable where*

$$E_W = \{\hat{x} \in Z | \text{there exists no } x, y \in \hat{x}, x \neq y \text{ such that } (x, y) \in I_M\}.$$

As this result suggests, we can construct a WD-observer as follows [OW2]: Start the observer at state Y and let it evolve. Whenever the observer trajectory enters E_W, we know that we will be able to determine the precise state at that time using future observations, thanks to the distinguishability of the states in E_W.

We may also use the pair automaton associated with A (together with indistinguishability) to construct a polynomial-time test for WD-observability:

Proposition 2.7 [OW2]. *A is WD-observable if and only if O_P is E_{DP}-stable where*

$$E_{DP} = \{(x, y) \notin I_M\}.$$

Finally, we state the following result which plays an important role in the development of Section 4:

Proposition 2.8. *Given $x, y \in Y$, $(x, y) \in I_M$ if and only if x and y share an output string of length greater than or equal to q^2.*

Proof. Straightforward since any path of length q^2 in O_P has a cycle embedded in it.

For future reference, let n_i denote the minimum number of observations required to distinguish between any pair of distinguishable states in A. What Proposition 2.8 states is that $n_i \leq q^2$. Of course, in many systems n_i can be much smaller than this bound.

2.5. *Resiliency*

An important aspect of our work concerns resiliency or error recovery. Specifically, suppose that the observed sequence of transitions includes errors corresponding to inserted events, missed events, or mistaken events. We terms an observer *resilient* if at the end of a finite burst of such measurement errors, the observer resumes correct behavior after a finite number of transitions, i.e., the current observer estimate includes the current state of the system. In [OW2] we construct a resilient observer as follows: The observer O as specified in (2.13) and (2.14) is defined only for event sequences that can actually occur in the system. When measurement error occurs, the resulting observed sequence may not be feasible. In this case the observer

at some point will be in a state such that the next observed event is not defined. In this case we reset the observer state to $\{Y\}$, i.e., to the condition of knowing nothing about the system state. Thus, for each state in Z and for all events that are not defined at that state, we add a transition to $\{Y\}$. In particular, we modify w and v as follows:

$$w_R(\hat{x}, \gamma) = \begin{cases} w(\hat{x}, \gamma) & \text{if} \quad \gamma \in v(\hat{x}), \\ \{Y\} & \text{otherwise,} \end{cases} \tag{2.19}$$

$$v_R(\hat{x}) = \Gamma, \tag{2.20}$$

and we thus construct the observer $O_R = (F, w_R, v_R, i)$. As before, the initial state of O_R is the state $\{Y\}$. We show in [OW2] that O_R is a resilient observer if A is observable or, in fact, WD-observable.

3. Invertibility

In this section we present and analyze two notions of invertibility: The first notion assumes that the initial state is known, while the second does *not*. Since a reconstruction in the latter case involves estimating the current state first, our second notion allows for a bounded error in the beginning of the reconstructed string.

3.1. Invertibility with a Delay

We consider first the problem in which the initial state x_0 of A is known. We assume that A is a minimal automaton generating the event language $L = L(A, x_0)$, so that all states are reachable from x_0, and no two states generate the same language. Furthermore, we assume that A is deterministic. Neither of these assumptions is restrictive since we are concerned with the estimation of elements in L, and we can always choose a minimal deterministic automaton (and initial state) that generates L. Specifically, given L, or equivalently, such an A and x_0, we are interested in whether or not we can reconstruct an event trajectory $s \in L$ when we only observe that part of the string that is in Γ, i.e., we observe $h(s)$ (see Fig. 3.1).

To begin, let h_L be the restriction of the output function h to the domain L so that $h_L^{-1}(r)$ is the set of strings in L that generate the output string $r \in h(L)$. Let us now formally define invertibility with a delay:

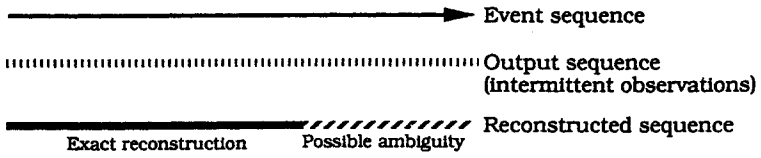

Fig. 3.1. Invertibility with a delay: given the output sequence, the event sequence is reconstructed exactly but with some delay. The ambiguity at the end of the reconstructed string will be resolved using future observations.

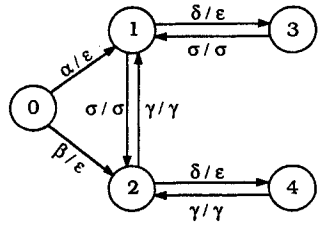

Fig. 3.2. Example for WD-invertibility: state 0 is the initial state.

Definition 3.1. L is *invertible with a delay* (or WD-invertible) if there exists an integer n_d such that, for all $s \in L$, there exists a prefix p of s such that $h_L^{-1}(h(s)) \subset p(\Sigma \cup \{\varepsilon\})^{n_d}$, where $p(\Sigma \cup \{\varepsilon\})^{n_d}$ is used to denote the set: p concatenated with arbitrary strings of length at most n_d.

What this definition states is that for a WD-invertible language, we can, at any time, use knowledge of the output sequence up to that time to reconstruct the full event sequence up to a point at most n_d events into the past (that is, p is uniquely specified).

Consider the system in Fig. 3.2, where state 0 is the initial state. In this case, L is WD-invertible with $n_d = 4$. Note that it is *not* invertible without delay (i.e., $n_d = 0$). For example, if we observe σ^2, the original input string could be $\alpha(\delta\sigma)^2$ or $\alpha(\delta\sigma)^2\delta$ or $\alpha\delta\sigma\sigma$, etc., but we know the first three events with certainty.

To begin our investigation of WD-invertibility, let us define two subsets of $L_f(A, x)$. Specifically, let $L_1(A, x)$ (or L_1 where it is clear from the context) consist of those strings of $L_f(A, x)$ that have only one observable event, and let $L_\sigma(A, x)$ (or L_σ) be the set of strings in L_1 that have $\sigma \in \Gamma$ as the observable event. We first need the following notion:

Definition 3.2. A is termed ambiguous if, for some $x \in X$ and $\gamma \in \Gamma$, there exist distinct $s, t \in L_\gamma(A, x)$ such that $f(x, s) = f(x, t)$.

The importance of this concept for invertibility is given by the following:

Proposition 3.3. *If A is ambiguous, then L is not WD-invertible.*

Proof. Let x, γ, s, and t be as in Definition 3.2. Since A is minimal and deterministic, find an event sequence p so that $f(x_0, p) = x$. Then the distinct sequences ps and pt have indetical outputs and drive x_0 to the same state. Thus, no future behavior will allow us to distinguish between these strings.

As an example, the system in Fig. 3.3 is ambiguous since both $\alpha\delta$ and $\beta\delta$, which produce the same output, take state 0 to state 3. Thus the language generated from 0 is not invertible.

Unambiguity alone is not sufficient for invertibility. For example, the automaton in Fig. 3.4, where 0 is the initial state, is unambiguous, but $L(A, x_0)$ is not invertible either. More specifically, event trajectories $(\beta\alpha)^*$ and $(\delta\alpha)^*$ both have the output α^*.

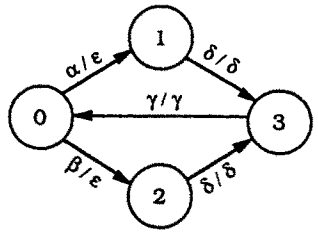

Fig. 3.3. Example for an ambiguous system.

In order to explore necessary and sufficient conditions for invertibility, let us first state the following recursive characterization of invertibility which follows from the fact that $R(A, x_0) = X$:

Proposition 3.4. *L is WD-invertible if and only if $L(A, x)$ is WD-invertible for each $x \in X$.*

What this result suggests is the following: Suppose that we have perfect knowledge of some state x in the past, and that we have an algorithm for reconstructing the event trajectory s that corresponds to the next observed event γ from x, i.e., $s \in L(A, x)$ and $h(s) = \gamma$. Then, thanks to determinism, we also have perfect knowledge of the state that s takes x to. Since we also have perfect knowledge of the initial state, we can reconstruct the entire event trajectory by applying this algorithm iteratively. We use such an approach below to present necessary and sufficient conditions for WD-invertibility which can be tested in polynomial time.

Consider an unambiguous A, any $x \in X$, and any $\sigma \in \Gamma$. Then distinct elements of $L_\sigma(A, x)$ correspond to distinct elements of

$$Q_{x,\sigma} = f(x, L_\sigma(A, x)). \tag{3.1}$$

If L is WD-invertible, then at some point we will be able to reconstruct the entire event sequence and hence the entire state trajectory through the transition into $Q_{x,\sigma}$ caused by the last observable event σ. That is, with the help of future event observations, we are able to distinguish between the elements of $Q_{x,\sigma}$. For example, in Fig. 3.2, A is unambiguous,

$$Q_{0,\sigma} = Q_{0,\gamma} = Q_{1,\sigma} = Q_{2,\gamma} = \{1, 2\},$$

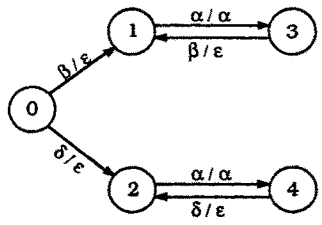

Fig. 3.4. Example for an unambiguous but not invertible system: state 0 is the initial state.

and (1, 2) is not an indistinguishable pair since state 1 produces only σ as output and state 2 produces only γ. Thus, we can distinguish between states 1 and 2 using the next observed event. Consequently, we have the following:

Proposition 3.5. *L is WD-invertible if and only if A is not ambiguous, and, for all $x \in Y \cup x_0$, $\sigma \in \Gamma$, and distinct $y, z \in Q_{x,\sigma}$, $(y, z) \notin I_M$. Furthermore, if L is WD-invertible, then $n_d \leq nq^2$.*

Proof. (necessity) From Proposition 3.3, we only need to check the second condition. Assume the contrary. Then, for some $x \in Y$ and $\sigma \in \Gamma$, there exist distinct s, $t \in L_\sigma$ such that $(f(x, s), f(x, t)) \in I_M$. Thus, using the future outputs, we cannot distinguish between s and t. Therefore, L cannot be WD-invertible and we establish a contradiction.

(sufficiency) We prove this inductively, and the proof also serves as a construction for an inverter. Clearly, in the beginning, we know that the system is in state x_0. Suppose that at some point in time we have inverted the output sequence up through some point and therefore know that the system was in some particular state x a finite number of transitions into the past. Suppose that γ is the first observable event after the system is in x. Since all pairs in $Q_{x,\gamma}$ are distinguishable, we can, using at most q^2 future outputs (corresponding to nq^2 events since any chain of unobservable events can have at most n events), determine which state $y \in Q_{x,\gamma}$ the system was in. Given x, y, γ and since A is unambiguous, we can exactly reconstruct that part of the input string which takes x to y and produces γ as the output. Therefore, the entire input sequence which takes x_0 to y can be reconstructed exactly, and by induction we have proven invertibility.

To summarize, our WD-inverter does the following:

- Starts with the initial state x_0.
- Given the next output γ, uses future outputs (at most q^2 of them will be necessary and this corresponds to at most nq^2 events) to distinguish between the states in $Q_{x_0,\sigma}$ (let y be the new state).
- Reconstructs the part of the state trajectory between x_0 and y.
- Repeats the process using the state y as the initial state and the output after y.

We conclude this section by providing a polynomial-time test for WD-invertibility. First, to construct a test for ambiguity, note that we only need to consider the states in Y, in addition to the initial state x_0, since all other states can be reached by states in $Y \cup \{x_0\}$ using only unobservable events. Furthermore, if there are no transitions defined to x_0, then $x_0 \in Y$, and otherwise x_0 can be reached by some state in Y using only unobservable events. Therefore, we only need to consider the states in Y. Let us pick some $x \in Y$ and consider the set of states $X_x = R(A|\overline{\Gamma}, x)$ which includes x itself. Recall that by assumption, there can be no loops that consist of only unobservable events. As the next result shows, ambiguity may then arise in three forms:

1. Two unobservable events define the same transition from some y to some z both in X_x.

2. For some distinct y and z in X_x there is an observable event defined at both y and z that takes both y and z to the same state (this is the case in Fig. 3.3).
3. For some distinct y and z in X_x, there are unobservable events defined from y and z that take y and z to the same state.

Specifically, we have the following:

Proposition 3.6. *A is ambiguous if and only if there exists some $x \in Y$, such that any of the following conditions is satisfied:*

1. *There exist some $y \in X_x$ and $\sigma_1, \sigma_2 \in \mathrm{d}(y) \cap \bar{\Gamma}$ such that $f(y, \sigma_1) = f(y, \sigma_2)$.*
2. *There exist distinct $y, z \in X_x$ and some $\gamma \in d(y) \cap d(z) \cap \Gamma$ such that $f(y, \gamma) = f(z, \gamma)$.*
3. *There exist distinct $y, z \in X_x$, some $\sigma_1 \in d(y) \cap \bar{\Gamma}$, and $\sigma_2 \in d(z) \cap \bar{\Gamma}$ such that $f(y, \sigma_1) = f(z, \sigma_2)$.*

Proof. (sufficiency) Obvious.

(necessity) Ambiguity implies that there exist some $w \in X$, $\gamma \in \Gamma$, and distinct r, $q \in L_\gamma(A, w)$ such that $f(w, r) = f(w, q)$. Since w can be reached by some $x \in Y$ using only unobservable events (i.e., $w \in X_x$), there exist distinct s, $t \in L_\gamma(A, x)$ such that $f(x, s) = f(x, t)$. Let $s = s'\gamma$ and $t = t'\gamma$. If $f(x, s') \neq f(x, t')$, then the second condition is satisfied. Suppose that $f(x, s') = f(x, t')$. Furthermore, suppose that there exist prefixes s'' of s' and t'' of t' such that $f(x, s'') \neq f(x, t'')$ but $f(x, s''\sigma_1) = f(x, t''\sigma_2)$ where σ_1 (resp. σ_2) is the next event in s' after s'' (resp. the next event in t' after t''). Then the third condition must be satisfied. Finally, if we *cannot* find such s'' and t'', then the state trajectories corresponding to s and t must be the same. Since s and t are distinct they must differ in at least one event, and thus there must exist some state y in the state trajectory so that the first condition is satisfied.

In order to simplify this test further let us pick some $x \in Y$. We first consider the second condition of the above proposition. For each $y \in f(X_x, \Gamma)$ and $\gamma \in d(X_x)$ let us define the following set:

$$F_{y,\gamma}^{-1} = \{z \in X_x | \gamma \in d(z) \text{ and } f(z, \gamma) = y\}. \tag{3.2}$$

Note that this set can be empty for some y and γ. It is obvious that the second condition is satisfied if and only if there exists some y and γ such that $|F_{y,\gamma}^{-1}| \geq 2$. We next consider a combined test for the first and third conditions. In this case, for each $y \in X_x$ let us define the following set:

$$D_y^{-1} = \{\sigma \in \bar{\Gamma} | \exists z \in X_x \text{ such that } \sigma \in d(z) \text{ and } f(z, \sigma) = y\}. \tag{3.3}$$

It is straightforward to show that either the first or the third condition is satisfied if and only if $|D_y^{-1}| \geq 2$ for some y. We thus have the following result:

Corollary 3.7. *A is unambiguous if and only if, for all $x \in Y$,*

(1) *for all $y \in f(X_x, \Gamma)$ and $\gamma \in d(X_x)$, $|F_{y,\gamma}^{-1}| \leq 1$, and*
(2) *for all $y \in X_x$, $|D_y^{-1}| \leq 1$.*

Let $\bar{n}_x$ be the maximum of $|R(A|\bar{\Gamma}, x)|$ over all $x \in Y$. Then, testing for ambiguity takes $O(\bar{n}_x q)$ time.

The following result shows that a necessary and sufficient test for the second condition of Proposition 3.5 is to test if (x_0, x_0) can only reach indistinguishable pairs in O_P:

Proposition 3.8. *Given A, we have that $(y, z) \notin I_M$ for all $x \in Y$, $\sigma \in \Gamma$ and distinct y, $z \in Q_{x,\sigma}$ if and only if the range of (x_0, x_0) in O_P is contained in $\{(x, y) \notin I_M\}$.*

Proof. Straightforward by contraposition.

The condition of the above proposition can be tested in $O(q^2)$ time. Therefore, WD-invertibility can be tested in $O(\bar{n}_x q + q^2)$ time.

3.2. Invertibility with Unknown Initial State

Let us now consider a related notion of invertibility which will be of particular importance when we consider inversion in the presence of observation errors. In particular, suppose that we do not have any information concerning the initial state. In this case, in general, we will not be able to reconstruct the entire event string because of some unresolvable ambiguity at the start. However, it may be possible for us to perform error-free reconstruction after the initial period of uncertainty. In this section we investigate this property.

Let $h_{L(A)}$ be the restriction of the output function h to the domain $L(A)$ so that $h^{-1}_{L(A)}(r)$ is the set of strings in $L(A)$ that generate the output string $r \in h(L(A))$:

Definition 3.9. A system A is *invertible with a delay with unknown initial state* (WDX-invertible) if there exists an integer n_d such that, for all $x \in X$, $s \in L(A, x)$, there exist p, q, r such that $s = pqr$, and $h^{-1}_{L(A)}(h(s)) \subset (\Sigma \cup \{\varepsilon\})^{n_d} q (\Sigma \cup \{\varepsilon\})^{n_d}$.

In order to derive necessary and sufficient conditions for WDX-invertibility, we first enrich the state space of A by including the event trajectory as part of the state. Thanks to our assumption that there are no loops of unobservable events, the number of strings in $L_1(A, x)$ is finite for any x. Thus, we can enrich the state space of A to include such event trajectories, while keeping the state space finite, by using strings in L_1. In particular, let the new state space be as follows:

$$X_s = \{(s, y) | s = \varepsilon \text{ or } s \in L_1(A, x) \text{ for some } x \in Y \text{ such that } y = f(x, s)\}. \quad (3.4)$$

We then define a system $A_s = (G_s, f_s, d_s, 1)$, where $G_s = (X_s, \Gamma, \Gamma)$, and

$$f_s((s, y), \gamma) = \{(t, z) | t \in L_\gamma(A, y) \text{ such that } z = f(y, t)\}, \quad (3.5)$$

$$d_s((s, y)) = \{\gamma \in \Gamma | \exists t \in L_1(A, y) \text{ such that } h(t) = \gamma\}. \quad (3.6)$$

Note that A_s is not necessarily deterministic. For example, A_s corresponding to Fig. 3.2 is illustrated in Fig. 3.5. Note that, for each state x, A_s has a state (ε, x). Since, for example, the string $\alpha\delta\sigma$ takes state 0 to state 1 in Fig. 3.2, the event σ takes state $(\varepsilon, 0)$ to state $(\alpha\delta\sigma, 1)$ in Fig. 3.5. Similarly, the event σ also takes $(\varepsilon, 0)$ to $(\alpha\sigma, 2)$.

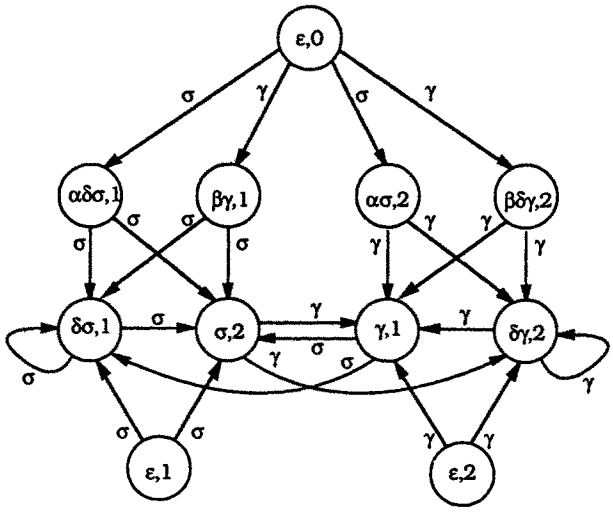

Fig. 3.5. A_s for Fig. 3.2.

WDX-invertibility can be characterized more easily using A_s. First, note that reconstructing the event trajectory in A corresponds to reconstructing the state trajectory in A_s, modulo the first component of the states; i.e., if two states in X_s have their first components equal, it is not important, for WDX-invertibility, to distinguish between these two states. However, if two states in X_s differ in their first components, then we need to be able to tell which state the trajectory has passed through in order to reconstruct the event trajectory in A, and, thus, the second components of these two states need to be distinguishable in A. Let $O_s = (H_s, w_s, v_s)$ where $H_s = (Z_s, \Gamma, \Gamma)$ denotes the observer for A_s. We characterize WDX-invertibility based on this observer. In particular, let E_s be those states $\hat{x}$ of O_s, such that, for all pairs of states (s, x), $(t, y) \in \hat{x}$, either $s = t$ or x and y are distinguishable in A, i.e.,

$$E_s = \{\hat{x} \in Z_s | \text{ for all } (s_1, y_1), (s_2, y_2) \in \hat{x}, s_1 = s_2, \text{ or } (y_1, y_2) \notin I_M\}. \quad (3.7)$$

In order to be able to reconstruct the event trajectory of A within a finite number of transitions, we need the observer to be E_s-pre-stable and indeed we would like the observer to *stay* in E_s. In the following result, let W_s be the maximal w_s-invariant subset of E_s in O_s:

Proposition 3.10. *A is WDX-invertible if and only if the observer O_s for A_s is W_s-prestable.*

Proof. (necessity) Straightforward by contraposition.

(sufficiency) We use the observer O_s as a basis for the inversion. Thanks to stability, the trajectory enters W_s in a finite number of transitions (Proposition 3.13 below provides a bound for the number of transitions it takes for the trajectory to enter O_s). The trajectory stays in W_s once it enters W_s and we can then invert the

event trajectory as follows: Let $\hat{x} \in W_s$ and partition $\hat{x}$ as

$$\hat{x} = \{(s_1, y_{11}), (s_1, y_{12}), \ldots\} \cup \{(s_2, y_{21}), (s_2, y_{22}), \ldots\} \cup \cdots.$$

Note that $h(s_1) = h(s_2) = \cdots = \gamma$ where γ is the last observed event before O_s entered $\hat{x}$. In order to invert that portion of the event trajectory corresponding to the observation γ, we need to be able to distinguish between y_{ij} and y_{kl} for $i \neq k$. Since by definition of E_s, $(y_{ij}, y_{kl}) \notin I_M$, for $i \neq l$, A is WDX-invertible.

The WDX-inverter motivated by the above proof can be outlined as follows:

- Trace the output trajectory of A in O_s until the trajectory in O_s enters W_s.
- Let $\hat{x}_0$ be the state that the trajectory in O_s enters when it enters W_s for the first time and let γ be the last observed event. Partition $\hat{x}_0$ as

  $$\hat{x}_0 = \{(s_1, y_{11}), (s_1, y_{12}), \ldots\} \cup \{(s_2, y_{21}), (s_2, y_{22}), \ldots\} \cup \cdots.$$

  Let $Y_i = \{y_{i1}, y_{i2}, \ldots\}$ for each i. Thanks to distinguishability, the states in Y_i and Y_j do not share any infinite length output sequences for $i \neq j$. Thus, using *future* observations, distinguish between Y_i or, equivalently, decide which s_i has occurred. Then that s_i is the actual trajectory in A corresponding to the observation γ.
- Repeat the previous step for all the subsequent states of the trajectory in O_s, following $\hat{x}_0$.

To test WDX-invertibility we use the pair automaton $O_{sP} = (Z_{sP}, w_{sP}, v_{sP})$ associated with A_s, in a manner similar to that used in [OW2] for testing observability (see Section 2). In what follows, let

$$E_{sP} = \{((s_1, y_1), (s_2, y_2)) \in P_s | s_1 = s_2 \text{ or } (y_1, y_2) \notin I_M\} \tag{3.8}$$

and let W_{sP} denote the maximal w_{sP}-invariant set in E_{sP}:

Proposition 3.11. *A is WDX-invertible if and only if O_{sP} is W_{sP}-stable.*

Proof. Follows from Propositions 3.10 and 2.7.

Let $\bar{l}_x$ be the maximum of $|L_1(A, x)|$ over all $x \in Y$, then WDX-invertibility can be tested in $O((\bar{l}_x q)^4)$ time since O_{sP} has at most $(\bar{l}_x q)^2$ states.

We now calculate a bound for n_d. Note that n_d is due to the ambiguity at the beginning and at the end of the inverted string. The ambiguity at the end depends on n_i, the minimum number of transitions it takes to distinguish between two states in A, and the ambiguity at the beginning depends on the number of transitions it takes a trajectory in O_s to enter W_s. Specifically, if we let n_w denote the length of the longest trajectory in O_s that starts from the initial state of O_s and ends as soon as the trajectory enters W_s, and if we let n_u denote the length of the longest chain of unobservable events in A,[2] we have the following:

[2] Note that $n_u \leq n - q$.

Proposition 3.12. *If A is WDX-invertible, then $n_d \leq n_u \max(n_i, n_w)$.*

Proof. Straightforward by construction of the WDX-inverter and the fact that A can have at most n_u unobservable events between observable events.

Recall that a bound on n_i, as stated in Proposition 2.8, is q^2, and we show below that the same is also a bound on n_w:

Proposition 3.13. *If A is WDX-invertible, then $n_w \leq q^2$.*

Proof. Assume the contrary, then there exists a path of at least $q^2 + 1$ states (corresponding to q^2 transitions) in O_s:

$$\hat{x}_0, \ldots, \hat{x}_{q^2},$$

such that there exists a path in O_{sP}:

$$p_1 = ((s_{01}, y_{01}), (s_{02}, y_{02})), \ldots, p_{q^2} = ((s_{q^2 1}, y_{q^2 1}), (s_{q^2 2}, y_{q^2 2}))$$

for which $(s_{i1}, y_{i1}), (s_{i2}, y_{i2}) \in \hat{x}_i$ for all i and such that $p_{q^2} \notin W_{sP}$. Then, for some integers j and k, say $j < k$, $\{y_{1j}, y_{2j}\} = \{y_{1k}, y_{2k}\}$. Let $\sigma \in v_{sP}(p_j)$ such that $p_{j+1} \in w_{sP}(p_j, \sigma)$, then $p_{j+1} \in w_{sP}(p_k, \sigma)$. Thus, there exists a cycle

$$p_{j+1}, \ldots, p_k, p_{j+1}$$

in O_{sP} which may reach p_{q^2}. Since $p_{q^2} \notin W_{sP}$ and W_{sP} is w_{sP}-invariant, no state in the above cycle can be in W_{sP} either. Thus, O_{sP} cannot be W_{sP}-stable, and we establish a contradiction. Therefore, $n_w \leq q^2$.

We conclude this section by presenting a result on distinguishability that plays an important role in the development of resilient WDX-invertibility in the next section. Before presenting this result, we need to introduce the following notion: We call a state *recurrent* if it can be reached from another state by an arbitrarily long string. We let A_r denote the recurrent part of A, which we construct as follows: Let D_0 denote the set of "dead" states in A^{-1}, i.e.,

$$D_0 = \{x \in X | d^{-1}(x) = \varnothing\}. \tag{3.9}$$

Now let D_1 be the set of states that can only reach D_0, with at most one transition, in A^{-1}, i.e.,

$$D_1 = D_0 \cup \{x \in X | f^{-1}(x, d^{-1}(x)) \subset D_0\}. \tag{3.10}$$

In general, we let

$$D_{i+1} = D_i \cup \{x \in X | f^{-1}(x, d^{-1}(x)) \subset D_i\} \tag{3.11}$$

and we let n_r be the smallest integer so that $D_{n_r+1} = D_{n_r}$. Then it is not difficult to check that $\bar{D}_{n_r}$ is the set of all recurrent states of A. We define A_r over the set $X_r = \bar{D}_{n_r}$ but with the same dynamics as A, and let A_{sr}, with state space X_{sr}, be the counterpart for A_r of A_s. The following result states a connection between WDX-invertibility and distinguishability in A_r^{-1}:

Proposition 3.14. *If A is WDX-invertible, then for all $p_1 = (s_1, x_1)$, $p_2 = (s_2, x_2) \in X_{sr}$ such that $(p_1, p_2) \notin E_{sP}$ and $h(s_1) = h(s_2)$, and for all $y_1, y_2 \in X_r$ such that $x_1 = f(y_1, s_1)$ and $x_2 = f(y_2, s_2)$, y_1 and y_2 must be distinguishable in A_r^{-1}.*

Proof. Let $\gamma = h(s_1) = h(s_2)$ and let us assume the contrary. Thus, there exists some p_1, p_2, y_1, y_2 that satisfy above conditions and such that y_1 and y_2 are indistinguishable in A_r^{-1}. Let O_{rP}^{-1} denote the pair automaton corresponding to A_r^{-1}. Since y_1 and y_2 are indistinguishable in A_r^{-1}, then, using similar reasoning to that in Proposition 2.8, we conclude that there exists a cycle

$$z_1, \ldots, z_k, z_1$$

in O_{rP}^{-1} that is reachable from (y_1, y_2). However, then the same cycle also exists (in reverse order) in O_P so that it may reach (y_1, y_2) in O_P. This in turn implies that if we represent z_i by (z_{i1}, z_{i2}), then there exists a cycle

$$((t_{11}, z_{11}), (t_{12}, z_{12})), \ldots, ((t_{11}, z_{11}), (t_{12}, z_{12}))$$

in O_{sP}, for some t_{ij}, which may reach $((r_1, y_1), (r_2, y_2))$, for some r_1 and r_2, in O_{sP}. Then this cycle may also reach (p_1, p_2) which is *not* in E_{sP}. Therefore, none of the elements of the above cycle can be in W_{sP} (since it is invariant) and we establish a contradiction since A cannot be WDX-invertible.

Finally, we let n_{ii} denote the minimum number of transitions it takes to distinguish between any two distinguishable states in A_r^{-1} and note that $n_{ii} \leq q^2$.

4. Resilient Inverters

As with the observability problem, we are interested in resilient inverters. Specifically, we wish to construct inverters that invert correctly after a finite number of transitions following an error burst. In addition, in contrast to the situation for resilient observability, we allow the reconstructed string to be incorrect in a bounded window *before and after* the burst. We represent this notion of invertibility pictorially by Fig. 4.1.

In contrast to the situation for observability and resilient observability, invertibility is not sufficient for resilient invertibility. For example, consider Fig. 4.2 where 0 is the initial state. There is no resilient inverter for this system since an erroneous insertion of β, say after the kth output, may lead to an infinite number of errors. If

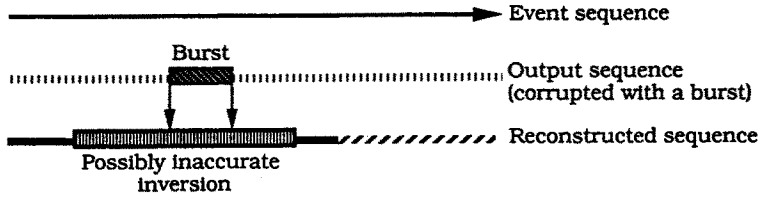

Fig. 4.1. Resilient WD-invertibility: inversion can only be wrong for a finite number of transitions before and after the burst.

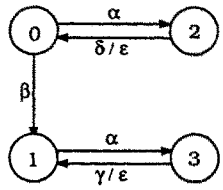

Fig. 4.2. Example for nonresilient inversion: state 0 is the initial state.

in fact β never occurs, the actual string would be $(\alpha\delta)^*$ whereas the inverted string is $(\alpha\delta)^*\beta(\alpha\gamma)^*$.

In order to define what we mean by resilient invertibility, we also need to define a notion to represent the discrepancy between two strings. Since the actual point that the burst ends is important for our definition of resiliency, we compare two strings from their beginning and we represent their discrepancy by how much they differ at the end. In particular, we say that the discrepancy between two strings s and t is of length at most i, denoted by

$$\xi(s, t) \leq i, \tag{4.1}$$

if there exists a prefix, p, of s and t such that $|s/p| \leq i$ and $|t/p| \leq i$.

As in the case of resilient observability, we allow the burst to be any string in Γ. Then the corrupted output is *not* necessarily an output string that can be generated by a state in X, and thus h_L^{-1} is undefined for this erroneous string. Therefore, we must define an inverter so that its response is defined for all such strings. We also require that the behavior of this inverter is equivalent to that of h_L^{-1} for uncorrupted strings:

Definition 4.1. A *WD-inverter* is a map $\mathbf{I}: \Gamma^* \to \Sigma^*$ so that for those strings that *are* in $L(A)$, $\mathbf{I}$ yields the same behavior as $h_{L(A)}^{-1}$, i.e., for all $s \in L(A)$, we require that $\mathbf{I}(h(s)) \subset h_{L(A)}^{-1}(h(s))$. Similarly, a *WDX-inverter* is a map $\mathbf{I}: \Gamma^* \to \Sigma^*$ so that for those strings that *are* in $L(A)$, $\mathbf{I}$ yields the same behavior as $h_{L(A)}^{-1}$, i.e., for all $s \in L(A)$, we require that $\mathbf{I}(h(s)) \subset h_{L(A)}^{-1}(h(s))$.

Note that we require that the behavior of $\mathbf{I}$ is a subset of the inversion (as opposed to equivalent to the inversion), since if L is invertible, that portion of the inverted string that is of interest to us is unique and the inverter output for the ambiguous portion is not relevant. We term an inverter resilient if, for all corrupted output strings, the inverter output, compared with the actual event trajectory, is only incorrect within a bounded window around the burst. In the following formal definition of resiliency, L_f denotes the set of strings in L that have an observable event as the last event:

Definition 4.2. A WD-inverter $\mathbf{I}$ is a *resilient WD-inverter* if there exists an integer n_b such that for all strings s in L, for all possible output strings t which can be generated by corrupting $h(s)$ with a finite length burst, i.e.,

- for all integers i,
- for all $t \in \Gamma^*$ such that $\xi(h(s), t) \leq i$ (let i' be the length of that part of s whose

output is corrupted by the burst, i.e., let $s' \in L_f$ be the prefix of s such that $|h(s/s')| = i$ then $i' = |s/s'|$),

and for all possible completions r of s, i.e., for all $r \in L$ such that s is a prefix of r, there exists

- a prefix p_1 of s which is free of inversion errors in spite of the burst, i.e., p_1 is that prefix of s for which $|s/p_1| \leq n_b + i'$,
- a prefix p_2 of r so that the inversion error and the ambiguity can be confined to p_2/p_1, i.e., $|p_2/p_1| \leq 2n_b + i'$, and
- a prefix p_3 of r so that the inversion delay, as before, can be confined to r/p_3, i.e., $|r/p_3| \leq n_b$,

such that

$$\mathbf{I}(th(r/s)) \subset p_1(\Sigma \cup \varepsilon)^{2n_b+i'}(p_3/p_2)(\Sigma \cup \{\varepsilon\})^{n_b}.$$

(Note that the observed string is $th(r/s)$, whereas r is what has actually occurred in the system.) L is *resiliently WD-invertible* if L is WD-invertible and a resilient WD-inverter exists.

Similarly,

Definition 4.3. A WDX-inverter $\mathbf{I}$ is a *resilient WDX-inverter* if there exists an integer n_b such that for all strings s in $L(A)$, for all possible output strings t which can be generated by corrupting $h(s)$ with a finite length burst, i.e.,

- for all integers i,
- for all $t \in \Gamma^*$ such that $\xi(h(s), t) \leq i$ (let $s' \in L_f(A)$ be the prefix of s such that $|h(s/s')| = i$ and let $i' = |s/s'|$),

and for all possible completions r of s, i.e., for all $r \in L(A)$ such that s is a prefix or r, there exists

- a prefix p_0 of p_1, representing the ambiguity in the beginning, such that $p_1/p_0 \leq n_b$,
- a prefix p_1 of s which is free of inversion errors in spite of the burst, i.e., p_1 is that prefix of s for which $s/p_1 \leq n_b + i'$,
- a prefix p_2 of r so that the inversion error and the ambiguity can be confined to p_2/p_1, i.e., $|p_2/p_1| \leq 2n_b + i'$, and
- a prefix p_3 of r so that the inversion delay, as before, can be confined to r/p_3, i.e., $|r/p_3| \leq n_b$,

such that

$$\mathbf{I}(th(r/s)) \subset (\Sigma \cup \varepsilon)^{n_b}(p_1/p_0)(\Sigma \cup \varepsilon)^{2n_b+i'}(p_3/p_2)(\Sigma \cup \{\varepsilon\})^{n_b}.$$

A is *resiliently WDX-invertible* if A is WDX-invertible and a resilient WDX-inverter exists.

As the following result shows, a sufficient condition for resilient WD-invertibility is WD-observability together with WD-invertibility, and we justify this as follows: If A is WD-observable, then, a finite number of transitions after a burst, the observer estimate is guaranteed to include the actual state of the system. Moreover, using

future outputs, we can exactly determine the actual state of the system. Since WD-invertibility implies that the language generated by any state is WD-invertible (see Proposition 3.4), we can invert correctly after a finite number of transitions:

Proposition 4.4. *If L is WD-invertible and A is WD-observable, then L is resiliently WD-invertible.*

Proof. Straightforward by the above reasoning using the fact that WD-observers are resilient, as noted in Section 2, and using Proposition 3.4.

The converse of Proposition 4.4 is not necessarily true. For example, consider the system illustrated in Fig. 4.3, where all events are observable and 0 is the initial state. This system is clearly resiliently invertible since all events are observable. However, it is not WD-observable since if only α occurs, we can never distinguish between states 1 and 2.

To find necessary and sufficient conditions for resilient WD-invertibility, we first address the problem of resilient WDX-invertibility, since, as we have noted in Section 2, observation errors may lead to a complete loss of current state information. It turns out that WDX-invertibility is necessary and sufficient for resilient WDX-invertibility. Since necessity is clear, we concentrate on showing sufficiency. In doing so, we assume WDX-invertibility and construct a resilient inverter. However, the construction for a resilient WDX-inverter is fairly complicated in this case. We start by using the WDX-inverter. If a burst never occurs, the inversion proceeds as before. In case of a burst, if the corrupted output trajectory is a feasible one, i.e., if it is in $h(L(A))$, the burst will never be detected. Later in this section we show that our WDX-inverter works correctly in this case. However, if the corrupted string is

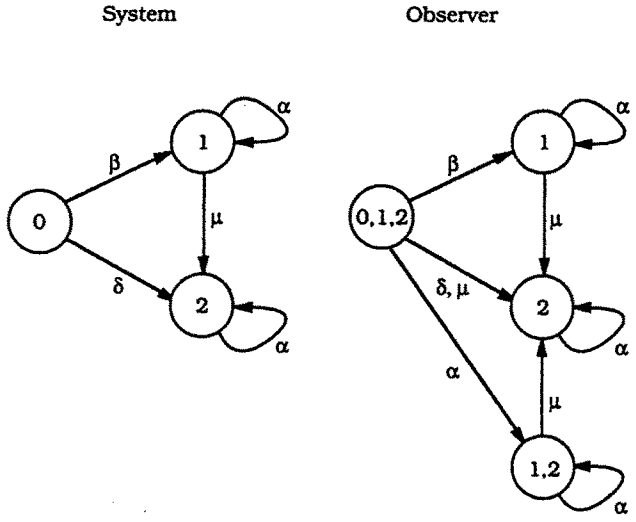

Fig. 4.3. Counterexample to necessity of WD-observability for resilient invertibility: all events are observable, 0 is the initial state.

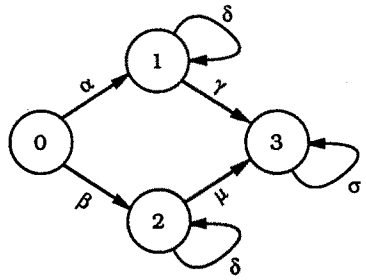

Fig. 4.4. Example for resilient invertibility.

not a feasible one, then we need to modify our WDX-inverter so that we can incorporate this possibility; the observer O_s used as a basis for the WDX-inverter must also be used somewhat differently.

Let us examine the effect of an error burst on the inverter. When a burst occurs, it is possible that this burst is never detected. For example, consider the system in Fig. 4.4, where all events are assumed to be observable so that inversion is trivial, i.e., the inverter is just the identity map. Suppose that $\alpha\delta^*$ occurs but $\beta\delta^*$ is observed. This observation error is never detected, but inversion is resilient because there are only a finite number of inversion errors (just one in this case) and they occur in close vicinity of the error burst (in this case coincident with the burst). This example illustrates the general situation for a WDX-invertible system when an error burst occurs that is not detectable. In this case, by the definition of WDX-invertibility, the WDX-inverter itself will provide correct inversion after a finite period following the burst (note that this is because the WDX-inverter is capable of performing the inversion even though the system state is unknown at the end of the burst).

However, as in the preceding example and as illustrated in Fig. 4.5, the point t_d at which we detect the inconsistency may occur after an arbitrarily large time following the measurement error burst interval $[t_a, t_b]$. What we would like to do, however, is to correct inversion errors (once we have detected a discrepancy) to obtain a correctly inverted string except for a region around the actual error burst;

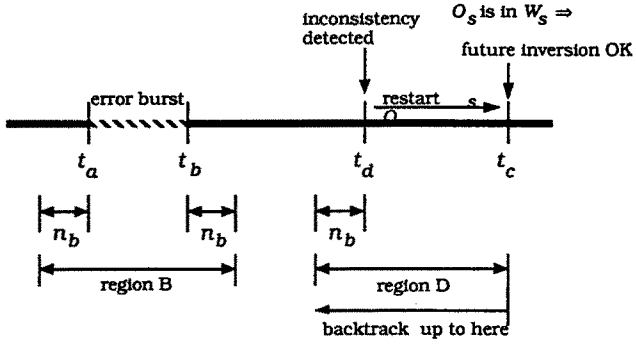

Fig. 4.5. Illustration of crucial points in time in resilient inversion: particular choice of time indices illustrates the case when the inconsistency is detected arbitrarily far from the burst.

i.e., we would like to find a number n_b and a *backtracking* procedure that guarantees that any inversion errors are confined to region B. Note that while the times t_d and t_c are known, t_a and t_b are *not*, so the backtracking procedure must have a stopping rule that works without this knowledge. Our backtracking procedure starts from the point t_c, at which we are sure that processing in the future will be correct, and works backward in time. We will see that we need to backtrack at most n_b steps before t_d in order to guarantee that inversion errors are contained in region B.

Before we present our main result, let us provide a more complete picture of the possible situations that must be analyzed:

1. It is possible that an inconsistent transition *never* occurs, so that our WDX-inverter continues without change. In this case region D in Fig. 4.5 does not exist and we must show that any inversion errors are necessarily confined to B.
2. The point t_d lies sufficiently far from $[t_a, t_b]$ so that regions B and D do not overlap. In this case what we must show is that the original inversion errors are confined to regions B and D and that our backtracking, which only covers region D, corrects the errors in D leaving any remaining errors confined to region B.
3. The point t_d lies outside $[t_a, t_b]$ but regions B and D overlap. In this case it is obvious that the original inversion errors are confined to regions B and D (since their union covers everything from t_a through t_c). Note also that the backtracking step will cover the part of region B overlapping D. Thus, because of the original measurement errors, it is possible that we will encounter an inconsistency during the backtracking step. In this case we can be sure that we have backtracked into region B and can stop backtracking (so that our full backtracking step proceeds until n_b events before t_d or until an inconsistency is detected, whichever comes first). What we must show in this case is that backtracking corrects errors in that part of D lying outside B.[3]
4. The point t_d lies inside $[t_a, t_b]$. The situation in this case is a bit more complex. Specifically, since there are still measurement errors following t_d, when we reset O_s at t_d and run it forward, we may observe *another* inconsistency before we reach t_c. In this case we know that the error burst extends beyond our original t_d, and we simply reset O_s again using this latest point of inconsistency as our new t_d. This continues until we have a reset that leads to a successful achievement of the point t_c. In this case the associated t_d may lie outside $[t_a, t_b]$, in which case we are in one of the first three cases. The only remaining situation is one in which t_d is still within $[t_a, t_b]$. Note that in this case, since there are measurement errors following t_d, the correction procedure, which consists of restarting O_s and propagating it forward to t_c followed by backtracking, may have errors in it. However, what we will see in this case is that region D is completely contained in region B so that inversion errors, even after our correction step, will be confined to B. Again, we may detect a subsequent

[3] Note that in some cases if n_b is sufficiently large and the error burst is short, we may backtrack to a point before t_a (but never outside B since $t_a \leq t_d$) so that the backtracking step may actually introduce new inversion errors, which, however, will be confined to B.

inconsistency after this point, and this will then correspond once again to one of these four cases.

Proposition 4.5. *A is resiliently WDX-invertible if and only if A is WDX-invertible. Furthermore, if A is resiliently WDX-invertible, then*

$$n_b \leq \max[n_u \max(n_w, n_i, n_{ii}+1), n_r].$$

Proof. (necessity) Obvious. As shown in the inverter construction below, n_b depends on

(a) the number of transitions it takes a trajectory from the initial state X_s of O_s to enter W_s,
(b) the number of transitions it takes to distinguish between two distinguishable states in A,
(c) one plus the number of transitions it takes to distinguish between two distinguishable states in A_r^{-1}, and
(d) the minimum number of transitions it takes any state in A to reach a recurrent state.

Thus, n_b is bounded by $\max[n_u \max(n_w, n_i, n_{ii}+1), n_r]$.

(sufficiency) This is a constructive proof, producing a resilient WDX-inverter. As described above, the resilient WDX-inverter is the same as the WDX-inverter if no inconsistencies are observed. When an inconsistency is observed, we restart O_s and wait until it enters W_s. If another inconsistency is observed after restarting O_s, we restart O_s again. After O_s enters W_s, we proceed in two directions:

(1) We use the WDX-inverter for future inversion.
(2) We backtrack until an inconsistency is detected or until we have reached a point n_b steps before t_d.

To prove the result we must pick n_b, specify the backtracking procedure, and analyze the four cases presented previously. We let $n_b = \max[n_u \max(n_w, n_i, n_{ii}+1), n_r]$ and we begin with case 1.

For case 1 suppose that no inconsistency is observed. From Proposition 3.12 we know that our WDX-inverter has made no errors any earlier than $n_u n_i$ events before t_a. What we will show is that it makes no errors any later than $n_u n_w$ events—or equivalently n_w observable events—after t_b. Suppose that, at time t_b, A_s is in some state x and O_s is in some state $\hat{x}$. If $x \in \hat{x}$, then the inversion will be correct for all the transitions following the burst. Suppose that $x \notin \hat{x}$. Consider the states to which A_s and O_s move after n_w observed transitions, and call these y and $\hat{y}$. Then since O_s must enter W_s in at most n_w transitions (see the proof of Proposition 3.13), there is some $\hat{z} \in W_s$ so that $\hat{y} \cup \{y\} \subset \hat{z}$ (se can take $\hat{z}$ to be the state reached in these n_w observed transitions starting from some initial state in O_s that contains $\hat{x} \cup \{x\}$—e.g., we can take the initial state to be *all* of X_s). If $y \in \hat{y}$, then subsequent inversions will be correct. Otherwise, if we let $y = (s, w)$, then for all $(p, v) \in \hat{y}$ either $p = s$ or w and v are distinguishable. Also, since we observe no inconsistencies, the string we observe after this point is shared by $L(A_s, y)$ and $L(O_s, \hat{y})$. Let r be the next n_i

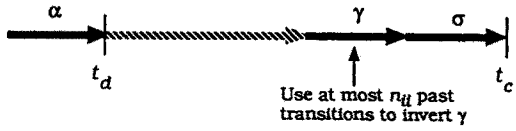

Fig. 4.6. Proof of resilient WD-invertibility (for case 2(b)): ordering of α, σ, and γ.

observations. Recall that our WDX-inverter, using r, eliminates the states in $\hat{y}$ that cannot generate r. Let $\hat{y}'$ be the set of states in $\hat{y}$ which can generate r. By WDX-invertibility, for all $(p_0, v_0), (p_1, v_1) \in \hat{y}'$, $p_0 = p_1$. Also, thanks to Proposition 2.8, $p = s$ for all $(p, v) \in \hat{y}'$. Therefore, the inverter will produce the correct inversion at this point, Since these same conditions hold for all subsequent states in the trajectories of A_s and O_s, the inversion proceeds correctly.

Consider next the three other cases. What we first wish to show is that the inversion errors, before backtracking, are confined to regions B and D. In cases 3 and 4 this is obvious, so we focus on case 2. At an intermediate point between the regions B and D, let A_s be in state y and let O_s be in state $\hat{y}$. Let r be the next n_i observations. Note that r ends before t_d and thus r can be generated by both y and $\hat{y}$. Let $y = (s, w)$. As in the preceding proof case 1, we use Proposition 2.8 and conclude that the inversion at this point must be correct. Since this reasoning holds for *any* point between the regions B and D, the inversion errors must be confined to these regions.

We now specify the precise procedure to be used when an inconsistency is detected. Let α be the inconsistent transition at time t_d (see Fig. 4.6). Since α is not defined at the current state of O_s, we start O_s from the state X_s after α occurs, and let it evolve. Let t_c be the point at which the state of O_s first enters W_s following the reset at t_d. From Proposition 3.13 we know that there are at most $n_u n_w$ transitions between t_d and t_c. We can then conclude that in case 4, i.e., when $t_a < t_d < t_b$, region D is completely contained in region B (see Fig. 4.5). Then we need only consider cases 2 and 3 in which $t_d > t_a$.

Let $\hat{x} \in W_s$ be the state of O_s at t_c and let σ be the event that caused this transition into W_s (see Fig. 4.6). Then, for all $(s_1, y_1), (s_2, y_2) \in \hat{x}$, either $s_1 = s_2$ or y_1 and y_2 are distinguishable (see Fig. 4.7 where the top ellipse denotes $\hat{x}$), and, in addition, $h(s_1) = h(s_2) = \sigma$. Thus, by distinguishability, using a finite number of future outputs (after σ), we can improve the estimate at t_c from $\hat{x}$, to, say $\hat{x}'$, such that, for all $(s_1, y_1), (s_2, y_2) \in \hat{x}'$, $s_1 = s_2$, and, therefore, we can reconstruct the string corresponding to the output σ—i.e., the segment of the event trajectory in A ending with σ and preceded by unobservable events only (see Fig. 4.7 where the second ellipse denotes $\hat{x}'$). We now begin the backtracking step: Let γ denote the last observable transition prior to σ (see Fig. 4.6), and let us construct the set of states $\hat{z}$ in A_s that may reach a state in $\hat{x}'$ with one observable transition (σ) and such that the output corresponding to the first component of the state is γ, i.e., $\hat{z}$ consists of all $(r, x) \in X_s$ such that $h(r) = \gamma$ and $f_s((r, x), \sigma) \cap \hat{x}' \neq \varnothing$ (see Fig. 4.7). Consider then any (r_1, x_1), $(r_2, x_2) \in \hat{z}$. If $(x_1, x_2) \notin I_M$ are distinguishable, we can use the n_i subsequent observations (γ, σ, and all but the last observation used to reconstruct σ) to decide between them. In this way we can reduce the size of $\hat{z}$ so that the remaining elements have $(x_1, x_2) \in I_M$.

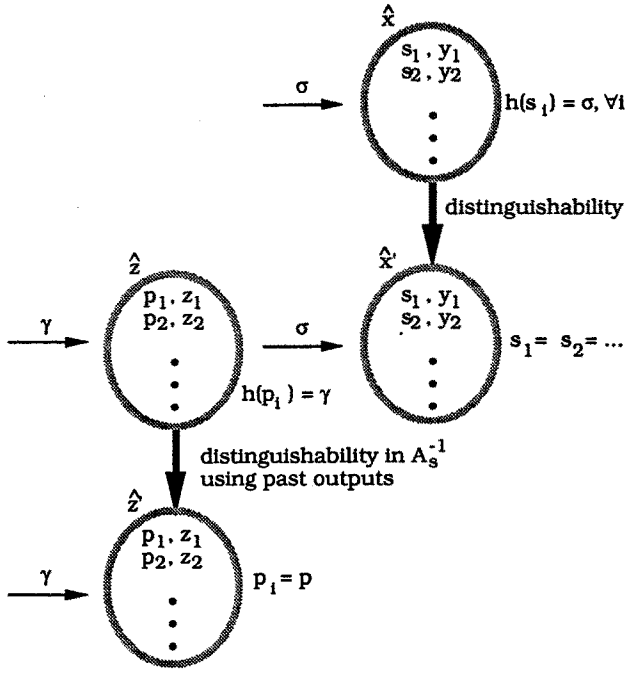

Fig. 4.7. Proof of resilient WD-invertibility (for case 2(b)): backtracking step.

Next, without loss of generality, we can eliminate all states in $\hat{z}$ that are *not* recurrent, for the following reason: If A_s is in a nonrecurrent state, then at most n transitions could have occurred in the system, and due to that part of region B prior to the measurement errors, if we make any inversion errors because of eliminating nonrecurrent states, then these errors will be in region B. So we pick a pair of recurrent states $p_1 = (s_1, x_1), p_2 = (s_2, x_2) \in \hat{z}$ such that $s_1 \neq s_2$. Since also $h(s_1) = h(s_2)$, and $(x_1, x_2) \in I_M$, p_1 and p_2 satisfy the conditions of Proposition 3.14. Thus, by this proposition, for all $y_1, y_2 \in X_r$ such that $x_1 = f(y_1, s_1)$ and $x_2 = f(y_2, s_2)$, y_1 and y_2 must be distinguishable in A_r^{-1}. Then we can also distinguish between x_1 and x_2 using n_{ii} observations prior to γ, or $n_{ii} + 1$ observations including γ. Thus, by considering $n_{ii} + 1$ outputs prior to $\hat{z}$, we can distinguish between different strings s_i in $\hat{z}$. Thus, using these prior outputs, we can construct a new set $\hat{z}'$ such that, for all $(s_1, x_1), (s_2, x_2) \in \hat{z}', s_1 = s_2$ (see Fig. 4.7). Therefore, we can reconstruct the string corresponding to the output γ and can then repeat the process going backward one observable event at a time. This backward reconstruction continues up to n_b transitions before t_d or until the observer estimate going backward encounters an inconsistency (i.e., the $\hat{z}$ constructed above is empty), whichever comes first.

Finally, note that in order to invert correctly an observation γ in the backward inversion process that we have described, we need at most n_{ii} observations prior to γ. Since, by the definition of region B, the last n_b/n_u observations in region B are free of any measurement errors and $n_b/n_u \geq n_{ii} + 1$, the inversion for all points in region D that are outside region B will be correct, proving our assertion for cases 2 and 3.

The following result immediately follows from this proposition and the fact that measurement errors may lead to the observation of inconsistent transitions:

Proposition 4.6. *Given a WD-invertible L, L is resiliently WD-invertible if and only if A is WDX-invertible. Furthermore, if L is resiliently WD-invertible, then* $n_b \leq \max[n_u \max(n_w, n_i, n_{ii} + 1), n_r]$.

5. Conclusions

In this paper we have introduced notions of invertibility and resiliency for discrete-event systems described by finite-state automata, and we have developed algorithms with polynomial-time complexity to test for invertibility, resiliency, and to construct resilient inverters. We have shown that the central element in these notions is the notion of stability that we considered in [OWA] and the notion of observability that we considered in [OW2].

The stability concepts that we introduced in [OWA] can be thought of as notions of error recovery or resiliency in that the system always returns to "good" states. In this paper we have carried this notion farther by presenting an approach for reconstructing system behavior resiliently in spite of observation errors. Motivated by problems such as schedule-following in a flexible manufacturing system, we can also formulate regulation or tracking problems for DEDS in which a feedback system is sought so that the DEDS produces a particular desired sequence of output events. This tracking problem can also be thought of as a dual of the problem of invertibility that we have addressed. Analyses addressing these and related problems will be the subjects of subsequent papers.

References

[CDFV] R. Cieslak, C. Desclaux, A. Fawaz, and P. Varaiya, Supervisory control of discrete-event processes with partial observations, *IEEE Trans. Automat. Control*, **33** (1988), 249–260.

[LW] F. Lin and W. M. Wonham, Decentralized supervisory control of disrete-event systems, *Inform. Sci.*, **44** (1988), 199–224.

[OW1] J. S. Ostroff and W. M. Wonham, A temporal logic approach to real time control, *Proceedings of the 24th IEEE Conference on Decision and Control*, Ft. Lauderdale, FL, 1985, pp. 656–657.

[OW2] C. M. Özveren and A. S. Willsky, Observability of discrete event dynamic systems, *IEEE Trans. Automat. Control*, **35** (1990), 797–806.

[OW3] C. M. Özveren and A. S. Willsky, Output stabilizability of discrete event dynamic systems, *IEEE Trans. Automat. Control*, **36** (1991), 925–935.

[OWA] C. M. Özveren, A. S. Willsky, and P. J. Antsaklis, Stability and stabilizability of discrete-event dynamic systems, *J. Assoc. Comput. Mach.*, **38** (1991), 730–752.

[PW] W. W. Peterson and E. J. Weldon, Jr., *Error-Correcting Codes*, MIT Press, Cambridge, MA, 1972.

[RW1] P. J. Ramadge and W. M. Wonham, Modular feedback logic for discrete-event systems, *SIAM J. Control Optim.*, **25** (1987), 1202–1218.

[RW2] P. J. Ramadge and W. M. Wonham, Supervisory control of a class of discrete-event processes, *SIAM J. Control Optim.*, **25** (1987), 206–230.

[VW] A. F. Vaz and W. M. Wonham, On supervisor reduction in discrete-event systems, *Internat. J. Control*, **44** (1986), 475–491.