Department of Electrical and Computer Engineering

State Estimation in Logical and Stochastic Discrete Event Systems

Christoforos Keroglou

A Dissertation

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

at the University of Cyprus

June, 2016

VALIDATION PAGE

Christoforos Keroglou

State Estimation in Logical and Stochastic Discrete Event Systems

The present Doctorate Dissertation was submitted in partial fulfillment of the requirements for the Degree of Doctor of Philosophy in the Department of Electrical and Computer Engineering, and was approved on June 24, 2016 by the members of the Examination Committee.

Committee Chair _____
Dr. Charalambos Charalambous

Research Supervisor _____
Dr. Christoforos N. Hadjicostis

Committee Member _____
Dr. Christos Panayiotou

Committee Member _____
Dr. Anna Philippou

Committee Member _____
Dr. Demosthenis Teneketzis

Declaration of Authorship

The present doctoral dissertation was submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy of the University of Cyprus. It is a product of original work of my own, unless otherwise mentioned through references, notes, or any other statements.

……

Christoforos Keroglou
Department of Electrical and Computer Engineering
University of Cyprus
June, 2016

Περίληψη

Η παρούσα διδακτορική διατριβή μελετά τη διαδικασία της εκτίμησης κατάστασης (state estimation) σε πολύπλοκα διασυνδεδεμένα συστήματα, τα οποία μοντελοποιούνται σαν discrete event systems ή stochastic discrete event systems. Ένα από τα σημαντικά κίνητρα για να ασχοληθούμε με τα προβλήματα που σχετίζονται με την εκτίμηση κατάστασης, είναι ο σημαντικός ρόλος που διαδραματίζουν σε θεωρητικά προβλήματα, όπως την κατηγοριοποίηση (classification) και την ανίχνευση σφαλμάτων (fault diagnosis). Ιδιαίτερα, η παρούσα διατριβή έχει σκοπό της να εξερευνήσει τις έννοιες της ακριβούς εκτίμησης κατάστασης (detectability), της ανίχνευσης σφαλμάτων, και της κατηγοριοποίησης, χρησιμοποιώντας τεχνικές εκτίμησης κατάστασης, κατάλληλες για discrete event systems και stochastic discrete event systems. Ερευνούμε αυτά τα προβλήματα σε μη ντετερμινιστικά και στοχαστικά πεπερασμένα αυτόματα, σε τρεις διαφορετικές περιπτώσεις (συγκεκριμένα, μονολιθικά, αποκεντρωμένα και κατανεμημένα συστήματα). Ειδικότερα, ερευνούμε την έννοια της ανίχνευσης σφαλμάτων σε κατανεμημένα συστήματα, για μη-ντετερμινιστικά πεπερασμένα αυτόματα, σε συνθήκες περιορισμένης επικοινωνίας. Επίσης, εισάγουμε και επαληθεύουμε στοχαστικές έννοιες για τα προβλήματα της ακριβούς εκτίμησης κατάστασης και ανίχνευσης σφαλμάτων αναλύοντας την ασυμπτωτική συμπεριφορά των συστημάτων. Τέλος, προτείνουμε και συζητούμε διάφορες μεθόδους στο πρόβλημα της κατηγοριοποίησης σε κρυφά μαρκοβιανά μοντέλα (hidden Markov models). Αναπτύσσουμε διάφορες μεθόδους υπολογισμού ανωτάτων ορίων για την πιθανότητα λάθους κατηγοριοποίησης, που ασυμπτωτικά τείνουν στο μηδέν, κάτω από καθορισμένες ικανές συνθήκες.

Abstract

This thesis studies state estimation in complex networked systems that are modeled as discrete event systems (DES) or stochastic discrete event systems (SDES). One of the main motivations for looking into state estimation problems is their crucial role in classification and fault diagnosis applications. Specifically, this thesis aims to explore the notions of detectability, diagnosability, and classification, using state estimation methods appropriate for DES and SDES. We pursue investigations of such problems in nondeterministic and probabilistic finite automata, under three different observation settings (namely, centralized, decentralized, and distributed). In particular, we explore diagnosability in distributed settings for nondeterministic finite automata under communication constraints. Furthermore, we introduce stochastic notions for the problems of detectability and diagnosability, and analyze the asymptotic behaviour of the resulting state estimation processes. Finally, we discuss the classification among two hidden Markov models (HMMs), and develop various methods for computing asymptotically tight bounds on the probability of misclassification, under established sufficient conditions.

Acknowledgments

I would like to express my deepest appreciation and thanks to my advisor, Professor Christoforos N. Hadjicostis, who has been a great advisor for me and his commitment to excellence has inspired me. Thank you so much for your excellent guidance, constant encouragement, patience and unlimited support during my postgraduate studies at the University of Cyprus. Your high expertise and knowledge in the field of discrete event systems have helped me to overcome all the obstacles that I faced during the implementation of the work presented in this dissertation. You inspired me to achieve so many things that I did not even dream about. You constantly pushed me to question my ideas, or my results so that finally I am able to pose the right questions by myself. Last but not least, I would like to thank you for being a role-model for me, into so many aspects, such as professionalism, or research ethics. I would like to thank my family. I would like to express my love to my wife Maria, who always believed in me, even if I did not believe in my self. I like to thank her, for giving me my son, Manolis, who changed my life perspective completely. My world now is less egocentric than before. I would like to thank my mother Eygenia, and my sister Smaroula, for their uncoditional love. I would like to remember my father Manolis, who was very proud of me, even if he did not acknowledge it. I would like to remember him, for his realistic approach to any matter. I would like to remember him, for being passionate with his job. I will remember him, when I will read any of these philosophical, archaelogical books, in his personal library, that I used to read, when I was a little kid.

Publications

Published journal publications

1. J. Chen, C. Keroglou, C. N. Hadjicostis, and R. Kumar, "Revised test for stochastic diagnosability of discrete-event systems," *IEEE Transactions on Automation Science and Engineering*, vol. PP, no. 99, pp. 1–5, 2016.

2. C. Keroglou and C. N. Hadjicostis, "Detectability in stochastic discrete event systems," *Systems & Control Letters*, vol. 84, pp. 21–26, 2015.

Published conference proceedings

1. C. Keroglou and C. N. Hadjicostis, "Probabilistic system opacity in discrete event systems," in *Proceedings of 13th IFAC-IEEE International Workshop on Discrete Event Systems (WODES)*, 2016, pp. 379–384.

2. ——, "Distributed diagnosis using predetermined synchronization strategies in the presence of communication constraints," in *IEEE International Conference on Automation Science and Engineering (CASE)*, 2015, pp. 831–836.

3. ——, "Distributed diagnosis using predetermined synchronization strategies," in *53rd IEEE Conference on Decision and Control*, 2014, pp. 5955–5960.

4. ——, "Opacity formulations and verification in discrete event systems," in *IEEE Emerging Technology and Factory Automation (ETFA)*, 2014, pp. 1–12.

5. ——, "Hidden markov model classification based on empirical frequencies of observed symbols," in *12th IFAC-IEEE International Workshop on Discrete Event Systems (WODES), ENS Cachan, France*, 2014, pp. 7–12.

6. ——, "Detectability in stochastic discrete event systems," in *12th IFAC-IEEE International Workshop on Discrete Event Systems (WODES), ENS Cachan, France*, 2014, pp. 27–32.

7. ——, "Initial state opacity in stochastic des," in *18th Conference on Emerging Technologies and Factory Automation (ETFA)*, 2013, pp. 1–8.

8. ——, "Bounds on the probability of misclassification among hidden Markov models," in *50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, 2011, pp. 385–390.

9. ——, "Bound on the probability of HMM misclassification," in *19th Mediterranean Conference on Control Automation (MED)*, 2011, pp. 449–454.

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

This thesis explores state estimation techniques in discrete event systems (DES) that can be modeled by nondeterministic finite automata (NFAs) or probabilistic finite automa (PFAs), under particular observation models (that typically inhibit the direct observation of the system state). Early instances of state estimation problems in discrete event systems appear in [30] and [34], both of which formulate observability as a system property that requires perfect knowledge of the current state of the system. The observability property was generalized to various notions of detectability in [45].

State estimation is key in many control engineering applications involving complex systems. For example, opacity [7, 40] requires that a given set of states (with certain properties of interest) remain opaque (non-identifiable) based on the generated sequence of observations, regardless of the underlying activity in the system. Enforcing opacity, e.g., by enabling or disabling events at appropriate instances of instances of time, has recently drawn attention in a variety of security applications [53]. Another related application is fault diagnosis [24, 42, 54] which requires discrimination (within a finite time interval following the occurrence of a fault) between the set of normal states (states that are possible under normal behaviour) and the set of faulty states (states that are possible under faulty behaviour), for every possible trace that can be executed in the system; disambiguation between these two sets of states requires state estimation techniques. A similar problem in stochastic discrete event systems (probabilistic finite automata) is the classification between two given

models (hidden Markov models or probabilistic finite automata) [1]. Though the properties of detectability, fault diagnosis, classification and opacity can be used in diverse applications to capture distinct features of the underlying systems, they can be verified with similar tools, if one applies state estimation techniques.

1.1.1 Fault Diagnosis

Fault diagnosis is important in the monitoring and control of discrete event systems, with applications ranging from transportation systems to heating/ventilation systems, and from communication networks to medical diagnostics. In centralized settings, fault detection and isolation, as well as the verification of diagnosability (i.e., the ability to diagnose faults after a finite number of observations following their occurrence) for a given (non-deterministic) finite automaton that models the system of interest can be performed by a single entity, called observer or diagnoser, typically designed as a (deterministic) finite automaton that is driven by observable events [3,29,42]. For fault diagnosis, we consider some specific events that (indicate faults or abnormal conditions and) need to be detected (more generally, they need to be classified or identified) after a finite (bounded) delay.

Following [42], many researchers have investigated algorithms for fault diagnosis and the verification of diagnosability. In particular, language diagnosability as defined in [42] can be verified with polynomial-time algorithm [54]. The approach in [54] assumes a deterministic system and constructs, for each fault type, a verifier i.e., a nondeterministic finite automaton that can be used to check eventual (i.e., within a finite number of events or observations) fault detection and/or isolation of the corresponding failure type. The system is shown to be diagnosable for a particular failure type if all cycles of the corresponding verifier have identical state labels. The approach in [54] assumes a nondeterministic system and constructs a verifier automaton with language specified by the natural projection map. In contrast to [54], the verifier in [54] also tracks labels of multiple faults. The system is shown to be diagnosable if the cycles of the verifier consist of states with identical failure labels.

Diagnosability has also been investigated in decentralized architectures [8,13,33, 36,48,51,52]. In particular, the authors in [13] propose three protocols for coordinated decentralized diagnosis. Protocols 1 and 2 assume unidirectional communication

from the local diagnosers to a coordinator, whereas Protocol 3 assumes no communication between them or to any coordinator.

We elaborate on these protocols a bit since they are useful for our analysis later on.

1. In Protocol 1, the diagnostic information at a local site is generated by an *extended diagnoser*, the states of which consist of both the predecessor and successor state estimates of an observable event along with its failure label. The decision rule of the coordinator is defined under different intersection operations applied on pairs of system state estimates and their matching normal/failure conditions.

2. Protocol 2 uses the *basic (standard) diagnoser* to generate the local diagnostic information, and system diagnosis is performed under the same communication and decision rules as in Protocol 1. Although the computational complexity of Protocol 2 is reduced compared to Protocol 1, the performance of the former is constrained to traces that adhere to the "well-ordering" property of the coordinator, also referred to as *failure-ambiguous* traces in [13].

3. Protocol 3 is directly linked to the so-called property of *co-diagnosability*. A system is codiagnosable (or decentralized diagnosable) if any occurrence of a failure is detected by at least one local diagnoser within a bounded interval of observations.

Polynomial complexity algorithms for the verification of Protocol 3 and variations of it can be found in [33, 36, 48, 51, 52]. In [52] decentralized diagnosability is studied under a framework of *conditional decisions* issued by the local sites, and polynomial tests are proposed for the verification of equivalent language-based notions of decentralized diagnosability. In [51] the authors propose polynomial algorithms to transform a problem of co-observability to the problem of co-diagnosability under the assumption of dynamic observations. They also propose a polynomial-time algorithm for testing co-diagnosability of the system based on *cluster automata*. The authors of [48] study co-diagnosability under the condition of state-dependence and nondeterminism of partial event observation. The algorithm proposed in [36] is based on constructing a testing automaton that, given a faulty trace, searches for corresponding indistinguishable non-faulty traces at each local site. Our contribution is the application of a novel communication protocol, that allows the exchange of local state estimates between the local sites, under communication constraints.

1.1.2 Detectability

An important task associated with state estimation is that of accurate characterization of the possible (compatible) current states following a (possibly long) observation sequence generated by the underlying discrete event system. In deterministic settings, a key concept is the notion of detectability which was introduced in [45]. In particular, the notion of *strong detectability* holds if all observation sequences lead to an accurate estimate of the current state (perfect knowledge of the system state) after a finite number of observations. Thus, the notion of detectability is primarily determined by finite observation sequences generated by the underlying discrete event system. The authors of [45] defined four different notions for detectability in discrete event systems that can be modeled as nondeterministic automata: strong detectability, detectability, strong periodic detectability, and periodic detectability.

In stochastic DES (SDES) we can relax the strong notion of perfect state estimation described above, by requiring that perfect state estimation is only achieved asymptotically. The availability of stochastic information allows us to compute the probability of any trace of events and determine not only if a state is a possible candidate as a current state, but also how probable it is. In other words, the available probabilistic information can be used by the estimator to determine the posterior likelihood of a certain state (conditioned on a particular sequence of observations) and, via the verification process, to determine the likelihood of problematic observation sequences. Thus, a number of possibilities open up, in terms of utilizing the probabilistic information to characterize detectability in SDES.

The authors of [46] introduced notions of detectability in PFAs. Whereas in deterministic settings the problematic system behaviour corresponds to sequences of observations that do not lead to perfect state estimation, the approach in [46] takes the viewpoint that the problematic behaviour generates sequences of observations that do not allow us to estimate the exact state with increasing certainty. Furthermore, the approach in [46] analyzes all possible observation sequences (infinite sequences), however improbable (as long as they are feasible), and declares the system *not* stochastically detectable, if there is at least one such problematic sequence that is feasible. Our contribution includes the introduction of the novel stochastic notions of A-detectability, and AA-detectability, and their verification.

1.1.3 Classification among HMMs

Classification among systems that can be modeled as hidden Markov models (HMMs or PFAs) is related to the ability to distinguish the correct HMM based on a sequence of observations that has been generated by underlying (unknown) activity in one of two (or more) known HMMs. The ability to distinguish the correct HMM is, of course, related to the probability of selecting an incorrect HMM model, which is measured by the probability of misclassification among two HMMs. The performance of the maximum *a posteriori* (MAP) classifier, which minimizes the probability of misclassification [1], is captured by the *a priori* probability of error, i.e., the probability of error before any observations are made. The precise calculation of the probability of error (for sequences of observations of a given finite length) is a combinatorial task of high complexity (typically exponential in the length of the sequences).

Classification can find application in many areas where HMMs are used, including speech recognition [2,23,37], pattern recognition [19], bioinformatics [15,27], and failure diagnosis in discrete event systems [1,11,31,49]. Classification is also related to approaches dealing with the *distance* or dissimilarity between two HMMs [17,25].

Our contribution involves methods that allow us to obtain an upper bound on the probability of misclassification with lower complexity than of the precise computation. Directly related previous work can be found in [1], which studies the probability of misclassification and obtains bounds that tend to zero under specific conditions.

1.2 Main Contributions and Thesis Organization

This thesis explores different state estimation problems in DES, such as detectability, fault diagnosis, and classification. These problems differ in their formulations and objectives, but share related verification methods. It is known that notions of detectability and diagnosability, in nondeterministic finite automata can be verified by employing similar constructions, called observers or diagnosers. The complexity of constructing an observer/diagnoser is exponential in the number of states of the original system. Verification methods with polynomial complexity in the number of the states of the system, have also been presented for detectability/diagnosability respectively. These methods employ finite automata called detectors/verifiers [45], [54].

This thesis studies state estimation problems in distributed settings and in stochastic settings. In distributed settings, this thesis studies diagnosability using prespecified communication protocols that involve exchanges of state estimates, among local agents that observe locally available events and can communicate (exchange information about state estimates). In stochastic settings we combine stochastic state estimation with classification for hidden Markov models or probabilistic finite automata. Using hypothesis testing techniques, we provide conditions for asymptotically tight bounds on the probability of misclassification among two HMMs. The explicit description of the contribution of each chapter follows.

1. In Chapter 3 we study diagnosis using a synchronization-driven intersection-based distributed diagnosis protocol (called Restricted Synchronization Intersection Based Diagnosis, or RS-IBDD). The RS-IBDD protocol allows local diagnostic information (namely, state estimates and associated normal/fault conditions) to be exchanged and refined among the local sites. The exchange and refinement of the diagnostic information takes place at synchronization points, that are predetermined at each observation site, by taking the intersection of the state estimates and associated normal/fault conditions provided by the local diagnosers. The fused information is then communicated back to the local diagnosers, which subsequently continue operation based on this refined diagnostic information. In this protocol there is no need for a coordinator, but the diagnostic information is exchanged and refined among the local sites. Furthermore, communication constraints dictate a general approach, which, among others, exploits the communication graph that represents the bidirectional communication capability between local sites. Diagnosability of the resulting distributed protocol is shown to be verifiable with polynomial complexity (in the size of the state space of the given nondeterministic finite automaton) and exponential in the number of observation sites.

2. In Chapter 4 we are interested in exploring state estimation techniques in stochastic discrete event systems (SDES) that can be modeled by probabilistic finite automata (PFAs) under particular observation models. The authors of [45] and [46] introduced notions of detectability in nondeterministic and stochastic settings respectively. When dealing with detectability in nondeter-

ministic finite automata in [45], the problematic system behaviour as far as detectability is concerned corresponds to sequences of observations that do not lead to exact state estimation (i.e., they do not lead to perfect state estimation, with no uncertainty). When dealing with detectability in PFA's in [46], the problematic behaviour is associated with sequences of observations that do not allow us to estimate the exact state with increasing certainty. The major contribution of this chapter is the introduction and verification of the notions of A-detectability and AA-detectability. Specifically, we provide necessary and sufficient conditions for A-detectability and AA-detectability, along with a proof that A-detectability is a PSPACE-hard problem. A-detectability concentrates on highly probable system behaviour and characterizes the given system's detectability accordingly. By considering only observation sequences that belong to the recurrent behavior of the system, A-detectability does not take into account observation sequences that are treated as problematic in previous notions of stochastic detectability. In some cases even if the system is not A-detectable (i.e., there exists a nonzero probability of generating observation sequences that correspond to possible estimates for more than one state), the probability of estimating the correct state for these observation sequences goes to one. These cases lead to the definition of AA-detectability.

3. In Chapter 5 we analyze the problem of classification among Hidden Markov models (HMMs).

   (a) In Section 1 we characterize a class of upper bounds on the *a priori* probability of error when classifying among two known HMMs. By introducing an appropriate deterministic finite automaton (DFA), we systematically merge different sequences of the same length in a way that allows easy computation of an upper bound on the probability of misclassification. Our approach also allows us to use Markov chain theory to obtain an upper bound for asymptotically large n (in all cases, the approach has complexity polynomial in the size of the two given HMMs and the size of the DFA that is used).

   (b) In Section 2 we characterize an upper bound on the probability of error,

when classifying among two HMMs, by constructing a finite automaton that captures the common behaviour of the two HMMs. We also establish necessary and sufficient conditions for this bound to tend to zero exponentially with increasing observation steps.

(c) In Section 3 we provide a method for obtaining an upper bound on the probability of misclassification between two competing HMMs using a suboptimal rule. We use a suboptimal decision rule which counts the number of times each output symbol appears, obtains the empirical (measured) frequency of each output symbol, and compares empirical frequencies against the expected frequencies in each of the two systems. We establish that the verification of the effectiveness of this rule is polynomial with respect to the number of states of the two HMMs. Specifically, we are able to discriminate between the two models using the suboptimal decision rule based on the empirical frequencies of output symbols, as long as the two systems are characterized, at steady-state, by different statistical properties for the occurrence of output symbols, which we refer to as stationary emission probabilities. Furthermore, we apply these results into a special case of probabilistic system opacity, so as to have a polynomial complexity verification algorithm.

Chapter 2

Notation and Background

2.1 Notation on Languages and Automata

Let Σ be an alphabet (set of events) and denote by Σ^* the set of all finite-length strings of elements of Σ (sequences of events), including the empty string ε (the length of a string s is denoted by $|s|$ with $|\varepsilon| = 0$). A language $L \subseteq \Sigma^*$ is a subset of finite-length strings in Σ^* [10] (i.e., sequences of events with the convention that the first event appears on the left). Given strings $s, t \in \Sigma^*$, the string st denotes the concatenation of s and t, i.e., the sequence of events captured by s followed by the sequence of events captured by t. For a string s, $\bar{s}$ denotes the *prefix-closure* of s, and is defined as $\bar{s} = \{t \in \Sigma^* \mid \exists t' \in \Sigma^* \{tt' = s\}\}$. For two string s and t, we also define $t \in s$, if $\exists t_1, t_2 \in \Sigma^*$, such as $t_1 t t_2 = s$.

Definition 1. *(Nondeterministic Finite Automaton (NFA)). A nondeterministic finite automaton is captured by $G = (X, \Sigma, \delta, X_0)$, where $X = \{1, 2, \ldots, N\}$ is the set of states, Σ is the set of events, $\delta : X \times \Sigma \to 2^X$ is the nondeterministic state transition function, and $X_0 \subseteq X$ is the set of possible initial states.*

For a set $Q \subseteq X$ and $\sigma \in \Sigma$, we define $\delta(Q, \sigma) = \cup_{q \in Q} \delta(q, \sigma)$; with this notation at hand, the function δ can be extended from the domain $X \times \Sigma$ to the domain $X \times \Sigma^*$ in the routine recursive manner: $\delta(x, \sigma s) := \delta(\delta(x, \sigma), s)$ for $x \in X$, $s \in \Sigma^*$ and $\sigma \in \Sigma$ (note that $\delta(x, \varepsilon) := \{x\}$). The behavior of G is captured by $L(G) := \{s \in \Sigma^* \mid \exists x_0 \in X_0 \{\delta(x_0, s) \neq \emptyset\}\}$. We use $L(G, x)$ to denote the set of all traces that originate from state x of G (so that $L(G) = \bigcup_{x_0 \in X_0} L(G, x_0)$).

Definition 2. *(Deterministic Finite Automaton (DFA)). A deterministic finite automaton is captured by $D = (X, \Sigma, \delta, x_0)$, where $X = \{1, 2, \ldots, N\}$ is the set of states, Σ is the set*

of events, $\delta : X \times \Sigma \to X$ is the (possibly partially defined) deterministic state transition function, and $x_0 \in X$ is the initial state.

The function δ can be extended from the domain $X \times \Sigma$ to the domain $X \times \Sigma^*$ in the routine recursive manner:

$$\delta(x, \sigma s) = \begin{cases} \delta(\delta(x, \sigma), s), & \text{if } \delta(x, \sigma) \text{ is defined,} \\ \text{undefined,} & \text{otherwise,} \end{cases}$$

for $x \in X$, $s \in \Sigma^*$ and $\sigma \in \Sigma$ (note that in this case $\delta(x, \varepsilon) := x$). The behavior of D is captured by $L(D) := \{s \in \Sigma^* \mid \delta(x_0, s) \text{ is defined}\}$.

In general, only a subset Σ_o ($\Sigma_o \subseteq \Sigma$) of the events can be observed, so that Σ is partitioned into the set of observable events Σ_o and the set of unobservable events $\Sigma_{uo} = \Sigma - \Sigma_o$. The natural projection $P_{\Sigma_o} : \Sigma^* \to \Sigma_o^*$ maps any trace executed in the system to the sequence of observations associated with it, and is defined recursively as $P_{\Sigma_o}(s\sigma) = P_{\Sigma_o}(s)P_{\Sigma_o}(\sigma),\ \sigma \in \Sigma, s \in \Sigma^*$, with

$$P_{\Sigma_o}(\sigma) = \begin{cases} \sigma, & \text{if} \quad \sigma \in \Sigma_o, \\ \varepsilon, & \text{if} \quad \sigma \in \Sigma_{uo} \cup \{\varepsilon\}, \end{cases}$$

where ε represents the empty observation trace [10]. In the sequel, the subscript Σ_o in P_{Σ_o} will be dropped when it is clear from context. The inverse projection of a string of observable events $\omega \in P(L(G))$ is given by

$$P^{-1}(\omega) = \{s \in L(G) :\ P(s) = \omega\}.$$

Definition 3. *(Possible states following a sequence of observations ($R : 2^{|X|} \times \Sigma_o^* \to 2^{|X|}$)). Suppose that a nondeterministic automaton $G = (X, \Sigma, \delta, X_0)$ is in a state in the set of possible states $X' \subseteq X$; the set of all possible states after observing $\omega \in \Sigma_o^*$ is $R(X', \omega) = \{x \in X \mid (\exists x' \in X')(\exists s \in \Sigma^*)\{P(s) = \omega \wedge x \in \delta(x', s)\}\}$.*

Note that using Definition 3, the unobservable reach of the set $X' \subseteq X$ [10] can be expressed as $UR(X') = R(X', \varepsilon)$ and it denotes the set of states that are reachable from a state in X' via zero, one, or more unobservable events.

2.2 Detectability in Nondeterministic Finite Automata

The authors of [45] define four different notions for detectability in discrete event systems that can be modeled as nondeterministic automata: strong detectability,

detectability, strong periodic detectability, and periodic detectability. We recall below the notion of strong detectability, which is of interest for the developments in this thesis, given that we are able to introduce constructions, such as the Detector, which are useful for the analysis of the stochastic notions, later in Chapter 4.

Definition 4. *(Strong Detectability) [45]. An NFA $G = (X, \Sigma, \delta, X_0)$ is strongly detectable with respect to natural projection map P for a set $\Sigma_{obs} \subseteq \Sigma$ of observable events if we can determine the current state and subsequent states of the system, after a finite number of observations, for all trajectories s, $s \in L(G)$, of the system greater than a certain length N, i.e.,*

$$(\exists N \in \mathbb{N})(\forall n \geq N)$$

$$(\forall s \in L(G))\{(|s| = n) \Rightarrow (|R(X_0, P(s))| \leq 1)\}.$$

Remark: For all $s \in L(G)$, we have $|R(X_0, P(s))| \geq 1$, thus in the above definition we could have said $|R(X_0, P(s))| = 1$.

As motivation for studying the notion of detectability, consider a vehicle capable of moving on a grid, such as the toy grid in Fig. 2.1(a). If we use the cell number to denote the state of the vehicle, then the trajectory that the vehicle follows corresponds to a sequence of states and the origin of the trajectory is captured by the initial state of the vehicle. The vehicle's possible movements are available via a kinematic model, i.e., a finite automaton whose states are associated with the state (cell) of the vehicle and whose transitions correspond to the movements of the vehicle that are allowed at each position (up, down, left, right, diagonal, etc. — the allowed movements will presumably depend on the underlying terrain that the grid is capturing). Fig. 2.1(b) depicts an example of a kinematic model for the vehicle that moves in the toy grid of Fig. 2.1(a).

Suppose specific sensors that detect events α and β are deployed in the grid. Sensor α detects the left and right movements of the vehicle. Sensor for β detects the up and down movements of the vehicle, along with movement that finalizes at the same cell. Fig. 2.2 depicts the (non-deterministic) finite automaton G that models both the kinematic model of the vehicle and the corresponding sensor readings for a particular set of sensor coverages. Essentially G is a non-deterministic finite automaton with partial observation on its transitions.

We can extend the above formulation to a probabilistic setting under which each transition is assigned a specific probability of occurrence as in Fig. 4.1. One of the

questions that might arise in the above context is that of understanding whether the sensory information that is available allows us to obtain important information about the present location (current state) of the vehicle. In a probabilistic setting this translates to the requirement that the probability of violating strings be under or above a specific threshold.

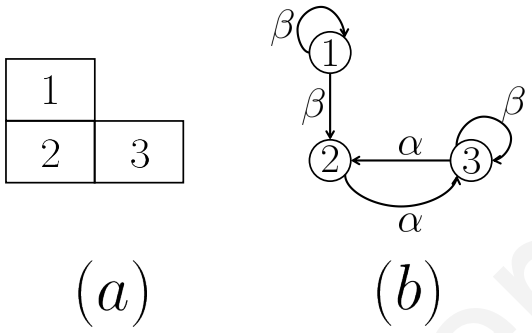

Figure 2.1: (a) Grid in which a vehicle can move; (b) Kinematic model for a vehicle in the grid in (a).

Definition 5. *(Observer or Current-state estimator) [9]. Given an NFA* $G = (X, \Sigma, \delta, X_0)$ *with set of observable events* $\Sigma_{obs} \subseteq \Sigma$ *under the natural projection map P, the observer (or current-state estimator) is a deterministic finite automaton (DFA)* $G_{obs} = (Q_{obs}, \Sigma_{obs}, \delta_{obs}, Q_{0,obs})$, *which captures the state estimates (following a sequence of observations* $\omega \in \Sigma^*_{obs}$*) and can be constructed as follows.*
(1) Each state of G_{obs} *is associated with a unique subset of states of the original NFA G (this means that* $Q_{obs} \subseteq 2^X$ *has at most* $2^{|X|}$ *states).*
(2) The initial state $Q_{0,obs}$ *of* G_{obs} *is the unobservable reach of* X_0 *(*$Q_{0,obs} = UR(X_0) = R(X_0, \varepsilon)$*).*
(3) From any state $Q \in Q_{obs}$ *of the current-state estimator, the next state for any* $\sigma \in \Sigma_{obs}$ *is captured by* $\delta_{obs}(Q, \sigma) = R(Q, \sigma)$ *(Definition 3).*

Verification of Strong Detectability

Strong detectability for NFA *G* can be verified easily by constructing its observer G_{obs} and checking whether it has loops with certain properties [45].

Theorem 1. *(Strong detectability: Necessary and sufficient conditions using observer* G_{obs}*) [45]. A nondeterministic automaton* $G = (X, \Sigma, \delta, X_0)$ *is strongly detectable with respect to*

a set of observable events Σ_{obs} iff its observer $G_{obs} = (Q_{obs}, \Sigma_{obs}, \delta_{obs}, Q_{0,obs})$ does not include loops that contain ambiguous states (i.e., states in Q_{obs} that involve more than one states of G) .

Example 1. *Consider NFA G in Fig. 2.2 (associated to PFA H in Fig. 4.1), and assume that $X_0 = \{x_1, x_2, x_3\}$, $\Sigma_{obs} = \{\alpha, \beta\}$ and $\Sigma_{uo} = \emptyset$. We construct its observer (on the right of Fig. 2.2) to verify strong detectability (Definition 4).*

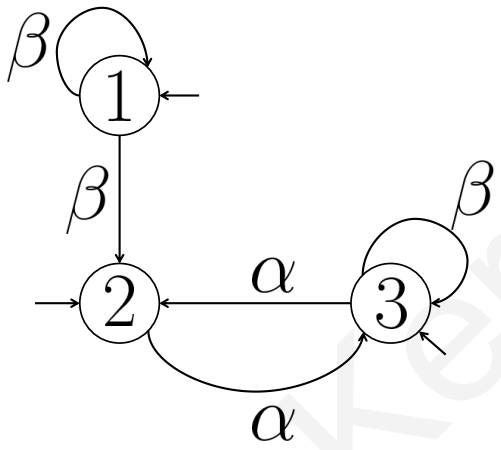

Figure 2.2: NFA G.

System G is not strongly detectable (Definition 4), because the sequence β^* does not allow us to detect the exact state of the system ($|R(X_0, \beta^*)| > 1$); this is easily seen from the observer because the sequence of observations β^* keeps us in state $\{x_1, x_2, x_3\}$ arbitrarily long.

Apart from verifying detectability using an observer (as captured by the conditions in Theorem 1), strong detectability for NFA G can also be verified with polynomial complexity using a detector. Next we quickly review the construction of a detector from [45] and then state necessary and sufficient conditions on the detector for strong detectability to hold.

Definition 6. *(Detector) [45]. Given an NFA $G = (X, \Sigma, \delta, X_0)$ under the natural projection map P with respect to $\Sigma_{obs} \subseteq \Sigma$, the detector $G_d = (X_d, \Sigma_{obs}, \delta_d, X_{0d})$ is a nondeterministic finite automaton, where*

(1) $X_{0d} = R(X_0, \varepsilon)$ is the set of all possible initial states for NFA G;

(2) $X_d = X_p \cup X_s \cup \{X_{0d}\}$ is the finite set of states, with $X_s = \{\{x_j\} | x_j \in X\}$ and $X_p \equiv \{x_{d_1}, x_{d_2}, ..., x_{d_D}\}$, where $D = |X \times X| - |X|$ with $x_{d_i} = \{x_l, x_m\} \in X_p$, $x_l \neq x_m$, $x_l, x_m \in X$;

(3) Σ_{obs} is the finite set of observable events;

(4) $\delta_d : X_d \times \Sigma_{obs} \rightarrow X_d$ captures the state transitions and is defined as follows:

$$\delta_d(x_d, \sigma) = \begin{cases} \{x_{d_i} \in X_p | x_{d_i} \subseteq R(x_d, \sigma)\}, & \text{if} \quad |R(x_d, \sigma)| > 1, \\ \{x_l\} \in X_s, & \text{if} \quad R(x_d, \sigma) = \{x_l\}, \\ \text{undefined}, & \text{if} \quad R(x_d, \sigma) = \emptyset. \end{cases}$$

Theorem 2. *[45] (Strong detectability: Necessary and sufficient conditions using detector G_d). An NFA G is strongly detectable iff its detector G_d does not include any loop that is reachable from the initial state and contains ambiguous states (i.e., states in X_p).*

Remark: The detector G_d (in Definition 6) can be used to verify strong detectability for NFA G with polynomial complexity (with respect to the size of the given NFA). The reason is that the number of states of the detector is at most $|X|^2 + 1$; this should be contrasted with the number of states of the observer which could be as high as $2^{|X|}$.

Example 2. *The detector G_d for the NFA G in Fig. 2.2 is shown in Fig. 2.3. Its number of states is polynomial in the size of G and can be used to verify strong detectability with polynomial complexity. In this case, the existence of the loops β^* in states $\{1,2\}$ and $\{1,3\}$ indicates that system G is not detectable. Note that the observer (shown in Fig. 2.2), also allows us to verify strong detectability but requires potentially exponential complexity (with respect to the size of the state space of the corresponding NFA).*

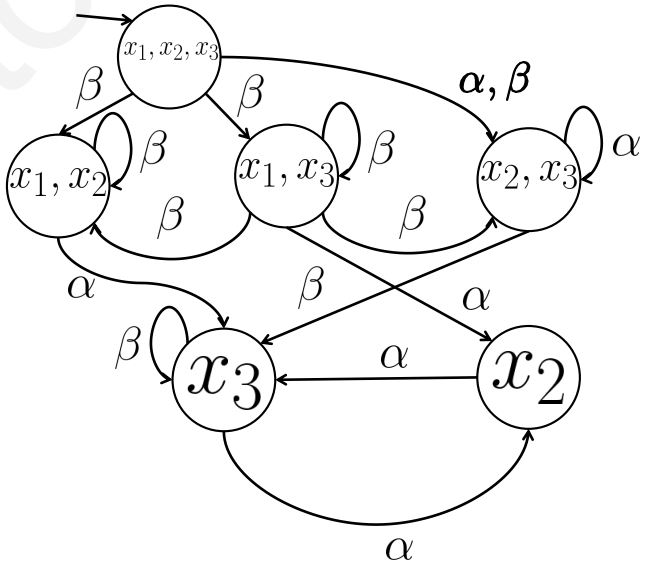

Figure 2.3: Detector for NFA G in Fig. 2.2, with initial state $\{x_1, x_2, x_3\}$.

2.3 Fault Diagnosis in Nondeterministic Finite Automata

For ease of presentation, we assume that there is only a single fault class $\Sigma_f \subseteq \Sigma$ and assume, without loss of generality, that $\Sigma_f \subseteq \Sigma_{uo}$ (otherwise, since an observable fault event is immediately visible, the task of fault diagnosis becomes trivial). We also denote the set of traces $s \in L(G)$ that end with the failure event σ_f, $\sigma_f \in \Sigma_f$, that occurs for the first time, by $\Psi(\sigma_f) = \{t\sigma_f \in L\colon t \in (\Sigma - \sigma_f)^*, \}$ and the post-language of L after s as $L \setminus s = \{t' \in \Sigma^*\colon st' \in L\}$.

Definition 7. *(Logical diagnosability in discrete event systems) [42]. Suppose we are given an NFA $G = (X, \Sigma, \delta, X_0)$ with a set of observable events Σ_o, $\Sigma_o \subseteq \Sigma$. We are also given fault event σ_f, $\sigma_f \in \Sigma_{uo}$ (where $\Sigma_{uo} \equiv \Sigma - \Sigma_o$). For $\sigma_f \in \Sigma_{uo}$, let $\Psi(\sigma_f) = \{s = t\sigma_f \in L(G) : \{t \in (\Sigma - \sigma_f)^*\}\}$. The live[1], prefix-closed language $L(G)$ is logically diagnosable with respect to fault σ_f, under the natural projection observation map P (with respect to Σ_o) if*

$$(\exists N_i \in \mathbb{N})[\forall s \in \Psi(\sigma_f)](\forall t \in L/s)[|t| \geq N_i \Rightarrow D(st) = 1]$$

where the diagnosability function $D : \Sigma^* \to \{0, 1\}$ is

$$D(st) = \begin{cases} 1, & \text{if } s' \in P^{-1}[P(st)] \Rightarrow \sigma_f \in s', \\ 0, & \text{otherwise.} \end{cases}$$

In other words, a diagnosable system implies that, once fault σ_f occurs, it gets diagnosed within a bounded observation interval ($\sigma_f \in s'$ means that σ_f appears in the string s' at least once).

Example 3. *Consider the NFA G in Fig. 2.4 where $X_0 = \{0\}$, $\Sigma_f = \{\sigma_f\}, \Sigma_{uo} = \Sigma_f$ and $\Sigma_o = \{a, b, c, d\}$. For the given Σ_o and Σ_f, we easily prove that $(\exists N_i \in \mathbb{N})[\forall s \in \Psi(\sigma_f) \equiv \{\sigma_f\}](\forall t \in L/s \equiv \{da(a + b + c)^*\})[|t| \geq N_i \Rightarrow D(\sigma_f da(a + b + c)^*) = 1]$; more specifically, in this case we can take $N_i = 1$. Thus, we conclude that G is logically diagnosable with respect to fault σ_f.*

For simplicity, below we repeat the diagnoser construction for a single fault.

Definition 8. *(Diagnoser) [42]. The Diagnoser $G_d = \{Q_d, \Sigma_o, \delta_d, Q_{0,d}\}$ is a deterministic finite automaton built from $G = (X, \Sigma, \delta, X_0)$.*

(i). The state space $Q_d \equiv 2^{X \times \Delta}$, where Δ is the complete set of possible labels: $\Delta = \{N, F\}$ where

[1]The notion live language refers to a language without terminating strings.

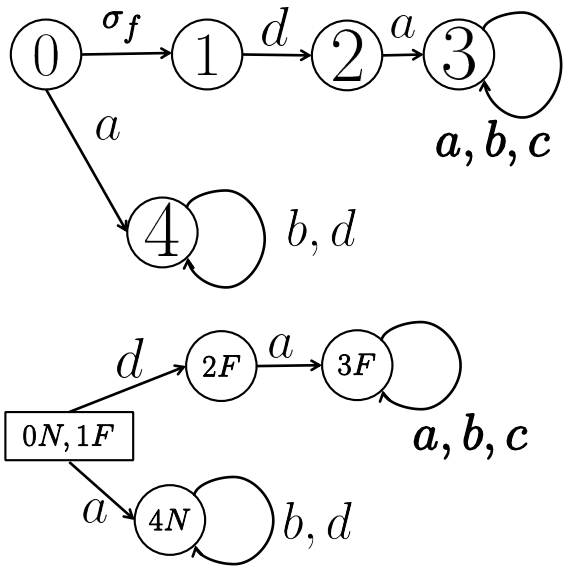

Figure 2.4: Automaton G (Up) and its Diagnoser (Down) used in Example 3.

F is the label that corresponds to the faulty behaviour and N is the label that corresponds to normal behaviour of the system. More specifically, a state $q_d \in Q_d$ is of the form $q_d \subseteq X \times \Delta$, i.e., q_d can be written as

$$q_d = \{(x_1, l_1), \ldots, (x_n, l_n)\},$$

with $x_i \in X$ and $l_i \in \Delta$ (repetitions of x_i are permitted[2]).

(ii). We will use the notation $URL(X)$ to denote the set of states (along with any failure labels that can be reached in G from states in the set X following any sequence of events $(\Sigma - \Sigma_o)^*$. The initial state of the diagnoser is given as $Q_{0,d} = URL(X_0)$ (the fault has not occurred at the initial states belong to X_0 so these are labeled as $\{N\}$ states) where $(x_{0,i}, l_{0,i}) \in URL(X_0)$ with $l_{0,i} = \{N\}$, if $\exists s \in \Sigma_{uo}^*$ s.t. $x_{0,i} \in R(X_0, s)$ and s does not contain the fault event σ_f. Similarly, $(x_{0,i}, l_{0,i}) \in Q_{0,d}$ with $l_{0,i} = \{F\}$, if $\exists s \in \Sigma_{uo}^*$ s.t. $x_{0,i} \in R(X_0, s)$ and s contains the fault event σ_f.

(iii). The transition function $\delta_d \colon 2^{X \times \Delta} \times \Sigma_o \to 2^{X \times \Delta}$ satisfies $q'_d = \delta_d(q_d, \sigma)$ and updates both the possible states and matching normal/failure condition(s) as defined in [42]; the current state q_d of the diagnoser captures the set of estimates of G with their corresponding labels. Let the next observed event be $\sigma \in \Sigma_o$; the new state of the diagnoser q'_d is computed following a three step process:

1) For every state estimate x, where $(x, l) \in q_d$, compute its reach due to σ defined to be $R(x, \sigma)$ (where $R(x, \sigma)$ was defined in Definition 3).

2) Let $x' \in R(x, \sigma)$ with $x' \in \delta(x, s_1 \sigma s_2)$ for some $s_1, s_2 \in \Sigma_{uo}^*$. The label l' associated to x' with

[2]For example, state q_d could be the set $\{(x_1, N), (x_1, F)\}$.

x is obtained based on the label l associated with state x (i.e., $(x,l) \in q_d$) and the following rules:

a) If $l = F$, then the label $l' = F$.

b) If $l = N$, and s_1, s_2 do not contain the failure event, then the label $l' = N$.

c) If $l = N$, and s_1 and/or s_2 contain the failure event, then the label $l' = F$.

3) Let $\delta_d(q_d, \sigma) = q'_d$ be the set of all (x', l') pairs computed following steps 1) and 2) above, for each (x, l) in q_d.

Note that, for simplicity of our exposition later on (and unlike common practice), we allow the diagnoser function δ_d to be defined everywhere[3] (i.e., $Q_d = 2^{X \times \Delta}$) even if certain states might not be necessary because they may not be reachable from the initial state $Q_{0,d}$. Verification of diagnosability using the diagnoser depends on the presence of F-uncertain states which are defined as follows [42].

Definition 9. *(F-uncertain states) [42]. A state $q_d \in Q_d$ of the diagnoser is called F-uncertain if $\exists (x,l), (x',l') \in q_d$, such that $F \in l$ and $F \notin l'$.*

System G is diagnosable with respect to failure type F if its diagnoser (for failure type F) contains no cycles composed exclusively of F-uncertain states. In addition, a system will generally [4] not be diagnosable if its diagnoser contains cycles composed exclusively of F-uncertain states.

2.4 Classification among Hidden Markov Models

This section describes the hidden Markov model (HMM) and the methods needed to compute the probability of misclassification among two HMMs.

As a motivating application of classification we present in Fig. 2.5-a an example from the context of fault diagnosis of a Discrete Event System, which translates to classification between two HMMs, capturing normal and faulty behavior. Specifically, the first HMM describes a system with normal behavior, whereas the second

[3] This is useful in Chapter 3 when we define the *synchronized parallel product of local diagnosers* in the context of distributed state estimation and diagnosis.

[4] It is possible that cycles with F-uncertain states are not executable after the occurrence of faults; since such cycles do not violate diagnosability, we also need to check against this possibility when such cycles are present [42].

HMM describes the same system but with faulty behavior. The Discrete Event System is shown in Fig. 2.5-a with probabilities attached to each transition. The set of observable events is $E_o = \{\alpha, \beta\}$ and the set of unobservable events is $E_{uo} = \{\sigma_{uo}, \sigma_f\}$, where σ_f a faulty event (i.e., $E_f = \{\sigma_f\}$).

We divide the initial system into two subsystems as in Fig. 2.5-b, where $S^{(1)}$ captures the normal behavior of the system, and $S^{(2)}$ captures the faulty behavior. Clearly, $P_1 = P_2 = 0.5$, because of the equal probability to go to state 2 or state 4, from initial state 1.

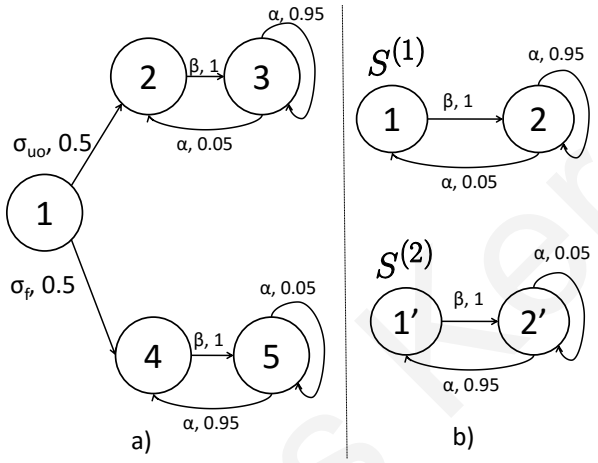

Figure 2.5: a) Discrete Event System, b) HMM $S^{(1)}$, capturing normal behavior and $S^{(2)}$, capturing faulty behavior of Discrete Event System in a).

Definition 10. *(HMM Model). An HMM is described by a five-tuple* $S = (Q, E, \Delta, \Lambda, \pi_0)$, *where* $Q = \{q_1, q_2, ..., q_{|Q|}\}$ *is the finite set of states;* $E = \{\sigma_1, \sigma_2, ..., \sigma_{|E|}\}$ *is the finite set of outputs;* $\Delta : Q \times Q \rightarrow [0\ 1]$ *captures the state transition probabilities;* $\Lambda : Q \times E \times Q \rightarrow [0\ 1]$ *captures the output probabilities associated with transitions; and* π_0 *is the initial state probability distribution vector. Specifically, for* $q, q' \in Q$ *and* $\sigma \in E$, *the output probabilities associated with transitions are given by*

$$\Lambda(q, \sigma, q') \equiv P(q[t+1] = q', E[t+1] = \sigma \mid q[t] = q) \ ,$$

and the state transition probabilities are given by

$$\Delta(q, q') \equiv P(q[t+1] = q' \mid q[t] = q) \ ,$$

where $q[t]$ *(*$E[t]$*) is the state (output/observation) of the HMM at time step t. The output function* $\Lambda(q, \sigma, q')$ *describes the conditional probability of observing the output* σ *associated*

with the transition to state q' from state q. The state transition function needs to satisfy

$$\Delta(q, q') = \sum_{\sigma \in E} \Lambda(q, \sigma, q') \tag{2.1}$$

and also $\sum_{i=1}^{|Q|} \Delta(q, q_i) = 1, \forall q \in Q$.

Given an HMM model $S = (Q, E, \Delta, \Lambda, \pi_0)$*, we also define for notational convenience the* $|Q| \times |Q|$ *matrix* A_σ*, associated with output* $\sigma \in E$*, as follows: the* $(k, j)^{th}$ *entry of* A_σ *captures the probability of a transition from state* q_j *to state* q_k *that produces output* σ*, i.e.,* $A_\sigma(k, j) = \Lambda(q_j, \sigma, q_k)$*. Note that* $A = \sum_{\sigma \in E} A_\sigma$ *is a column stochastic matrix whose* $(k, j)^{th}$ *entry denotes the probability of taking a transition from state* q_j *to state* q_k*, without regard to the output produced.*

Suppose that we are given two HMMs, captured by $S^{(1)} = (Q^{(1)}, E^{(1)}, \Delta^{(1)}, \Lambda^{(1)}, \pi_0^{(1)})$ and $S^{(2)} = (Q^{(2)}, E^{(2)}, \Delta^{(2)}, \Lambda^{(2)}, \pi_0^{(2)})$, with prior probabilities for each model given by P_1 and $P_2 = 1 - P_1$, respectively. Given $E^{(j)} = \{\sigma_1^{(j)}, \sigma_2^{(j)}, ..., \sigma_{|E^{(j)}|}^{(j)}\}$, $j = \{1, 2\}$, for the two HMMs, we define for notational convenience $E = E^{(1)} \cup E^{(2)}$ with $E = \{\sigma_1, \sigma_2, ..., \sigma_{|E|}\}$, and let $A_{\sigma_i}^{(j)}$ be the transition matrix for $S^{(j)}$, $j = \{1, 2\}$, under the output symbol $\sigma_i \in E$. We set $A_{\sigma_i}^{(j)}$ to zero if $\sigma_i \in E - E^{(j)}$. If we observe a sequence of n outputs $Y_1^n = y[1], y[2], ..., y[n]$, with $y[t] \in E$, that is generated by one of the two underlying HMMs, the classifier that minimizes the probability of error needs to implement the maximum *a posteriori* probability (MAP) rule. Specifically, the MAP classifier compares

$$P(S^{(1)} \mid Y_1^n) \underset{<}{\overset{>}{}} P(S^{(2)} \mid Y_1^n) \Rightarrow \frac{P(Y_1^n \mid S^{(1)})}{P(Y_1^n \mid S^{(2)})} \underset{<}{\overset{>}{}} \frac{P_2}{P_1},$$

and decides in favor of $S^{(1)}$ ($S^{(2)}$) if the left (right) quantity is larger. When we decide in favor of one or the other model, we incur a probability of error proportional to the probability of the model that was not selected; with some algebra, it can be shown that $P(\text{error}, Y_1^n) = \min\{P_1 \cdot P(Y_1^n \mid S^{(1)}), P_2 \cdot P(Y_1^n \mid S^{(2)})\}$. Clearly, if $E^{(1)} \neq E^{(2)}$, at least one symbol σ_i is unique to $S^{(1)}$ (i.e., $\sigma_i \in E - E^{(2)}$) or to $S^{(2)}$ (i.e., $\sigma_i \in E - E^{(1)}$), and if we happen to observe σ_i (i.e. $y[t] = \sigma_i$, for some t) then we will choose the model with nonzero probability of error and will make an error with zero probability. More generally, however, the probability of error will not be zero.

To calculate the *a priori* probability of error before any sequence of observations of length n is observed, we need to consider all possible observation sequences of

length n, so that

$$P(\text{error at } n) \;=\; \sum_{Y_1^n \in E^n} P(\text{error}, Y_1^n), \tag{2.2}$$

where E^n is the set of all sequences of length n with outputs from E. We arbitrarily index each of the d^n ($d = |E|$) sequences of observations via $Y(i)$, $i \in \{1, 2, ..., d^n\}$, and use $P_i^{(j)}$ to denote $P_i^{(j)} = P(Y(i)|S^{(j)})$, $j \in \{1, 2\}$. Note that some of these sequences may have zero probability under one of the two models (or even both models). The probability of misclassification between the two systems, after n steps, can then be expressed as

$$\begin{aligned} P(\text{error at } n) &= \sum_{i=1}^{d^n} P(\text{error}, Y(i)) \\ &= \sum_{i=1}^{d^n} \min\{P_1 \cdot P_i^{(1)}, P_2 \cdot P_i^{(2)}\}. \end{aligned} \tag{2.3}$$

We can calculate $P_i^{(j)} = P(Y(i)|S^{(j)})$ with an iterative algorithm, a detailed description of which can be found in [1, 19]. Specifically, given sequence $Y_1^n = y[1], y[2], ..., y[n]$ we calculate $\rho_n^{(j)} = A_{y[n]}^{(j)} A_{y[n-1]}^{(j)} ... A_{y[1]}^{(j)} \pi_0^{(j)}$, which is essentially a vector whose k^{th} entry captures the probability of reaching state $q_k \in Q^{(j)}$ while generating the sequence of outputs Y_1^n (i.e., $\rho_n^{(j)}(k) = P(q[n] = q_k, Y_1^n|S^{(j)})$). If we sum up the entries of $\rho_n^{(j)}$ we obtain $P_{Y_1^n}^{(j)} = P(Y_1^n \mid S^{(j)}) = \sum_{k=1}^{|Q^{(j)}|} \rho_n^{(j)}(k)$.

Utilizing the above algorithm, we can certainly compute the probability of error at n by explicitly calculating $P_j \cdot P_i^{(j)}$ for each sequence $Y(i)$. The obvious problem with this approach is that it has to enumerate all d^n sequences.

Example 4. *Suppose we are given the two HMMs shown in Fig. 2.6, with* $E^{(1)} = E^{(2)} = E = \{\alpha, \beta\}$, $\pi_0^{(1)} = \pi_0^{(2)} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T$, *and* $P_1 = P_2 = 0.5$. *The corresponding* $A_\alpha^{(1)}, A_\beta^{(1)}, A_\alpha^{(2)}, A_\beta^{(2)}$ *are as follows:*

$$\mathcal{A}_\alpha^{(1)} = \begin{bmatrix} 0 & 0.95 \\ 0 & 0.05 \end{bmatrix}, \mathcal{A}_\beta^{(1)} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix},$$

$$\mathcal{A}_\alpha^{(2)} = \begin{bmatrix} 0 & 0.05 \\ 0 & 0.95 \end{bmatrix}, \mathcal{A}_\beta^{(2)} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}.$$

If the sequence $Y(\ell) = \beta\alpha\beta\alpha$ *is observed, we have* $P_\ell^{(1)} = \sum_{k=1}^{|Q^{(1)}|} \rho_4^{(1)}(k) = 0.05$, *where* $\rho_4^{(1)} = A_\alpha^{(1)} A_\beta^{(1)} A_\alpha^{(1)} A_\beta^{(1)} \pi_0^{(1)}$, *and* $P_\ell^{(2)} = \sum_{k=1}^{|Q^{(2)}|} \rho_4^{(2)}(k) = 0.95$, *where* $\rho_4^{(2)} = A_\alpha^{(2)} A_\beta^{(2)} A_\alpha^{(2)} A_\beta^{(2)} \pi_0^{(2)}$.

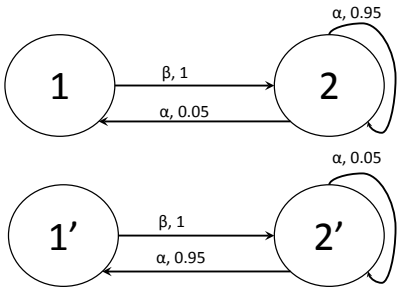

Figure 2.6: $S^{(1)}$ (up) and $S^{(2)}$ (down) in Example 1.

Thus, the probability of error between the two models if this specific sequence is observed is $P(error, Y(\ell)) = 0.025$.

Chapter 3

Fault Diagnosis in Decentralized and Distributed Settings

3.1 Introduction

We explore different state estimation problems in DES in three different settings (centralized/decentralized/modular). In Fig. 3.1 we can see that in all settings, we construct, for each problem, specific automata (*es*) whose function is to perform state estimation. In the centralized approach, we have a monolithic system G, then we construct its state estimator (*es*), based on which we make a decision. In the decentralized setting, we have several local modules of the system G (OM_i), each assigned with a state estimator (es_i), and all state estimators send their local decisions to a coordinator, which decides for the system (G). The distributed setting lacks a coordinator, and allows instead communication between the local estimators.

We consider distributed fault diagnosis in a discrete event system, modeled as a nondeterministic finite automaton and observed at multiple observation sites, each with distinct observation capabilities. The majority of previous work in this setting has focused on *separately* implementing local diagnosers at each observation site, without attempting, at any point, to refine the diagnostic information of these observation sites by exchanging information among them. Instead, these local diagnosers typically rely exclusively on communicating their decision (fault or no fault) to a coordinator that is responsible for making the ultimate diagnosis decision (e.g., co-diagnosability).

Diagnosability has been investigated in decentralized architectures [8, 13, 33, 36,

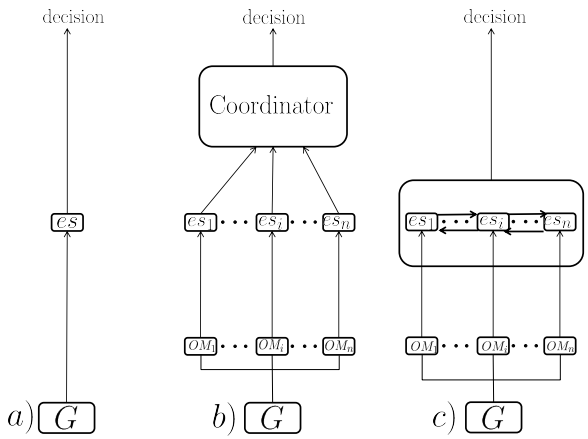

Figure 3.1: a) Centralized, b) decentralized, and c) modular) architectures for state estimation.

48, 51, 52]. In particular, the authors in [13] propose three protocols for coordinated decentralized diagnosis. Protocols 1 and 2 assume unidirectional communication from the local diagnosers to a coordinator, whereas Protocol 3 assumes no communication between them or to any coordinator. Polynomial complexity algorithms for the verification of Protocol 3 and variations of it can be found in [33, 36, 48, 51, 52]. In [52] decentralized diagnosability is studied under a framework of *conditional decisions* issued by the local sites, and polynomial tests are proposed for the verification of equivalent language-based notions of decentralized diagnosability. In [51] the authors propose polynomial algorithms to transform a problem of co-observability to the problem of co-diagnosability under the assumption of dynamic observations. They also propose a polynomial-time algorithm for testing co-diagnosability of the system based on *cluster automata*. The authors of [48] study co-diagnosability under the condition of state-dependence and nondeterminism of partial event observation. The algorithm proposed in [36] is based on constructing a testing automaton that, given a faulty trace, searches for corresponding indistinguishable non-faulty traces at each local site.

In this chapter we study diagnosis using a synchronization-driven intersection-based distributed diagnosis (called Restricted Synchronization Intersection-Based Diagnosis, or RS-IBDD). The RS-IBDD protocol allows local diagnostic information (namely, state estimates and associated normal/fault conditions) to be exchanged and refined among the local sites. The exchange and refinement of the diagnostic information takes place at synchronization points, that are predetermined at each

observation site, by taking the intersection of the state estimates and associated normal/fault conditions provided by the local diagnosers. The fused information is then communicated back to the local diagnosers, which subsequently continue operation based on this refined diagnostic information. In this protocol there is no need for a coordinator, but the diagnostic information is exchanged and refined among the local sites. Furthermore, the presence of communication constraints dictates a more general approach, which, among others, exploits the communication graph that represents the bidirectional communication capability between local sites. The diagnosability of the resulting distributed protocol is shown to be verifiable with polynomial complexity (in the size of the state space of the given nondeterministic finite automaton) and exponential in the number of observation sites (which is not foreseen to be a prohibitive factor in practical applications).

Thus, unlike the majority of previous work that does not allow exchange of diagnostic information among local observation sites, in this chapter, we allow information flow from one local site to another (which occurs when, at the coordinator, we take the intersection of the sets of states estimates and associated normal/fault conditions of all local diagnosers). Among other implications, this means that we do not have a completely decentralized setup any more, but rather a (special case of a) distributed one. Other works that have allowed information to flow from one observation site to another include [16, 47] which focus on modular systems (with a local observer for each module) and explore iterative strategies that involve exchange of diagnostic information and intersection operations for refinement among neighboring local sites.

3.2 Description of RS-IBDD Protocol in the Presence of Communication Constraints

We consider a system $G = (X, \Sigma, \delta, X_0)$ (modeled as an NFA) that is observed by multiple observers, each with its own natural projection map with respect to its set of observable events. More specifically, there are m observation sites, $1, 2, ..., m$, with observation site j having observable events $\Sigma_{o_j} \subseteq \Sigma$. These m observation sites are allowed to exchange information according to the communication links that connect them, which are captured by a *communication graph*, described by an

undirected graph $C = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V} = \{1, 2, ..., m\}$ is the set of vertices or nodes of the graph, that correspond to the local sites, and $\mathcal{E} \subseteq \{\{i, j\} \mid i, j \in \mathcal{V}, i \neq j\}$ is the set of bidirectional edges. In particular, edge $\{i, j\} \in \mathcal{E}$ if communication is possible between observation site i and observation site j. In other words, the *communication graph* $C = \{\mathcal{V}, \mathcal{E}\}$ captures the communication capabilities between different nodes (observation sites). Notice that communication is possible among all local sites or $\mathcal{E} = \{\{i, j\} \mid i, j \in \mathcal{V}, i \neq j\}$ iff the *communication graph* is a complete undirected graph (i.e., there exists an edge between any pair of nodes $i, j \in \mathcal{V}$).

Algorithm 1 RS-IBDD Protocol

Input: Consider the setting described in Section 3.2. A sequence of events $s \in \Sigma^*$ generates a sequence of observations $\omega_j = P_{\Sigma_{o_j}}(s) \in \Sigma_{o_j}^*$ at each observation site j. Since this is a distributed algorithm, we describe what happens from the perspective of observation site j.

1) **Initialization:** Each node $j \in \mathcal{V}$ sets $q_{d_j} = URL_j(X_0)$ and $\omega_j = \varepsilon$.

2) **Runtime Operation:** Following an event $\sigma \in \Sigma$, the following actions are taken by each observation site:

i) $q_{d_j} = \delta_{d,j}(q'_{d_j}, P_{\Sigma_{o_j}}(\sigma))$ where q'_{d_j} is the current state and q_{d_j} is the next state of the local diagnoser Q_{d_j} at observation site j (notice that if $P_{\Sigma_{o_j}}(\sigma) = \varepsilon$, then nothing will change since nothing is observed at the observation site j).

ii) $\omega_j = \omega'_j P_{\Sigma_{o_j}}(\sigma)$ where ω'_j is the sequence of observations seen so far and ω_j is the updated version of it (again nothing happens if σ is not observable at observation site j).

iii) **Synchronization:** If $|\omega_j| = 0 \mod k_j$ (and ω'_j did not satisfy $|\omega'_j| = 0 \mod k_j$), then j initiates a **synchronization under communication constraints** as described in Definition 11. If let $J_s = \{j_1, j_2, ..., j_s\} \subseteq \{1, 2, ..., m\}$ be the observation sites that initiate a synchronization following σ.

3) It waits for the next event σ to occur and repeats (goes back to Step 2).

Now we describe the proposed protocol in Algorithm 1. Having defined the constraints imposed in communication exchanges, we next describe the proposed synchronization strategy for exchanging information among observation sites in Definition 11. We will use the notation $URL_j(X')$ to denote the set of states (along with any fault labels that can be reached in G from states in the set X' following any sequence of events $(\Sigma - \Sigma_{o_j})^*$ that is unobservable at observation site j.

Definition 11. *(Synchronization under communication constraints). Consider a sequence of events* $s \in \Sigma^*$ *that generates the sequence of observations* $\omega_j = P_{\Sigma_{o_j}}(s)$ *at the* j^{th} *observation site. If there was no exchange of information, the diagnostic information at local site j would consist of all possible states that could be reached (given* ω_j*) along with their matching label(s)* $l \in \Delta$*. In other words, the diagnostic information at local site j would be given by the set* $q_{d_j} = \delta_{d_j}(Q_{0,d_j}, \omega_j)$ *of the diagnoser* Q_{d_j} *associated with local site j. Notice that diagnoser* $G_{d_j} = \{Q_{d_j}, \Sigma_{o_j}, \delta_{d_j}, Q_{0,d_j}\}$ *is constructed as in Definition 8 with* $\Sigma_o = \Sigma_{o_j}$*. The synchronization strategy modifies the diagnostic information at each site as follows:*

A priori, *a nonnegative integer constant* k_j *is chosen for each site. The constants* k_1, k_2, *...,* k_m *govern how synchronization among the observation sites is performed. Specifically, local site j initiates the exchange of diagnostic information with the observation sites it can communicate with (i.e., sites in the set* $\mathcal{N}_j = \{i \mid \{i, j\} \in \mathcal{E}\}$*) when it observes* k_j *events, since the last synchronization it initiated. In other words, observation site j initiates a synchronization when* $|\omega_j| = |P_{\Sigma_{o_j}}(\sigma)| \equiv 0 \mod k_j$ *(i.e., the length of* ω_j *is a multiple of* k_j*). Let* $J_s = \{j_1, j_2, ..., j_s\} \subseteq \{1, 2, ..., m\}$ *denote the observation sites that initiate a synchronization following* σ*. Each observation site* $j \in J_s$ *does the following:*

S.i) Downloading phase: Local site $j \in J_s$ downloads *the diagnostic information* S_i *(state estimates along with associated normal/fault conditions) from all neighboring sites (*$i \in \mathcal{N}_j$*). In other words, local site j obtains* $\{S_i \mid i \in \mathcal{N}_j\}$.

S.ii) Refining phase: Local site $j \in J_s$ refines *the diagnostic information by intersecting its local diagnostic estimate with all downloaded diagnostic estimates and taking the unobservable reach of the resulting set of estimates with respect to its own unobservable events; the resulting diagnostic information at site* $j \in J_s$ *is given by*

$$S_j^{(r)} = URL_j(\cap_{i \in \mathcal{N} \cup \{j\}} S_i).$$

S.iii) Sending phase: Local site $j \in J_s$ sends *the refined diagnostic estimate* $S_j^{(r)}$ *to all connected local sites in the set* $\mathcal{N}_j$.

S.iv) All sites take the intersection of the sets of refined estimates they might receive with their own set of estimates, as well as the unobservable reach with respect to their own set of unobservable events. They then continue operation from this refined diagnostic information. In other words, for all observation sites i

$$S_i^{(f)} = URL_i(S_i^{(r)} \cap (\cap_{j \in (\mathcal{N}_j \cap J_s)} S_j^{(r)})).$$

Note that $S_i^{(r)} = S_i$ for nodes in the set $i \in \{1, 2, ..., m\} - J_s$. Moreover, $S_i^{(f)} = S_i^{(r)} = S_i$ for all nodes in the set $\{1, 2, ..., m\} - J_s - \cup_{j \in J_s} N_j$. Also, if there are two or more local sites that simultaneously initiate synchronization, then these three steps (Downloading, Refining, and Sending) run simultaneously for the initiating local sites.

Diagnosis decision: When at least one local site j positively verifies the presence of a fault (or the presence of a specific class of faults) in the system (i.e., $S_i^{(f)}$ contains exclusively states with label F – or exclusively faults from a specific class of faults), then that fault (or fault class) is detected (or identified).

Remark: Note that the above synchronization strategy implies that, following a sequence of events s, the state of observer j will not necessarily be identical to the state observer j would be in following the sequence of observations $\omega_j = P_{\Sigma_{o_j}}(s)$ (i.e., $q_{j,i} = \delta_{d_j}(Q_{0,d_j}, \omega_j)$) because of refinements that might take place due to earlier synchronization exchanges (which could be initiated by either this particular diagnoser or its neighbors).

Example 5. *The overall procedure is graphically illustrated in the example in Figs. 3.2 and 3.3. We have a communication graph $C = \{\mathcal{V}, \mathcal{E}\}$ with four local sites O_1, O_2, O_3, O_4. Suppose that at a particular time instant, the diagnostic estimate before initiation of a synchronization is captured by sets (of state estimates and corresponding normal/failure conditions) S_1, S_2, S_3, S_4, respectively for sites O_1, O_2, O_3, O_4 (note that S_i necessarily satisfies $S_i = URL_i(S_i)$). Suppose the protocol is initiated simultaneously by local sites O_1 and O_3 because the latest event was observed at both O_1 and O_3, and brought their counters respectively to k_1 and k_3. First, each local site downloads the diagnostic estimates from all neighboring local sites. Then, each initiating site refines its diagnostic estimate (e.g., for O_3 this refinement results in $S_3^{(r)} = URL_3(S_2 \cap S_3 \cap S_4)$). Finally, each site sends its refined diagnostic estimate to its neighbors. Note that O_2 receives refined diagnostic estimates from both initiating local sites, which means that overall the diagnostic estimate for that site is the unobservable reach (with respect to the events that are unobservable at site 3) of the intersection of the received diagnostic estimate from $S_1^{(r)}$ and $S_3^{(r)}$.*

Though there are many motivating examples for distributed diagnosis in discrete event systems (e.g., from automated manufacturing), we discuss here a setting that involves joint tasks for vehicles that communicate with each other. Consider, for example, the vehicles O_1, O_2, O_3, O_4 in Fig. 3.4 and assume that their goal is to perform

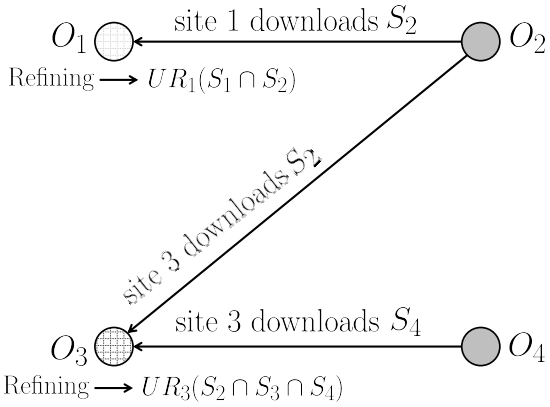

Figure 3.2: Downloading phase and refining phase for the two initiating local sites, O_1 and O_3.

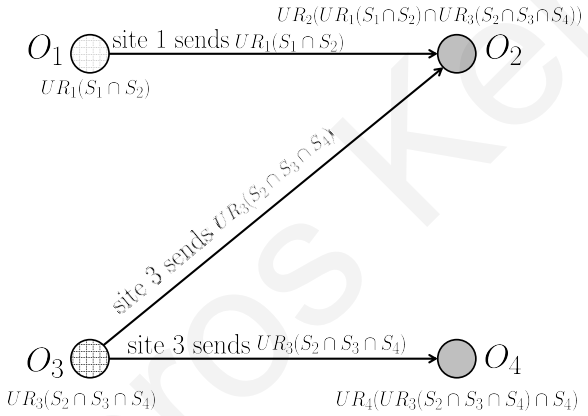

Figure 3.3: Sending phase for the two initiating local sites O_1 and O_3.

a sequence of tasks (events) that is described by the language of a known nondeterministic automaton. We assume that all abnormalities (faults) are also captured in this NFA. The goal is to use the local diagnosers associated with each vehicle (viewed as local observation sites), that presumably observe different subsets of the various events, to diagnose the possible occurrence of faults. In the following example, we illustrate the synchronization strategy under two different communication graphs.

As a motivating example, we are given NFA G in Fig. 2.4 and four observation sites, with locally observable event sets $\Sigma_{o_1} = \{a, b\}$, $\Sigma_{o_2} = \{b, c\}$, $\Sigma_{o_3} = \{b, d\}$, and $\Sigma_{o_4} = \{c, d\}$, under local natural projection maps P_1, P_2, P_3, and P_4, respectively. Suppose the sequence of events $st = \sigma_f dab^n$, $n \geq 1 \in \mathbb{N}$, occurs with $s = \sigma_f$ and $t = dab^n$.

Notice that in case of co-diagnosability, there does not exist a local observer able

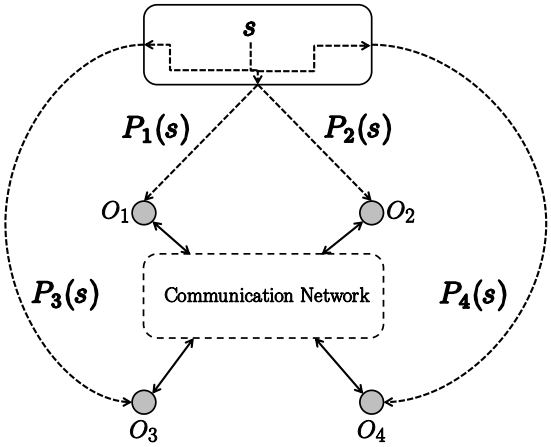

Figure 3.4: The output of the process (dotted lines) is a sequence of events s from an NFA G. Depending on the set of observable events at each local site, each local observer sees a different projection of s, denoted by $P_j(s)$. A communication graph captures the constraints in terms of information exchanges allowed between the four local sites.

to diagnose this fault occurrence on its own. To see this, we can check what happens at each observation site. Considering the first local site with observable event set Σ_{o_1}; it observes $P_1(st) = ab^n$ and, for all n, we can find $t' = ab^n \in P^{-1}[ab^n] \cap L(G)$, such that $\sigma_f \notin t'$. Thus, the first local site cannot decide about the fault occurrence. Using the same reasoning, with all remaining local sites, we see that there does not exist a single local observer able to diagnose the fault on its own. It follows from the definition of co-diagnosability that the system is not co-diagnosable.

In the case of the IBDD protocol in [35], the fault will also not be detected. To see this, we can construct the four local diagnosers (Definition 8) $G_{d,i} = (Q_{d,i}, \Sigma_{o_i}, \delta_{d,i}, Q_{0,i})$ for $i \in \{1, 2, 3, 4\}$ with the corresponding Σ_{o_i}. The initial states are given by $Q_{0,1} = \{0N, 1F, 2F\}$, $Q_{0,2} = \{0N, 1F, 2F, 3F, 4N\}$, $Q_{0,3} = \{0N, 1F, 4N\}$, and $Q_{0,4} = \{0N, 1F, 4N\}$. For $st = \sigma_f dab^n$, we have (for $n \geq 1$) that $\delta_{d,1}(Q_{0,1}, \omega_1) = \delta_{d,2}(Q_{0,2}, \omega_2) = \delta_{d,3}(Q_{0,3}, \omega_3) = \{3F, 4N\}$, and $\delta_{d,4}(Q_{0,2}, \omega_4) = \{2F, 3F, 4N\}$, with $\omega_1 = ab^n$, $\omega_2 = \omega_3 = b^n$, and $\omega_4 = db^n$ (note that $\omega_i = P_{\Sigma_{o_i}}(st)$ for $i = 1, 2, 3, 4$). This means that, $\forall n \in \mathbb{N}$, after the intersection of the four sets of diagnostic information (state estimates along with labels), the coordinator remains confused about the occurrence of the fault. Thus, the system is not IBDD diagnosable (the intersection between state estimates is $\{3F, 4N\}$).

We now consider RS-IBDD under communication graph C_1 in Fig. 3.5 with $k_1 = k_2 = k_4 = 10$, and $k_3 = 1$, being the numbers of events that govern the synchronization

protocol at each local site. We illustrate the execution of the protocol for ambiguous faulty string $st = \sigma_f dab^n$. The third site initiates the synchronization protocol, when it observes the event d, i.e., after the sequence of events $s = \sigma_f d$ is executed. According to the synchronization protocol under the constraints of the given communication graph C_1, the local site 3 downloads the diagnostic estimates from local sites 2 and 4. It refines its diagnostic estimate to

$$URL_3(\delta_{d,3}(Q_{0,3}, P_3(s)) \cap \delta_{d,2}(Q_{0,2}, P_2(s)) \cap \delta_{d,4}(Q_{0,4}, P_4(s))) =$$

$$URL_3(\delta_{d,3}(Q_{0,3}, d) \cap \delta_{d,2}(Q_{0,2}, \varepsilon) \cap \delta_{d,4}(Q_{0,4}, d)) = \{2F, 3F, 4N\}$$

(where $s = \sigma_f d$ and $P_2(s) = \varepsilon$ and $P_3(s) = P_4(s) = d$), and sends this refined diagnostic estimate to local sites 2 and 4. The same procedure is followed for subsequently observed symbols until the last symbol of the observed sequence. The key difference from previous works in decentralized settings is that some (but not neccessarily all) local sites continue their estimation from a refined set of state estimates and associated normal/fault conditions. Specifically, the synchronization initiated at local site 3 after observing d will force local site 2 to state $\{2F, 3F, 4N\}$ (note that local site 4 actually does not benefit from this synchronization step). In this example, the synchronization protocol is not able to diagnose the fault, because it can be shown that after observing $st = \sigma_f dab^n$, for $n = 1$, the local diagnosers enter a cycle, where all local estimates remain $\{3F, 4N\}$. Therefore, any future initiation of the protocol, by any local site, will be unsuccesful into further refining the diagnostic estimates; thus, the approach will be unable to identify the fault.

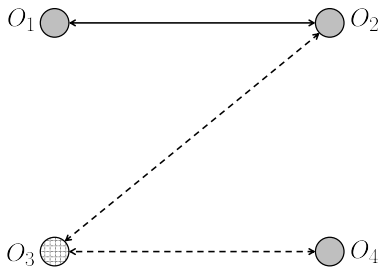

Figure 3.5: Communication graph (C_1), representing communication between local observation sites in Example 2 (the line between nodes O_1 and O_2 represents the edge $\{1, 2\} \in \mathcal{E}$). The initiating local site is 3, and the active links are represented by the dotted lines.

Example 6. *As in Example 2, suppose we are given NFA G in Fig. 2.4 and four observation sites, with locally observable event sets* $\Sigma_{o_1} = \{a, b\}$, $\Sigma_{o_2} = \{b, c\}$, $\Sigma_{o_3} = \{b, d\}$, *and* $\Sigma_{o_4} = \{c, d\}$ *under local projection maps* P_1, P_2, P_3, *and* P_4 *respectively, and* $k_1 = k_2 = k_4 = 10$, *and* $k_3 = 1$. *We assume that the communication graph* C_2 *is the graph in Fig. 3.6. We illustrate our protocol for* $st = \sigma_f dab^n$, *which is the only problematic string from the set of faulty behaviours* ($st = \sigma_f da(b + c)^n$), *because after observing c for the first time. The fault will be diagnosed locally (by local site* 2*). When* $s = \sigma_f d$ *occurs, d is observed at observation sites* 3 *and* 4, *whose local diagnostic information is given by* $\delta_{d,3}(Q_{0,3}, d) = \delta_{d,4}(Q_{0,4}, d) = \{2F, 3F, 4N\}$. *The first and the second diagnoser remain in the initial states* $\delta_{d,1}(Q_{0,1}, P_1(s)) = Q_{0,1} = \{0N, 1F, 2F\}$, $\delta_{d,2}(Q_{0,2}, P_2(s)) = Q_{0,2} = \{0N, 1F, 2F, 3F, 4N\}$ (*d is unobservable at the first and second observation sites, so we have* $P_1(s) = P_2(s) = \varepsilon$*).*

Since $k_3 = 1$, *the third site initiates a synchronization protocol, immediately after observing the event d. According to the synchronization protocol and under the constraints of the communication graph, the following occur: (i) local site* 3 *downloads the diagnostic information (state estimates and matching labels) from local sites* 1 *and* 4*; (ii) subsequently, local sites* 1 *and* 4 *obtain refined information from site* 3*; more specifically, local site* 3 *refines its diagnostic information to* $URL_3(\delta_{d,3}(Q_{0,3}, P_3(s)) \cap \delta_{d,1}(Q_{0,1}, P_1(s)) \cap \delta_{d,4}(Q_{0,4}, P_4(s))) = \{2F\}$, *and sends this refined diagnostic estimate to local sites* 1 *and* 4. *The same procedure is followed for subsequently observed symbols until the last symbol of the observed sequence. In this example, we have already detected the fault, because at least one local site unambiguously diagnoses the fault. More specifically, local sites* 1, 3 *and* 4 *are now in position to diagnose the fault because all of their state estimates are associated with the fault label F. In fact, at later synchronizations that involve other observation sites, this certainty about the occurrence of a fault will propagate to other (and eventually to all) observation sites. Also note, that since the faulty behavior was the only challenging behavior in terms of diagnosis, this discussion has also verified that the system is RS-IBDD diagnosable (under this particular communication graph* C_2 *and constants* k_1, k_2, k_3 *and* k_4*).*

From the above, we conclude that our proposed synchronization-driven intersection-based distributed diagnosis protocol in the presence of communication restrictions (RS-IBDD) diagnoses the fault, at least when $k_3 = 1$ (*i.e., local site* 3 *initiates a synchronization each time it observes an event). It can be shown that RS-IBDD is unsuccessful in diagnosing the fault when* $k_1 = k_2 = k_4 = 10$ *and* $k_3 > 1$*; in such cases, the local diagnosers enter an* F_i*-uncertain cycle, and, no further initiation of the protocol can make the system exit from that cycle.*

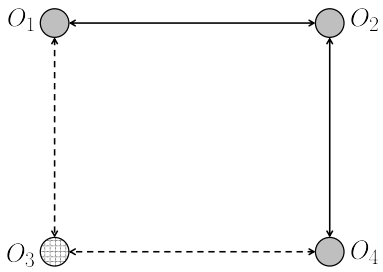

Figure 3.6: Communication graph (C_2), representing communication between local observation sites in Example 3 (the line between nodes O_1 and O_2 represents the edge $\{1,2\} \in \mathcal{E}$). The initiating local site is 3, and the active links are represented by the dotted lines ($\{1,3\}$, $\{1,4\}$).

Now we express formally the notion RS-IBDD diagnosability, which roughly is the diagnosability for a system on which we apply the RS-IBDD protocol.

Definition 12. *(RS-IBDD diagnosability in discrete event systems). Suppose we are given a system $G = (X, \Sigma, \delta, X_0)$ (modeled as an NFA) that is observed by multiple observers, each with its own natural projection map with respect to its set of observable events. More specifically, there are m observation sites, $1, 2, ..., m$, with observation site j having observable events $\Sigma_{o_j} \subseteq \Sigma$. These m observation sites are allowed to exchange information according to the communication links that connect them, which are captured by a* communication graph*, described by an undirected graph $C = \{\mathcal{V}, \mathcal{E}\}$. For a given fault event σ_f, $\sigma_f \in \Sigma_{uo}$ (where $\Sigma_{uo} \equiv \Sigma - \Sigma_o$ where $\Sigma_o = \cup_{\forall i} \Sigma_{o_i}$), let $\Psi(\sigma_f) = \{s = t\sigma_f \in L(G) : \{t \in (\Sigma - \sigma_f)^*\}\}$. The live, prefix-closed language $L(G)$ is RS-IBDD diagnosable with respect to fault σ_f, if after applying the RS-IBDD protocol, the following holds:*

$$(\exists N \in \mathbb{N})(\forall s \in \Psi(\sigma_f))(\forall t \in L/s : |t| \geq N)(\exists j \in \{1, 2, ..., m\})[D_j^{RS-IBDD}(st) = 1]$$

where the diagnosability function $D_j^{RS-IBDD} : \Sigma^* \rightarrow \{0, 1\}$ is referred to the ability of the local site j of locally diagnosing (or not) the fault, when all observation sites follow the RS-IBDD protocol. Applying the RS-IBDD protocol, we define the local state estimate, for a local site j, following an event sequence[1] s, as $RS_j(s)$. The formal definition of $D_j^{RS-IBDD}$ follows:

[1]In the next section, we introduce the constructions of System Diagnoser and System Verifier, which can be used to reconstruct the $RS_j(s)$ for an event sequence s.

$$D_j^{RS-IBDD}(st) = \begin{cases} 1, & \text{if } \{\forall (x,l) \in RS_j(st)\} \Rightarrow \{l = \{F\}\}, \\ 0, & \text{otherwise.} \end{cases}$$

In other words, RS-IBDD diagnosability needs at least one local site, able to distinguish the fault for all continuations for the given faulty behaviour of a system (this local site could be different, for different continuations).

Remark: note that $RS_j(st)$ (and thus $D_j^{RS-IBDD}(st)$) are functions of the sets of observable events at each observation site as well as the constants $k_1, k_2, ..., k_m$ that have govern the synchronization protocol.

3.3 Verification of RS-IBDD Diagnosability using a Synchronized Product of Local Diagnosers

In this section we describe the verification of diagnosability for the RS-IBDD protocol. The main steps of the verification process are described below. (i) Build each enhanced local diagnoser with side information that tracks how many events the corresponding local site has observed so far since the last synchronization it initiated (Definition 13). (ii) Take the parallel product of all enhanced local diagnosers (Definition 14). (iii) Enforce diagnostic information refinement (in terms of the downloading, intersection and refinement operations) whenever an event is observed that makes the counter modulo k_j at observation site j zero. (iv) Take the product of the system with the parallel product of all diagnosers to eliminate behavior in the parallel product of diagnosers that is not actually allowed by the system (Definition 15). (v) Check for interminate cycles in the resulting structure. (In general, the existence of an F_i indeterminate cycle indicates the presence of behavior that does not allow the detection of fault F_i.)

We are given an NFA $G = (X, \Sigma, \delta, X_0)$ and m local sites, each with set of locally observable events Σ_{o_j}, $\Sigma_{o_j} \subseteq \Sigma$, under a local natural projection map P_j, $j = 1, 2, \ldots, m$. Furthermore, observation site j is associated with a positive integer k_j that is used to govern its synchronization operation. We are also given a *communication graph* $C = \{\mathcal{V}, \mathcal{E}\}$ that captures the bidirectional communication capabilities between the local sites $\mathcal{V} = \{1, 2, ..., m\}$ via the edges in the set $\mathcal{E}$.

Definition 13. *(Local diagnosers with synchronized events). We construct the local diagnoser* $D_j = \{Q_{d,j}, \Sigma_{o_j}, \delta_{d,j}, Q_{0,j}\}$ *according to Definition 8. The local diagnoser with synchronized events is given by* $D_{sj} = \{Q_{s,j}, \Sigma, \delta_{s,j}, Q_{0,sj}\}$. *It has states that are pairs of the form* $(q_{d,j}, z_j) \in Q_{s,j} = Q_{d,j} \times \{0, 1, ..., k_j - 1\}$, *where* $q_{d,j} \in Q_{d,j} \equiv 2^{X \times \Delta}$ *and* z_j *is an integer in* $\{0, 1, 2, \ldots, k_j - 1\}$. *The initial state is* $Q_{0,sj} = \{(q_{d,j}, 0) \mid q_{d,j} \in Q_{0,j}\}$. *The state transition function, for* $\sigma \in \Sigma$, *is defined below*

$$\delta_{s,j}((q_{d,j}, z_j), \sigma) = \begin{cases} (\delta_{d,j}(q_{d,j}, \sigma), (z_j + 1) \pmod{k_j}), & \sigma \in \Sigma_{o_j}, \\ (q_{d,j}, z_j), & \text{otherwise,} \end{cases}$$

where no change takes place if the event that occurs is unobservable to observation site j.

Definition 14. *(Synchronized Parallel Product of Local Diagnosers* $(D_{\|})$*). We construct the local diagnosers* $D_{sj} = \{Q_{s,j}, \Sigma, \delta_{s,j}, Q_{0,sj}\}$ *according to Definition 13 and obtain the synchronized parallel product of diagnosers* $D_{\|} = \{Q_{\|}, \Sigma, \delta_{\|}, Q_{0\|}\}$ *as a deterministic finite automaton, where*

(i) The state space $Q_{\|} \subseteq \prod_{j \in \{1,...m\}} Q_{s,j}$ *contains states of the form*

$$q_{\|} = ((q_{d,1}, z_1), \ldots, (q_{d,m}, z_m)) \equiv (q_{\|,d}, z_{\|}),$$

where $q_{\|,d} = (q_{d,1}, \ldots, q_{d,m})$ *and* $z_{\|} = (z_1 (\text{mod } k_1), \ldots, z_m (\text{mod } k_m))$.

(ii) When $\delta_{s,j}((q_{d,j}, z_j), \sigma) = (q'_{d,j}, z'_j)$, *we define*

$$sync_j((q_{d,j}, z_j), \sigma) = \begin{cases} 1, & \text{if } \{z'_j = 0\} \wedge \{z'_j \neq z_j\}, \\ 0, & \text{otherwise.} \end{cases}$$

(iii) The transition function is defined as

$$\delta_{\|}(q_{\|}, \sigma) = (\delta_{\cap}(q_{\|,d}, \sigma), (z_{\|} + 1) \pmod{k_{\|}})$$

where

i) $z_{\|} + 1 (\text{mod } k_{\|}) = (z_1 + f_1(\sigma) (\text{mod } k_1), \ldots, z_m + f_m(\sigma) (\text{mod } k_m))$, *where* $f_i(\sigma)$ *is* 1, *when* $\sigma \in \Sigma_{o_i}$, *and* 0 *otherwise.*

ii) $\delta_{\cap}(q_{\|,d}, \sigma) = (SE_1, \cdots, SE_m)$, *where*[2] $SE_i = URL_i(\cap_{\{k \in \mathcal{N}_i\} \wedge \{sync_k((q_{d,k}, z_k), \sigma) = 1\}} DRL_k \cap DRL_i)$ *with*

$$DRL_j = \begin{cases} URL_j(\cap_{i \in \mathcal{N}_j \cup \{j\}} (\delta_{q_{d,i}}, \sigma)), \text{ if } sync_j((q_{d,j}, z_j), \sigma) = 1, \\ \delta_{d,j}(q_{d,j}, \sigma), \text{otherwise,} \end{cases}$$

[2]We use the notation URL_j for the unobservable reach for observation site j.

Remark: Note that some of the states of the parallel product of local diagnosers may have been unreachable from the initial state in the local diagnosers without synchronized events (then become reachable with the refinement in state estimates that occurs due to the intersection operation). Also note that after each intersection operation of local state estimates, we need to take the unobservable reach (*URL*) with respect to that observation site.

Definition 15. *(RS-IBDD diagnoser (D_p)). We construct the parallel product of local diagnosers with synchronized events $D_{\|} = \{Q_{\|}, \Sigma, \delta_{\|}, Q_{0\|}\}$. The system diagnoser $D_p = \{Q_p, \Sigma, \delta_p, Q_{0,p}\}$ is a (generally nondeterministic) finite automaton that has states $Q_p \equiv X \times Q_{\|}$, initial states $Q_{0,p} \equiv X_0 \times Q_{0\|}$ and transition function $\delta_p(q_p, \sigma)$ for $q_p = (x_i, q_{\|})$ and $\sigma \in \Sigma$ defined as*

$$\delta_p(q_p, \sigma) = \{(x, \delta_{\|}(q_{\|}, \sigma)) \mid x \in \delta(x_i, \sigma)\}.$$

Remark: The system diagnoser D_p eliminates from the synchronized parallel product of local diagnosers $D_{\|}$ behavior that cannot be possibly generated from the underlying system G. Note that unobservable events (i.e., events in $\Sigma - \cup_{i=1}^{m} \Sigma_{o_i}$ that cannot be observed at any site) are included in $D_{\|}$ but appear as self-loops. Unobservable events do not necessarily appear as self loops in D_p because they generally cause changes in the system state.

Definition 16. *State $q_p = \{x_i, \{(q_{d,1}, ..., q_{d,m}), (z_1, ..., z_m)\}\} \in Q_p \equiv X \times Q_{\|}$ is RS-IBDD F_i-uncertain if its $q_{\|}$ component includes, for each local diagnoser j, subsets of the form $\{(x_j, l), (x'_j, l')\} \subseteq q_{d,j}$ and $F_i \in l$ but $F_i \notin l'$.*

We continue from Example 3. The local diagnosers for the system in Fig. 1 are illustrated in Fig. 3.7 and a part of the system diagnoser is illustrated in Fig. 3.8. Suppose that the communication graph describing the allowable communication exchanges between observation sites is C_2. Consider again the sequence $\sigma_f dab^n$. It should be clear from the four local diagnosers in Fig. 3.7 that none of them is in position to detect the fault; however, as seen earlier, if the four local diagnosers follow the RS-IBDD protocol, they will be able to detect the fault. In the construction of the RS-IBDD diagnoser, when σ_f occurs, nothing changes (apart from the system state) because σ_f is unobservable to all observation sites. When d occurs, this is observable to observation sites 3 and 4, and causes a change in the states of those two diagnosers. In fact, because $k_3 = 1$, the occurrence of d also causes a synchronization

to be initiated by observation site 3. Eventually, sites 1, 3, and 4 refine their estimates following this synchronization operation. Notice that, at this point, the fault has been diagnosed at observation sites 1, 3, and 4. Also note that the resulting state is a new state, not previously reachable in any of the local diagnosers at observation sites 1, 3, and 4.

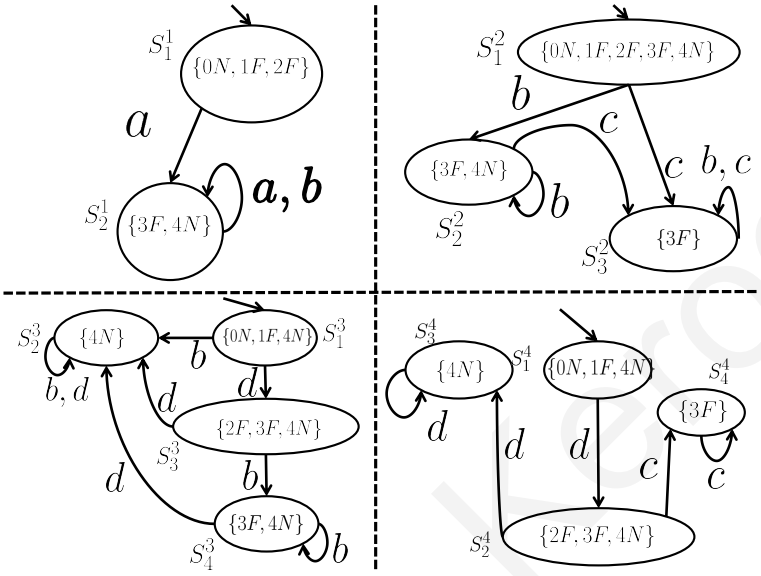

Figure 3.7: Local diagnosers for the local sites O_1, O_2, O_3, O_4 (with $\Sigma_{o_1} = \{a, b\}$, $\Sigma_{o_2} = \{b, c\}$ $\Sigma_{o_3} = \{b, d\}$ $\Sigma_{o_4} = \{c, d\}$) for the system in Fig. 2.4. For conciseness, the figure only shows states that are reachable from the initial state of each diagnoser.

We are particularly interested in RS-IBDD F_i-uncertain states that exist within cycles of the *system diagnoser*. Such cycles are called *RS-IBDD F_i-confused cycles*.

Theorem 3. *Suppose we are given a system $G = (X, \Sigma, \delta, X_0)$ (modeled as an NFA) that is observed by multiple observers, each with its own natural projection map with respect to its set of observable events. More specifically, there are m observation sites, $1, 2, ..., m$, with observation site j having observable events $\Sigma_{o_j} \subseteq \Sigma$. These m observation sites are allowed to exchange information according to the communication links that connect them, which are captured by a* communication graph*, described by an undirected graph $C = \{\mathcal{V}, \mathcal{E}\}$. For a given fault event σ_f, $\sigma_f \in \Sigma_{uo}$ (where $\Sigma_{uo} \equiv \Sigma - \Sigma_o$ with $\Sigma_o = \cup_{\forall i}\Sigma_{o_i}$), let $\Psi(\sigma_f) = \{s = t\sigma_f \in L(G) : \{t \in (\Sigma - \sigma_f)^*\}\}$. The live, prefix-closed language $L(G)$ is RS-IBDD diagnosable with respect to fault σ_f, iff in the RS-IBDD diagnoser, there are no reachable RS-IBDD F_i–uncertain states that exist within RS-IBDD F_i–confused cycles.*[3]

[3]As in the case of centralized diagnosis, we also need to verify that these F_i–confused cycles can indeed be executed after the occurrence of a fault [42].

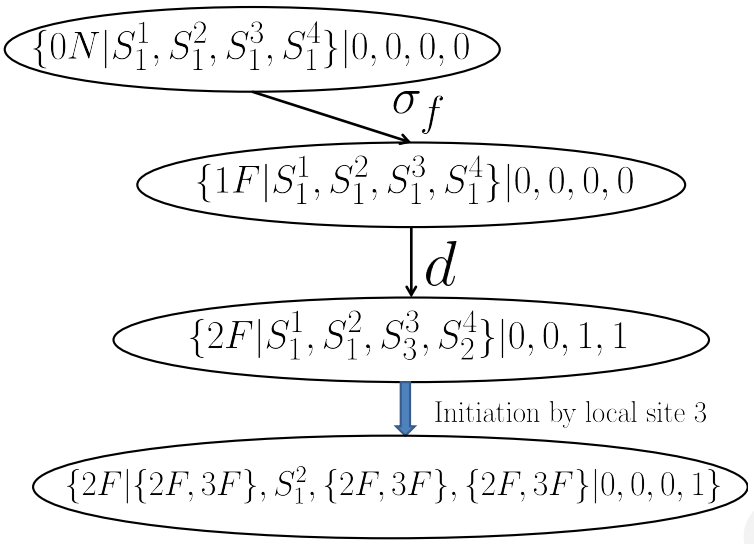

Figure 3.8: Part of the system diagnoser in Example 3. After the execution of $s = \sigma_f d$, local site 3 initiates the synchronization protocol, which refines the state estimates of all local sites, as shown in the figure.

Proof. Let us assume that there does not exist a reachable $F_{\|}$–confused cycle in the RS-IBDD diagnoser. This means that there does not exist an $F_{\|}$–uncertain state in any cycle in the Parallel Product of Local Diagnosers. If this is true, then for any sequence of events s that contains a fault there exists some (at least one) local site i, such that after executing the RS-IBDD protocol observation site i does not remain confused and it is able to diagnose the fault. This means that the system is RS-IBDD diagnosable.

If there exists at least one reachable $F_{\|}$–confused cycle, then there exists at least one $F_{\|}$–uncertain state in this cycle. Let t be a sequence of events that gets us to this F-uncertain state after the occurrence of σ_f, and let s be a sequence of events that gets us from this F-uncertain state back to it. Then, of any integer $n, n >= 0$, the string ts^n, reaches this F-uncertain state, i.e., for an arbitrarily long sequence of events following the fault, all local diagnosers are kept confused. Thus, the system is not RS-IBDD diagnosable. □

Remark: The complexity of the proposed verification method requires exponential complexity in the number of states of the given automaton G and exponential complexity in the number of observation sites.

3.4 Verification of RS-IBDD Diagnosability using a Synchronized Product of Local Verifiers

In this section, we provide an alternative verification method that relies on a product of verifiers. The method proposed in this section has complexity polynomial in the number of states and exponential in the number of observation sites.

The main steps of the verification process are described below.
(i) Build, for each observation site, a local verifier (Definition 17), and enhance it with side information that tracks how many events it has observed so far since the last synchronization it initiated (Definition 18).
(ii) Take the parallel product of all local verifiers (Definition 19).
(iii) Enforce diagnostic information refinement whenever an event is observed that makes the counter modulo k_j at observation site j zero; this is accomplished via appropriate intersection and unobservable reach operations of the available diagnostic information at the observation sites that are involved in this particular synchronization step.

We are given an NFA $G = (X, \Sigma, \delta, X_0)$ and m local sites $j = 1, 2, \ldots, m$. Local site j has locally observable events Σ_{o_j}, $\Sigma_{o_j} \subseteq \Sigma$, under a local natural projection map P_j. We assume that observation site j is associated with a positive integer k_j that is used to govern its synchronization operation. We are also given a *communication graph* $C = \{\mathcal{V}, \mathcal{E}\}$ that captures the bidirectional communication capabilities between the local sites $\mathcal{V} = \{1, 2, ..., m\}$ via the edges in the set $\mathcal{E}$. The RS-IBDD protocol is used to diagnose the occurrence of faults in the set Σ_f where (without loss of generality) $\Sigma_f \subseteq \cup_{i=1}^{m} \Sigma_{o,i}$. Our goal is to verify diagnosability under the RS-IBDD protocol.

Definition 17. *(Local Verifier) The local verifier at observation site j, denoted by $V_j = (Q_j, \Sigma_{o_j}, \delta_j, Q_{0,j})$, is a non-deterministic finite automaton constructed from the given non-deterministic system G as follows:*

1. $Q_j = (X \times \Delta) \times (X \times \Delta)$ is the set of states, where $\Delta = \{N, F\}$;

2. Σ_{o_j} is the set of observable events at site j;

3. $Q_{0,j}$ is the initial state given by $Q_{0,j} = \{((x_0, l_0), (x_0', l_0'))|\ (x_0, l_0), (x_0', l_0') \in URL_j(X_0)\}$where URL_j was defined in Definition 8 (all initial states are labeled as $\{N\}$ states and the unobservable reach with labels is taken with respect to to the set of observable events Σ_{o_j} at local site j). Note that (x_i, l_i) for notational convenience is also written as $x_i l_i$; thus, $((x_i, l_i), (x_i', l_i'))$

is also written as $(x_i l_i, x'_i l'_i)$;

4. δ_j *is the transition rule defined as* $\delta_j((x_i l_i, x'_i l'_i), \sigma) =$

$$
\begin{aligned}
&= \{(\{x_{i+1} l_{i+1}, x'_{i+1} l'_{i+1})\} \mid \exists s_1, s_2 \in \Sigma^*, \\
&P_j(s_1) = P_j(s_2) = \sigma, x_{i+1} \in \delta(x_i, s_1), x'_{i+1} \in \delta(x'_i, s_2), \\
&l_{i+1} = f(l_i, s_1), l'_{i+1} = f(l'_i, s_2)\},
\end{aligned}
$$

where, for $l_i \in \Delta = \{N, F\}$ *and* s_h, $h = 1, 2$, *the label function* $f : \Delta \times \Sigma^* \to \Delta$ *is defined as*

$$
f(l_i, s_h) = \begin{cases} F, & l_i = F, \\ F, & l_i = N \text{ and } \exists f \in \Sigma_f, \sigma_f \in s_h, \\ N, & l_i = N \text{ and } \forall f \in \Sigma_f, \sigma_f \notin s_h. \end{cases}
$$

Note that V_j is a nondeterministic finite automaton and its number of states is at most $(2|X|)^2$, where $|X|$ is the number of states of the given system G. More generally, in the case of multiple fault classes it will be $(|X| \times 2^{|L|})^2$ where L is the number of fault labels (each corresponding to a fault class).

A part of the local verifier for local site 1 with $\Sigma_{o_1} = \{a, b\}$ for the system in Fig. 1 is illustrated at the top of Fig. 3.9.

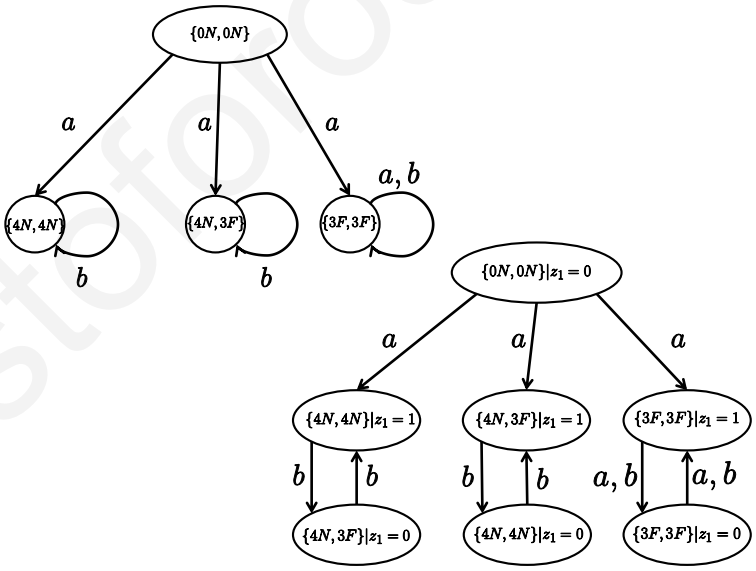

Figure 3.9: Part of the local verifier (up) and part of the local verifier with synchronized events (bottom) for the first local site O_1 (with $\Sigma_{o_1} = \{a, b\}$) for the system in Fig. 2.4.

Definition 18. *(Local verifiers with synchronized events). We construct the local verifier with synchronized events* $V_j = (Q_j, \Sigma_{o_j}, \delta_j, Q_{0,j})$ *according to Definition 17. The local verifier*

with synchronized events is given by $V_{sj} = \{Q_{s,j}, \Sigma, \delta_{s,j}, Q_{0,sj}\}$. *It has states that are pairs of the form* $(q_j, z_j) \in Q_{s,j} = Q_j \times \{0, 1, ..., k_j - 1\}$, *where* z_j *is an integer in* $\{0, 1, 2, \ldots, k_j - 1\}$. *The set of initial states is* $Q_{0,sj} = \{(q_{d,j}, 0) \mid q_j \in Q_{0,j}\}$. *The state transition function, for* $\sigma \in \Sigma$, *is defined below*

$$\delta_{s,j}((q_j, z_j), \sigma) = \begin{cases} (\delta_j(q_j, \sigma), (z_j + 1) \pmod{k_j}), & \sigma \in \Sigma_{o_j}, \\ (q_j, z_j), & \textit{otherwise.} \end{cases}$$

Definition 19. *(Synchronized Parallel Product of Local Verifiers* $(V_{\|})$*). We construct the local verifiers* $V_{sj} = \{Q_{s,j}, \Sigma, \delta_{s,j}, Q_{0,sj}\}$ *according to Definition 18 and obtain the synchronized parallel product of verifiers* $V_{\|} = \{Q_{\|}, \Sigma, \delta_{\|}, Q_{0\|}\}$ *as a nondeterministic finite automaton, where*

(i) The state space $Q_{\|} \subseteq \prod_{j \in \{1,...m\}} Q_{s,j}$ *has states of the form*

$$q_{\|} = ((q_1, z_1), \ldots, (q_m, z_m)) \equiv (q_{\|,d}, z_{\|}),$$

where $q_{\|,d} = (q_1, \ldots, q_m)$ *and* $z_{\|} = (z_1, \ldots, z_m)$.

(ii) When $\delta_{s,j}((q_j, z_j), \sigma) = (q'_j, z'_j)$, *we define*

$$sync_j((q_j, z_j), \sigma) = \begin{cases} 1, & \textit{if } \{z'_j = 0\} \wedge \{z'_j \neq z_j\}, \\ 0, & \textit{otherwise.} \end{cases}$$

(iii) The transition function is defined as

$$\delta_{\|}(q_{\|}, \sigma) = (\delta_{\cap}(q_{\|,d}, \sigma), (z_{\|} + 1) \pmod{k_{\|}})$$

where

i) $z_{\|} + 1 (\text{mod } k_{\|}) = (z_1 + f_1(\sigma) (\text{mod } k_1), \ldots, z_m + f_m(\sigma) (\text{mod } k_m))$, *where* $f_i(\sigma) = 1$, *when* $\sigma \in \Sigma_{o_i}$, *and* $f_i(\sigma) = 0$ *otherwise.*

ii) if we let

$$DR_j = \begin{cases} \delta_j(q_j, \sigma), \textit{ if } sync_j((q_j, z_j), \sigma) = 0, \\ URL_j(\cap_{i \in \mathcal{N}_j \cup \{j\}}(\delta_i(q_i, \sigma)), \textit{otherwise}, \end{cases}$$

we can define the transition function as ii) $\delta_{\cap}(q_{\|,d}, \sigma) = \{(q_1, ..., q_m) \mid q_i = \{((x, l), (x', l'))\} \subseteq SE_i$, *where*

$$SE_i = URL_i((\cap_{\{k \in \mathcal{N}_i\} \wedge \{sync_k((q_k, z_k), \sigma) = 1\}} DR_k) \cap DR_i)\}.$$

This conditional intersection describes mathematically the outcome in terms of pairs of state/fault estimates that are possible after the completion of the steps from the synchronization protocol i) downloading, ii) refining (the sets of states DR_j), and iii) sending (the sets of states SE_j).

Lemma 1. *Suppose we are given a system $G = (X, \Sigma, \delta, X_0)$ (modeled as an NFA) that is observed by multiple observers, each with its own natural projection map with respect to its set of observable events. More specifically, there are m observation sites, $1, 2, ..., m$, with observation site j having observable events $\Sigma_{o_j} \subseteq \Sigma$. These m observation sites are allowed to exchange information according to the communication links that connect them, which are captured by a* communication graph*, described by an undirected graph $C = \{\mathcal{V}, \mathcal{E}\}$. Following the execution of the RS-IBDD protocol, where $RS_i(s) = S_i$, we have the following property: we can reconstruct the state estimation for this sequence (state estimations for all local sites), in the Synchronized Parallel Product of Local Verifiers*

Proof. For simplicity we represent a state as x, which combines the information of the state, label and synchronization variable (for a local site). The specific actions that are part of the RS-IBDD protocol and do not prohibit us from using the Synchronized Parallel Product of Local Verifiers are:

i) **the intersection for local state estimations**,

ii) **the unobservable reach for local state estimations**.

i) Consider an observation sequence s in the Synchronized Parallel Product of Local Verifiers without applying RS-IBDD related actions (intersection and unobservable reach), if the set of the local state estimates for s for local site $i \in \{1, 2, ..., m\}$ is $S_i = \{x_1, x_2, ..., x_k\}$, then each element of this set is present in the verifier as states of the form $q_i = \{x_i, x_j\} \subseteq S_i$. In the case, of RS-IBDD protocol, we need to argue that the construction is consistent with the downloading, refining, and sending phases of the protocol that occurs at various points during the execution of s. In downloading and refining we need to capture the intersection for two or more sets of estimates.

For simplicity let the two sets of state estimates (for two local sites 1 and 2) be $S_1 = \{x_1^1, x_2^1, ..., x_k^1\}$, $S_2 = \{x_1^2, x_2^2, ..., x_k^2\}$. The verifier of the initiating node (with state estimate S_1) will have states $q_1 = \{x_m^1, x_n^1\} \subseteq S_1$ and the verifier for the other local site will have states of the form $q_2 = \{x_m^2, x_n^2\} \subseteq S_2$. We define the function $f(q_1, q_2) = \{x_m, x_n\}$ if $q_1 = q_2 = \{x_m, x_n\}$ and $\emptyset$, otherwise. It is easy to argue that $S = S_1 \cap S_2 = \cup_{\forall q_1, q_2} f(q_1, q_2)$. In this case, following the observation sequence s in the

Synchronized Parallel Product of Local Verifiers, we are able to reconstruct the local state estimate for any s, for any local site, after any intersection with any other local site.

ii) Another operation in the RS-IBDD protocol is the operation of unobservable reach of a set of local state estimates $UR(S)$ for any observation sequence s. The unobservable reach will be given again as $\cup_{\forall q \in S}(UR(q))$. This means that we can reconstruct the total unobservable reach from the unobservable reaches that are computed into a verifier (Synchronized Parallel Product of Local Verifiers), from all subsets $q = \{x_1, x_2\}$ of the original set S. Finally, we can reconstruct any state estimate after the execution of the RS-IBDD protocol via the Parallel Product of Local Verifiers, given that we count also the exact number of event occurrences at each observation site, which is the case in our construction. The proof is completed.

□

Definition 20. *($F_{\|}$–uncertain states and $F_{\|}$–confused cycles). A state $q_{\|} = ((q_1, z_1), ..., (q_m, z_m)) \in Q_{\|}$ of the synchronized product of local verifiers is called $F_{\|}$-uncertain if $\{\forall q_i = ((x_i, l_i), (x_i', l_i'))\} \Rightarrow \{l_i \neq l_i'\}$. $F_{\|}$–confused cycle is a cycle in the Parallel Product of Local Verifiers that includes at least one $F_{\|}$–uncertain state (this is equivalent to saying that all states in the cycle are $F_{\|}$–uncertain states).*

Theorem 4. *(RS-IBDD Verification: Necessary and sufficient conditions). We are given an NFA $G = (X, \Sigma, \delta, X_0)$ and m local sites, each with set of locally observable events Σ_{o_j}, $\Sigma_{o_j} \subseteq \Sigma$, under a local natural projection map $P_{\Sigma_{o_j}}$, $j = 1, 2, \ldots, m$. Furthermore, we assume that each observation site is associated with a positive integer k_j that is used to govern its synchronization operation. We are also given a* communication graph *$C = \{\mathcal{V}, \mathcal{E}\}$ that captures the bidirectional communication capabilities between the local sites $\mathcal{V} = \{1, 2, ..., m\}$ via the edges in the set $\mathcal{E}$. The system is RS-IBDD diagnosable with respect to a set of faults $\Sigma_f \subseteq \cup_{i=1}^{m} \Sigma_{o_i}$ iff, in the Synchronized Parallel Product of Local Verifiers, there are no $F_{\|}$–uncertain states that exist within $F_{\|}$–confused cycles.*

Proof. Let us assume that there does not exist an $F_{\|}$–confused cycle. This means that there does not exist an $F_{\|}$–uncertain state in any cycle in the Parallel Product of Local Verifiers. If this is true, then for any $q_{\|,d} = ((q_{\|}, z_{\|})$, with $q_{\|} = (q_1, q_2, ..., q_m)$, with $q_i = (x_i, l_i), (x_i', l_i')$, for any local site i, there are no two sequences s_1, s_2, with $P_i(s_1) = P_i(s_2)$, $\forall i \in \{1, ..., m\}$, such that after executing RS-IBDD observation site is

confused the fault occurred in exactly one of these sequences. This means that the system is RS-IBDD diagnosable.

If there exists at least one $F_{\|}$–confused cycle, then there exists at least one $F_{\|}$–uncertain state in this cycle, $q_{\|,d} = (q_{\|}, z_{\|})$, with $q_{\|} = (q_1, q_2, ..., q_m)$, with $q_i = ((x_1, l_i), (x_i', l_i'))$, for any local site i. There are at least two sequences s_1^i, s_2^i, with $|s_1^i|, |s_2^i| > |X^2|$, for any local site i, with $P_i(s_1^i) = P_i(s_2^i) = \omega_i$, that create the uncertainty in i^{th} component of the $F_{\|}$–uncertain state. If we take any repetition of these sequences for any N_0, $(s_1^i)^{N_0}$, $(s_2^i)^{N_0}$, then the uncertainty will remain. This means that $(\forall N_0 \in \mathbb{N})$, we can find $N' > N_0$, and sequences of length N', that cause the presence of a $F_{\|}$–confused cycle. Thus, all diagnosers are kept confused and we cannot be certain about the occurrence of the fault if we apply the RS-IBDD protocol. The system is not RS-IBDD diagnosable.

□

Remark: The complexity of the proposed verification method requires polynomial complexity in the number of states and exponential complexity in the number of observation sites.

Chapter 4

Detectability in Discrete Event Systems

4.1 Introduction

In this chapter we are interested in exploring state estimation techniques in stochastic discrete event systems (SDES) that can be modeled by probabilistic finite automata (PFAs) under particular observation models. The authors of [45] and [46] introduced notions of detectability in nondeterministic and stochastic settings respectively. In the approach for detectability in nondeterministic finite automata in [45], the problematic system behaviour corresponds to sequences of observations that do not lead to exact state estimation (i.e., they do not lead to perfect state estimation with no uncertainty). In the approach for detectability in PFA's in [46], the problematic behaviour is associated with sequences of observations that do not allow us to estimate the exact state with increasing certainty. More specifically, the notion of stochastic detectability in [46] takes into account all possible observation sequences (infinite sequences) and declares the system *not* stochastically detectable when such problematic sequences are present.

The major contribution of this chapter is the introduction and verification of the notions of A-detectability and AA-detectability. Specifically, we provide necessary and sufficient conditions for A-detectability, polynomially verifiable necessary and sufficient conditions for AA-detectability, and a proof that A-detectability is a PSPACE-hard problem.

The chapter is organized as follows: in Section 2 we revisit notation on proba-

bilistic finite automata), languages and Markov chains. In Section 3 we repeat the notion of stochastic detectability intoduced in [46]. In Section 4 we introduce the notion of A-detectability and its associated necessary and sufficient conditions. In Section 5 we establish that A-detectability and A-diagnosability are PSPACE-hard. In Section 6 we introduce the notion of AA-detectability and its associated necessary and sufficient conditions. During this development of the material we also provide several examples.

4.2 Notation and Background

Definition 21. *(Probabilistic Finite Automaton (PFA)). A stochastic discrete event system (SDES) is modeled in this paper as a probabilistic finite automaton (PFA)* $H = (X, \Sigma, p, \pi_0)$*, where* $X = \{x_1, x_2, \ldots, x_{|X|}\}$ *is the set of states (also denoted for simplicity as* $X = \{1, 2, 3, ..., |X|\}$*),* Σ *is the set of events,* π_0 *is the initial-state probability distribution vector, and* $p(i', \sigma|i)$ *is the state transition probability defined for* $i, i' \in X$*, and* $\sigma \in \Sigma$*, as the conditional probability that event* σ *occurs and the system transitions to state* i' *given that the system is in state* i*.*

We can assign a probability to each trace in Σ^* with the interpretation that this value determines the probability of occurrence of this trace: if $\Pr(i', s)$ denotes the probability that s is executed in the system and the end state of the system is state i', then we can define for $\sigma \in \Sigma, s \in \Sigma^*$,

$$\left.\begin{aligned} \Pr(i, \epsilon) &= \pi_0(i) \\ \Pr(i, s\sigma) &= \sum_{i' \in X} p(i, \sigma|i') \Pr(i', s) \\ \Pr(s\sigma) &= \sum_{i \in X} \Pr(i, s\sigma) \end{aligned}\right\} \tag{4.1}$$

Definition 22. *(Probability of an observation sequence* ω*). Suppose we are given a PFA* $H = (X, \Sigma, p, \pi_0)$*, with* $\Sigma_{obs} \subseteq \Sigma$ *being the set of observable events with respect to the natural projection map P. For any observation sequence* $\omega = \omega_1\omega_2...\omega_n \in \Sigma^*_{obs}$*, of length n, the state probability of state* $i \in X$ *is*

$$\pi_\omega(i) = \sum_{s \in \Sigma^*: (P(s) = \omega) \wedge (\nexists t \in \bar{s}: P(t) = \omega)} \Pr(i, s),$$

where $\pi_\omega(i)$ *is the probability of occurrence of observation sequence* ω *leading to state* $i \in X$*.*

The probability of sequence ω is

$$\pi(\omega) = \sum_{i \in X} \pi_\omega(i).$$

More generallly, for $X' \subseteq X$,

$$\pi_\omega(X') = \sum_{i \in X'} \pi_\omega(i).$$

Note that if one of two strings s and t (with $P(s) = P(t) = \omega$) is a prefix of the other (say $t \in \bar{s}$), then to obtain the probability π_ω we only include the probability of the prefix string.

An example of a PFA can be seen on the left of Fig. 4.1. When $p(i', \sigma \mid i) = 0$, state i' is not reachable from state i via event σ (in the diagram representing the given PFA, we do not draw such transitions). Clearly, we have $\sum_{i' \in X} \sum_{\sigma \in \Sigma} p(i', \sigma \mid i) = 1, \forall i \in X$.

Definition 23. *(Unique NFA from a PFA). Given a PFA $H = (X, \Sigma, p, \pi_0)$ we can associate with it a unique NFA $G = (X, \Sigma, \delta, X_0)$ where the state transition function $\delta : X \times \Sigma \to 2^X$ is defined for $i \in X$, $\sigma \in \Sigma$ as*

$$\delta(i, \sigma) = \{ i' \mid p(i', \sigma \mid i) > 0\},$$

and the set of possible initial states is defined as $X_0 = \{i \mid \pi_0(i) > 0\}$. In this way, the behavior of the PFA H is mapped to the behavior of the associated NFA G, i.e., $L(H) = L(G)$ (where $L(H) = \{s \in \Sigma^* \mid \Pr(s) > 0\}$).

The following PFA is needed in later sections, where we perform probabilistic classification.

Definition 24. *(Observable PFA associated with a given PFA). We construct PFA $H_o = (X, \Sigma_{obs}, p_o, \pi_0)$ by omitting the unobservable events. The transition probability matrix $p_0(j, \sigma \mid i)$ for $i, j \in X$ and $\sigma \in \Sigma_{obs}$ is obtained by setting $\pi_0(i) = 1$, and calculating $p_o(j, \sigma \mid i) = \sum_{s \in \Sigma^*_{uo}} \Pr(j, s\sigma)$.*

Now we provide definitions from classification of PFAs, that will become useful later in the paper.

Definition 25. *(Probability of Misclassification among two PFAs using the MAP rule). Suppose that we are given two PFAs, captured by $H_1 = (X_1, \Sigma_1, p_1, \pi_{0,1})$ and $H_2 = (X_2, \Sigma_2, p_2, \pi_{0,2})$, and $\Sigma_{obs} \subseteq \Sigma_1 \cup \Sigma_2$ is the set of observable events with respect to a natural projection map P. We are given prior probabilities for each model given by P_1 and*

$P_2 = 1 - P_1$, *respectively. To calculate the* a priori *probability of error when using the MAP rule, before any sequence of observations* ω *of length* n *is observed, we need to consider the probability of error for all possible observation sequences of length* n*, so that*

$$\begin{aligned} \Pr(\textit{error at } n, H_1, H_2) &= \sum_{\omega \in \Sigma_{obs}^n} \Pr(\textit{error}, \omega) = \\ &= \sum_{\omega \in \Sigma_{obs}^n} \min\{\Pr(\omega, H_1), \Pr(\omega, H_2)\}. \end{aligned}$$

Definition 26. *(Probability of Misclassification when using the MAP rule, among m PFAs). Suppose we have* m *different PFAs* $H_j = (X_j, \Sigma_j, p_j, \pi_{0,j})$*, with* $j \in \{1, 2, \ldots, m\}$*, and* $\Sigma_{obs} \subseteq \cup_j \Sigma_j$ *is the set of observable events with respect to a natural projection map* P*. We are given a prior probability* P_j *for each model* H_j*, such that* $\sum_j P_j = 1$*. The* a priori *probability of misclassification when using the MAP rule (before any sequence of observations* ω *of length* n *is observed) is given below*

$$\begin{aligned} \Pr(\textit{error at } n) &= \sum_{\omega \in \Sigma_{obs}^n} \Pr(\textit{error}, \omega) = \\ &= \sum_{\omega \in \Sigma_{obs}^n} (\Pr(\omega) - \max_{j \in \{1,\ldots,m\}} \{\Pr(\omega | H_j) P_j\}), \end{aligned}$$

where $\Pr(\omega) = \sum_{j=1}^{m} \Pr(\omega | H_j) P_j$.

Definition 27. *(Markov chain M). Given a PFA* $H = (X, \Sigma, p, \pi_0)$ *we can associate with it a Markov chain* $M = (X, A, \pi_0)$*, where* X *is the set of states;* A *is the state transition matrix defined so that its* (j, i)*th entry captures the probability of a transition from state* i *to state* j *(given by* $p_M(j \mid i) = \sum_{\sigma \in \Sigma} p(j, \sigma \mid i)$*, and* π_0 *is the initial state probability distribution vector, defined so that its* i*th entry* $\pi_0(i)$ *captures the probability that the Markov chain starts from state* i *at start up.*

Example 7. *The following example is used to clarify the notation. Consider the PFA* H *depicted on the left of Fig. 4.1 with* $X = \{1, 2, 3\}$*,* $\Sigma = \{\alpha, \beta\}$*,* δ *as defined by the transitions in the figure (along with their probabilities), and*[2] $\pi_0 = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]'$ *(i.e., each state is equally likely at the initialization of the system). Consider also the underlying Markov chain* M *of PFA* H*, at the right of Fig. 4.1. The unique NFA* $G = (X, \Sigma, \delta, X_0)$ *associated with PFA* H *has* δ *as defined by the transitions in Fig. 4.1, and* $X_0 = \{1, 2, 3\}$*.*

[2]Note that $'$ denotes matrix or vector transposition.

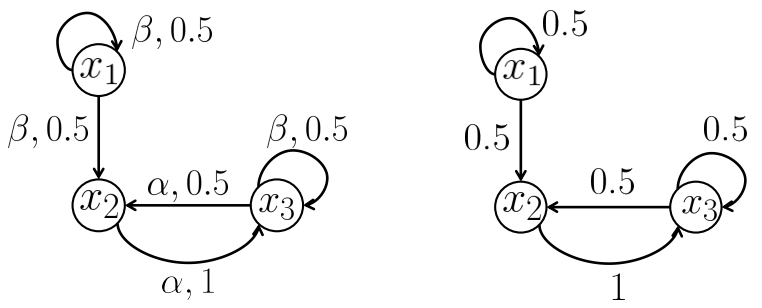

Figure 4.1: PFA H (left) and its underlying Markov chain M (right).

We now provide results from Markov chain theory that will help us deal with the problem of detectability in PFAs later in this chapter.

Definition 28. *[9, 44] (Irreducible or strongly connected Markov chain). A Markov chain $MC = (Q, A, \pi_0)$ (where $Q = \{q_1, q_2, ..., q_{|Q|}\}$ or $Q = \{1, 2, ..., |Q|\}$ is the set of states) is irreducible if for all $j, i \in Q$, there exists some $n \in \mathbb{N}$ such that $A^n(j, i) > 0$, where $A^n(j, i)$ captures the transition probability from state i to state j in exactly n steps (given by the (j, i)th entry of matrix A^n).*

Definition 29. *[9, 44] (Aperiodic Markov chain). A state $q_i \in Q$ of a Markov chain $MC = (Q, A, \pi_0)$ is said to be periodic if the greatest common divisor d of the set $\{n > 0 : \Pr(q[n] = q_i \mid q[0] = q_i) > 0\}$ is $d \geq 2$ (note that $q[t] = q_i$ denotes the event that the state of the Markov chain at step t is q_i). If $d = 1$, state q_i is said to be aperiodic. The Markov chain is said to be aperiodic if all states $q_i \in Q$ are aperiodic.*

Lemma 2. *[9] If a Markov chain $MC = (Q, A, \pi_0)$ is irreducible, then all its states have the same period. It follows that if $d = 1$ for any state of an irreducible Markov chain, then all states are aperiodic.*

Definition 30. *Let $\pi[t]$ be a $|Q|$-dimensional vector whose jth entry denotes the probability of being in state q_j after t steps. We have $\pi[0] = \pi_0$ and $\pi[t] = A\pi[t-1] = A^t\pi_0$ for $t = 1, 2, ...$*

Definition 31. *[4, 9] (Recurrence Time). Given a Markov chain $MC = (Q, A, \pi_0)$, the*

recurrence time of state $q_i \in Q$ is defined as

$$T_i = \inf\{n > 0 : q[n] = q_i\}.$$

Definition 32. *[4, 9] (Recurrent States of a Markov Chain). Given a Markov chain $MC = (Q, A, \pi_0)$, a state q_i is called recurrent ($q_i \in Q_R$, where $Q_R \subseteq Q$ is the set of recurrent states) if*

$$\Pr(T_i < \infty \mid q[0] = q_i) = 1.$$

Thus, recurrence implies that a state is visited infinitely often.

Definition 33. *[9] (Stationary distribution of a Markov chain). If the Markov chain $MC = (Q, A, \pi_0)$ is irreducible and aperiodic, then $\lim_{t\to\infty} \pi[t]$ exists and is called the stationary distribution of the Markov chain denoted by $\pi_s = [\pi_s(q_1), \pi_s(q_2), ..., \pi_s(q_{|Q|})]'$.*

Note that the stationary state probability vector for a PFA $H = (X, \Sigma, p, \pi_0)$ (assuming that one exists) is the same as the stationary state probability vector of its associated Markov chain $MC = (X, A, \pi_0)$, i.e., it is the unique probability vector that satisfies $\pi = A\pi$.

4.3 Detectability in Stochastic Discrete Event Systems

We first revisit some definitions from [46], which provide relevant background for our development. Throughout this section, we consider a PFA $H = (X, \Sigma, p, \pi_0)$ with set of observable events $\Sigma_{obs} \subseteq \Sigma$ under the natural projection map P.

Definition 34. *($\rho_\omega(x_i)$). Given $x_i \in X$, and $\omega \in P(L(H))$*

$$\rho_\omega(x_i) \equiv \frac{\pi_\omega(x_i)}{\sum_{\forall x_i} \pi_\omega(x_i)},$$

is the conditional probability of occurrence of state x_i given that observation sequence ω has occurred (recall that $\pi_\omega(x_i)$ was defined in Definition 22).

Definition 35. *(Convergent sequence). Consider an infinite sequence of observations $\omega = \omega_1\omega_2...\omega_i...$ and let $\omega_1^n = \omega_1\omega_2...\omega_n$ be its prefix of length n. Let*

$$\rho(\omega_1^n) = \max(\rho_{\omega_1^n}(x_1), \rho_{\omega_1^n}(x_2), ..., \rho_{\omega_1^n}(x_{|X|}));$$

then, ω is convergent if $\lim_{n\to\infty} \rho(\omega_1^n) = 1$.

Below we provide the definition of (strong) stochastic detectability introduced in [46].

Definition 36. *(Strong (Stochastic) Detectability) [46]. A stochastic discrete event system captured by a PFA $H = (X, \Sigma, p, \pi_0)$ is strongly (stochastic) detectable with respect to a set of observable events $\Sigma_{obs} \subseteq \Sigma$ if from equally likely initial states (i.e., $\pi_0 = \frac{1}{|X|}\mathbf{1}$) all infinite sequences are convergent (Definition 35). This means*

$$(\forall 0 < \alpha < 1)(\exists N \in \mathbb{N})(\forall n \geq N)$$

$$\{(s \in \Sigma^* : \Pr(s) > 0 \wedge |s| = n) \rightarrow (\rho(P(s)) \geq \alpha)\}.$$

4.4 A-Detectability in Stochastic Discrete Event Systems

In this section we introduce the notion of A-detectability; we also develop a methodology to verify it using observer based techniques, and prove that A-detectability is a PSPACE-hard problem.

Definition 37. *(A-Detectability). A stochastic discrete event system captured by PFA $H = (X, \Sigma, p, \pi_0)$ is A-detectable from initial probability distribution π_0 with respect to a set of observable events $\Sigma_{obs} \subseteq \Sigma$ if*

$$(\forall \epsilon > 0)(\exists N \in \mathbb{N})$$

$$\Pr(\{s \in \Sigma^* : ||s|| = n \geq N, |R(X_0, P(s))| > 1\}) < \epsilon,$$

where $R(X_0, P(s))$ is taken with respect to the NFA G associated with PFA H.

Remark: A comparison between Strong Detectability and A-detectability shows that if the NFA associated with a given PFA is strongly detectable then the PFA is A-detectable. The proof is outlined below. Suppose the nondeterministic automaton $G = (X, \Sigma, \delta, X_0)$, associated with a PFA $H = (X, \Sigma, p, \pi_0)$, is strongly detectable with respect to natural projection map P for a set $\Sigma_{obs} \subseteq \Sigma$ of observable events. Then, the PFA H can be shown to be A-detectable. From Definition 4, we have

$$(\exists N \in \mathbb{N})$$

$$\{(\forall s \in \Sigma^* : ||s|| = n \geq N) \Rightarrow |R(X_0, P(s))| \leq 1\} \Rightarrow$$

$$\Pr(\{s \in \Sigma^* : \|s\| = n \geq N, |R(X_0, P(s))| > 1\}) = 0,$$

which, according to Definition 37, means that PFA H is A-detectable. Note, however, that the converse is not necessarily true. A counter example is the PFA in Figure 1, which is used as a running example in the thesis.

Next we discuss the verification of A-detectability, which relies on the construction of a stochastic observer.

4.4.1 Verification of A-Detectability

Next we describe a useful extension of observer G_{obs} from Definition 5, based on the NFA G that is associated with the given PFA H.

Definition 38. *(G_{obs} with unobservable self-loops ($\hat{G}_{obs}$)). Given an NFA $G = (X, \Sigma, \delta, X_0)$ with set of observable events $\Sigma_{obs} \subseteq \Sigma$ under the natural projection map P, the observer (or current-state estimator) is a deterministic finite automaton (DFA) $G_{obs} = (Q_{obs}, \Sigma_{obs}, \delta_{obs}, Q_{0,obs})$ constructed as in Definition 5. Adding a self-loop to each state of DFA G_{obs} for each label in the set $\Sigma_{uo} = \Sigma - \Sigma_{obs}$, we create the DFA $\hat{G}_{obs} = (Q_{obs}, \Sigma, \hat{\delta}_{obs}, Q_{0,obs})$. More specifically $\hat{\delta}_{obs}$ extends δ_{obs}, as follows: for $Q \in Q_{obs}$ and $\sigma \in \Sigma$*

$$\hat{\delta}_{obs}(Q, \sigma) = \begin{cases} \delta_{obs}(Q, \sigma), & \text{if } \sigma \in \Sigma_{obs}, \\ Q, & \text{if } \sigma \in \Sigma_{uo}. \end{cases}$$

Definition 39. *(Stochastic Observer H_{obs}). Given a PFA $H = (X, \Sigma, p, \pi_0)$ and a natural projection map P with respect to the set of observable events $\Sigma_{obs} \subseteq \Sigma$, H_{obs} is constructed as follows:*
(1) We construct the (deterministic) observer $G_{obs} = (Q_{obs}, \Sigma_{obs}, \delta_{obs}, Q_{0,obs})$, and then $\hat{G}_{obs} = (Q_{obs}, \Sigma, \hat{\delta}_{obs}, Q_{0,obs})$ with respect to the NFA $G = (X, \Sigma, \delta, X_0)$ associated with H.
(2) We construct the PFA $H_{obs} = H \times \hat{G}_{obs} := (X \times Q_{obs}, \Sigma, p_{obs}, \pi_{0,obs})$, where $X \times Q_{obs}$ is the set of states, $p_{obs}(x^{i'}_{j'}, \sigma | x^i_j)$ is the state transition probability defined for $x^i_j = (x_j, Q_i) \in X \times Q_{obs}$ and $x^{i'}_{j'} \in X \times Q_{obs}$ (i.e., $x_j \in X$, $Q_i \in Q_{obs}$, $x_{j'} \in X$ and $Q_{i'} \in Q_{obs}$) and $\sigma \in \Sigma$, as $p_{obs}(x^{i'}_{j'}, \sigma | x^i_j) = p(x_{j'}, \sigma | x_j)$ if $Q_{i'} = \hat{\delta}_{obs}(Q_i, \sigma)$, and $p_{obs}(x^{i'}_{j'}, \sigma | x^i_j) = 0$, otherwise; $\pi_{0,obs}$ is the initial-state probability distribution vector given by a column vector with $\pi_{0,obs}(x^i_j) = \pi_0(x_j)$ if $Q_i = Q_{0,obs}$ and zero otherwise.

Definition 40. *(Markov chain MC of stochastic observer H_{obs}). Given a PFA $H = (X, \Sigma, p, \pi_0)$, its associated NFA $G = (X, \Sigma, \delta, X_0)$, and its deterministic observer $G_{obs} =$*

$(Q_{obs}, \Sigma_{obs}, \delta_{obs}, Q_{0,obs})$, *we construct the stochastic observer* $H_{obs} = H \times \hat{G}_{obs} := (X \times Q_{obs}, \Sigma, p_{obs}, \pi_{0,obs})$. *The Markov chain* $MC = (X \times Q_{obs}, T_{obs}, \pi_{0,obs})$ *associated with the PFA* H_{obs}, *is the Markov Chain with state transition probabilities* $p_M(x_{j'}^{i'} \mid x_j^i) = \sum_{\sigma \in \Sigma} p_{obs}(x_{j'}^{i'}, \sigma \mid x_j^i)$ *for* $x_j^i, x_{j'}^{i'} \in X \times Q_{obs}$ *(indexing the states in the order* $(x_1^1, x_2^1, \cdots, x_{|X|-1}^{|Q_{obs}|}, x_{|X|}^{|Q_{obs}|})$, *the entries of the state transition matrix* T_{obs} *are given by* $T_{obs}(|Q_{obs}|(i'-1)+j', |Q_{obs}|(i-1)+j) = p_M(x_{j'}^{i'} \mid x_j^i)$*).*

Example 8. *Given PFA H in Fig. 4.1, the corresponding* H_{obs}, *with* $Q_1 = \{x_1, x_2, x_3\}, Q_2 = \{x_2, x_3\}, Q_3 = \{x_3\}, Q_4 = \{x_2\}$, *is as shown in Fig. 4.2. Ordering the states from left-right, and top-bottom as* $(x_1^1, x_2^1, x_3^1, x_2^2, x_3^2, x_3^3, x_2^4)$, *the state transition probability matrix of the underlying Markov chain is*

$$A_{obs} = \begin{bmatrix} 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0.5 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0.5 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0.5 & 0 \end{bmatrix}$$

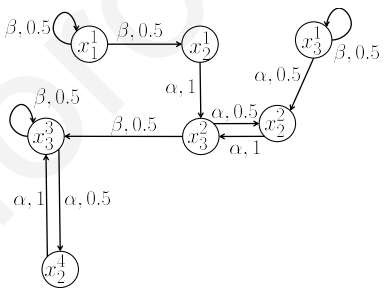

Figure 4.2: H_{obs} used in Example 8.

We now recall a useful property for a finite state Markov chain (see, for example, [4, 11]).

Lemma 3. *Let X be the finite state space of Markov chain* $MC = (X, A, \pi_0)$ *and* $X = X_R \dot{\bigcup} X_T$, *where* X_R *and* X_T *denote the non-intersecting sets of recurrent and transient states. We have that*

$$(\forall \epsilon > 0)(\exists N \in \mathbb{N})(\forall n \geq N)$$
$$\pi_n^T \triangleq \sum_{x_j \in X_T} \pi_n(x_j) < \epsilon \ ,$$

which clearly implies that for any $x_j \in X_T, \pi_n(x_j) < \epsilon$.

Remark: Lemma 3 implies that as the number of transitions increases, the probability of the Markov chain being in a transient state approaches zero. If a state $x_i \in X_R$ (equivalent to $x_i \notin X_T$), then it is easily proved (Lemma 3) that

$$(\exists \epsilon > 0)(\forall N \in \mathbb{N})(\exists n \geq N)$$

$$\pi_n(x_i) \geq \epsilon,$$

as long as the recurrent states are reachable (with a nonzero probability) from the set of possible initial states (this can be easily ensured by trimming all states that are not reachable from possible initial states of the given Markov chain).

Theorem 5. *(A-detectability using stochastic observer H_{obs}: Necessary and sufficient conditions). Given a PFA $H = (X, \Sigma, p, \pi_0)$, its associated NFA $G = (X, \Sigma, \delta, X_0)$, we construct its observer $G_{obs} = (Q_{obs}, \Sigma_{obs}, \delta_{obs}, Q_{0,obs})$, its stochastic observer $H_{obs} = H \times \hat{G}_{obs} := (X \times Q_{obs}, \Sigma, p_{obs}, \pi_{0,obs})$, and its associated NFA $G_{Hobs} = (X \times Q_{obs}, \Sigma, \delta_{Hobs}, Q_{0,Hobs})$, and Markov chain $MC = (X \times Q_{obs}, T_{obs}, \pi_{0,obs})$ as in Definition 39.*

Then, PFA H is A-detectable iff the Markov chain MC has the following property:

$$(\forall x_j^i \equiv (x_j, Q_i) \in X_R \subseteq X \times Q_{obs}) \rightarrow (|Q_i| = 1)\ ,$$

where X_R is the set of recurrent states of Markov chain MC as defined in Lemma 3.

Theorem 5 implies that for a PFA to be A-detectable, we need all recurrent states $x_j^i \in X \times Q_{obs}$ of its underlying Markov chain to be associated with state estimates that involve a single state (i.e., have $|Q_i| = 1$).

Proof. ($\rightarrow$): Suppose that there exists at least one recurrent state $x_j^i = (x_j, Q_i)$, where $Q_i \in Q_{obs}$, with $|Q_i| > 1$.

Clearly, $\{x_j^i \in X_R\}$ means that $(\exists \epsilon > 0)(\forall N \in \mathbb{N})(\exists n \geq N)$ such that $\Pr(\{s \in \Sigma^* : \|s\| = n \geq N, |R(X_0, P(s))| > 1\}) \geq \pi_n(x_j^i) \geq \epsilon$ (Remark 4.4.1). Thus, the system is not A-detectable (Definition 37).

($\leftarrow$): Suppose that all recurrent states x_j^i are associated with singleton states Q_i, then all non singleton states $x_{j'}^{i'} \in X_T$. Clearly, $(\forall \epsilon > 0)(\exists N \in \mathbb{N})(\forall n \geq N)$ such that $\Pr(s : \|s\| = n \geq N, |R(X_0, P(s))| > 1) \leq \pi_n^T < \epsilon$, where X_T is the set of all transient states (Lemma 3). Thus the system is A-detectable (Definition 37).

Example 9. *According to Definition 37 the system is A-detectable because all recurrent states in the associated Markov chain of H_{obs} (shown in Fig. 4.2) are singleton states (namely,*

states x_3^3 and x_2^4). Although the PFA H is A-detectable, the associated NFA G, is not strongly detectable because the observer G_{obs} has loops which involve nonsingleton states, e.g., $\{x_2, x_3\}$ is involved in self loop a^* (see the observer in Fig.13).

Remark: In an A-detectable system, following any (long enough) sequence of events s, such that the set of possible state estimates associated with its projection $P(s)$ includes at least one recurrent state, then there always exists at least one continuation t, such that the state estimate of $P(st)$ includes only one state, which is recurrent (otherwise the system would not be A-detectable). This line of thought connects the problem of A-detectability to a problem of language equivalence (between the language associated to the transient behaviour of the system versus the language capturing its recurrent behaviour).

4.5 Complexity Comparison between Verification of A-detectability and A-Diagnosability

It is worth discussing a bit differences and similarities between A-detectability and A-diagnosability [49], which is a similar notion in fault diagnosis. The main idea in both notions is that the most probable observation sequences allow us to resolve a specific property of the system with always increasing certainty. The difference between the two notions is that A-detectability resolves the exact state of the system (the ambiguity occurs when the state estimate involves at least two different states of the system), whereas in A-diagnosability the ambiguity for an observation sequence occurs when the state estimate involves at least one state from two different sets of states (the normal set of states, which represents the normal behaviour of the system, and the faulty set of states, which indicate that the fault has occurred). Another difference is the monotonicity property which is present in fault diagnosis but not in detectability. More specifically, when the state estimate involves only states in the set of the faulty states, then for any new observation the state estimate remains in the set of faulty states, and the fault diagnosis problem is resolved. This is not true in general for the detectability problem, because even if the state estimate is a single state at a certain point, it is possible that a new observation may drive the estimator to a state estimate that involves multiple states; thus, the problem of exact detection of a state is not resolved. In this section we prove that A-detectability

and A-diagnosability are PSPACE-hard by introducing a polynomial-time reduction of each instance of the universality problem for a given NFA to an instance of the A-detectability and A-diagnosability problems; since the universality problem for an NFA is PSPACE-complete, these reductions prove that A-detectability and A-diagnosability problems are PSPACE-hard.

4.5.1 A-Detectability is PSPACE-Hard

Given a nondeterministic finite automaton G over the alphabet Σ, the universality problem asks if the language of G contains all finite words over Σ, that is, if $L(G) = \Sigma^*$ [26].

Definition 41. *(Universality problem for NFA with all states initial) [26]. Given an NFA $G = (X, \Sigma, \delta, X)$ over an alphabet Σ, having the property[1] that all states are initial ($X_0 = X$), do we have $L(G) = \Sigma^*$?*

The Universality problem with all states initial is shown in [26] to be PSPACE-complete, when $|\Sigma| \geq 2$. We now establish a reduction of the universality problem with all states initial to an instance of the A-detectability problem. Suppose that we are given an instance of the universality problem for $G_T = (X_T, \Sigma_o, \delta_T, X_T)$. We construct the following instance of the A-detectability problem (refer to Fig. 4.3). PFA $H = (X, \Sigma, p, \pi_0)$ has $X = X_T \cup \{x_R\}$ where the set of states $X_T = \{x_1, x_2, ..., x_{|X_T|}\}$ and x_R is a new state (not in X_T). The set $\Sigma = \Sigma_o \cup \{\delta_{uo}\}$ is the set of events, where Σ_o is the set of observable events (events of G_T) with $|\Sigma_o| \geq 2$ and δ_{uo} is a new event (not in Σ_o) that is unobservable. We assign probabilities as follows:
i) The $(|X_T| + 1) \times 1$ column-vector $\pi_0 = \frac{1}{|X_T|+1}[1, 1, ..., 1]'$ is the initial-state probability distribution vector, where the order of states is taken to be $(x_1, x_2, ..., x_{|X_T|}, x_R)$.
ii) The state transition probability $p(x_{i'}, \sigma \mid x_i)$ is defined for $x_i, x_{i'} \in X$ and $\sigma \in \Sigma$ as follows:

a) $\forall x_i \in X_T$, $\forall x_{i'} \in X$ and $\forall \sigma \in \Sigma$, if $x_{i'} \in \delta(x_i, \sigma)$, then $p(x_{i'}, \sigma \mid x_i) = \dfrac{1}{\sum_{\sigma \in \Sigma} |\delta(x_i, \sigma)|}$,

otherwise $p(x_{i'}, \sigma \mid x_i) = 0$;

[1] In [26], they call G an automaton with all states both initial and final because they consider the marked language of the automaton with respect to the set of final (marked) states.

b) $\forall \sigma \in \Sigma_o$, $p(x_R, \sigma \mid x_R) = \frac{1}{|\Sigma_o|}$.

Remark: Note that i) the set of states X_T is the set of transient states of PFA H and state x_R is the only recurrent state; ii) there exists a transition (via the unobservable event) from every state $x_i \in X_T$ to state x_R.

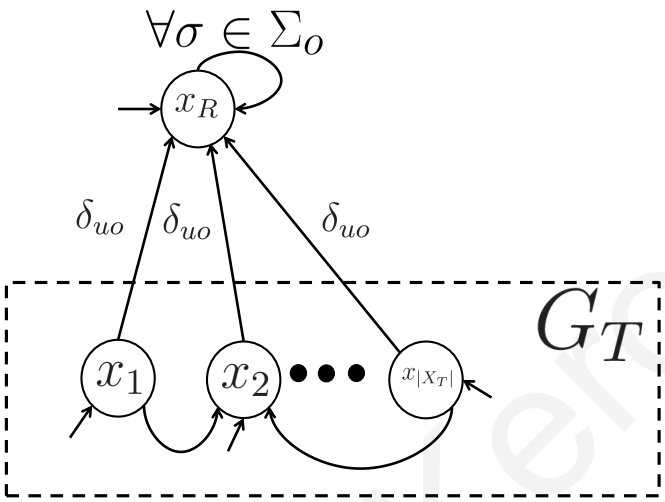

Figure 4.3: Instance of A-detectability.

The following theorem shows that every instance of the language universality problem of an NFA ($G_T = (X_T, \Sigma_o, \delta_T, X_T)$) with all states initial, can be reduced to an instance of A-detectability problem (as it was described in previous paragraphs). Thus, the A-detectability problem is PSPACE-hard.

Theorem 6. *A-detectability is PSPACE-hard.*

We argue that language universality of NFA G_T is equivalent to PFA H (as in Fig. 4.3 and as described above) not being A-detectable.

($\rightarrow$) If $L(G_T) = \Sigma_o^* \Rightarrow (\forall s : |R(X, P(s)| > 1) \Rightarrow (\forall N \in \mathbb{N})(||s|| = n \geq N) \Pr(s : |R(X, P(s)| > 1) = 1$. This means that the system H is not A-detectable.

($\leftarrow$) If $L(G_T) \neq \Sigma_o^* \Rightarrow (\exists K \in \mathbb{N})(\exists s$ s.t. $P(s) = \omega \in \Sigma_o^K : \omega \notin L(G_T)$. Notice that $\forall \omega_1 \in \Sigma_o^* : \omega_1\omega \notin L(G_T)$, because $\delta(X_T, \omega_1\omega) = \delta(\delta(X_T, \omega_1), \omega) \subseteq \delta(X_T, \omega) = \emptyset$.

Let $N' = NK$, where $N \in \mathbb{N}$. Then, $\Pr(s : ||P(s)|| = n' \geq N' \wedge |R(X, P(s))| > 1)) \leq \Pr(s : P(s) = \omega = \omega^{(1)}\omega^{(2)}...\omega^{(N)}$ (with $||\omega^{(j)}|| = K$ and $||\omega|| = N'$) $\wedge\, \omega^{(j)} \neq \omega_K$ for $j = 1, 2, ..., N)) \leq (1 - \frac{1}{(|\Sigma|+1)|X|}\frac{1}{|\Sigma|^K})^N \Rightarrow (\forall \epsilon > 0)(\exists N' = NK \in \mathbb{N}) \Pr(s : ||s|| = n' \geq N' \wedge |R(X, P(s))| > 1) < \epsilon$ (namely, $N' = NK$, with $N \geq \lceil \frac{\log \epsilon}{\log(1 - \frac{1}{(|\Sigma|+1)|X|}\frac{1}{|\Sigma|^K})} \rceil$). This means that H is A-detectable.

This establishes the reduction and we conclude that the A-detectability problem is PSPACE-hard.

4.5.2 A-Diagnosability is PSPACE-Hard

Definition 42. *[49](A-diagnosability). Suppose we are given a PFA $H = (X, \Sigma, p, \pi_0)$ with a set of observable events Σ_{obs}, $\Sigma_{obs} \subseteq \Sigma$. For $\Sigma_f \in \Sigma_{uo} \equiv \Sigma - \Sigma_{obs}$, let $\Psi(\sigma_f) = \{s = t\sigma_f \in L(H) : \{t \in (\Sigma - \sigma_f)^*\}\}$. The live, prefix-closed language $L(H)$ is A-diagnosable with respect to a fault σ_f, under the natural projection observation map P (with respect to a set of observable events $\Sigma_{obs} \subseteq \Sigma$) if*

$$(\forall \epsilon > 0)(\exists N \in \mathbb{N})(\forall s \in \Psi(\sigma_f) \wedge n \geq N)$$
$$\Pr(\{t : D(st) = 0\} | t \in \frac{L}{s} \wedge ||t|| = n) < \epsilon,$$

where the diagnosability function D is defined as

$$D(st) = \begin{cases} 1, & \text{if } \omega \in P^{-1}[P(st)] \Rightarrow \sigma_f \in \omega, \\ 0, & \text{otherwise.} \end{cases}$$

Given a PFA H, the A-diagnosability problem (Definition 42) is PSPACE-hard. We prove that A-diagnosability is PSPACE-hard by introducing a polynomial-time reduction of each instance of the universality problem for a given NFA to an instance of the A-diagnosability problem; since the universality problem for an NFA is PSPACE-complete, this proves that the A-diagnosability problem is PSPACE-hard.

Given a nondeterministic finite automaton G over the alphabet Σ, the universality problem asks if the language of G contains all finite words over Σ, that is, if $L(G) = \Sigma^*$ [26]. We now establish a reduction of the Universality problem (Definition 41) with all states initial to an instance of the A-diagnosability problem. Suppose that we are given an instance of the Universality problem for $G_N = (X_N, \Sigma_o, \delta_N, X_N)$. We construct the following instance of the A-diagnosability problem (refer to Fig. 4.4). PFA $H = (X, \Sigma, p, \pi_0)$ has $X = \{x_0\} \cup X_N \cup \{x_f\}$ where x_0 and x_f are new states (not in X_N). The set of states $X_N = \{x_1, x_2, ..., x_{|X_N|}\}$ can be seen as the set of states of G that are consistent with the normal behaviour of H and x_f can be seen as the single state that is consistent with faulty behaviour of H. The set $\Sigma = \Sigma_o \cup \{\sigma_{uo}, \sigma_f\}$ is the set of events, where Σ_o is the set of observable events (events of G_N) with $|\Sigma_o| \geq 2$ and σ_f, σ_{uo} are new events (not in Σ_o) that are unobservable. We assign probabilities as follows:

i) The $(|X_N| + 2) \times 1$ column-vector $\pi_0 = [1, 0, ..., 0]'$ is the initial-state probability distribution vector, where the order of states is taken to be $(x_0, x_1, x_2, ..., x_{|X_N|}, x_f)$.
ii) The state transition probability $p(i', \sigma|i)$ is defined for $i, i' \in X$ and $\sigma \in \Sigma$ as follows:
a) $\forall x_{i'} \in X_N$, $p(x_{i'}, \sigma_{uo}|x_0) = \frac{1}{|X_N|+1}$ and $p(x_f, \sigma_f|x_0) = \frac{1}{|X_N|+1}$;
b) $\forall \sigma \in \Sigma_o$, $p(x_f, \sigma|x_f) = \frac{1}{|\Sigma_o|}$ and $\forall \sigma \in \Sigma$, $p(x_f, \sigma|x_i) = 0$ for $x_i \neq x_f$;
c) $\forall x_i, x_{i'} \in X_N$ and $\forall \sigma \in \Sigma_o$, if $x_{i'} \in \delta(x_i, \sigma)$, then $p(x_{i'}, \sigma|x_i) = \frac{1}{\sum_{\sigma \in \Sigma_o} |\delta(x_i, \sigma)|}$, otherwise $p(x_{i'}, \sigma|x_i) = 0$.

Remark: Note that i) The NFA G that can be associated to PFA H can be seen as the union of $G_N = (X_N, \Sigma_o, \delta_N, X_N)$ and the single state NFA $G_F = (\{x_f\}, \Sigma_o, \delta_f, \{x_f\})$, whose language $L(G_F) = \Sigma_o^*$; ii) there exists a transition (with unobservable event) from x_0 to each state in X_N and a transition (with fault event) to x_f.

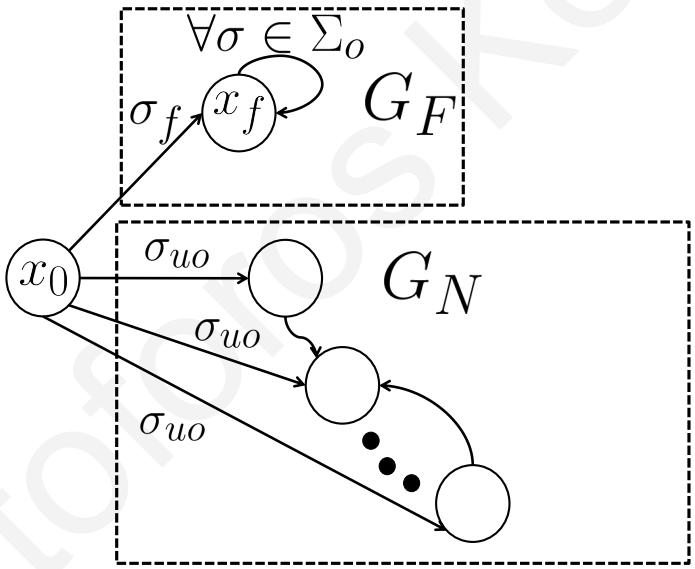

Figure 4.4: Instance of A-diagnosability given NFA G_N

The following theorem shows that every instance of the language universality problem of an NFA ($G_N = (X_N, \Sigma_o, \delta_N, X_N)$) with all states initial, can be reduced to an instance of A-diagnosability problem (as it was described in previous paragraph). Thus, the A-diagnosability problem is PSPACE-hard.

Theorem 7. *A-diagnosability is PSPACE-hard.*

Proof. We argue that language universality of an NFA G_N is equivalent to PFA H (as in Fig.4.4 and as described above) not being A-diagnosable.

($\rightarrow$) If $L(G_N) = \Sigma_o^* \Rightarrow L(G_N) = L(G_F) \Rightarrow (\forall \omega \in \Sigma_o^* : D(\omega) = 0) \Rightarrow (\forall n \in \mathbb{N})(\Pr(\{\omega : D(\omega) = 0 | \omega \in L(G_F) \wedge \|\omega\| = n\}) = \Pr(\omega \in \Psi(\sigma_f) \wedge \|\omega\| = n) = 1)$. This means that the system G is not A-diagnosable.

($\leftarrow$) If $L(G_N) \neq \Sigma_o^* \Rightarrow (\exists k \in \mathbb{N})(\exists \omega \in \Sigma_o^k : \omega \notin L(G_N)$. Notice that $\forall \omega_1 \in \Sigma_o^* : \omega_1 \omega \notin L(G_N)$, because $\delta(X_N, \omega_1 \omega) = \delta(\delta(X_N, \omega_1), \omega) \subseteq \delta(X_N, \omega) = \emptyset$.

Let $n' = nk$, where $n, k \in \mathbb{N}$. Then, $\Pr(\omega \in \Psi(\sigma_f) : \|\omega\| = n' \wedge D(\omega) = 0)) \leq \Pr(\omega \in \Psi(\sigma_f) : \omega = \omega^{(1)}\omega^{(2)}...\omega^{(n)}$ (with $\|\omega^{(j)}\| = k$ and $\|\omega\| = n'$) $\wedge \omega^{(j)} \neq \omega_k$ for $j = 1, 2, ..., n)) = (1 - \frac{1}{|\Sigma_o|^k})^n \Rightarrow (\forall \epsilon > 0)(\exists n' = nk \in \mathbb{N}) \Pr(\omega \in \Psi(\sigma_f) : \|\omega\| \geq n' \wedge D(\omega) = 0) < \epsilon$ (namely, $n' = nk$, with $n \geq \lceil \frac{\log \epsilon}{\log(1 - \frac{1}{|\Sigma_o|^k})} \rceil$). This means that G is A-diagnosable. This establishes the reduction and we conclude that the A-diagnosability problem is PSPACE-hard. □

4.6 AA-Detectability

In some cases even if the system is not A-detectable (i.e., there exists a nonzero probability of generating observation sequences that correspond to possible estimates for more than one state), the probability of estimating the correct state for these observation sequences goes to one (Definition 35). These cases lead to the definition of AA-detectability, which is similar to the notion of AA-diagnosability in [49]. The differences and similarities between the two notions (AA-diagnosability/AA-detectability) can be understood in the context of differences and similarities between the problems of detectability and diagnosability as discussed in previous Section. Interestingly enough, it is currently unknown whether AA-diagnosability can be verified with polynomial complexity, whereas AA-detectability is shown in this section to be polynomially verifiable.

Definition 43. *(AA-detectability). A stochastic discrete event system captured by PFA $H = (X, \Sigma, p, \pi_0)$ is AA-detectable with respect to a set of observable events $\Sigma_{obs} \subseteq \Sigma$ if*

$$(\forall \epsilon > 0)(\forall \alpha > 0)(\exists N \in \mathbb{N})(\forall n \geq N)$$

$$\{\Pr(s \in \Sigma^* : |s| = n \wedge \rho(P(s)) \geq \alpha) < \epsilon\},$$

where $\rho(P(s))$ was defined in Definition 35.

Example 10. *The following example is used to clarify the notion of AA-detectability. Consider the PFA $H = (X, \Sigma, p, \pi_0)$ depicted in Fig. 4.5 with $\pi_0 = [1, 0, 0, 0, 0, 0]'$, where*

$X = \{1, 2, 3, 4, 5, 6\}$, $\Sigma = \{\alpha, \beta\}$, *p is as defined by the transitions in the figure, and* $\Sigma_{obs} = \Sigma$. *According to Definition 37 the system is not A-detectable because* $\forall s \in \Sigma^* : |R(X_0, P(s))| > 1$. *Note also that the infinite sequence* $s = \alpha\beta^n$, *where n is arbitrarily large is not convergent (Definition 35). We have* $\omega = P(s) = P(\alpha\beta^n)$ *and* $\rho(P(\omega)) = \rho_{P(\omega)}(3) = \frac{\pi_\omega(3)}{\pi_\omega(3)+\pi_\omega(5)} = \frac{1/3}{1/3+2/15} < 1$, *because* $\pi_\omega(3) = \frac{2(0.5)^n}{3}$ *and* $\pi_\omega(5) = \frac{2(0.5)^n}{15}$. *Although there exists an infinite sequence which is not convergent, the system in this example is, in fact, AA-detectable, because AA-detectability is related to the overall probability of non-convergent sequences. Formal verification of this fact will not be provided until Section 4.*

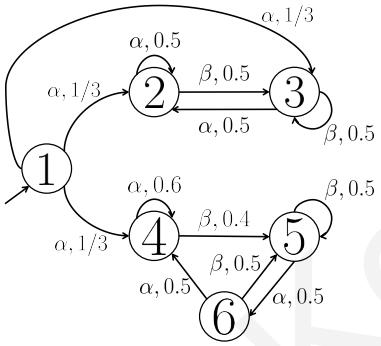

Figure 4.5: PFA used in Example 10.

In this section we establish that the property of AA-Detectability for PFA H can be verified with polynomial complexity. We first argue that the AA-detectability problem for a PFA H relies on finding the continuations (t) of any string s, *after* we reach the recurrent states of H (since we can decrease the probability that the underlying Markov chain of the system remains in the set of transient states, by increasing the number of observations). In other words, we can focus on the recurrent behavior of H; in fact, by waiting for more observations we can be certain, within whatever threshold we choose, that we reach a recurrent state or, equivalently, a closed strongly connected component of the given PFA.

Definition 44. *(Closed strongly connected component (CSCC)) [11]. Given a PFA* $H = (X, \Sigma, p, \pi_0)$ *a closed strongly connected component is a PFA* $H_i = (X_i, \Sigma, p_i, \pi_{0,i})$ *with* $X_i \subseteq X$ *such that* $\forall x, x' \in X_i$ *and* $\forall \sigma \in \Sigma, p_i(x', \sigma|x) = p(x', \sigma|x)$, *whenever the former is defined (zero otherwise) and* $\pi_{0,i}$ *is an* $|X_i|$*-dimensional probability vector that captures the initial probabilities associated with states in* X_i. H_i *is a strongly connected component or irreducible if its associated Markov chain is strongly connected or irreducible.*

Remark: Note that $H_i = (X_i, \Sigma, p_i, \pi_{0,i})$ being a CSCC implies that $\sum_{\sigma\in\Sigma} \sum_{x'\in X_i} P(x', \sigma \mid x) =$

1 for $x \in X_i$.

4.6.1 Polynomial Verification of AA-Detectability

The first step of the proposed verification algorithm is the identification of all CSCCs of PFA H. The CSCCs depend only on the graph structure of the state transition diagram of the given PFA. Polynomial graph algorithms for the identification of the strongly connected components and CSCCs of a given strongly connected graph can be found in [11], [12]. Let $H_1, H_2, \cdots, H_m$ denote the strongly connected components of the given PFA H. Furthermore, note that if we have to distinguish between the different recurrent components, we need to classify between them by choosing the most probable component (which is also a PFA on its own). Naturally, this leads us to consider methods of computing the probability that we made the wrong choice (probability of error) when performing this classification. As discussed in Section 2 the optimal rule to do this is the MAP (maximum a posteriori) rule.

We first explore necessary and sufficient conditions for the probability of error when classifying among a given set of PFAs to tend to zero eventually (with increasing number of observations, the probability of classification error goes to zero). Necessary and sufficient conditions are difficult in the general case (see, for example, Chapter 5), but we will argue that the PFAs have certain properties (namely, they are associated to NFAs that have an underlying deterministic transition mechanism) that can be exploited to establish necessary and sufficient conditions for the probability of classification error to go to zero. If these conditions hold, we are able to correctly identify the CSCC in which the system state lies with increasing certainty. If, in addition, we are able to pinpoint, with increasing certainty, the exact state estimate within the chosen CSCC, then we can establish AA-detectability. We use this two-stage approach to establish necessary and sufficient conditions for AA-detectability. In the following theorem we prove a key result for the development of a polynomial complexity verification algorithm.

Theorem 8. *(A necessary condition for AA-detectability). Given a PFA $H = (X, \Sigma, p, \pi_0)$ and its associated underlying Markov chain $M = (X, A, \pi_0)$, where $X = X_R \cup X_T$ with X_R being the set of recurrent states and X_T being the set of transient states, a necessary condition for AA-detectability is that there do not exist $x_i, x_k, x_l \in X_R$ and $s^k, s^l \in \Sigma_{uo}^*$ and $\sigma \in \Sigma_o$, such that $p(x_k, s^k\sigma \mid x_j) > 0$ and $p(x_l, s^l\sigma \mid x_j) > 0$.*

Proof. Let us suppose that the given PFA is AA-detectable, and let us assume that there exists at least one CSCC. For $s \in L(G)$, such that it reaches a state $x \in X_R$ in the CSCC H, with some nonzero probability and $0 < \alpha < 1$. Let event $A(N, \alpha) = \{s \in L(G) : |s| = n \geq N \wedge \rho(P(s)) \geq \alpha)\}$, $A^C(N, \alpha)$ be the complement of A, and for $x_i \in X$, let $A(N, \alpha, x_i) = \{s \in L(G) : |s| = n \geq N \wedge \rho_{P(s)}(x_i) \geq \alpha\}$. The system being AA-detectable means,

$$(\forall \epsilon > 0)(\forall 0 < \alpha < 1)(\exists N \in \mathbb{N})(\forall n \geq N)$$
$$\Pr(A(N, \alpha)) \geq 1 - \epsilon.$$

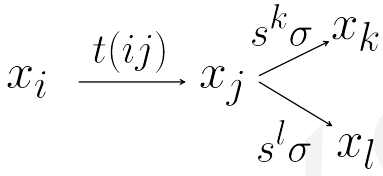

Figure 4.6: Example of non-deterministic transition used in Theorem 8.

The recurrent states are strongly connected, which means that $\forall x_i, x_j \in X_R, x_i \neq x_j$ there exists an acyclic path from x_i to x_j. Let us denote these paths by $t(ij)$, with length $n(ij)$, where $n(ij) \leq |X|$.

The proof is by contradiction. Suppose that the theorem statement does not hold and that from x_j there exists a path $s^k\sigma$ (or $s^l\sigma$), with length $k + 1$ (or $l + 1$), where the state estimate of the projection of this string includes the states x_k and x_l (refer to Fig. 4.6). Following this path, we split the probability of reaching a single state, to two states. It turns out that if we choose a large enough α, for all strings s, then all the continuations st', will result to a $\rho(P(st')) < \alpha$. To see this, take the worst case, where $|t(ij)| = |X|$, and $p(t(ij)) = p_{min}^{|X|}$, for $x_i, x_j \in X$, where $p_{min} = \min_{x_i, x_j, \sigma}\{p(x_j \mid x_i, \sigma) \mid p(x_j \mid x_i, \sigma) \neq 0\}$. Then, given $\rho_{P(s)}(x_i) \geq \alpha$, a sufficient condition for the state x_j to be the most probable state after the execution of sequence $st(ij)$, is that $\alpha \cdot p_{min}^{|X|} > 1 - \alpha$, or equivalently $\alpha > \frac{1}{p_{min}^{|X|}+1}$.

Now, we try to contradict the fact that the system is AA-detectable, by finding appropriate $N' > N$, such that $\Pr(A^C(N', \alpha) \wedge A(N, \alpha)) \geq \epsilon'$ for ϵ' to be specified. Note that $\Pr(A^C(N', \alpha) \wedge A(N, \alpha)) = \Pr(A^C(N', \alpha) \mid A(N, \alpha)) \Pr(A(N, \alpha)) = \sum_{\forall x \in X_R} \Pr(A^C(N', \alpha) \mid A(N, \alpha, x)) \Pr(A(N, \alpha, x) \mid A(N, \alpha)) \Pr(A(N, \alpha))$.

Clearly, there exists $x_i \in X_R$, for which $\Pr(A(N,\alpha,x_i) \mid A(N,\alpha)) \geq \frac{1}{|X|}$. For the chosen x_i, following the line of thought of the previous paragraph, we can find an acyclic path $t' = t(ij)s^k\sigma$, with length $|t(ij)| + k + 1$, which means that we can choose $N' = N + |t(ij)| + k + 1$, such that $\Pr(A^C(N',\alpha) \mid A(N,\alpha,x_i)) \geq p_{min}^{2|X|}$ ($2|X|$ is the maximum length of a possible t'). Thus, $\Pr(A^C(N',\alpha)) \geq \frac{p_{min}^{2|X|}}{|X|}(1-\epsilon)$. Due to the AA-detectability property, we need to have $\frac{p_{min}^{2|X|}}{|X|}(1-\epsilon) \leq \Pr(A^C(N',\alpha)) < \epsilon$, this double inequality holds if and only if $\epsilon \geq \epsilon_0$, where $\epsilon_0 = \dfrac{\dfrac{p_{min}^{2|X|}}{|X|}}{\dfrac{p_{min}^{2|X|}}{|X|}+1}$. In particular, for all $\epsilon < \epsilon_0$, the inequality does not hold, and therefore, for $\epsilon < \epsilon_0$, we cannot find $N \in \mathbb{N}$, so that AA-detectability holds. Thus, we have reached a contradiction, and the proof is completed.

□

Remark: According to Theorem 8 it is necessary for all CSCCs $H_i, i = 1, ..., m$ of PFA H, to have associated NFAs, $G_i, i = 1, ..., m$ that are DFAs. This is very important for the polynomial verification of AA-detectability.

Consider m different PFAs $H_i = (X_i, \Sigma_i, p_i, \pi_{0,i})$, with $i \in \{1, 2, \ldots, m\}$, and let $\Sigma_{obs} \subseteq \cup_i \Sigma_i$ be the set of observable events with respect to a natural projection map P. We are given a priori probability P_i for each model H_i, such that $\sum_i P_i = 1$. Note that O_n is an observation sequence of length n, which can be generated by at least one of the m PFAs, with $\Pr(O_n, H_i)$ be the probability that O_n has occurred from PFA H_i. $H_{max}(O_n) = \arg\max_{H_i}\{\Pr(O_n, H_i)\}$ is the PFA, with the maximum probability of occurrence for O_n.

Lemma 4. *The probability of misclassification among m PFAs tends to zero iff the probability of misclassification among all pairs of PFAs (H_i, H_j), $i, j \in \{1, 2, \cdots, m\}$ tends to zero.*

Proof. Notice that $\Pr($ error at $n) =$

$$\sum_{O_n} \sum_{H_i \neq H_{max}(O_n)} \Pr(O_n, H_i).$$

Notice also that probability of error among the pair H_i and H_j is denoted as

$$\Pr(\text{ error at } n, H_i, H_j) = \frac{1}{P_i + P_j} \sum_{O_n} \min\{\Pr(O_n, H_i), \Pr(O_n, H_j)\}.$$

Let

$$P_{min} = \min\{P_1, ..., P_m, \frac{P_1}{P_1 + P_2}, ..., \frac{P_{m-1}}{P_{m-1} + P_m}\}$$

and

$$P_{max} = \max\{P_1, ..., P_m, \frac{P_1}{P_1 + P_2}, ..., \frac{P_{m-1}}{P_{m-1} + P_m}\}.$$

Then we have,

$$\frac{P_{min}}{P_{max}} \Pr(\text{ error at } n) = \frac{P_{min}}{P_{max}} \sum_{O_n} \sum_{H_i \neq H_{max}(O_n)} \Pr(O_n, H_i) \leq$$

$$\sum_{O_n} \frac{1}{P_i + P_{H_{max(O_n)}}} \sum_{H_i \neq H_{max}(O_n)} \Pr(\text{ error } O_n, H_i, H_{max})$$

$$\leq \sum_{H_i \neq H_j} \Pr(\text{ error at } n, H_i, H_j) \leq m^2 \frac{P_{max}}{P_{min}} \Pr(\text{ error at } n).$$

□

When multiple CSCCs are present, AA-detectability hinges on our ability to distinguish among different CSCCs of the system that are reachable under sequences with identical projections. By construction, the CSCCs are irreducible PFAs, thus the verification of AA-detectability is transformed into a problem of classification between irreducible PFAs that are simultaneously reachable under each possible observation sequence (for AA-detectability to hold the probability of error between these PFAs needs to tend to zero eventually).

It is important to realize that when we know the exact recurrent component, we know also the exact state (Theorem 8 says that eventually, with high probability, one state will dominate). Clearly, one could use the observer to resolve all possible state combinations (and combinations of recurrent components) that are reachable under different strings but with the same natural projection. From the Lemma 4, we observe that the probability of error among m PFAs tends to zero iff the probability of error for any pair (H_i, H_j) among these m PFAs also tends to zero. This detail is crucial because it allows us to use the detector instead of the observer. In order to develop a method for verifying AA-detectability, we establish a polynomial complexity method for classification between two different PFAs, whose associated NFA's are restricted to be DFA's. Therefore, the second step for a verification algorithm is to explore, using the detector for the associated NFA G of PFA H, all the pairs of closed strongly connected components that are reachable, possibly under different strings but with the same projection (i.e., (H_i, H_j)). Next, we construct the associated PFAs,

without the unobservable events, as in Definition 24 (for simplicity we keep the same symbols H_i and H_j), and perform classification between all pairs of closed strongly connected components that we found at the second step (the method of classification is described in detail in the following subsection).

Theorem 9. *(Necessary and sufficient conditions for AA-detectability). A PFA $H = (X, \Sigma, p, \pi_0)$ with m CSCCs, $H_1, H_2, ..., H_m$, $m \geq 1$, is AA-detectable iff for all CSCCs $H_{i_1}, H_{i_2}, ..., H_{i_k}, k \in \{1, 2, ...\}$ that are reachable under strings $s_1, s_2, ..., s_k$, with $P(s_1) = P(s_2) = ... = P(s_k)$, the probability of error when trying to classify between these different PFAs tends asymptotically to zero.*

Proof. Now we describe a method that allows us to find if the probability of error for two PFAs $H_i = (X_i, \Sigma_{obs}, p_i, \pi_{0,i})$ and $H_j = (X_j, \Sigma_{obs}, p_j, \pi_{0,j})$ tends to zero. This problem is open at the moment for the general case; however, in our case, the underlying logical structure involves only deterministic finite automata, and we show that the problem can be solved with a polynomial complexity algorithm, with respect to the size of the state–space of the two PFAs.

In our solution, it is important to capture the common behaviour of the two PFAs which can be done with a detector (defined in Definition 6). The second step is to assign probabilities to that detector, which is difficult in the general case, but can be done easily in our case, due to the deterministic nature of the underlying logical structure of the two PFAs. Finally, we construct two derived Markov chains; checking whether the probability of error tends to zero when classifying among the given PFAs is shown to be equivalent to a problem of classification between these derived Markov chains.

Definition 45. *(PFAs $H_{d_{ij},i}$ and $H_{d_{ij},j}$ with associated detector $G_{d,ij}$). Consider two PFAs $H_i = (X_i, \Sigma_{obs}, p_i, \pi_{0,i})$ and $H_j = (X_j, \Sigma_{obs}, p_j, \pi_{0,j})$ with associated Markov chains that are irreducible (note that the PFAs are constructed from the initial PFAs by removing the unobservable events, as Remark 24).*

Suppose that the finite automata associated with these PFAs are DFA's $D_i = (X_i, \Sigma_{obs}, \delta_i, X_{0,i})$ and $D_j = (X_j, \Sigma_{obs}, \delta_j, X_{0,j})$.

(1) We construct $D_{ij} = (X_{ij}, \Sigma_{obs}, \delta_{ij}, X_{0,ij})$, with $X_{ij} = X_i \cup X_j$ (assume, without loss of generality, that $X_i \cap X_j = \emptyset$), $\delta_{ij}(x, \sigma) = x'$ iff $\{x, x' \in X_i \wedge \delta_i(x, \sigma) = x'\} \vee \{x, x' \in X_j \wedge \delta_j(x, \sigma) = x'\}$ and $X_{0,ij} = X_{0,i} \cup X_{0,j}$.

(2) We construct the detector (Definition 6) $G_{d,ij} = (X_{d,ij}, \Sigma_{obs}, \delta_{d,ij}, X_{0d,ij})$ of DFA D_{ij}.

(3) We construct the PFAs $H_{d_{ij},i} = (X_{d_{ij}}, \Sigma_{obs}, p_{d_{ij},i}, \pi_{0,d_{ij}i})$ and $H_{d_{ij},j} = (X_{d_{ij}}, \Sigma_{obs}, p_{d_{ij},j}, \pi_{0,d_{ij}j})$, by assigning probabilities over the detector DFA $G_{d,ij}$. We have $p_{d_{ij},i}((x'_k, x'_l), \sigma|(x_k, x_l)) = p_i(x'_k, \sigma|x_k)$, where $x_k, x'_k \in D_i$ and $x_l, x'_l \in D_j$ (we can also have perhaps $x_k = \emptyset$ as a value, or the same state for the two components, which in the definition of the detector is equivalent to a state with a single component). Similarly, we construct PFA $H_{d_{ij},j}$.

Example 11. *Consider the PFAs $H_1 = (X_1, \Sigma, p_1, \pi_{0,1})$ and $H_2 = (X_2, \Sigma, p_2, \pi_{0,2})$ depicted in Fig. 4.5 with $\pi_{0,1} = [1, 0]'$, $\pi_{0,2} = [1, 0, 0]'$ where $X_1 = \{2, 3\}$ and $X_2 = \{4, 5, 6\}$, $\Sigma = \{\alpha, \beta\}$, p_1 and p_2 are defined by the transitions in the figure, and $\Sigma_{obs} = \Sigma$. The DFA's of interest are D_1, D_2 and D_{12} (the latter can be seen as the union of D_1 and D_2).*

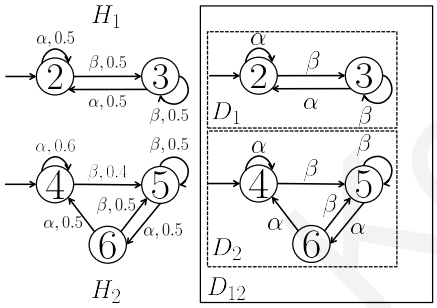

Figure 4.7: PFAs used in Example 11 (left) and associated DFA's (right).

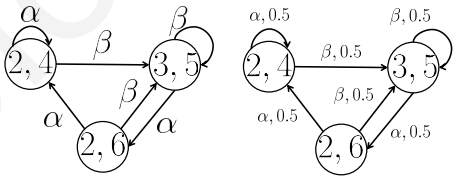

Figure 4.8: $G_{d,12}$ (left) and PFA $H_{d_{12},1}$ (right) used in Example 11 .

Definition 46. *(One-step transition probability matrix for a Markov chain) [14,28]. Given an irreducible Markov chain $MC = (Q, A, \pi_0)$, and its stationary distribution[2] $\pi_s = [\pi_s(q_1), \pi_s(q_2), ..., \pi_s(q_{|Q|})]'$, then for $q_k, q_l \in Q$ the corresponding element of the one-step transition probability matrix is given as $\mathbb{P}(k, l) = A(k, l) \times \pi_s(l)$.*

[2]In the case of a periodic Markov chain with period d, there exist $\pi_{s,r} = \lim_{t\to\infty} A^{dt+r}\pi_0$, $r \in \{0, ..., d-1\}$. In that case, we use the average stationary distribution $\pi_s = \frac{1}{d}\sum_{r=0}^{r=d-1} \pi_{s,r}$.

Lemma 5. *Given two irreducible PFAs $H_{d_{ij},i} = (X_{d_{ij},i}, \Sigma_{obs}, p_{d_{ij},i}, \pi_{0,i})$ and $H_{d_{ij},j} = (X_{d_{ij},j}, \Sigma_{obs}, p_{d_{ij},j}, \pi_{0,j})$ with the same set of allowed transitions, and thus the same transition diagram, according to [28] (Part III, Chapter 12) the probability of misclassification between the two PFAs (equivalently for two completely observable Markov chains) tends asymptotically to zero iff the underlying Markov chains ($M_{d_{ij},i} = (X_{d_{ij},i}, A_{d_{ij},i}, \pi_{0,i})$ and $M_{d_{ij},j} = (X_{d_{ij},j}, A_{d_{ij},j}, \pi_{0,j})$) are irreducible, and the one-step transition probability matrices ($\mathbb{P}^1$ and $\mathbb{P}^2$) are different.*

We create two new PFAs $H_{d_{ij},i}$ and $H_{d_{ij},j}$, with associated DFA the detector $G_{d,ij}$ and the probabilities are assigned according the definition above. In the general case, the detector's associated Markov chains, are reducible. We can easily deal with Markov chains which are reducible, finding all of their CSCCs and applying Theorem 5 for all CSCCs: we decide that the PFA is AA-detectable if for all associated Markov chains of the CSCCs, the one-step transition probability matrices are different.

A special case arises when some irreducible components of the detector $G_{d,ij}$, involve a state $x_k \in X_{d,ij}$, with $|x_k| = 1$ (singleton state), then the state estimation problem is effectively resolved for these CSCCs, because we reach the singleton state with increasing certainty. In that case, we do not assign probabilities into these CSCCs.

Example 12. *For the PFAs in Fig. 4.7 we easily construct the associated Markov chains of PFAs $H_{d_{12},1}$ and $H_{d_{12},2}$ by dropping the label of each transition from PFAs. The Markov chains $M_{d_{12},1}$ and $M_{d_{12},2}$ are depicted in Fig. 4.9.*

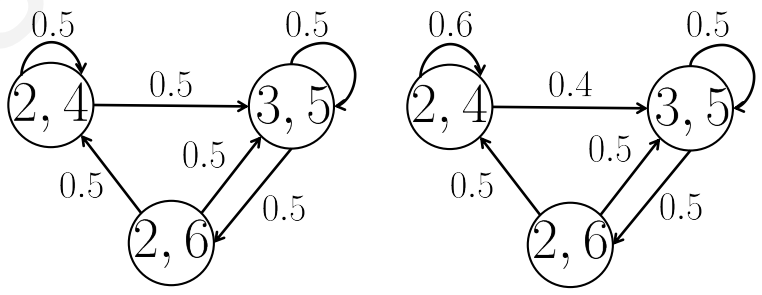

Figure 4.9: $M_{d_{12},1}$ (left) and $M_{d_{12},2}$ (right) used in Example 11.

Finally, we decide that H is AA-detectable iff the probability of misclassification for all pairs tends asymptotically to zero, which is formally summarized in Theorem 9. □

We now describe the proposed verification procedure for AA-detectability. The proposed verification algorithm involves the following steps:

- Identification of all CSCCs $H_1, H_2, ..., H_m$ of the given PFA H. Note that Theorem 8 implies that the finite automaton associated to each CSCC is a DFA.
- Classification between pairs of CSCCs (that are simultaneously reachable with sequences of events that have the same projection on the set of observable events), in order to find the most probable component (see Theorem 38). Note that if a CSCC is simultaneously entered via different paths and entry points, then we need to consider classification of a pair of CSCCs that involves the same CSCC but with different initial conditions.
    1. We construct the Detectors (Definition 6) of any pairs of CSCCs that are simultaneously reachable with strings that have the same projection on the set of observable events.
    2. Again we identify all possible CSCCs that exist in the Detectors.
    3. We assign probabilities to all CSCCs of all Detectors, by constructing two different PFAs for each pair of CSCCs (Definition 45).
    4. We compare the one-step transition probability matrices for the two different underlying Markov chains, related to the PFAs we constructed in the previous step (Definition 46).

Remark: The identification of all CSCCs and the construction of the detectors are of polynomial complexity regarding the number of the states of the PFA [11], [45]. The comparison for the one-step transition probabilities is also of polynomial complexity, because it is based on the computation of the steady-state probabilities for two Markov chains. Overall the verification procedure for AA-detectability is of polynomial complexity with respect to the number of states of the PFA.

Chapter 5

Classification among Hidden Markov Models

5.1 Introduction

We consider classification among systems that can be modeled as hidden Markov models (HMMs), based on a sequence of observation symbols that has been generated by underlying (unknown) activity in one of two known HMMs. The performance of the maximum *a posteriori* (MAP) classifier, which minimizes the probability of misclassification [1], is captured by the *a priori* probability of error, i.e., the probability of error before any observations are made. The precise calculation of the probability of error (for sequences of observations of a given finite length) is a combinatorial task of high complexity (typically exponential in the length of the sequences). In this chapter we propose a number of different ways of upper bounding this probability of error (using methods of much less computational complexity); we also establish necessary and sufficient conditions under which there exists an upper bound on the probability of error that tends to zero (at least asymptotically).

Our analysis and bounds can find application in many areas where HMMs are used, including speech recognition [2,23,37], pattern recognition [19], bioinformatics [15, 27], and failure diagnosis in discrete event systems [1, 11, 31, 49]. Our analysis also relates to approaches dealing with the *distance* or dissimilarity between two HMMs [17,25]. Directly related previous work can be found in [1], which studies the probability of misclassification and obtains bounds that tend to zero under specific conditions.

5.2 Notation and Background

Theorem 10. *[9] (Stationary distribution of a Markov chain). If the Markov chain is irreducible and aperiodic, then* $\lim_{t\to\infty} \pi[t]$ *exists and is called the stationary distribution of the Markov chain denoted by* $\pi_s = [\pi_s(q_1), \pi_s(q_2), ..., \pi_s(q_{|Q|})]'$.

Definition 47. *(Stationary emission probabilities of HMM). Given an HMM* $S = (Q, E, \Delta, \Lambda, \pi_0)$ *the stationary emission probability* $\pi_e^{(j)}(e_i)$, $\forall e_i \in E$, *can be expressed as*

$$\pi_e(e_i) = R_{|Q|} \times (A_{e_i} \times \pi_s),$$

where $R_{|Q|}$ *is the* $|Q|$*-dimensional row vector with ones in all entries (and* $\times$ *denotes matrix-matrix or matrix-vector multiplication).*

We define for notational convenience the $|Q| \times |Q|$ matrix A_{e_i}, associated with output $e_i \in E$, as follows: the $(k, j)^{th}$ entry of A_{e_i} captures the probability of a transition from state q_j to state q_k that produces output e_i (see also Section 2.4).

Note that the stationary state probability vector for an HMM S is the same as the stationary state probability vector of its associated Markov chain $MC = (Q, A, \pi_0)$.

5.3 Calculation of Upper Bound via a DFA

We establish a class of upper bounds for the probability of error among two HMMs, via the construction of a deterministic finite automaton (DFA). The key idea is that observation sequences of a specific length, that can be generated by at least one of the two HMMs, are distributed to the states of this DFA. Then one can find an upper bound of the probability of error, by comparing the probability of the states of this DFA. The key advantage is that this comparison can be done with polynomial complexity with respect to the number of states of the DFA.

We prove the following lemma, which will be useful later:

Lemma 6. *If we have two sequences* $Y(1)$ *and* $Y(2)$ *of length* n*, that can be generated by HMMs* $S^{(1)}$ *and* $S^{(2)}$ *(see Section 2.4), where the a priori probabilities are* P_1 *and* P_2*, and* $P_i^{(j)} = P(Y(i) \mid S^{(j)})$ *for* $j \in \{1, 2\}$*, then we can obtain an upper bound on the probability of*

error for these sequences as follows:

$$P(error, \{Y(1), Y(2)\}) = \sum_{i=1}^{2} \min\{P_1 \cdot P_i^{(1)}, P_2 \cdot P_i^{(2)}\}$$
$$\leq \min\{P_1 \cdot \sum_{i=1}^{2} P_i^{(1)}, P_2 \cdot \sum_{i=1}^{2} P_i^{(2)}\} \ . \tag{5.1}$$

Proof. The above can be shown easily by considering the different cases and observing that $\min\{a_1, a_2\} + \min\{b_1, b_2\} \leq \min\{a_1 + b_1, a_2 + b_2\}$. We can easily generalize the above discussion to any number of merged sequences of the same length. The next step is to find an upper bound for the probability of error at n steps. In particular, if we take any partition of the index set $I = \{1, 2, ..., d^n\}$, into subsets $D_1, D_2, ..., D_m$ (such that $D_i \cap D_j = \emptyset$ for $i \neq j$ and $\cup_{i=1}^{m} D_i = I$), then we have

$$\begin{aligned} P(\text{error at } n) &= \sum_{\ell=1}^{d^n} P(\text{error}, Y(\ell)) \\ &= \sum_{k=1}^{m} \sum_{\ell \in D_k} \min\{P_1 \cdot P_\ell^{(1)}, P_2 \cdot P_\ell^{(2)}\} \\ &\leq \sum_{k=1}^{m} \min\{\sum_{\ell \in D_k} P_1 \cdot P_\ell^{(1)}, \sum_{\ell \in D_k} P_2 \cdot P_\ell^{(2)}\}. \end{aligned} \tag{5.2}$$

□

We now discuss how we can obtain a partition of the index set I, via a deterministic finite automaton (DFA) G_d with language E^*. The reason we consider this particular partitioning of I will become clearer later when we discuss efficient ways of calculating the quantities $\sum_{\ell \in D_k} P_1 \cdot P_\ell^{(j)}$, $j = 1, 2$.

A DFA G_d is described by a four-tuple (X, E, δ, x_0), where $X = \{x_1, x_2, ..., x_{|X|}\}$ is the finite set of states; $E = \{\sigma_1, \sigma_2, ..., \sigma_{|E|}\}$ is the finite set of inputs (alphabet); $\delta : X \times E \to X$ is the transition function; and $x_0 \in X$ is the initial state. For a sequence of events $s = s[n]s[n-1]...s[1]$, $s[i] \in E$, $i = 1, 2, ..., n$, we define $\delta(q, s) = \delta(...\delta(\delta(q, s[1]), s[2]), ..., s[n])$.

A sufficient condition for the requirement that the language of G_d is E^* is that δ is defined for all pairs of states $x \in X$ and outputs $\sigma \in E$. For notational simplicity, we assume this sufficient condition holds. Consider the following subsets of sequences of observations of length n: $D_k = \{s \in E^n \mid \delta(x_0, s) = x_k\}$, $k = 1, 2, ..., |X|$. It is not hard to argue that D_k, where $k = 1, 2, ..., |X|$, form a partition of E^n.

For each $\sigma \in E$, we can construct the binary transition matrix T_σ of G_d, following the rule that if $\delta(x_i, \sigma) = x_{i'}$, then $T_\sigma(i', i) = 1$, otherwise $T_\sigma(i', i) = 0$. This matrix

captures all possible transitions from a state to another, under event σ; since G_d is deterministic, T_σ for $\sigma \in E$ is a binary matrix with exactly a single "1" in each column. We can also define the binary column vector π_0' to have a single nonzero element with value "1" at its i^{th} location, if $x_0 = x_i$ (in other words, π_0' is an indicator vector for the initial state of G_d). With this notation at hand, $\delta(x_0, s) = x_k$ for $s = s[n]s[n-1]...s[1]$ is equivalent to $\pi_n' = \underbrace{T_{s[n]}T_{s[n-1]}...T_{s[1]}}_{T_s}\pi_0'$ being a vector with all zero entries except a single "1" at the k^{th} location. This is easy to establish by induction.

More generally, the entries of the matrix $T_s = T_{s[n]}T_{s[n-1]}...T_{s[1]}$ are such that $T_s(k,i) \in \{0,1\}$ and $T_s(k,i) = 1$ if and only if $\delta(x_i, s) = x_k$. If we let the two vectors $c^{(j)} = P_j[1...1]$, of size $1 \times |Q^{(j)}|$ for $j = 1, 2$, we can show that the probability of error in Eq. (5.2) is smaller or equal to

$$\sum_{k=1}^{|X|} \min\{\sum_{s\in D_k} c^{(1)} A_s^{(1)} \pi_0^{(1)}, \sum_{s\in D_k} c^{(2)} A_s^{(2)} \pi_0^{(2)}\} \tag{5.3}$$

where for $s = s[n]s[n-1]...s[1]$ we have $A_s^{(j)}\pi_0^{(j)} = A_{s[n]}^{(j)}A_{s[n-1]}^{(j)}...A_{s[1]}^{(j)}\pi_0^{(j)}$. We now discuss how the above bound can be computed rather efficiently.

We define the matrix $\mathcal{A}^{(j)} = \sum_{\sigma\in E} T_e \otimes A_\sigma^{(j)}$, $j = 1, 2$, where $T_e \otimes A_\sigma^{(j)}$ denotes the Kronecker product defined as the $(|X||Q^{(j)}|) \times (|X||Q^{(j)}|)$ matrix

$$\begin{bmatrix} T_\sigma(1,1)A_\sigma^{(j)} & T_\sigma(1,2)A_\sigma^{(j)} & \cdots & T_\sigma(1,|X|)A_\sigma^{(j)} \\ T_\sigma(2,1)A_\sigma^{(j)} & T_\sigma(2,2)A_\sigma^{(j)} & \cdots & T_\sigma(2,|X|)A_\sigma^{(j)} \\ \vdots & \vdots & \ddots & \vdots \\ T_\sigma(|X|,1)A_\sigma^{(j)} & T_\sigma(|X|,2)A_\sigma^{(j)} & \cdots & T_\sigma(|X|,|X|)A_\sigma^{(j)} \end{bmatrix}.$$

Note that each $T_\sigma(i',i)A_\sigma^{(j)}$, $x_i, x_{i'} \in X$, is a matrix of size $(|Q^{(j)}|) \times (|Q^{(j)}|)$. We also define the (i',i) block of $\mathcal{A}^{(j)}$ as $\mathcal{A}^{(j)}(B_{i'}, B_i) = \mathcal{A}^{(j)}(b_i^{(j)} : f_i^{(j)}, b_{i'}^{(j)} : f_{i'}^{(j)})$, i.e., a $(|Q^j|) \times (|Q^j|)$ submatrix starting from row $b_i^{(j)} = (i-1)Q^{(j)} + 1$ to row $f_i^{(j)} = iQ^{(j)}$, and from column $b_{i'}^{(j)} = (i'-1)Q^{(j)}$ to column $f_{i'}^{(j)} = i'Q^{(j)}$. Letting $p_0^{(j)} = \pi_0' \otimes \pi_0^{(j)}$, we can

write[1] (for $s = s[n]s[n-1]...s[1] \in E^n$)

$$\begin{aligned} p_n^{(j)} &= (\mathcal{A}^{(j)})^n p_0^{(j)} \\ &= \left(\sum_{\sigma \in E} T_\sigma \otimes A_\sigma^{(j)} \right)^n (\pi_0' \otimes \pi_0^{(j)}) \\ &= \sum_{s \in E^n} (T_{s[n]} ... T_{s[1]}) \pi_0' \otimes (A_{s[n]}^{(j)} ... A_{s[1]}^{(j)}) \pi_0^{(j)} \\ &= \sum_{k=1}^{|X|} \sum_{s \in D_k} T_s \pi_0' \otimes \rho_{n,s}^{(j)} \\ &= \sum_{k=1}^{|X|} \sum_{s \in D_k} u_k \otimes \rho_{n,s}^{(j)} \\ &= \sum_{k=1}^{|X|} u_k \otimes \sum_{s \in D_k} \rho_{n,s}^{(j)} , \end{aligned}$$

where u_k is a column vector of size $|X| \times 1$, with zeros on all of its entries except a single one at its k^{th} entry, and $\rho_{n,s}^{(j)}$ is the vector $\rho_n^{(j)}$ for the sequence of observations s.

If we focus on the k^{th} block of $p_n^{(j)}$ of size $|Q^{(j)}| \times 1$ (i.e., entries $(k-1)Q^{(j)} + 1$ to $kQ^{(j)}$), we see that

$$p_n^{(j)}(B_k) = \sum_{s \in D_k} \rho_{n,s}^{(j)} = \sum_{s \in D_k} A_s^{(j)} \pi_0^{(j)} .$$

Following Eqs. (5.2) and the bound in (5.3), we can write

$$P(\text{error at } n) \leq \sum_{k=1}^{|X|} \min\{c^{(1)} p_n^{(1)}(B_k), c^{(2)} p_n^{(2)}(B_k)\} , \tag{5.4}$$

which can be used to compute an upper bound on the probability of error between the two systems ($S^{(1)}$ and $S^{(2)}$) by taking advantage of how the DFA G_d creates the partitions D_k, $k = 1, 2, ..., |X|$.

Example 13. *Consider the two HMMs in Fig. 2.6 and the DFA G_d in Fig. 5.1, with $X = \{1, 2, 3\}$, language $E^* = (\alpha + \beta)^*$, and initial state $x_0 = 1$ (which means that $\pi_0' = [1\ 0\ 0]^T$). Assume that the priors are $P_1 = 0.6$, $P_2 = 0.4$, so that*

$$c^{(1)} = \begin{bmatrix} 0.6 & 0.6 \end{bmatrix}, c^{(2)} = \begin{bmatrix} 0.4 & 0.4 \end{bmatrix},$$

and also that

$$\pi_0^{(1)} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T, \pi_0^{(2)} = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}^T .$$

[1] One of the properties of the Kronecker product is that $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$ for matrices A, B, C, D of appropriate sizes [38].

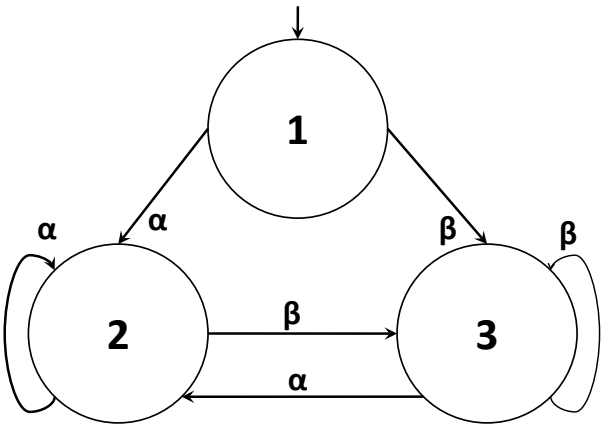

Figure 5.1: DFA G_d for Example 2.

We create, according to the previous definitions, the matrices T_α, T_β for H_s as

$$T_\alpha = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}, T_\beta = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

and obtain the matrices $\mathcal{A}^{(1)}$, $\mathcal{A}^{(2)}$ as

$$\mathcal{A}^{(1)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.05 & 0 & 0.05 & 0 & 0.05 \\ 0 & 0.95 & 0 & 0.95 & 0 & 0.95 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix},$$

$$\mathcal{A}^{(2)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.95 & 0 & 0.95 & 0 & 0.95 \\ 0 & 0.05 & 0 & 0.05 & 0 & 0.05 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}.$$

Similarly, we obtain $p_0^{(j)} = \pi_0' \otimes \pi_0^{(j)}$, for $j = 1, 2$, as

$$p_0^{(1)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T,$$

$$p_0^{(2)} = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 & 0 & 0 \end{bmatrix}^T.$$

For a sequence of observations of length n, we can write

$$P(\text{error at } n) \leq \sum_{i\in\{1,2,3\}} \min\{c^{(1)}p_n^{(1)}(B_i), c^{(2)}p_n^{(2)}(B_i)\}\ ,$$

where $p_n^{(j)} = (\mathcal{A}^{(j)})^n \pi_0^{(j)}$, $j = 1, 2$. *The plot of the bound as a function of n is provided in Fig. 5.2. As n becomes infinite, this bound stabilizes at* 0.2349.

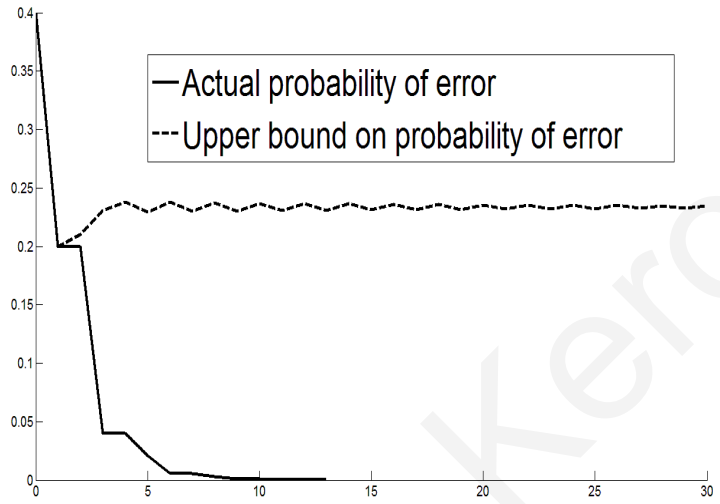

Figure 5.2: Actual probability of error (continuous line) and upper bound (dashed line) with DFA H_s in Fig. 5.1.

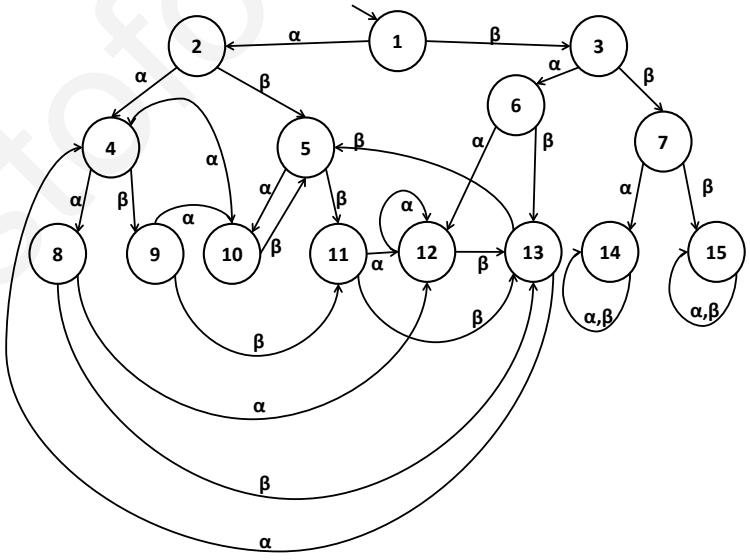

Figure 5.3: DFA G_d in Example 3.

Example 14. *We can extend the construction of the previous example to the larger DFA* G_d *in Fig. 5.3 with* $X = \{1, 2, ..., 15\}$, *language* $E^* = (\alpha + \beta)^*$, *and initial state* $x_0 = 1$ *(which*

means that $\pi'_0 = [1\ 0\ 0\ \ldots\ 0]^T$*). We omit the details of the construction since the steps are identical to the steps in Example 2.*

The resulting upper bound on the probability of error is plotted in Fig. 5.4 as a function of the number of observations. As $n \to \infty$*, we see that this upper bound tends to the constant value* 0.0166*. Note that this bound can perhaps be reduced by employing a DFA with more states and/or different transition functionality (to try and achieve a better partitioning of the set of possible sequences). In this particular example, in order to find this* G_d*, we tried all possible DFAs of* 15 *states, and presented the one that asymptotically results in the least upper bound.*

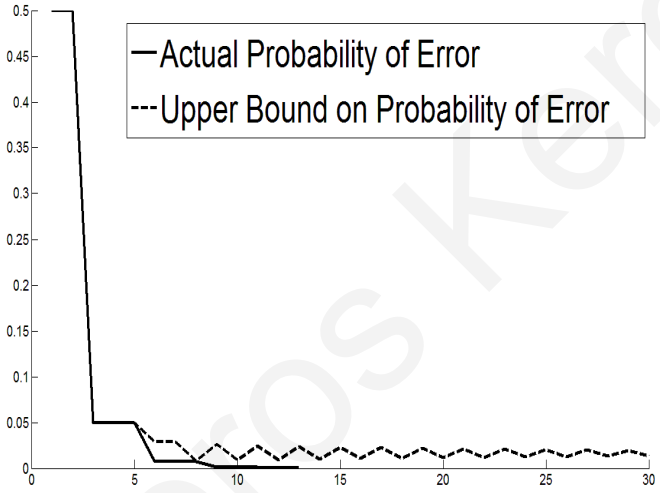

Figure 5.4: Actual probability of error (continuous line) and upper bound (dashed line) with the DFA G_d in Fig. 5.3.

Remark: In the previous examples, the upper bound did not tend to zero eventually. This happens, because the upper bound, stabilizes as the underlying Markov chain converges to steady-state. In the general case, there exists at least two recurrent states, that are reachable by the same observation sequence, with nonzero probability, by both models. Therefore, in the general case, the upper bound does not tend to zero.

5.3.1 Connections to a Stochastic Diagnoser

We can reduce the number of states or even the size of all transition submatrices $A_e^{(j)}$, $j = 1, 2$, for each model ($S^{(1)}$, $S^{(2)}$) if we are able to remove all states that are not reachable under specific conditions (e.g., unreachability from a specific starting

state). An example of such a deterministic finite automaton was the stochastic diagnoser introduced in [49], for the purpose of fault diagnosis. We describe this connection via the following example, where we use an appropriate DFA to create the stochastic diagnoser for the two models shown in Fig. 2.6.

Example 15. *Suppose that the models in Fig. 2.6 capture the Normal (S_1) and Faulty (S_2) behaviour of a system. Also we define $Q^{(1)} = \{1N, 2N\}$, and $Q^{(2)} = \{1F, 2F\}$, with priors $P_1 = P_2 = 0.5$, and initial states, $q_0^{(1)} = \{1N\}$, $q_0^{(2)} = \{1F\}$. We want to find all transition matrices for the stochastic diagnoser, and relate them to the previous analysis (the original work in [49] uses the transpose of the matrices we use here). We analyze the system using the previous method, with the only difference being that the construction of the matrices A_e, $e \in E$, considers the behavior in each system simultaneously, e.g.,*

$$A_b = \begin{bmatrix} \mathcal{A}_b^{(1)} & \mathbf{0} \\ \mathbf{0} & \mathcal{A}_b^{(2)} \end{bmatrix} = \begin{bmatrix} & 1N & 2N & 1F & 2F \\ 1N & 0 & 0 & 0 & 0 \\ 2N & 1 & 0 & 0 & 0 \\ 1F & 0 & 0 & 0 & 0 \\ 2F & 0 & 0 & 1 & 0 \end{bmatrix}.$$

If we only keep elements on nonzero rows and columns, we obtain the reduced matrix

$$A_b^{(s)} = \begin{bmatrix} & 1N & 1F \\ 2N & 1 & 0 \\ 2F & 0 & 1 \end{bmatrix}.$$

Following this approach, we can create all possible different states and apply the reduced transition matrices. The stochastic diagnoser for our example is shown in Fig. 5.5. We can create the S matrix which includes all submatrices, according to each state $\{X_1, X_2, X_3\}$ (e.g., $S(1,7)$ captures the transition probability from state X_1^{1N} to X_3^{1F}). If the states are ordered as follows: state 1→ X_1^{1N}, state 2→ X_1^{1F}, state 3→ X_2^{1N},..., state 8 → X_3^{2F}, the matrix S is given

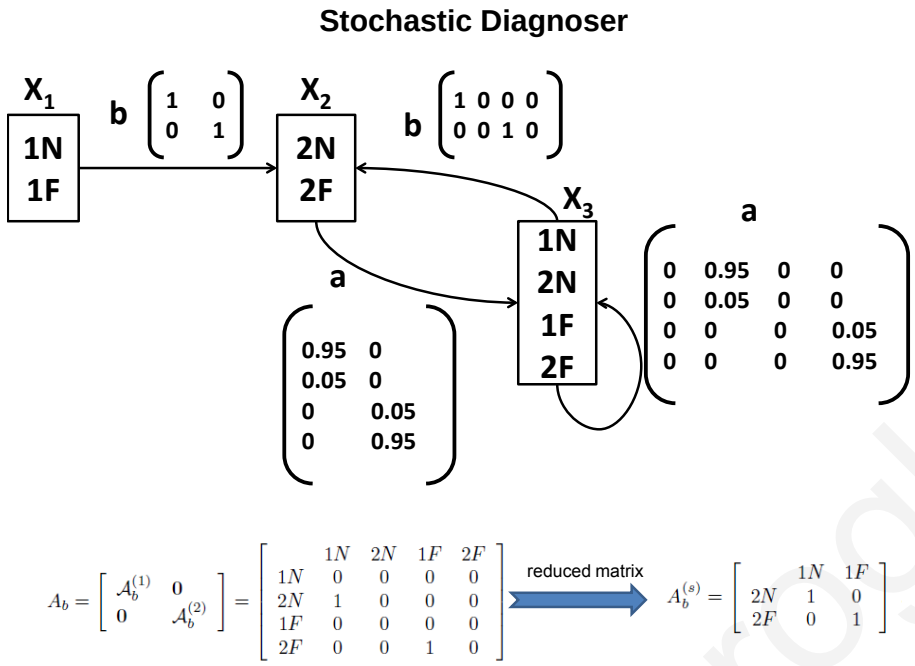

Figure 5.5: Stochastic Diagnoser for $S^{(1)}$ and $S^{(2)}$.

by

$$S = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0.95 & 0 & 0 & 0.95 & 0 & 0 \\ 0 & 0 & 0.05 & 0 & 0 & 0.05 & 0 & 0 \\ 0 & 0 & 0 & 0.05 & 0 & 0 & 0 & 0.05 \\ 0 & 0 & 0 & 0.95 & 0 & 0 & 0 & 0.95 \end{bmatrix}.$$

Using S we can compute the upper bound of the probability of error as in the previous example (using, however, blocks of different sizes, due to the fact that entries that are zero in each block are dropped). Alternatively, we can use the automaton shown in Fig. 5.6 and follow the approach in the previous section to obtain $p_n^{(j)} = (\mathcal{A}^{(j)})^n p_0^{(j)}$. *Note that by construction, a stochastic diagnoser checks if an output symbol is possible or not, so that the underlined symbols in Fig. 5.6 do not appear in the stochastic diagnoser in Fig. 5.5. For large n, we find the upper bound to be* 0.2802.

A probabilistic finite automaton that is AA-stochastically diagnosable [49] is essentially an automaton for which the probability of misclassification[2] goes to zero

[2]Strictly speaking AA-stochastic diagnosability is only concerned with faulty behavior that might

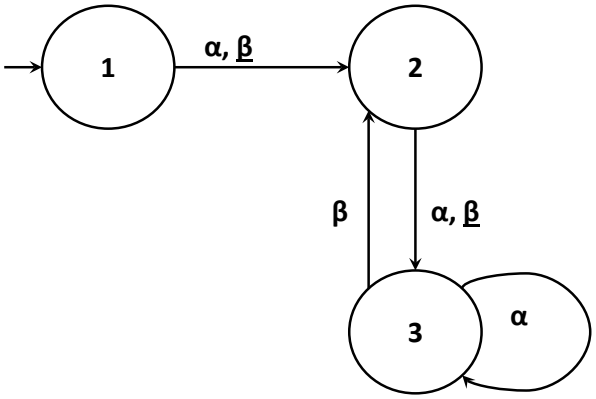

Figure 5.6: Equivalent DFA to Stochastic Diagnoser in Example 4.

as the number of observations becomes asymptotically large. It is evident that our method can be used to establish whether the probability of misclassification goes to zero (by determining whether its upper bound goes to zero) using constructions quite distinct from a stochastic diagnoser. Thus, a sufficient condition for AA-stochastic diagnosability would be the existence of a DFA that leads to an upper bound on the probability of misclassification that goes to zero as the number of observations increases.

Remark: The complexity of computing the exact probability of error is an exponential function of n (it is of $O(n \times d^n \times (|Q^{(1)}|^2 + |Q^{(2)}|^2))$). In obtaining the upper bound, we only require complexity linear in n (the complexity is of $O(n \times |X|^2 \times (|Q^{(1)}|^2 + |Q^{(2)}|^2))$). In addition, for an arbitrarily large number of observations, we can compute the asymptotic upper bound with complexity of $O(|X|^3 \times (|Q^{(1)}|^3 + |Q^{(2)}|^3))$ by employing eigenvalue decomposition to obtain the steady-state of the Markov chains with transition matrices $\mathcal{A}^{(j)}$, $j = 1, 2$.

5.4 Establishing an Upper Bound for the Probability of Error via a Stochastic Verifier

Now we establish an upper bound on the probability of error, which is computed with polynomial complexity. The verification algorithm is based on the construction

be considered as non-faulty (and whether its probability goes to zero as the number of observations increases); thus, one should exclude the probability of misclassification that arises from strings generated by the non-faulty system that are more likely to have been generated by the faulty system.

of a stochastic verifier. We are also able to give necessary and sufficient conditions under which the upper bound tends eventually to zero.

Theorem 11. *Suppose we have two HMMs ($S^{(1)}$, $S^{(2)}$), as defined in the previous section, and let $D = |E|$, be the number of different output symbols (in either $S^{(1)}$ or $S^{(2)}$). Arrange all d^n sequences of output symbols in some arbitrary order and call them $Y(1)$, $Y(2)$, ..., $Y(d^n)$, and let $P_i^{(1)} = P(Y(i) \mid S^{(1)})$ and $P_i^{(2)} = P(Y(i) \mid S^{(2)})$ for $i = 1, 2, ..., d^n$. If we use the optimal classifier to minimize the probability of error after a sequence of n observations, the* a priori *probability of error (after n observations) satisfies:*

$$P(\text{error at } n) \leq \sqrt{d^n \cdot P_1 \cdot P_2} \cdot \sqrt{\sum_{i=1}^{d^n} P_i^{(1)} \cdot P_i^{(2)}}. \tag{5.5}$$

Proof. We can obtain the probability of error of the classifier (that minimizes the probability of error) by calculating and comparing the state distributions of the two models for all possible sequences of observations of length n. We can represent the computation in terms of two d-ary trees of depth n, as shown in Fig. 5.7. Each node represents $\rho_j^{(n)}$, $j \in \{1, 2\}$, after a specific sequence of n events has occured (see previous section). For each node at level L, we create d child-nodes, and we repeat this procedure until having n levels in the tree.

Once we expand these trees to n levels, each leaf node corresponds to a unique sequence of outputs of length n, which, in the worst case scenario, can be produced by both HMMs. We assign to each leaf-node a probability of occurring $P_i^{(j)} = P(Y_1^n = Y(i) \mid S^{(j)})$, where $j \in \{1, 2\}$ represents the model and $i \in \{1, 2, ..., d^n\}$ is the index of the length-n output sequence.[3] We can express the probability of error for the two systems, after n steps, as

$$\begin{aligned} P(\text{error at } n) &= \sum_{i=1}^{d^n} P(\text{error}, Y(i)) \\ &= \sum_{i=1}^{d^n} \min\{P_1 \cdot P_i^{(1)}, P_2 \cdot P_i^{(2)}\} \end{aligned} \tag{5.6}$$

Clearly, from Eqs. (4.2) and (5.6) we have that

$$P(\text{error}, Y(i)) \leq P_1 \cdot P_i^{(1)} \tag{5.7}$$

$$P(\text{error}, Y(i)) \leq P_2 \cdot P_i^{(2)} \tag{5.8}$$

[3]Note that we use the D- ary tree in our derivation of the exact value of the probability of error, but we actually do not need it to derive the simple upper bound that we present in the next section.

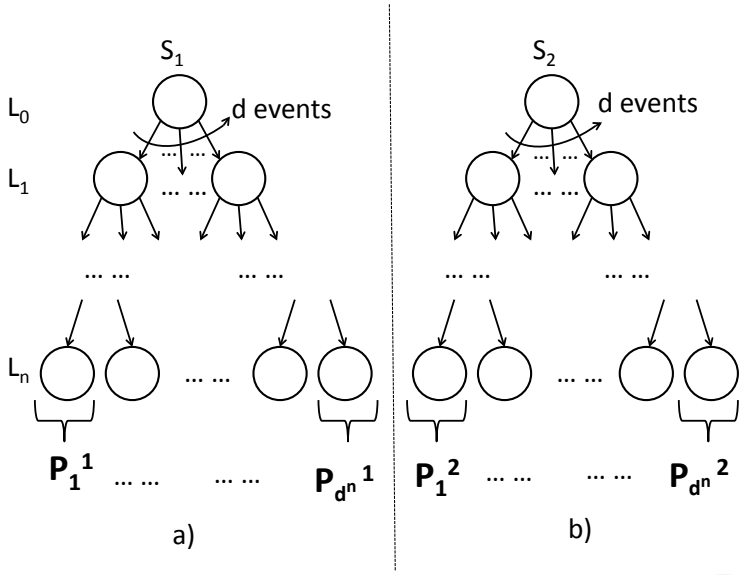

Figure 5.7: Two d-ary trees for $S^{(1)}$ and $S^{(2)}$.

If we combine the two inequalities above, we can conclude that

$$(P(\text{error}, Y(i)))^2 \leq (P_1 \cdot P_2) \cdot (P_i^{(1)} \cdot P_i^{(2)}) \tag{5.9}$$

Summing up all d^n of the above inequalities we reach the following inequality

$$\sum_{i=1}^{d^n} P(\text{error}, Y(i))^2 \leq (P_1 \cdot P_2) \cdot \sum_{i=1}^{d^n} (P_i^{(1)} \cdot P_i^{(2)}) \tag{5.10}$$

Our goal is to bound the probability of error for n steps; to do this we make use of the following equation:

$$(P(\text{error at } n))^2 = \left(\sum_{i=1}^{d^n} P(\text{error}, Y(i)) \right)^2 =$$

$$\sum_{i=1}^{d^n} \sum_{j=1}^{d^n} P(\text{error}, Y(i)) \cdot P(\text{error}, Y(< i + j - 1 >)), \tag{5.11}$$

where $< k > \equiv k \mod (d^n)+1$. The rearrangement inequality [22] states the following: let $\alpha_1 \leq \alpha_2 \leq ... \leq \alpha_n$ and $\beta_1 \leq \beta_2 \leq ... \leq \beta_n$ be sequences of real numbers and π any permutation of the set $\{1, 2, ..., n\}$. Then,

$$\sum_{i=1}^{n} \alpha_i \beta_{n-i+1} \leq \sum_{i=1}^{n} \alpha_i \beta_{\pi_i} \leq \sum_{i=1}^{n} \alpha_i \beta_i. \tag{5.12}$$

Without loss of generality we can assume that

$$P(\text{error}, Y(1)) \leq P(\text{error}, Y(2)) \leq ... \leq P(\text{error}, Y(d^n))$$

and from Eq. (5.12), we can infer that $\forall j \in \{2, ..., d^n\}$, we have

$$\sum_{i=1}^{d^n} P(\text{error}, Y(i)) \cdot P(\text{error}, Y(< i + j - 1 >)) \leq$$

$$\sum_{i=1}^{d^n} P(\text{error}, Y(i)) \cdot P(\text{error}, Y(i)). \tag{5.13}$$

Thus from Eqs. (5.11) and (5.13), we conclude that

$$\begin{aligned} (P(\text{error at } n))^2 &\leq d^n \cdot (\sum_{i=1}^{d^n} P(\text{error}, Y(i))^2) \\ &\leq d^n P_1 P_2 \sum_{i=1}^{d^n} P_i^{(1)} \cdot P_i^{(2)}, \end{aligned} \tag{5.14}$$

so that

$$P(\text{error at } n) \leq \sqrt{d^n \cdot P_1 \cdot P_2} \cdot \sqrt{\sum_{i=1}^{d^n} P_i^{(1)} \cdot P_i^{(2)}}. \tag{5.15}$$

At this point, the proof is complete. □

Remark: We suppose that we have two HMMs, $S^{(1)}$ and $S^{(2)}$, with languages (i.e., sequences of outputs with nonzero probability), $L(S^{(1)})$ and $L(S^{(2)})$, respectively. We define also $L_n(S^{(1)}) = \{t \in L(S^{(1)}) |$ length of t equals $n\}$; similarly, we define $L_n(S^{(2)})$. If we can bound the number N of the sequences of observations of length n that can be produced by both models ($N = |L_n(S^{(1)}) \cap L_n(S^{(2)})|$) then, we can refine the bound in Eq. (5.15) by replacing d^n by N. The next section is devoted to the quantity $\sqrt{\sum_{i=1}^{d^n} (P_i^{(1)} \cdot P_i^{(2)})}$ in Eq. (5.15).

5.4.1 Construction of a Stochastic Verifier

We can see that

$$P(\text{identical sequence after } n \text{ steps}) = \sum_{i=1}^{d^n} (P_i^{(1)} \cdot P_i^{(2)}) \tag{5.16}$$

is the probability that the two HMMs, when assumed independent produce an identical sequence after n steps. In order to find an upper bound for this probability, we follow the method of Massey in [32]. The general idea is to capture the common behavior of the two HMMs with a new carefully constructed product HMM. Once we construct this product HMM, an upper bound for the probability in Eq. (5.16) can be obtained as a function of the eigenvalue with the second largest magnitude of the transition matrix of the product HMM. In the remainder of this section we present the construction of the product HMM.

*Step 1:*In this step of the construction we are interested, within each HMM, to be able to discriminate the transition to the same state but under a different event. The easiest way to do this is to create replicas of each state, depending on the event under which one reaches this state. Thus, for each state $q_h^{(j)} \in Q^{(j)}$, we check if there is at least one state $q_{h'}^{(j)} \in Q^{(j)}$ such that $A_{e_i}^{(j)}(q_h^{(j)}, q_{h'}^{(j)}) > 0$ for some event $e_i \in E$ (actually, we also have $e_i \in E^{(j)}$); if this is the case, we create a new state which is called $q_{h,e_i}^{(j)}$ and represents state $q_h^{(j)}$ when reached under the output symbol $e_i^{(j)}$. The transitions out of this state remain the same as the transitions out of $q_h^{(j)}$. Clearly, we need only create at most $|E^{(j)}|$ replicas for each $q_h^{(j)}$ and we can end up with at most $|Q^{(j)}| \times |E^{(j)}|$ states. We use $Q'^{(j)}$, $j \in \{1, 2\}$, to denote the set of all states (including newly constructed states) and define the transition matrices for these states by $A_{e_i}'^{(j)}$ where $A_{e_i}'^{(j)}(q_{h,e_i}^{(j)}, q_{h',e_s}^{(j)}) = A_{e_i}^{(j)}(q_h^{(j)}, q_{h'}^{(j)})$, for e_i, $e_s \in E$. Note that when we expand the set of states of HMM, we also need to redefine their initial state distribution. The simplest thing to do is to set the initial probability of state q_{h,e_i} for some e_i to be equal to the initial probability for state q_h and zero for all q_{h,e_j}, $e_j \neq e_i$. Other ways to do this also exist.

*Step 2:*Combining all possible pairs of new states from the two HMMs in step 1, we create a Stochastic Verifier which is a product Markov chain $H_p = (Q_{H_p}, E_{H_p}, \Delta_{H_p}, \pi_0)$, where $Q_{H_p} = Q'^{(1)} \times Q'^{(2)}$ is the finite set of states, $E_{H_p} = E^{(1)} \cup E^{(2)}$ is the finite set of outputs, π_0 is the initial probability distribution vector, and Δ_{H_p} is the transition probability function. The state transition matrix associated with H_p is $\mathcal{A}_{H_p}$, where[4]

$$\mathcal{A}_{H_p}((k_1, k_2), (l_1, l_2)) = \sum_{e_i \in E_{H_p}} A_{e_i}'^{(1)}(k_1, l_1) \times A_{e_i}'^{(2)}(k_2, l_2) \ ,$$

[4]We are abusing notation a bit by using (k_1, k_2) and (l_1, l_2) to index the entries of matrix $\mathcal{A}_{H_p}$; (k_1, k_2) and (l_1, l_2) should be seen as the index that corresponds to a state of the product Markov chain H_p.

with $k_1, l_1 \in Q'^{(1)}$, $k_2, l_2 \in Q'^{(2)}$, and $e_i \in E_{H_p}$. The initial probability distribution vector is chosen so that $\pi_0((l_1, l_2)) = \pi_0^{(1)}(l_1)\pi_0^{(2)}(l_2)$. From each state (ℓ_1, ℓ_2) of H_p if the sum of the probabilities of the transitions out of this state is not unity, we add a transition to state *NC* so that the sum of the probabilities out of (ℓ_1, ℓ_2) is unity. The *NC*-state is the state that captures the non-consistent behavior, i.e., when two different sequences are produced by the two HMMs. We also add a return transition with probability one from *NC*-state to itself. Note that if the *NC*-state is the only absorbing state, it will be reached with probability one. Note that the *NC* state may not be present (for example, when we have two identical HMMs with a single transition out of each state).

Example 16. *As an application of our bound we present on the left of Fig. 5.8 an example in the context of fault diagnosis in discrete event systems (DES). The problem translates to classification between two HMMs, capturing normal and faulty behavior. Specifically, the first HMM describes a system under normal behavior, whereas the second HMM describes the same system but under faulty behavior. The discrete event system is shown on the left of Fig. 5.8 with probabilities attached to each transition. The set of observable events is* $E_o = \{\alpha, \beta\}$ *and the set of unobservable events is* $E_{uo} = \{\sigma_{uo}, \sigma_f\}$*, where* σ_f *is a fault event (i.e.,* $E_f = \{\sigma_f\}$*).*

We divide the initial system into two subsystems as shown on the right of Fig. 5.8, where $S^{(1)}$ *captures the normal behavior of the system, and* $S^{(2)}$ *captures the faulty behavior. Clearly,* $P_1 = P_2 = 0.5$*, because of the equal probability to go to state 2 or state 4, from initial state 1. We now illustrate the method we described earlier in this section, creating the product Markov chain* H_p *from these two HMMs with the same language.*

We have $E^{(1)} = E^{(2)} = E = \{\alpha, \beta\}$*, and we define* $A_\alpha^{(1)}, A_\beta^{(1)}, A_\alpha^{(2)}, A_\beta^{(2)}$ *as follows:*

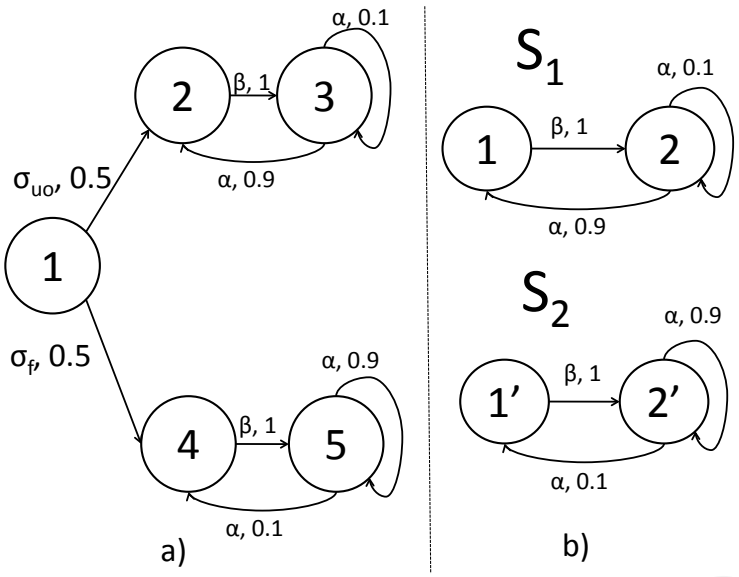

Figure 5.8: Example of a discrete event system (left) and HMM $S^{(1)}$, capturing normal behavior, and $S^{(2)}$, capturing faulty behavior of the discrete event system (right).

$$\mathcal{A}_{\alpha}^{(1)} = \begin{bmatrix} 0 & 0.9 \\ 0 & 0.1 \end{bmatrix},$$

$$\mathcal{A}_{\beta}^{(1)} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix},$$

$$\mathcal{A}_{\alpha}^{(2)} = \begin{bmatrix} 0 & 0.1 \\ 0 & 0.9 \end{bmatrix},$$

$$\mathcal{A}_{\beta}^{(2)} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}.$$

Step 1:*We create the Markov chains $S'_{(1)}$, and $S'_{(2)}$ in Fig. 5.9, with matrices $A'^{(1)}_{\alpha}, A'^{(1)}_{\beta}, A'^{(2)}_{\alpha}, A'^{(2)}_{\beta}$, where we have $Q'^{(j)} = \{q^{(j)}_{1,\alpha}, q^{(j)}_{2,\alpha}, q^{(j)}_{2,\beta}\}$, as follows:*

$$\mathcal{A}'^{(1)}_{\alpha} = \begin{bmatrix} & (1/\alpha) & (2/\alpha) & (2/\beta) \\ (1/\alpha) & 0 & 0.9 & 0.9 \\ (2/\alpha) & 0 & 0.1 & 0.1 \\ (2/\beta) & 0 & 0 & 0 \end{bmatrix},$$

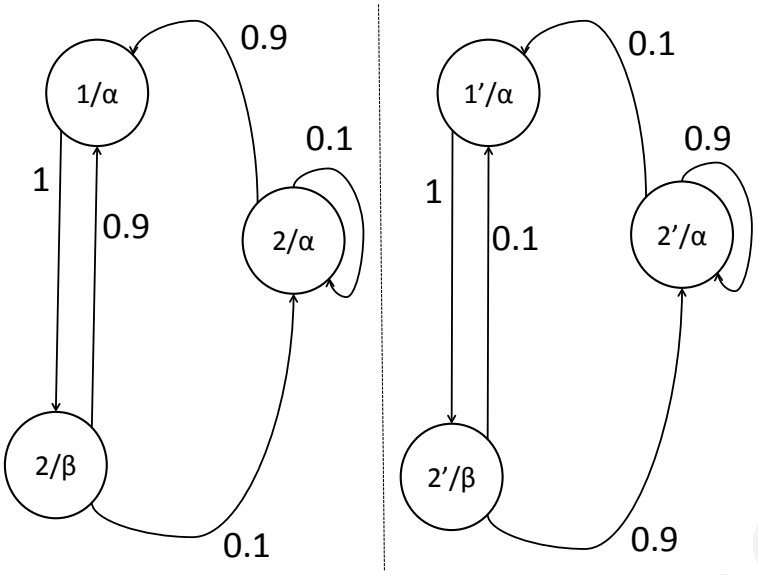

Figure 5.9: Markov chains S_1' (left) and S_2' (right).

$$\mathcal{A}'^{(1)}_{\beta} = \begin{bmatrix} & (1/\alpha) & (2/\alpha) & (2/\beta) \\ (1/\alpha) & 0 & 0 & 0 \\ (2/\alpha) & 0 & 0 & 0 \\ (2/\beta) & 1 & 0 & 0 \end{bmatrix},$$

$$\mathcal{A}'^{(2)}_{\alpha} = \begin{bmatrix} & (1/\alpha) & (2/\alpha) & (2/\beta) \\ (1/\alpha) & 0 & 0.1 & 0.1 \\ (2/\alpha) & 0 & 0.9 & 0.9 \\ (2/\beta) & 0 & 0 & 0 \end{bmatrix},$$

$$\mathcal{A}'^{(2)}_{\beta} = \begin{bmatrix} & (1/\alpha) & (2/\alpha) & (2/\beta) \\ (1/\alpha) & 0 & 0 & 0 \\ (2/\alpha) & 0 & 0 & 0 \\ (2/\beta) & 1 & 0 & 0 \end{bmatrix}.$$

Step 2:*We construct the product Markov chain H_p in Fig. 5.10, and the transition probability matrix of H_p (A_{H_p}), combining all possible pairs of states between the two HMMs, where $Q_{H_p} = \{h_1, h_2\}$, with $h_j \in Q'^{(j)}$, where $j \in \{1, 2\}$. In $\mathcal{A}_{H_p}$, the i^{th} column indicates the transitions from state q_i, to any other state, where $Q_{H_p} = \{q_1, q_2, ..., q_9, q_{10}\} = \{(1/\alpha, 1'/\alpha), (1/\alpha, 2'/\alpha), ..., (2/\beta, 2'/\beta), NC\}$. Note that in Fig. 5.10 we have not included states 3 and 7 (corresponding to $(1/\alpha, 2'/\beta)$ and $(2/\beta, 1'/\alpha)$ respectively) because they are not reachable from states with nonzero initial probability in H_p. We can see from the structure*

of H_p that from states 2 and 4, we can only move to the NC state (because the underlying pairs of states do not have a common output).

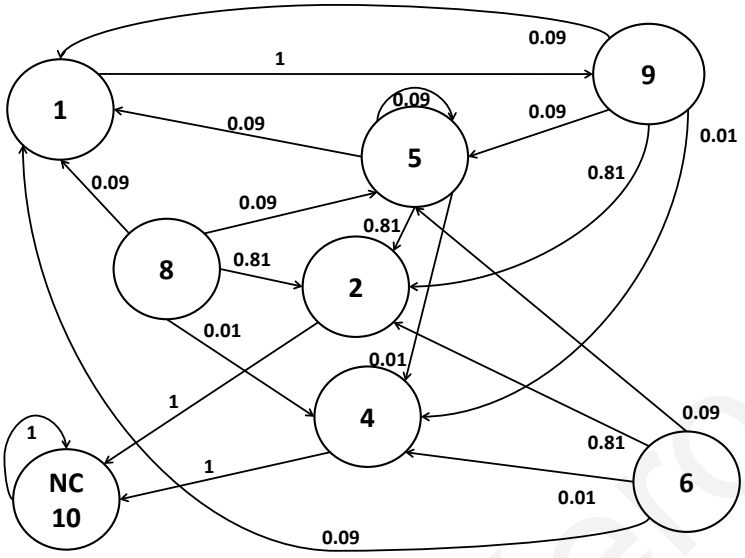

Figure 5.10: Product Markov chain H_p, with state 1 corresponding to $(1/\alpha, 1'/\alpha)$ and state 10 corresponding to the *NC*-state.

$$\mathcal{A}_{H_p} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0.09 & 0.09 & 0 & 0.09 & 0.09 & 0 \\ 0 & 0 & 0 & 0 & 0.81 & 0.81 & 0 & 0.81 & 0.81 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.01 & 0.01 & 0 & 0.01 & 0.01 & 0 \\ 0 & 0 & 0 & 0 & 0.09 & 0.09 & 0 & 0.09 & 0.09 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Note that the initial distribution of the product Markov chain H_p is $\pi[0] = [1, 0, 0, ..., 0]'$ which indicates that we start in state 1 (corresponding to $(1/\alpha, 1'/\alpha)$) with probability one. Note that at the next step, $\pi[1] = [0, 0, ..., 1, 0]'$ because we move to state 9 (corresponding to $(2/\beta, 2'\beta)$) with probability one. Thus, in the next section to simplify notation we assume (without any loss of generality) that we start in state 9.

5.4.2 Conditions for the Upper Bound to tend to zero

We can categorize the states of the Markov chain H_p in two main categories, the absorbing state (NC-state) and the remaining states. We can express the probability of having the same sequence after n steps, as the probability of staying in states other than the NC state of the Markov chain. If we denote the probability distribution state vector of the MC after n steps by π_n, we have

$$\pi_{n+1} = \mathcal{A}_{H_p} \cdot \pi_n$$

with π_0 representing the initial state probability distribution. We also define the probability of being in the NC state, as $c_n = C \cdot \pi_n$, where C, is a row indicator vector with a single "1" on the entry corresponding to the NC state, and zero's everywhere else. Then, we can write the probability of having the same sequence after n steps as

$$P(\text{same sequence after } n \text{ steps}) = 1 - c_n. \tag{5.17}$$

We now turn our attention to cases where $c_n \to 1$ (in fact, exponentially). Suppose that $\mathcal{A}_{H_p}$ has unique eigenvalue $\lambda_1 = 1$ and the remaining eigenvalues $\lambda_2, \lambda_3, ..., \lambda_n$ satisfy $1 > ||\lambda_2|| \geq ||\lambda_3|| \geq ... \geq ||\lambda_n||$; then, there exist a unique stationary distribution π on Q_{H_p} [39]. Moreover, given an initial distribution π_0 and state $q_i \in Q_{H_p}$ (note that $q_{|Q_{H_p}|} = NC$), there is a constant $M_i > 0$, which corresponds to state q_i ($i \in \{1, 2, ..., |Q_{H_P}|\}$), such that:

$$\pi_n(i) \leq \pi(i) + M_i \cdot n^{J-1} \cdot ||\lambda_2||^{n-J+1} , \tag{5.18}$$

where J is the size of the largest Jordan block of $\mathcal{A}_{H_p}$.

Finally for M_{NC} (the constant corresponding to state NC) and π (the unique stationary distribution column vector with $\pi(i) = 0$, for $q_i \neq q_{NC}$, and $\pi(q_{NC}) = 1$), we have $P(\text{same sequence after } n \text{ steps}) \leq M_{NC} \cdot n^{J-1} \cdot ||\lambda_2||^{n-J+1}$.

In the special case that $\mathcal{A}_{H_p}$ is diagonalizable, then $J = 1$, and

$$P(\text{same sequence after } n \text{ steps}) \leq M_{NC} \cdot ||\lambda_2||^n. \tag{5.19}$$

Incorporating all of this in Eq. (5.15), the upper bound can be written as:

$$P(\text{error at } n) \leq \sqrt{P_{(1)} \cdot P_{(2)} \cdot M_{NC}} \cdot (d \cdot ||\lambda_2||)^{n/2} \tag{5.20}$$

When $\lambda_2 < 1/d$, $\lambda_1 = 1$ is unique, and the NC state is present, then $\lim_{n\to\infty} P(\text{error at } n) = 0$.

Example 17. *Using Eq. (5.20), we are able to find an upper bound for H_p in Fig. 5.10 with $\lambda_2 = 0.3484$ and $D = 2$ which means that*

$$P(\textit{error at } n) \leq (0.5 \cdot \sqrt{M_{NC}}) \cdot (0.6968)^{n/2} .$$

Since $\lambda < 1/D$, the upper bound tends to zero exponentially with the number of observations. The next step is to find M_{NC}, i.e., M_{10} (as $NC = q_{10}$). For initial state distribution π_0 such that $\pi_0(i) = 0$, for $q_i \neq q_9$, and $\pi_0(9) = 1$, we can express this initial state distribution as a linear combination of the right eigenvectors (V_1, V_2, V_3), that correspond to the three nonzero eigenvalues of the matrix ($\lambda_1 = 1$, $\lambda_2 = 0.3484$, and $\lambda_3 = -0.2584$). Then, we can write the initial probability for the NC state as:

$$\pi_0(10) = 1 \cdot V_1(10) + 2.51 \cdot V_2(10) - 1.82 \cdot V_3(10). \tag{5.21}$$

From Eq. (5.21), we obtain the following inequality:

$$P(\textit{same sequence after n steps}) = 1 - \pi_n(10)$$

$$= 2.07 \cdot (0.3484)^n - 1.07 \cdot (-0.25)^n$$

$$< ||2.07 + 1.07|| \cdot (0.3484)^n$$

so that

$$P(\textit{same sequence after n steps}) \leq 3.14 \cdot (0.3484)^n \tag{5.22}$$

(i.e., $M_{NC} = 3.14$). Then, from Eq. (5.20), we can write

$$\begin{aligned} P(\textit{error at } n) &< (0.5 \cdot \sqrt{3.14}) \cdot (0.6968)^{n/2} \\ &< 0.886 \cdot (0.6968)^{n/2} . \end{aligned} \tag{5.23}$$

In Fig. 5.11, we plot the upper bound in Eq. (5.23), together with the actual probability of error (obtained after exhaustive calculation of each possible sequence of observations with length ranging from 1 to 13). As expected, the upper bound goes to zero eventually. It is worth pointing out that the given discrete event system does not belong to the class of logically diagnosable systems. In this case, our approach allows us to conclude that we

are able to diagnose the error with increasingly smaller probability of error, exponentially decreasing with the number of observations.

In this example we found an upper bound on the probability of error, when classifying a sequence of outputs between the two given HMMs. The complexity of computing the exact probability of error is an exponential function of n (it is of $O(d^n n(|Q^{(1)}|^2 + |Q^{(2)}|^2))$*). Applying our method, we lower this complexity by finding an upper bound for an arbitrarily large number of n observations. We can compute the upper bound with complexity* $(|Q'^{(1)}| \times |Q'^{(2)}|)^3 = (|Q^{(1)}| \times |E^{(1)}| \times |Q^{(2)}| \times |E^{(2)}|)^3$ *using eigenvalue-eigenvector decomposition of the transition matrix* $\mathcal{A}_{H_p}$ *to obtain the steady-state of the product Markov chain.*

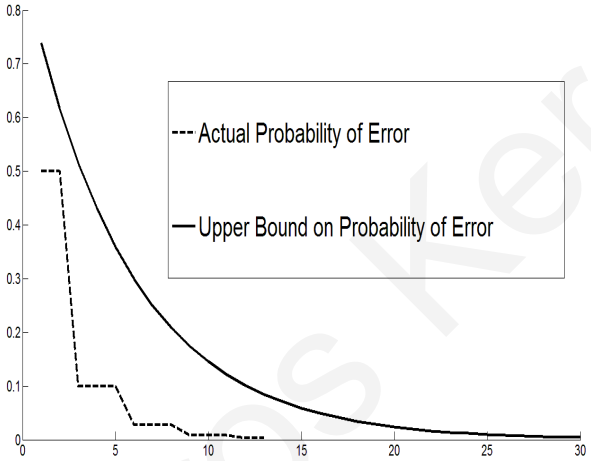

Figure 5.11: Actual probability of error (dashed line) and upper bound (continuous line).

5.5 Classification Rule Based on Empirical Frequencies of Event Sequences

In this section we first define a suboptimal rule based on the empirical frequencies of events and then find an upper bound on the probability of error for this rule. We are also able to give necessary and sufficient conditions under which, the upper bound using this rule tends asymptotically to zero.

Definition 48. *(Fraction of times event* e_i *appears* $(m_n(e_i))$*). Suppose we are given an observation sequence of length n* $(Y_1^n = y[1] \cdots y[n])$*. We define* $m_n(e_i) = \frac{1}{n} \sum_{t=1}^{n} g_{e_i}(y[t])$*, where*

$$g_{e_i}(y[t]) = \begin{cases} 1, & \text{if } y[t] = e_i, \\ 0, & \text{otherwise.} \end{cases}$$

In other words, $m_n(e_i)$ is the fraction of times event e_i appears in observation sequence Y_1^n.

Definition 49. *(Distance in variation $d_V(v, v')$ between two probability vectors v, v'). The distance in variation [14] between two $|E|$-dimensional probability vectors v, v' is defined as*

$$d_V(v, v') = \frac{1}{2} \sum_{j=1}^{|E|} |v(j) - v'(j)| \geq 0 \ ,$$

where $v(j)$ ($v'(j)$) is the jth entry of vector v (v').

Let the stationary emission probabilities for HMM $S^{(1)}$ ($S^{(2)}$) be denoted by the $|E|$-dimensional vector $\pi_e^{(1)} = [\pi_{e_1}^{(1)}, ..., \pi_{e_{|E|}}^{(1)}]'$ (respectively by $\pi_e^{(2)} = [\pi_{e_1}^{(2)}, ..., \pi_{e_{|E|}}^{(2)}]'$). Then, we have $d_V(\pi_e^{(1)}, \pi_e^{(2)}) = \frac{1}{2} \sum_{j=1}^{|E|} |\pi_{e_j}^{(1)} - \pi_{e_j}^{(2)}|$.

Definition 50. *(Empirical Rule). Given two irreducible and aperiodic HMMs $S^{(1)}$ and $S^{(2)}$ and a sequence of observations $Y_1^n = y[1]y[2] \cdots y[n]$, we perform classification using the following suboptimal rule. We first compute $m_n = [m_n(e_1), m_n(e_2), \cdots, m_n(e_{|E|})]'$ as in Definition 48. We then set $\theta = \frac{1}{2} d_V(\pi_e^{(1)}, \pi_e^{(2)})$, where $\pi_e^{(j)}$, $j \in \{1, 2\}$, is the stationary emission probability vector for $S^{(j)}$, and compare*

$$d_V(m_n, \pi_e^{(1)}) \gtrless \theta \ . \tag{5.24}$$

We decide in favor of $S^{(1)}$ ($S^{(2)}$) if the right (left) quantity is larger.

Remark: The empirical rule is a suboptimal rule, which means that even if we compute exactly the probability of error using the empirical rule, this remains an upper bound on the probability of error using the optimal rule (in Section 2.4). In subsequent sections of this paper we obtain a bound on the probability of error using the empirical rule. This bound generally is not tight.

Remark: Using the empirical rule has some advantages over the optimal rule. In Section 5.5.2 we provide necessary and sufficient conditions for a bound on the probability of error using the empirical rule to be asymptotically tight. These conditions can be verified with low computational complexity (polynomial complexity). Another advantage is that the system needs to keep only the number of events that are observed and not the whole observation sequence. This can lead to lower memory requirements for the system.

5.5.1 Upper Bound on the Probability of Misclassification using the Empirical Rule

The following theorem is discussed thoroughly in the remainder of this chapter.

Theorem 12. *(*Upper bound on probability of error using the empirical rule*). Consider classification among two HMMs $S^{(j)} = (Q^{(j)}, E^{(j)}, \Delta^{(j)}, \Lambda^{(j)}, \pi_0^{(j)})$, $j = 1, 2$, with corresponding Markov chains $MC^{(j)} = (Q^{(j)}, A^{(j)}, \pi_0^{(j)})$ that are irreducible and aperiodic. If $d_V(\pi_e^{(1)}, \pi_e^{(2)}) > 0$ (where $\pi_e^{(j)}$ is the stationary emission probability vector for HMM $S^{(j)}$, see Definition 47), then we can find a function $F(n)$ (defined in Eqs.* (5.26) *and* (5.27)*) that is exponentially decreasing in the number of steps n such that*

$$P(\text{Error after } n \text{ observations using the empirical rule}) \leq F(n) \,. \tag{5.25}$$

Example 18. *We apply the above empirical rule to the two HMM models $S^{(1)}$ and $S^{(2)}$ in Fig. 2.6. First, we compute $\theta = \frac{1}{2} d_V(\pi_e^{(1)}, \pi_e^{(2)}) = 0.2031$. Then, we compute the bound on the probability of error, which is exponentially decreasing and described by $F(n) = K \times e^{-an}$, with $K = 1.4574$, $a = 0.002347$. Notice that in this example the bound is not useful for $n \leq 1500$, because it is greater than one.*

Now we discuss and prove Theorem 12. First we define a function of the states of the underlying Markov chain of the two HMMs $S^{(1)}$ and $S^{(2)}$, that counts the occurrences of each event $e_i \in E$, with which we arrive at that state. This is not necessarily possible in $S^{(j)}$, $j \in \{1, 2\}$, because in general we can reach a state via different events. The reason we need to define a function of the states is so that we can analyze the empirical rule (Definition 50) and using existing techniques for Markov chain analysis. Therefore, we introduce new enhanced models $\widetilde{S}^{(1)}$ and $\widetilde{S}^{(2)}$ in which each state can be reached with a single (specific) event. We then prove that the stationary emission probabilities for $\widetilde{S}^{(j)}$ and $S^{(j)}$ for $j \in \{1, 2\}$ are equal and we show how to obtain the upper bound on the probability of error in Theorem 13.

In our analysis we will deal with classification between two competing HMM models. First, we obtain, for each of the given HMMs, an enhanced construction that allows us to discriminate the transition to the same state but via different events. We prove that our enhanced construction inherits the properties of irreducibility and aperiodicity (the two conditions needed to apply Theorem 13) from the corresponding original HMM. The two enhanced HMM models are denoted by

$\widetilde{S}^{(j)} = \{\widetilde{Q}^{(j)}, E, \widetilde{\Delta}^{(j)}, \widetilde{\Lambda}^{(j)}, \widetilde{\pi_0}^{(j)}\}$, $j = \{1, 2\}$. The enhanced construction creates replicas of each state, depending on the event via which one reaches this state. Thus, for each state $q_h \in Q^{(j)}$, we create states $q_{h,e_i} \in \widetilde{Q^{(j)}}$, $e_i \in E$, to represent that we reach state $q_h \in Q^{(j)}$ under the output symbol e_i. Clearly, we end up with at most $|\widetilde{Q}^{(j)}| = |Q| \times |E|$ states.

The following discussion applies to each original HMM and its enhanced model (we drop j, $j \in \{1, 2\}$, to simplify notation). In the state probability vectors $\pi[k]$, $\widetilde{\pi}[t]$, where t is the current state epoch, states are indexed in the order shown below

$$\pi[t] = \begin{bmatrix} \pi[t](q_1) \\ \pi[t](q_2) \\ \vdots \\ \pi[t](q_{|Q|}) \end{bmatrix}, \widetilde{\pi}[t] = \begin{bmatrix} \widetilde{\pi}[t](q_{1,e_1}) \\ \widetilde{\pi}[t](q_{1,e_2}) \\ \vdots \\ \widetilde{\pi}[t](q_{1,e_{|E|}}) \\ \widetilde{\pi}[t](q_{2,e_1}) \\ \vdots \\ \widetilde{\pi}[t](q_{|Q|,e_{|E|}}) \end{bmatrix}.$$

The matrix $\widetilde{A}_{e_i}$, $e_i \in E$, satisfies $\widetilde{A}_{e_i}(q_{h,e_i}, q'_{h,e'_i}) = A_{e_i}(q_h, q'_h)$, $\forall e'_i \in E$ and $\forall q_h, q'_h \in Q$ (zero otherwise). We also have for $e_i \in E$ and $q_{h,e_i}, q'_{h,e'_i} \in \widetilde{Q}$, $\widetilde{\Lambda}(q'_{h,e'_i}, e_i, q_{h,e_i}) = \widetilde{A}_{e_i}(q_{h,e_i}, e_i, q'_{h,e'_i})$ (zero otherwise). We observe that matrix $\widetilde{A}_{e_i}$ is constructed by blocks of matrix A_{e_i}. If we define row-vector $R_{|E|} = \underbrace{[11\cdots1]}_{|E|-times}$ and let

$$R_{i,|E|} = \underbrace{[0\cdots0\,1\,0\cdots0]}_{\text{single one at } i\text{th position}},$$

then the state transition matrix $\widetilde{A}^{(j)}_{e_i}$ for the enhanced model $\widetilde{S}^{(j)}$ can be written as

$$\widetilde{A}^{(j)}_{e_i} = A^{(j)}_{e_i} \otimes (R^T_{i,|E|} \otimes R_{|E|}).$$

Example 19. *We create the enhanced HMM models $\widetilde{S}^{(1)}$ (shown in Fig. 5.12) and $\widetilde{S}^{(2)}$ for $S^{(1)}$ and $S^{(2)}$ respectively (shown in Fig. 2.6). We note that the underlying state transition matrix, for each enhanced model, is irreducible and aperiodic (as we will see $\widetilde{S}^{(j)}$ will be irreducible and aperiodic as long as $S^{(j)}$ is irreducible and aperiodic). The corresponding $\widetilde{A}^{(1)}_{\alpha}, \widetilde{A}^{(1)}_{\beta}, \widetilde{A}^{(2)}_{\alpha}, \widetilde{A}^{(2)}_{\beta}$ are as follows:*

$$\widetilde{A}_{\alpha}^{(1)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0.50 & 0.50 & 0.50 & 0.50 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \widetilde{A}_{\beta}^{(1)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0.50 & 0.50 & 0.25 & 0.25 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.25 & 0.25 \end{bmatrix},$$

$$\widetilde{A}_{\alpha}^{(2)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0.60 & 0.60 & 0.06 & 0.06 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \widetilde{A}_{\beta}^{(2)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0.40 & 0.40 & 0.04 & 0.04 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.90 & 0.90 \end{bmatrix}.$$

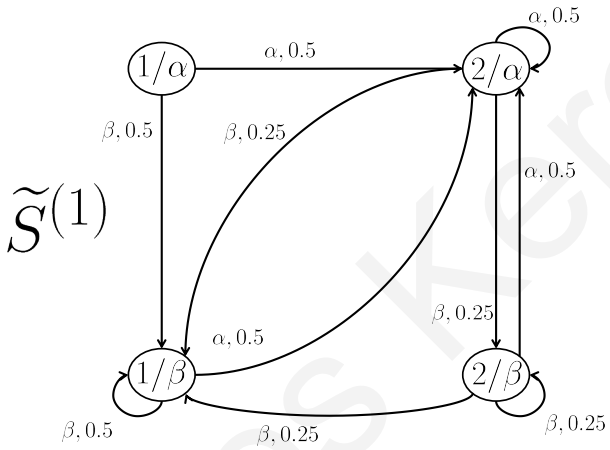

Figure 5.12: Enhanced model $\widetilde{S}^{(1)}$ for HMM model $S^{(1)}$ in Fig. 2.6.

Definition 51. *(Hoeffding's Inequality on Enhanced HMM Model). Consider an enhanced HMM,* $\widetilde{S}^{(j)} = \{\widetilde{Q}^{(j)}, E, \widetilde{\Delta}^{(j)}, \widetilde{\Lambda}^{(j)}, \widetilde{\pi}_0{}^{(j)}\}$, $j \in \{1, 2\}$, *with an underlying irreducible and aperiodic finite-state Markov chain with* $|E|$ *events and transition matrix* $\widetilde{A}^{(j)}$. *Assuming the Markov chains that correspond to the enhanced models* $\widetilde{S}^{(j)}$, $j = 1, 2$, *are irreducible and aperiodic, we denote their stationary distributions by* $\widetilde{\pi}^{(j)} > 0$ *and stationary emission distribution for events* $e_i \in E$ *as* $\widetilde{\pi}_e^{(j)} > 0$.

Using the enhanced models ($\widetilde{S}^{(1)}$ *and* $\widetilde{S}^{(2)}$*) for each* $e_i \in E$, *we define the indicator functions* $f_{e_i}(q_{h,e_j})$, $\forall q_{h,e_j} \in \widetilde{Q}$, *as*

$$f_{e_i}(q_{h,e_j}) = \begin{cases} 1, & \text{if } e_j = e_i, \\ 0, & \text{otherwise.} \end{cases}$$

Let $m_n(e_i) = \frac{1}{n} \sum_{t=1}^{n} f_{e_i}(q[t])$, *i.e., the* $|E|$*-dimensional vector* $m_n = [m_n(e_1), m_n(e_2), ..., m_n(e_{|E|})]'$ *denotes the empirical frequencies with which each event appears in the given observation win-*

dow of length n. Let M_j, be the smallest integer such that $(\widetilde{A}^{(j)})^{M_j} > 0$, element-wise, and $\lambda_j = \min_{l,l'}\{\frac{(\widetilde{A}^{(j)})^{M_j}(l,l')}{\widetilde{\pi}^{(j)}(l)}\}$, where $\widetilde{\pi}^{(j)}(l)$ is the stationary distribution of $\widetilde{S}^{(j)}$. As long as the enhanced model $\widetilde{S}^{(j)}$ is irreducible and aperiodic, it can be shown [20, 21] that the following is true for $n > \frac{2M_j}{\lambda_j \epsilon}$, and for each event e_i ($1 \leq i \leq |E|$):

$$Pr(m_n(e_i) - \widetilde{\pi}_e^{(j)}(e_i) \geq \epsilon) \leq \underbrace{\exp\left(-\frac{{\lambda_j}^2(n\epsilon - \frac{2M_j}{\lambda_j})^2}{2n{M_j}^2}\right)}_{F^{(j)}(n)}. \tag{5.26}$$

In order to use Eq. (5.26), we need $\widetilde{S}^{(j)}$ to correspond to an irreducible (Definition 28) and aperiodic (Definition 29) Markov chain. We now show that $\widetilde{S}^{(j)}$ is irreducible and aperiodic iff $S^{(j)}$ is irreducible and aperiodic. Also, we establish that $\widetilde{\pi}_e^{(j)} = \pi_e^{(j)}$.

We provide below the proof for one direction (if $S^{(j)}$ is irreducible and aperiodic then $\widetilde{S}^{(j)}$ is also irreducible and aperiodic); the other direction can be proved by similar reasoning).

Lemma 7. *If HMM $S^{(j)} = (Q^{(j)}, E^{(j)}, \Delta^{(j)}, \Lambda^{(j)}, \pi_0^{(j)})$ is irreducible (Definition 28), then the enhanced HMM $\widetilde{S}^{(j)} = \{\widetilde{Q}^{(j)}, E, \widetilde{\Delta}^{(j)}, \widetilde{\Lambda}^{(j)}, \widetilde{\pi_0}^{(j)}\}$ is also irreducible.*

Proof. We prove irreducibility by establishing the property that any state $q_{h,e_i} \in \widetilde{Q}^{(j)}$ that does not belong to the set of strongly connected states (Definition 28), may exhibit outgoing transitions but will have no incoming transition. Consider in the enhanced model the set of states

$$\widetilde{Q}_{ss} = \{q_{m,e} \in \widetilde{Q} | \exists q_{m'} \in Q, \exists e \in E \text{ s.t. } \Lambda(q_{m'}, e, q_m) > 0\}$$

Since the set of states Q in the original system is strongly connected, we can easily show that the states in $\widetilde{Q}_{ss}$ are strongly connected: given $q_{m,e}, q_{m',e'} \in \widetilde{Q}$ we can find a path to connect them as follows: Let $q_{m''}$ be such that $\Lambda(q_{m''}, e', q_{m'}) > 0$. Then, we can find a path

$$q_m \xrightarrow{e_{i_1}} q_{i_1} \xrightarrow{e_{i_2}} q_{i_2} \rightarrow$$

(because the original HMM is irreducible). Therefore

$$q_{m,e} \xrightarrow{e_{i_1}} q_{i_1,e_1} \xrightarrow{e_{i_2}} q_{i_2,e_{i_2}} \rightarrow \cdots \xrightarrow{e_{i_t}} q_{i_t} = q_{m''} \xrightarrow{e'} q_{m',e'}$$

is a path that connects $q_{m,e} \in \widetilde{Q}$ to $q_{m',e'} \in \widetilde{Q}$. We finally conclude that the states that do not belong to the set of strongly connected states, have only outgoing transitions.

Therefore, by choosing an appropriate initial distribution function that excludes all these transient states,[5] we can ensure that all of these transient states will never be visited. □

Lemma 8. *If HMM* $S^{(j)} = (Q^{(j)}, E^{(j)}, \Delta^{(j)}, \Lambda^{(j)}, \pi_0^{(j)})$ *is aperiodic (Definition 29), then the enhanced HMM* $\widetilde{S}^{(j)} = \{\widetilde{Q}^{(j)}, E, \widetilde{\Delta}^{(j)}, \widetilde{\Lambda}^{(j)}, \widetilde{\pi_0}^{(j)}\}$ *is also aperiodic.*

Proof. We now show that if the enhanced model $\widetilde{S}^{(j)}$ is periodic with period k, this contradicts the fact that $S^{(j)}$ is aperiodic. Suppose that $\widetilde{S}^{(j)}$ is periodic with period k (Definition 29 and Lemma 2). This means we can group all possible states of $\widetilde{S}^{(j)}$ to k groups $(\widetilde{C}_1, \widetilde{C}_2, \cdots, \widetilde{C}_k)$ such that for a state $q_{l,e} \in \widetilde{C}_m$, there exist one-step transitions only to states in $\widetilde{C}_{m'}$, where $m' = m + 1 \mod k$.

Due to the construction of enhanced models, the outgoing behaviour of $q_{l,e}$ states $\forall e \in E$ are copies of the outgoing behaviour of $q_l \in Q$. We can easily see that if there exists $q_{l,e} \in \widetilde{Q}$, that belongs to $\widetilde{C}_m$, then also $q_{l,e'} \in \widetilde{Q}$ belongs to $\widetilde{C}_m$, for all $e, e' \in E$ (due to the same outgoing behaviour). Thus, we can also group $q \in Q$ into $C_i, i \in \{1, 2, ..., k\}$, classes. Thus $S^{(j)}$ is periodic, with period k, which is a contradiction. □

We now show that in the enhanced model $\widetilde{S}^{(j)}$, the stationary emission probabilities of each event are consistent with the original model $S^{(j)}$ for $j = 1, 2$.

Lemma 9. *The computed stationary emission probabilities for symbols in the enhanced model* $\widetilde{S}^{(j)}$, $j = 1, 2$ *which is denoted respectively by* $\widetilde{\pi}_e^{(j)}$ *is identical to* $\pi_e^{(j)}$ *corresponding to* $S^{(j)}$.

Proof. Let $\widetilde{\pi}_s^{(j)}$ denote the steady-state distribution vector in the enhanced model j. Then, we have that under each model

$$\pi_s^{(j)} = (I_n \otimes R_{|E|}) \times \widetilde{\pi}_s^{(j)}, j \in \{1, 2\}.$$

For $S^{(j)}$ and $\forall e_i \in E$, the stationary emission probability $\pi_e^{(j)}(e_i)$ can be expressed as $\pi_e^{(j)}(e_i) = R_n \times (A_{e_i}^{(j)} \times \pi_s^{(j)})$, whereas for the enhanced model $\widetilde{S}^{(j)}$, we have

[5] We can always do this since subsequent behavior of the enhanced model does not depend on whether we start from state $q_{h,e}$ or $q_{h',e'}$.

$$
\begin{aligned}
\widetilde{\pi}_e^{(j)}(e_i) &= R_{n|E|} \times \widetilde{A}_{e_i}^{(j)} \times \widetilde{\pi}_s^{(j)} \\
&= R_{n|E|} \times (A_{e_i}^{(j)} \otimes (R_{i,|E|}^T \otimes R_{|E|})) \times \widetilde{\pi}_s^{(j)} \\
&= (R_n \otimes R_{|E|}) \times (A_{e_i}^{(j)} \otimes (R_{i,|E|}^T \otimes R_{|E|})) \times \widetilde{\pi}_s^{(j)} \\
&= (R_n \times A_{e_i}^{(j)}) \otimes (R_{|E|} \times (R_{i,|E|}^T \otimes R_{|E|})) \times \widetilde{\pi}_s^{(j)} \\
&= ((R_n \times A_{e_i}^{(j)}) \otimes R_{|E|}) \times \widetilde{\pi}_s^{(j)} \\
&= (R_n \times A_{e_i}^{(j)}) \otimes (R_1 \times R_{|E|}) \times \widetilde{\pi}_s^{(j)} \\
&= R_n \times (A_{e_i}^{(j)} \otimes R_{|E|}) \times \widetilde{\pi}_s^{(j)}.
\end{aligned}
$$

Moreover, we have

$$
\begin{aligned}
\pi_e^{(j)}(e_i) &= R_n \times (A_{e_i}^{(j)} \times \pi_s^{(j)}) \\
&= R_n \times (A_{e_i}^{(j)} \times (I_n \otimes R_{|E|}) \times \widetilde{\pi}_s^{(j)}) \\
&= R_n \times (A_{e_i}^{(j)} \otimes R_1) \times (I_n \otimes R_{|E|}) \times \widetilde{\pi}_s^{(j)} \\
&= R_n \times (A_{e_i}^{(j)} \times I_n) \otimes (R_1 \times R_{|E|}) \times \widetilde{\pi}_s^{(j)} \\
&= R_n \times (A_{e_i}^{(j)} \otimes R_{|E|}) \times \widetilde{\pi}_s^{(j)} \\
&= \widetilde{\pi}_e^{(j)}(e_i),
\end{aligned}
$$

which allows us to conclude that $\pi_e^{(j)}(e_i) = \widetilde{\pi}_e^{(j)}(e_i)$, $\forall e_i \in E$. □

Given two HMMs $S^{(1)}$ and $S^{(2)}$ (each irreducible and aperiodic), we construct the corresponding enhanced HMM models $(\widetilde{S}^{(1)}, \widetilde{S}^{(2)})$ with underlying irreducible and aperiodic Markov chains $\widetilde{MC}^{(1)} = (\widetilde{Q}^{(1)}, \widetilde{A}^{(1)}, \widetilde{\pi}_0{}^{(1)})$ and $\widetilde{MC}^{(2)} = (\widetilde{Q}^{(2)}, \widetilde{A}^{(2)}, \widetilde{\pi}_0{}^{(2)})$ (i.e., this means that $\widetilde{A}^{(1)}$ and $\widetilde{A}^{(2)}$ are primitive matrices). Suppose we have $d_V(\widetilde{\pi}_e^{(1)}, \widetilde{\pi}_e^{(2)}) > 0$ or equivalently $d_V(\pi_e^{(1)}, \pi_e^{(2)}) > 0$ (Lemma 9). Then, if we apply the empirical rule and use Hoeffding's inequality (Definition 51), we obtain the upper bound on the probability of error using the empirical rule (see Theorem 13) where $F(n)$ is given by

$$
F(n) = \max\{F^{(1)}(n), F^{(2)}(n)\} \,. \tag{5.27}
$$

Proof

We consider two error cases :

[3] In the following discussion we use the following well known properties of the Kronecker product: 1. $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$; 2. $(A \otimes B) \otimes C = A \otimes (B \otimes C)$ for matrices A, B, C, D of appropriate dimensions [5].

Case 1: Decide $S^{(1)}$ when the system is $S^{(2)}$;

Case 2: Decide $S^{(2)}$ when the system is $S^{(1)}$.

Case 1:

The decision of $S^{(1)}$ is equivalent to the event

$$H^{(1)} : d_V(m_n, \pi_e^{(1)}) < \theta,$$

which necessarily implies $d_V(m_n, \pi_e^{(2)}) \geq \theta$ (for $\theta = \frac{1}{2} d_V(\pi_e^{(1)}, \pi_e^{(2)})$). Otherwise, we reach a contradiction, because $d_V(m_n, \pi_e^{(2)}) < \theta$ and $d_V(m_n, \pi_e^{(1)}) < \theta$ imply

$$\begin{aligned} d_V(\pi_e^{(1)}, \pi_e^{(2)}) &< d_V(m_n, \pi_e^{(1)}) + d_V(m_n, \pi_e^{(2)}) \\ &= 2\theta \\ &= d_V(\pi_e^{(1)}, \pi_e^{(2)}) \, . \end{aligned}$$

Thus, $H^{(1)}$ implies $d_V(m_n, \pi_e^{(1)}) \geq \theta$, which requires

$$H_k^{(1)} : \{\exists e_k \in E \text{ such that } |m_n(e_k) - \pi_e^{(2)}(e_k)| > \frac{\theta}{|E|}\} \, .$$

Therefore, we have to consider two different subcases:

a) $m_n(e_k) - \pi_e^{(2)}(e_k) > 0$,

b) $m_n(e_k) - \pi_e^{(2)}(e_k) < 0$.

The probability of error for Case 1 and subcase a), after n observations, is

$$\begin{aligned} P(\text{error at } n, \text{ Case 1}) &= P(H^{(1)}|S^{(2)}) P(S^{(2)}) \\ &\leq P(H_k^{(1)}|S^{(2)}) P(S^{(2)}) \\ &\leq F^{(2)}(n) P(S^{(2)}) \, , \end{aligned}$$

where $P(H_k^{(1)}|S^{(2)}) \equiv \Pr(m_n(e_k) - \pi_e^{(2)}(e_k) > \frac{\theta}{|E|}) \leq F_n^{(2)}$, for $\epsilon = \frac{\theta}{|E|}$.

In case 1a) we can immediately apply Eq. (5.26), but in case 1b) in order to find a positive measure we choose to count the number of appearances of all elements in $k^c = \{e \in E \mid \text{ s.t. } e \neq e_k\}$, i.e., all possible events except $e_k \in E$. Then $m_n(e_{k^c}) - \pi_e^{(2)}(e_{k^c}) = (1 - m_n(e_k)) - (1 - \pi_e^{(2)}(e_k)) > 0$, and we can apply Eq. (5.26), which leads us to the same bound.

Case 2:

With the same reasoning as in Case 1, we establish the following inequality

$$\begin{aligned} P(\text{error at } n,\ \text{Case 2}) &= P(H^{(2)}|S^{(1)})P(S^{(1)}) \\ &\leq P(H^{(2)}_{k'}|S^{(1)})P(S^{(1)})) \\ &\leq F^{(1)}(n)P(S^{(1)}) \ , \end{aligned}$$

where $H^{(2)} : d_V(m_n, \pi_e^{(1)}) > \theta$, which implies that $H^{(2)}_{k'}$: {There exists at least one $e_{k'}$ such that $|m_n(e_{k'}) - \pi_e^{(1)}(e_{k'})| > \frac{\theta}{|E|}$, where $e_{k'} \in E$ }. The claim follows using similar arguments as in Case 1.

Finally, we prove that

$$\begin{aligned} P(\text{error at } n) &= P(\text{error at } n,\ \text{Case 1}) + P(\text{error at } n,\ \text{Case 2}) \\ &= P(H^{(1)}|S^{(2)})P(S^{(2)}) + P(H^{(2)}|S^{(1)})P(S^{(1)})) \\ &\leq P(H^{(1)}_{k}|S^{(2)})P(S^{(2)}) + P(H^{(2)}_{k'}|S^{(1)})P(S^{(1)})) \\ &\leq F(n)(P(S^{(1)}) + P(S^{(2)})) \equiv F(n) \end{aligned}$$

This concludes the proof of Theorem 13.

5.5.2 Necessary and Sufficient Conditions for Upper Bound to Tend to Zero

The suboptimal rule provides us with a bound on the probability of error that decreases exponentially with n iff

$$d_V(\pi_e^{(1)}, \pi_e^{(2)}) > 0.$$

This requires at least one (actually two) $e_i \in E$, such that

$$\pi_e^{(1)}(e_i) \neq \pi_e^{(2)}(e_i)$$

and is a condition that can be easily checked.

An interesting extension of the proposed empirical rule is to consider the empirical frequencies of events in E^k, i.e., all possible sequences of output symbols $e_j \in E$, of length k. The verification of conditions in this case is more complicated, but one can verify (with polynomial complexity) whether the resulting bound on the probability of error will tend to zero exponentially with the length of the sequence.

Consider $e_{k,i} = e[1]e[2]\cdots e[k] \in E^k$, where $e[j] \in E$. We can compute the stationary probability $\pi_e^{(j)}(e_{k,i})$, as

$$\pi_e^{(j)}(e_{k,i}) = R_n \times (A^{(j)}_{e[k]} \cdots \times A^{(j)}_{e[2]} \times A^{(j)}_{e[1]}) \times \pi_s^{(j)}.$$

We need to find a way to easily compare all possible $\pi_e^{(j)}(e_{k,i}) \in E^k$, for any given k. We can relate this problem to that of probabilistic equivalence between two probabilistic automata[6] [50]. Suppose (for simplicity) that the initial distribution for the two HMMs ($S^{(1)}$ and $S^{(2)}$) is the steady-state distribution. Then, the possible generated observation sequences are simply all event sequences $e_{k,i} \in E^k$, for the given k. These sequences are described by a probability which is identical to the stationary emission probability $\pi_e^{(j)}(e_{k,i})$. Thus, we can always apply the proposed empirical rule iff there exists at least one $e_{k,i} \in E^k$ which is generated with different probabilities from the two HMMs. This can be verified by applying the probabilistic equivalence algorithm in [50], which runs in polynomial time. Another interesting feature of this algorithm is that, if the two systems are not probabilistically equivalent, it outputs a specific event sequence, which is generated with different probability for the two probabilistic automata (HMMs in our case); the length of this sequence is always less than $Q^{(1)} + Q^{(2)} - 1$.

5.6 Application in Probabilistic System Opacity for Discrete Event Systems

Motivated by the increased reliance of many applications on shared cyber-infrastructures (ranging from defense and banking to health care and power distribution systems), various notions of *security and privacy* have received considerable attention from researchers. A number of such notions focus on characterizing the *information flow* from the system to the intruder [18]. *Opacity* falls in this category and aims at determining whether a given system's *secret* behavior (i.e., a subset of the behavior of the system that is considered critical and is usually represented by a predicate) is kept opaque to outsiders [6, 41]. More specifically, this requires that the intruder

[6]Two probabilistic automata are equivalent if for any string s, the two automata accept s with equal probability. We can use the probabilistic equivalence algorithm also for two HMMs, which is our case.

(modeled as a passive observer[7] of the system's behavior) never be able to establish the truth of the predicate.

Now we consider an application that combines the notions of classification among HMMs and opacity. Probabilistic system opacity considers the following setting: we are given m HMMs, denoted by $S^{(i)}$ for $i \in \{1, 2, ..., m\}$. The prior probability of $S^{(i)}$ is P_i, $P_i > 0$, and the prior probabilities satisfy $\sum_{i=1}^{m} P_i = 1$. A user is supposed to choose one of these models, say $S^{(i)}$, and would like to keep an observer (eavesdropper) confused about the chosen HMM, for any sequence that might occur in the chosen HMM, regardless of the sequence of observations generated by it and regardless of how long the observer is willing to wait. This means that for any observation sequence that can be generated by the chosen HMM, the observer must not be able to take a decision about the chosen HMM, at least not with absolute certainty or with certainty that tends asymptotically to unity.

The formal definition of probabilistic system opacity follows.

Definition 52. *(Probabilistic System Opacity). Consider a set of m HMMs, $S^{(i)} = (Q^{(i)}, E^{(i)}, \Delta^{(i)}, \Lambda^{(i)}, \pi_0^{(i)})$, for $i \in \{1, ..., m\}$, with corresponding Markov chains $MC^{(i)} = (Q^{(i)}, A^{(i)}, \pi_0^{(i)})$ that are irreducible and aperiodic and with initial probability distribution $\pi_0^{(i)} > 0$. Probabilistic system opacity holds if there exists an $\alpha_0 > 0$, such that for any chosen $S^{(i)}$ and for any observation sequence ω that could be generated by $S^{(i)}$, we have*

$$\alpha(\omega) = \frac{\sum_{k \neq i} P_k P_\omega^{(k)}}{\sum_{k=1}^{m} P_k P_\omega^{(k)}} \geq \alpha_0 \ .$$

Remark: Note that in Definition 52, we take as initial probability distribution $\pi_0^{(i)}$ a strictly positive vector. This means that all initial states are possible among all m HMMs. If this is the case, we will argue that probabilistic system opacity can be verified with polynomial complexity. The complexity and the verification algorithm in the more general case, where $\pi_0^{(i)}$ is not necessarily strictly positive, remains an open problem.

[7] A passive observer is one that does not have any decision-making authority in the system (i.e., it cannot influence the operation of the system).

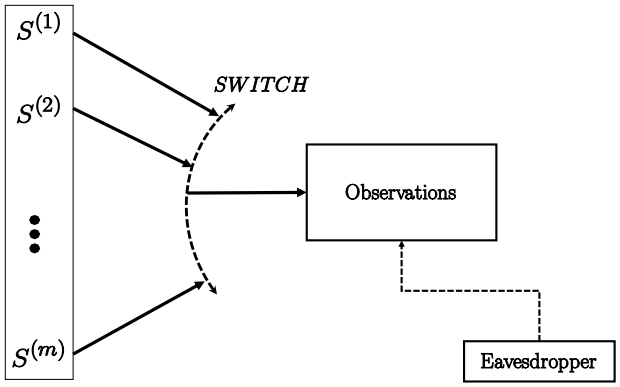

Figure 5.13: We choose one HMM out of m different HMMs, $S^{(1)}, S^{(2)}, ..., S^{(m)}$. An Eavesdropper knows the exact structure of these HMMs and also observes the observation sequence that is generated. The designer's aim is to keep the Eavesdropper confused about the true identity of the HMM that generates the observation.

5.6.1 Polynomial Verification of Probabilistic Opacity

Definition 53. *Probabilistic Equivalence for HMMs [50]. Two HMMs,* $S^{(i)} = (Q^{(i)}, E^{(i)}, \Delta^{(i)}, \Lambda^{(i)}, \pi_0^{(i)})$*, with* $E = E^{(1)} = E^{(2)}$ *for* $i \in \{1, 2\}$ *are probabilistically equivalent iff for any string* $\omega \in L(S)$ $(L(S) = L(S^{(1)}) \cup L(S^{(2)}))$ *the two HMMs, accept the string with equal probability.*

Remark: Two HMMs can be tested for probabilistic equivalence with an algorithm of polynomial complexity [50].

Remark: We can say that the two HMMs are *probabilistically equivalent from steady–state* iff the two HMMs $S^{(i)} = (Q^{(i)}, E^{(i)}, \Delta^{(i)}, \Lambda^{(i)}, \pi_s^{(i)})$, for $i \in \{1, 2\}$, where $\pi_s^{(i)}$ is the steady–state probability are probabilistically equivalent.

In the following definition, we simplify the problem of m HMMs to a problem of two HMMs, which are probabilistically opaque. It will become obvious in the proof of probabilistic system opacity, that the conditions for m HMMs, to be probabilistically opaque, are based on the conditions for two HMMs, to be probabilistically opaque.

Definition 54. *(Pairwise Probabilistic Opacity). Two HMMs,* $S^{(i)} = (Q^{(i)}, E, \Delta^{(i)}, \Lambda^{(i)}, \pi_0^{(i)})$ *for* $i \in \{1, 2\}$ *and prior probabilities*[8] P_1 *and* P_2 *are Probabilistically Opaque if* $(\exists 0 < \alpha_0 < 1/2)$

[8]Usually $P_1 + P_2 = 1$, but in our case we keep the priors as the two HMMs, were part of a system of m HMMs, as it is described in Definition 52. This helps us to avoid notational overhead involving renormalizations of priors (namely, $P_i' = P_i/(P_1 + P_2)$ for $i = 1, 2$) when we deal with two, or more HMMs.

such that

$$(\forall \omega \in L(S^{(1)}) \cup L(S^{(2)})) \text{ we have } \alpha(\omega) \geq \alpha_0 \ ,$$

with

$$\alpha(\omega) = \min\{\frac{P_1 P_\omega^{(1)}}{P_1 P_\omega^{(1)} + P_2 P_\omega^{(2)}}, \frac{P_2 P_\omega^{(2)}}{P_1 P_\omega^{(1)} + P_2 P_\omega^{(2)}}\} \ .$$

Recall that $P_j P_\omega^{(j)}$, $j = 1, 2$, is the probability that observation ω is generated by HMM $S^{(j)}$.

Definition 55. *(Probability of Error Among Two HMMs Tends to Zero). Consider two HMMs, $S^{(i)} = (Q^{(i)}, E^{(i)}, \Delta^{(i)}, \Lambda^{(i)}, \pi_0^{(i)})$, for $i \in \{1, 2\}$, with corresponding Markov chains $MC^{(i)} = (Q^{(i)}, A^{(i)}, \pi_0^{(i)})$ that are irreducible and aperiodic; it is known (see Section 5.5), that if $S^{(1)}$ and $S^{(2)}$ are not probabilistically equivalent from steady–state, then*

$$(\forall \epsilon > 0)(\exists n_0 \in \mathbb{N}) \text{ such that for } n \geq n_0 \ \Pr(\text{error at } n) < \epsilon,$$

where $\Pr(\text{error at } n)$ *is the probability of misclassification for the two HMMs.*

In other words, if the two HMMs are not probabilistically equivalent[9] from steady–state, then the probability of error among the two HMMs tends, at least asymptotically, to zero. Behind this result lies the fact that we are able to discriminate between the two HMM models using a suboptimal decision rule (Definition 50) based on the empirical frequencies of output symbols, as long as the two systems are characterized, at steady-state, by different statistical properties for the occurrence of output symbols or different statistical properties of finite sequences of consecutive output symbols (this means that the two HMMs are not probabilistically equivalent from steady–state). The theoretical analysis in Section 5.5 establishes an upper bound on the misclassification probability, which is described by a function that decreases exponentially with the length of the observation sequence (as long as the two systems are characterized, at steady-state, by different statistical properties for the stationary emission probabilities (Definition 47) or stationary emission probabilities for a finite number of consecutive output symbols).

In order to establish the conditions for probabilistic system opacity, we need to establish the conditions for two probabilistically opaque HMMs. In the following theorem, we explore the necessary and sufficient conditions needed for two HMMs to be probabilistically opaque.

[9]Note that probabilistic equivalence can be checked with polynomial complexity [50].

Theorem 13. *(*Conditions for Pairwise Probabilistic Opacity (Definition 54)*). Consider two HMMs $S^{(j)} = (Q^{(j)}, E^{(j)}, \Delta^{(j)}, \Lambda^{(j)}, \pi_0^{(j)})$, $j = 1, 2$, with corresponding Markov chains $MC^{(j)} = (Q^{(j)}, A^{(j)}, \pi_0^{(j)})$ that are irreducible and aperiodic. These two HMMs are probabilistically opaque iff they are probabilistically equivalent from steady-state.*

Proof. Let us use the following notation:

- $\omega = \omega[1]\omega[2]...\omega[n]$, where $\omega[t] \in E$ for $t \in \{1, ..., n\}$;
- $\mathbf{1}^T = [1...1]$ is a row vector with n identical elements equal to 1;
- $A_\omega^{(1)} = A_{\omega[n]}^{(1)} \cdots A_{\omega[1]}^{(1)}$ and $A_\omega^{(2)} = A_{\omega[n]}^{(2)} \cdots A_{\omega[1]}^{(2)}$;
- For a vector π, $\min\{\pi\}$ is the minimum element of the vector and $\max\{\pi\}$ is the maximum element of the vector;
- $\alpha(\omega) = \min\{\frac{P_1 P_\omega^{(1)}}{P_1 P_\omega^{(1)} + P_2 P_\omega^{(2)}}, \frac{P_2 P_\omega^{(2)}}{P_1 P_\omega^{(1)} + P_2 P_\omega^{(2)}}\}$, where $P_j P_\omega^{(j)}$, $j = 1, 2$, is the probability that observation ω is generated by HMM $S^{(j)}$.

($\rightarrow$). Suppose that the two HMMs are probabilistically opaque; we need to show that the two HMMs are probabilistically equivalent from steady-state. We know that if the probability of error does not tend to zero, then the two HMMs are probabilistically equivalent from steady–state according to the contraposition of Definition 55. It remains to prove that if the two HMMs are probabilistically opaque, then the probability of error among the two HMMs does not tend to zero. If the two HMMs are probabilistically opaque, we argue that the probability of error when trying to classify between $S^{(1)}$ and $S^{(2)}$ based on a sequence of observations satisfies

$$(\exists\ 0 < \alpha_0 < 1\ )(\forall n \in \mathbb{N}) \text{ such that } \Pr(\text{error at } n) \geq \alpha_0\ .$$

This is proved easily because we know that $(\exists \alpha_0)(\forall \omega \in L(S^{(1)}) \cup L(S^{(2)}))$, we have $\min\{\frac{P_1 P_\omega^{(1)}}{P_1 P_\omega^{(1)} + P_2 P_\omega^{(2)}}, \frac{P_2 P_\omega^{(2)}}{P_1 P_\omega^{(1)} + P_2 P_\omega^{(2)}}\} \geq \alpha_0$. Therefore, for each $n \in \mathbb{N}$

$$\begin{aligned}
\Pr(\text{error at } n) &= \sum_{\forall \omega : |\omega| = n} (\min\{P_1 P_\omega^{(1)}, P_2 P_\omega^{(2)}\}) \\
&\geq \sum_{\forall \omega : |\omega| = n} \alpha_0 (P_1 P_\omega^{(1)} + P_2 P_\omega^{(2)}) \\
&= \alpha_0 (P_1 \sum_{\forall \omega : |\omega| = n} P_\omega^{(1)} + P_2 \sum_{\forall \omega : |\omega| = n} P_\omega^{(2)}) = \alpha_0\ .
\end{aligned}$$

This proves that the probability of error does not tend to zero; therefore, the two HMMs are not probabilistically equivalent from steady–state.

($\leftarrow$). Suppose that the two HMMs are probabilistically equivalent from steady–state; then, for any ω, we have

$$\mathbf{1}^T A_\omega^{(1)} \pi_s^{(1)} = \mathbf{1}^T A_\omega^{(2)} \pi_s^{(2)} =: \pi_{\omega,s} \ .$$

We next prove that the two HMMs are Probabilistically Opaque. Four useful inequalities with $i \in \{1, 2\}$ are the following:

$$\begin{aligned} P_\omega^{(i)} =& \mathbf{1}^T A_\omega^{(i)} \pi_0^{(i)} \\ \geq& \mathbf{1}^T A_\omega^{(i)} \min\{\pi_0^{(i)}\} \mathbf{1} \\ \geq& \min\{\pi_0^{(i)}\} \pi_{\omega,s} \ , \end{aligned}$$

$$\begin{aligned} P_\omega^{(i)} =& \mathbf{1}^T A_\omega^{(i)} \pi_0^{(i)} \\ \leq& \max\{\pi_0^{(i)}\} \mathbf{1}^T A_\omega^{(i)} \mathbf{1} \\ \leq& \frac{\max\{\pi_0^{(i)}\}}{\min\{\pi_s^{(i)}\}} \mathbf{1}^T A_\omega^{(i)} \pi_s^{(i)} \\ \leq& \frac{\max\{\pi_0^{(i)}\}}{\min\{\pi_s^{(i)}\}} \pi_{\omega,s} \ . \end{aligned}$$

In summary, we have

$$\min\{\pi_0^{(i)}\} \pi_{\omega,s} \leq P_\omega^{(i)} \leq \frac{\max\{\pi_0^{(i)}\}}{\min\{\pi_s^{(i)}\}} \pi_{\omega,s} \ .$$

From the previous inequalities we can rewrite $\alpha(\omega) = \min\{\frac{P_1 P_\omega^{(1)}}{P_1 P_\omega^{(1)} + P_2 P_\omega^{(2)}}, \frac{P_2 P_\omega^{(2)}}{P_1 P_\omega^{(1)} + P_2 P_\omega^{(2)}}\} \geq \min\{c_1, c_2\}$, where $c_1 < 1$ and $c_2 < 1$, with $c_i = \frac{P_i \min\{\pi_0^{(i)}\}}{P_1 \frac{\max\{\pi_0^{(1)}\}}{\min\{\pi_s^{(1)}\}} + P_2 \frac{\max\{\pi_0^{(2)}\}}{\min\{\pi_s^{(2)}\}}}$. Which proves that for any ω, for any length n, the observer is uncertain with a threshold of at least $\alpha_0 = \min\{c_1, c_2\}$ threshold. □

Theorem 14. *(Necessary and Sufficient conditions for Probabilistic System Opacity). Consider a set of m HMMs, $S^{(i)} = (Q^{(i)}, E^{(i)}, \Delta^{(i)}, \Lambda^{(i)}, \pi_0^{(i)})$, for $i \in \{1, ..., m\}$, with corresponding Markov chains $MC^{(i)} = (Q^{(i)}, A^{(i)}, \pi_0^{(i)})$ that are irreducible and aperiodic and with initial probability distribution $\pi_0^{(i)} > 0$. The property of probabilistic system opacity as described in Definition 52 holds iff for any chosen $S^{(i)}$ there exists at least a pairwise probabilistically opaque (Definition 54) HMM $S^{(j)}$, $j \neq i$.*

Proof. ($\rightarrow$.) We need to show that for any system $S^{(i)}$ and for any observation sequence ω that can be generated by $S^{(i)}$, we have

$$\alpha(\omega) = \frac{\sum\limits_{k\neq i} P_k P_\omega^{(k)}}{\sum\limits_{k=1}^{m} P_k P_\omega^{(k)}} \geq \alpha_0 \ .$$

Suppose $S^{(j)}$ is the HMM that is pairwise probabilistically opaque with $S^{(i)}$. Then, from Definition 54, there exists an α_0, such that $\min\{\frac{P_i P_\omega^{(i)}}{P_i P_\omega^{(i)} + P_j P_\omega^{(j)}}, \frac{P_j P_\omega^{(j)}}{P_i P_\omega^{(i)} + P_j P_\omega^{(j)}}\} \geq \alpha_0$. Thus, for any observation sequence ω that could be generated by $S^{(i)}$, we have

$$\begin{aligned}\alpha(\omega) =& \frac{\sum\limits_{k\neq i} P_k P_\omega^{(k)}}{\sum\limits_{k=1}^{m} P_k P_\omega^{(k)}} = 1 - \frac{P_i P_\omega^{(i)}}{\sum\limits_{k=1}^{m} P_k P_\omega^{(k)}} \\ \geq& 1 - \frac{P_i P_\omega^{(i)}}{P_i P_\omega^{(i)} + P_j P_\omega^{(j)}} = \frac{P_j P_\omega^{(j)}}{P_i P_\omega^{(i)} + P_j P_\omega^{(j)}} \geq \alpha_0 \ .\end{aligned}$$

Therefore, probabilistic system holds if, for any chosen $S^{(i)}$, there exists another system $S^{(j)}$, such that $S^{(i)}$ and $S^{(j)}$ are pairwise probabilistically opaque (Definition 54).

($\leftarrow$.) We want to prove that {If there is at least a chosen $S^{(i)}$ such that there is no HMM $S^{(j)}$ with $j \neq i$ such that $S^{(i)}$ and $S^{(j)}$ are a probabilistically opaque pair} $\Rightarrow$ {The probability of error when classifying among m HMMs with $S^{(i)}$ as the chosen system tends to zero} $\Rightarrow$ {Probabilistic system opacity does not hold}. It is easier to prove the contrapositive proposition which is: {If probabilistic system opacity holds} $\Rightarrow$ {The probability of error among m HMMs with $S^{(i)}$ as the chosen system, does not tend to zero} $\Rightarrow$ {$\forall S^{(i)}$, there is at least one other system $S^{(j)}$ such that $S^{(i)}$ and $S^{(j)}$ are a probabilistically opaque pair}.

If probabilistic system opacity holds, then $(\exists \alpha_0)$ such that $\{(\forall S^{(i)})(\forall \omega \in L(S^{(i)})\} \rightarrow \{\alpha_\omega \geq \alpha_0\}$. The probability of error for m HMMs, with $S^{(i)}$ chosen, satisfies

$$\begin{aligned}\Pr(\text{error at } n \ , \ S^{(i)}) \ &= \ \sum_{\forall \omega:|\omega|=n} \sum_{k\neq i} P_k P_\omega^{(k)} \\ &\geq \ \alpha_0 \sum_{\forall \omega:|\omega|=n} \sum_{k=1}^{m} P_k P_\omega^{(k)} = \alpha_0\end{aligned}$$

due to the property of probabilistic system opacity. This proves that the probability of error, with $S^{(i)}$ chosen (for any $S^{(i)}$), does not tend to zero. It remains to prove that $\forall S^{(i)}$ there exists at least one pair of probabilistically opaque HMMs. If all pairs

$S^{(i)}$ and $S^{(j)}$ are not probabilistically opaque, then according to the proof of pairwise probabilistic opacity, we have the following: if we chose ($S^{(i)}$ then $\forall S^{(j)}$)($\forall \alpha_0$)($\exists n \geq n_0$)

$$\Pr(\text{pairwise error at } n \ , \ S^{(i)}) = \sum_{\forall \omega : |\omega| = n} P_j P_\omega^{(j)} < \alpha_0 \ .$$

Now we prove that the probability of error among m HMMs, with $S^{(i)}$ chosen, tends to zero. We see that ($\forall \alpha_0$)($\exists n \geq n_0$)

$$\begin{aligned} \Pr(\text{error at } n \ , \ S^{(i)}) &= \sum_{\forall \omega : |\omega| = n} \sum_{k \neq i} P_k P_\omega^{(k)} \\ &\leq \alpha_0 \sum_{k \neq i} P_k = \alpha_0 (1 - P_i) \ , \end{aligned}$$

i.e., the probability of error tends to zero. This proves that $\forall S^{(i)}$ there exists at least one other system $S^{(j)}$ such that the pair $S^{(i)}$ and $S^{(j)}$ form probabilistically opaque HMMs; otherwise, the probability of error would tend to zero eventually. □

Chapter 6

Conclusions

In this thesis we dealt with state estimation problems using novel state estimation techniques. Specifically, we verified a variety of discrete event system properties, relating to fault diagnosis, detectability, opacity and classification. Below, we provide a summary of the main contributions of the thesis, classified into different categories.

- Contributions to fault diagnosis: We studied distributed fault diagnosis in DES using synchronization-driven intersection-based distributed diagnosis (RS-IBDD) strategies in the presence of communication constraints. RS-IBDD allows the exchange of diagnostic information (namely, state estimates and associated normal/fault conditions) at predetermined synchronization points between neighboring sites in a distributed observation setting. We have provided a verification method for RS-IBDD diagnosability that relies on a parallel product of the local verifiers along with the synchronization operation. This approach has complexity that is polynomial in the number of states of the given system and exponential in the number of observation sites. In the future, we plan to further study distributed observation settings like the one describe in this thesis, by allowing the exchange of additional diagnostic information between observation sites (e.g., the exchange of sequences of observations or summaries of such sequences of observations) and ways to verify them in an efficient manner.

- Open problem in fault diagnosis: The development of an adaptive distributed protocol which minimizes the number of communication exchanges between the local sites, while still guaranteeing that the system remains diagnosable. Problems of communication usually require that communication among agents

be minimized in some way. Studies in minimizing communication are motivated by reducing network bandwidth, for conserving power when only limited battery power is available or for security purpose [43]. The major open problem is to develop an algorithm, such as to minimize the communication between local sites, also establishing the property of fault diagnosis.

- Contributions to detectability: In this thesis we studied detectability in discrete event systems modeled by PFAs. We defined and analyzed two notions of stochastic detectability, namely A-detectability, and AA-detectability which were inspired by analogous notions in stochastic diagnosability [49]. We showed that A-detectability is a PSPACE-hard problem and applied observer-based techniques to verify it. We applied methods closely related to those used in classification of PFAs to verify AA-detectability with polynomial complexity.

- Open problems in detectability: Possible extensions could be to the cases of distributed stochastic detectability, and/or periodic stochastic detectability. A possible future research direction is the computation of bounds on the probability of error in state estimation problems in stochastic DES.

- Contributions to classification: In this thesis we obtained a bound on the probability of misclassification between two HMMs based on a sequence of observations. We developed three methods

  - First method: Calculation of Upper bound via a DFA

    * Contribution: In this method we used a specific class of DFAs to split the sequences of observations into different partitions and apply Markov chain theory to efficiently compute an upper bound on the *a priori* probability of misclassification among the two HMMs for sequences in each partition. The choice of DFA affects the partitioning which in turn affects the tightness of the upper bound.
    * Open problems: An open problem is the choice of a specific DFA (of a fixed number of states) that results in the least upper bound.

  - Second method: Construction of a Stochastic Verifier

    * Contribution: In this method we used a technique which captures the common behavior of the two HMMs and constructs an appropriate

product Markov chain.

* Open problems: Many researchers are intrigued by the problem of finding a suitable measure for the distance between HMMs. The relation between this upper bound and the second largest eigenvalue of the transition probability matrix of the produced Markov chain is a result that looks promising for further research, particularly as an approximation of the dissimilarity between two HMMs.

– Third method: Classification Rule Based on Empirical Frequencies of Event Sequences

* Contribution: We developed a decision rule (empirical rule) that relies on the frequencies with which output symbols are observed. We established necessary and sufficient conditions under which this rule provides us with an upper bound that tends to zero exponentially with the length of the observation sequence.

* Open problems: An open problem is to bridge the difference between the optimal MAP rule and the rule analyzed here. One way to accomplish this is to explicitly state the necessary and sufficient conditions under which the probability of misclassification tends to zero. Many applications that depend on classification could potentially benefit from this approach, including decision making and fault diagnosis in distributed systems. A possible extension of the work is the application of the empirical rule in the classification problem for more than two hidden Markov models.

- We developed algorithms for performing asymptotically optimal classification, and we measured their efficiency, by computing an upper bound on the probability of error, and establishing necessary and sufficient conditions under which this upper bound asymptotically tends to zero. Finally, we used results from the classification problem to solve a probabilistic opacity problem with polynomial complexity.

- Open problems: The characterization of necessary and sufficient conditions under which the probability of error tends asymptotically to zero and the computation of more efficient upper bounds.

Bibliography

[1] E. Athanasopoulou and C. N. Hadjicostis, "Probability of error bounds for failure diagnosis and classification in hidden Markov models," in *Proceedings of the IEEE Conference on Decision and Control*, 2008, pp. 1477–1482.

[2] L. R. Bahl, F. Jelinek, and R. L. Mercer, "Readings in speech recognition," A. Waibel and K.-F. Lee, Eds. Morgan Kaufmann Publishers Inc., 1990, ch. A maximum likelihood approach to continuous speech recognition, pp. 308—319.

[3] S. Bavishi and E. Chong, "Automated fault diagnosis using a discrete event systems framework," in *IEEE Symposium on Intelligent Control*, 1994, pp. 213–218.

[4] P. Bremaud, *Markov chains : Gibbs fields, Monte Carlo simulation and queues*, ser. Texts in Applied Mathematics. Springer, 1999.

[5] J. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Transactions on Circuits and Systems*, vol. 25, no. 9, pp. 772–781, 1978.

[6] J. W. Bryans, M. Koutny, L. Mazare, and P. Ryan, "Opacity generalised to transition systems," in *Proceedings of the 3rd Int. Workshop on Formal Aspects in Security and Trust*, July 2005, pp. 81–95.

[7] J. W. Bryans, M. Koutny, L. Mazar, and P. Ryan, "Opacity generalised to transition systems," Selected Papers of the 3rd International Workshop on Formal Aspects in Security and Trust, Tech. Rep., 2005.

[8] M. Cabasino, A. Giua, A. Paoli, and C. Seatzu, "A new protocol for the decentralized diagnosis of labeled Petri nets," in *10th Int. Workshop on Discrete Event Systems (WODES)*, 2010, pp. 123–128.

[9] C. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Kluwer Academic Publishers, 1999.

[10] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Kluwer, 1999.

[11] J. Chen and R. Kumar, "Polynomial test for stochastic diagnosability of discrete event systems," in *Proceedings of IEEE Conference on Automation Science and Engineering*, 2012, pp. 521–526.

[12] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms, Third Edition*, 3rd ed. The MIT Press, 2009.

[13] R. Debouk, S. Lafortune, and D. Teneketzis, "Coordinated decentralized protocols for failure diagnosis of discrete event systems," *Discrete Event Dynamic Systems*, vol. 10, no. 1, pp. 33–86, 2000.

[14] A. Dembo and O. Zeitouni, *Large Deviations Techniques and Applications*. Springer-Verlag Berlin Heidelberg, 2010.

[15] R. Durbin, S. R. Eddy, A. Krogh, and G. Mitchison, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1998.

[16] E. Fabre, A. Benveniste, C. Jard, L. Ricker, and M. Smith, "Distributed state reconstruction for discrete event systems," in *39th IEEE Conference on Decision and Control (CDC)*, vol. 3, 2000, pp. 2252–2257.

[17] M. Falkhausen, H. Reininger, and D. Wolf, "Calculation of distance measures between hidden Markov models," in *Proceedings of Eurospeech*, 1995, pp. 1487–1490.

[18] R. Focardi and R. Gorrieri, "A taxonomy of trace–based security properties for CCS," in *Proceedings of the 7th Workshop on Computer Security Foundations*, June 1994, pp. 126–136.

[19] K. S. Fu, *Syntactic Pattern Recognition and Applications*. Prentice-Hall, 1982.

[20] P. Glynn and D. Ormoneit, "Hoeffding's inequality for uniformly ergodic Markov chains," *Statistics and Probability Letters*, vol. 56, pp. 143–146, 2002.

[21] C. N. Hadjicostis, "Probabilistic detection of FSM single state-transition faults based on state occupancy measurements," *IEEE Transactions on Automatic Control*, vol. 50, no. 12, pp. 2078–2083, 2005.

[22] G. Hardy, H. Littlewood, and G. Polya, *Inequalities*. Cambridge University Press, 1988.

[23] F. Jelinek, *Statistical methods for speech recognition*. MIT Press, 1997.

[24] S. Jiang, Z. Huang, V. Chandra, and R. Kumar, "A polynomial algorithm for testing diagnosability of discrete-event systems," *IEEE Transactions on Automatic Control*, vol. 46, no. 8, pp. 1318–1321, 2001.

[25] B.-H. Juang and L. Rabiner, "A probabilistic distance measure for hidden Markov models," *AT&T Technical Journal*, pp. 391–408, 1985.

[26] J.-Y. Kao, N. Rampersad, and J. Shallit, "On NFAs where all states are final, initial, or both," *Theor. Comput. Sci.*, vol. 410, no. 47-49, pp. 5010–5021, 2009.

[27] T. Koski, *Hidden Markov Models of Bioinformatics*. Kluwer Academic Publishers, 2001.

[28] B. C. Levy, *Principles of Signal Detection and Parameter Estimation*. Springer, 2008.

[29] F. Lin, "Diagnosability of discrete event systems and its applications," *Discrete Event Dynamic Systems*, vol. 4, no. 2, pp. 197–212, 1994.

[30] F. Lin and W. Wonham, "On observability of discrete-event systems," *Information Sciences*, vol. 44, no. 3, pp. 173–198, 1988.

[31] J. Lunze and J. Schröder, "State observation and diagnosis of discrete event systems described by stochastic automata," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 11, no. 4, pp. 319–369, 2001.

[32] J. L. Massey, in *New Directions in Signal Processing in Communication and Control*.

[33] M. V. Moreira, T. C. Jesus, and J. C. Basilio, "Polynomial time verification of decentralized diagnosability of discrete event systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 7, pp. 1679–1684, 2011.

[34] C. M. Ozveren and A. S. Willsky, "Observability of discrete event dynamic systems," *IEEE Transactions on Automatic Control*, vol. 35, no. 7, pp. 797–806, 1990.

[35] M. Panteli and C. N. Hadjicostis, "Intersection based decentralized diagnosis: Implementation and verification," in *52nd IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, 2013, pp. 6311–6316.

[36] W. Qiu and R. Kumar, "Decentralized failure diagnosis of discrete event systems," *IEEE Transactions on Systems Man and Cybernetics Part A: Systems and Humans*, vol. 36, no. 2, pp. 384–395, 2006.

[37] L. R. Rabiner, "Readings in speech recognition," A. Waibel and K.-F. Lee, Eds. Morgan Kaufmann Publishers Inc., 1990, ch. A tutorial on hidden Markov models and selected applications in speech recognition, pp. 267–296.

[38] P. A. Regalia and S. K. Mitra, "Kronecker products, unitary matrices and signal processing applications," *Society for Industrial and Applied Mathematics*, vol. 31, no. 4, pp. 586–613, December 1989.

[39] J. S. Rosenthal, "Convergence rates of markov chains," *SIAM Review*, vol. 37, pp. 387–405, 1995.

[40] A. Saboori and C. N. Hadjicostis, "Verification of infinite-step opacity and complexity considerations," *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1265–1269, 2012.

[41] ——, "Notions of security and opacity in discrete event systems," in *Proceedings of 46th IEEE Conference on Decision and Control*, 2007, pp. 5056–5061.

[42] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Diagnosability of discrete event systems," *IEEE Transactions on Automatic Control*, vol. 40, no. 9, pp. 1555–1575, 1995.

[43] D. Sears and K. Rudie, "Minimal sensor activation and minimal communication in discrete-event systems," *Discrete Event Dynamic Systems*, vol. 26, pp. 295–349, 2016.

[44] E. Seneta, *Non-negative Matrices and Markov Chains*. Springer Series in Statistics, 2006.

[45] S. Shu, F. Lin, and H. Ying, "Detectability of discrete event systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 12, pp. 2356–2359, 2007.

[46] S. Shu, F. Lin, H. Ying, and X. Chen, "State estimation and detectability of probabilistic discrete event systems," *Automatica*, vol. 44, no. 12, pp. 3054–3060, dec 2008.

[47] R. Su and W. M. Wonham, "Global and local consistencies in distributed fault diagnosis for discrete-event systems," *IEEE Transactions on Automatic Control*, vol. 50, no. 12, pp. 1923–1935, 2005.

[48] S. Takai and T. Ushio, "Verification of codiagnosability for discrete event systems modeled by Mealy automata with nondeterministic output functions," *IEEE Transactions on Automatic Control*, vol. 57, no. 3, pp. 798–804, 2012.

[49] D. Thorsley and D. Teneketzis, "Diagnosability of stochastic discrete event systems," *IEEE Transactions on Automatic Control*, vol. 50, no. 4, pp. 476–492, 2005.

[50] W.-G. Tzeng, "The equivalence and learning of probabilistic automata," in *30th Annual Symposium on Foundations of Computer Science*, 1989, pp. 268–273.

[51] W. Wang, A. R. Girard, S. Lafortune, and F. Lin, "On codiagnosability and coobservability with dynamic observations," *IEEE Transactions on Automatic Control*, vol. 56, no. 7, pp. 1551–1566, 2011.

[52] Y. Wang, T.-S. Yoo, and S. Lafortune, "Diagnosis of discrete event systems using decentralized architectures," *Discrete Event Dynamic Systems*, vol. 17, no. 2, pp. 233–263, 2007.

[53] Y.-C. Wu and S. Lafortune, "Synthesis of insertion functions for enforcement of opacity security properties," *Automatica*, vol. 50, no. 5, pp. 1336–1348, 2014.

[54] T.-S. Yoo and S. Lafortune, "Polynomial-time verification of diagnosability of partially observed discrete-event systems," *IEEE Transactions on Automatic Control*, vol. 47, no. 9, pp. 1491–1495, 2002.