

On Test Derivation from Partial Specifications

Alex Petrenko and Nina Yevtushenko [‡]

*CRIM, Centre de Recherche Informatique de Montréal
550 Sherbrooke West, Suite 100
Montreal, H3A 1B9, Canada
phone: +1 (514) 840-1290, fax: +1 (514) 840-1244
petrenko@crim.ca*

[‡] *Tomsk State University
36 Lenin str., Tomsk, 634050, Russia
yevtushenko.RFF@elefot.tsu.ru*

Abstract

The paper addresses the problem of test derivation from partially defined specifications. A specification is modeled by an Input/Output FSM such that transitions from some states on some inputs are not specified (a partial FSM). Tests have to be derived for a weak conformance, i.e., using the quasi-equivalence relation between FSMs as a conformance relation. The paper further elaborates the state-counting approach by providing an insight into the structure of tests with fault coverage for partial (deterministic) machines with compatible, quasi-equivalent, and distinguishable states and by offering risk-free optimizations which reduce the length of resulting tests. Based on this approach, a method for deriving tests with fault coverage guarantee (checking experiments) is proposed. It is demonstrated that the method is superior to other test derivation methods for deterministic state machines.

Keywords

I/O FSMs, partially defined machines, test derivation, weak conformance testing, fault detection, checking experiments.

1 INTRODUCTION

There are not many papers on test derivation that deal with partial specifications, in general, and with partial Input/Output FSMs, in particular. One of the reasons is that a classical state identification approach pioneered in [Hen64] does not extend to an arbitrary partially defined FSM, as its states may no longer be distinguishable using only "core" or specified transitions. The core transitions have exclusively to be used to check for a *weak* conformance, opposed to the *strong* conformance (equivalence) used for complete machines. A pragmatic solution that sometimes works is to complete a partial specification in one way or another. A complete specification can then be minimized so that the state identification approach becomes applicable; however, some conforming implementations may now be recognized as nonconforming as already argued, e.g., in [PBD93], [YaL95]. Note that in [YaL95] it is assumed that partial machine is reduced, which seems to be only a special case. In a more general case, partial machine may not be reduced and another approach is required. As a solution to the problem, we attempted to develop such an approach, called a state-counting approach, in a number of papers [YePe89], [YePe90], [Petr91], [PYL93], [PYB96]. This approach relies on counting states traversed by test sequences to decide when a particular sequence could be trimmed. It subsumes the classical state identification approach of Hennie [Hen64] and his followers, see, e.g., [Vas73], [Cho78], [FBK91], [LBP94], [ReU95].

The goal of this paper is to further advance the state-counting approach by providing an insight into the structure of tests with fault coverage for partial (deterministic) machines with compatible, quasi-equivalent, and distinguishable states and by offering risk-free optimizations which reduce the length of resulting tests.

The rest of the paper is organized as follows. In Section 2, we give necessary basic notions. In Section 3, we discuss the structure of tests for various types of FSMs, which may have compatible, quasi-equivalent and distinguishable states and propose a method for test derivation from partial FSMs with complete fault coverage. We also discuss the complexity of test suites obtained with the proposed method. In Section 4, we summarize the contributions of the paper and indicate possible future work.

2 BASIC DEFINITIONS AND NOTATIONS

A finite state machine (FSM), often simply called a machine throughout this paper, is an initialized deterministic Mealy machine which can be formally defined as follows. An FSM A is a 7-tuple $(S, s_0, X, Y, D_A, \delta, \lambda)$, where S is a finite set of states with the initial state s_0 , X is a finite set of inputs, Y is a finite set of outputs, D_A is a subset of $S \times X$, called a *specification domain*, and δ and λ are a transition function $\delta: D_A \rightarrow S$ and an output function $\lambda: D_A \rightarrow Y$, respectively.

An FSM A is said to be *completely* specified (a complete FSM) if $D_A = S \times X$. We will omit the specification domain D_A in the case of complete machines. If D_A is a proper subset of $S \times X$ then A is called a *partially* specified machine (a partial FSM). Note that, sometimes [GrL65], the functions δ and λ are defined over different specification domains, however, to simplify our discussions we will not consider here such cases. Moreover, we note that, for example, in typical protocol applications, both functions of protocol machines are defined over the same domain.

Given a string $\alpha = x_1 \dots x_k$ of the set X^* of all possible input sequences, α is said to be an *acceptable* input sequence at state $s \in S$ if there exist states s_1, \dots, s_k, s_{k+1} , where $s_1 = s$, such that $(s_i, x_i) \in D_A$ and $\delta(s_i, x_i) = s_{i+1}$ for any $i = 1, \dots, k$. In words, α is an acceptable input sequence at state s if the behavior of A for the sequence α is defined. We use $X_A^*(s)$ to denote the set of all acceptable input sequences for state s and X_A^* for the state s_0 , i.e. for A . The set $X_A^*(s)$ may be empty for state s , but hereinafter we assume that the set X_A^* is not empty.

We extend the transition and output functions from input symbols to acceptable input sequences, including the empty word ε , as usual. Let $\delta(s, \varepsilon) = s$ and $\lambda(s, \varepsilon) = \varepsilon$ for any $s \in S$. Suppose that β is an acceptable input sequence for state s and $\delta(s, \beta) = s'$ and $\lambda(s, \beta) = \gamma$. Then for any $x \in X$ such that $(s', x) \in D_A$ we define $\delta(s, \beta x) = \delta(s', x)$ and $\lambda(s, \beta x) = \gamma \lambda(s', x)$. For convenience, we use the same notations δ and λ for the extended functions, as well, since in our discussions, this does not imply any contradiction.

Given states $s, s' \in S$, a sequence $\alpha \in X_A^*(s)$ such that $\delta(s, \alpha) = s'$ is a *transfer* sequence from s to s' . For any state s , the empty sequence ε is a transfer sequence from s to s . A transfer sequence, if we do not specify from which state it is, starts, by default, from the initial one of a given FSM.

We say that the sequence α , applied at state s , *traverses* state s' if there exists a non-empty prefix β of α such that β is a transfer sequence from s to s' . We also say that a set of sequences *traverses* a given state if it contains at least one sequence that traverses state. We consider here only connected machines. FSM $A = (S, s_0, X, Y, D_A, \delta, \lambda)$ is said to be *connected* if all its states are reachable from the initial state, i.e., for any state $s \in S$, there exists a transfer sequence $\alpha \in X_A^*$ from s_0 to s .

A *state cover* of FSM A with n states is defined as a set of n transfer sequences that take the machine A from the initial state to every its state. An input sequence γ is said to *cover* a transition from state s with input x if it can be represented as a concatenation $\gamma = \alpha\beta$ such that α is a transfer sequence to s . We also say that a given set of sequences covers a given transition if it has at least one sequence that covers the transition.

Given state s of an FSM $A = (S, s_0, X, Y, D_A, \delta, \lambda)$ and state t of an FSM $B = (T, t_0, X, Y, D_B, \Delta, \Lambda)$, states s and t are said to be *compatible* if either $X_A^*(s) \cap X_B^*(t) = \emptyset$ or $\lambda(s, \alpha) = \varphi(t, \alpha)$ for any input sequence $\alpha \in X_A^*(s) \cap X_B^*(t)$. Otherwise, i.e., when there exists an input sequence $\alpha \in X_A^*(s) \cap X_B^*(t)$ such that $\lambda(s, \alpha) \neq \Lambda(t, \alpha)$, states s and t are said to be *distinguishable* (by α), written $s \not\equiv t$ ($s \not\equiv_\alpha t$). Given a set $W \subseteq X_A^*(s) \cap X_B^*(t)$, we also use $s \not\equiv_W t$ to denote the fact that there exists $\alpha \in W$ such that $s \not\equiv_\alpha t$.

In the case of a complete FSM, two distinguishable states can already be distinguished by input sequence α of length $|\alpha| \leq n - 1$, where n is the number of states in the given machine. In the case of a partial machine, the corresponding bound is $n(n - 1)/2$ [PBD93]. An FSM is *reduced* if any two of its states are distinguishable; a machine with compatible states is said to be *unreduced*.

An unreduced FSM may have quasi-equivalent or equivalent states. Given two compatible states, s of an FSM $A = (S, s_0, X, Y, D_A, \delta, \lambda)$ and t of an FSM $B = (T, t_0, X, Y, D_B, \Delta, \Lambda)$, state t of B is said to be *quasi-equivalent* to state s of A , written $t \sqsupseteq s$, if $X_B^*(t) \supseteq X_A^*(s)$. The quasi-equivalence relation, introduced in [Gil62], is also called weak conformance in [YaL95] and

other works. If states s and t are quasi-equivalent to each other, we refer to these states as to *equivalent* states, written $s \cong t$.

Given FSMs A and B over the same input alphabet, B is said to be *quasi-equivalent* to A , written $B \sqsupseteq A$, if the initial state of FSM B is quasi-equivalent to the initial state of FSM A . The two machines are *distinguishable*, written $B \not\cong A$, if their initial states are distinguishable.

If a reduced FSM B is quasi-equivalent to A then it is called a *reduced form* of A . Methods for constructing a reduced form from a partial machine are more involved than that for complete machines. The reason is that, in general, the relation of state compatibility is not transitive for partial FSMs, i.e. an appropriate state of an implementation FSM may be quasi-equivalent to different compatible states of a specification FSM. This feature explains why test derivation from partial machines is a more difficult problem than that for complete machines. We use the quasi-equivalence relation (weak conformance) for deriving tests directly from a given FSM without first minimizing it.

Let X be the input alphabet of a (partial or complete) specification FSM A . A *test suite* for FSM A is a finite set of acceptable input sequences (tests) of A . We thus assume that a reliable reset is available in any potential implementation machine to bring it back to the initial state to resume testing. We assume also that all implementations are represented by a set $\mathcal{F}(X)$ of complete FSMs defined over the input alphabet X , the set $\mathcal{F}(X)$ is often called a *fault domain*. A test suite TS is said to be *complete* for A w.r.t. the fault domain $\mathcal{F}(X)$ if, for all $B \in \mathcal{F}(X)$, $B \not\cong A$ implies $B \not\cong_{TS} A$. Let $\mathcal{F}_m(X)$ be a universal set of all complete FSMs with m states over the input alphabet X . A test suite is said to be *m -complete* if it is complete w.r.t. the fault domain $\mathcal{F}_m(X)$. Clearly, m -complete test suite is also k -complete for any $k < m$.

3 TEST DERIVATION

3.1 Complete Test Suite and Distinguishing Machine

Let A be a given specification FSM, partially or completely defined, and B be a completely defined implementation FSM, $B \in \mathcal{F}_m(X)$. To verify whether or not the FSM B is quasi-equivalent to A , we construct a designated FSM, called a distinguishing machine of A and B , whose states are pairs of states of A and B . Its initial state is the pair of initial states of the two machines, the remaining states are determined by performing a reachability analysis.

Formally, we define the distinguishing machine of FSM $A = (S, s_0, X, Y, D_A, \delta, \lambda)$ and FSM $B = (T, t_0, X, Y', \Delta, \Lambda)$ as an FSM with the state set $Q \subseteq S \times T$, the input set X , the output set $Y \cup \{fail\}$, $fail \notin (Y \cup Y')$, the transition function ψ , the output function ϕ , and the initial state (s_0, t_0) , using the following rules:

$$q_0 = (s_0, t_0),$$

$$\psi(st, x) = (\delta(s, x), \Delta(s, x)), \text{ if } (s, x) \in D_A, \text{ otherwise, } \psi(st, x) \text{ is undefined;}$$

$$\phi(st, x) = \lambda(s, x) \text{ if } \lambda(s, x) = \Lambda(t, x), \text{ and}$$

$$\phi(st, x) = fail \text{ if } \lambda(s, x) \neq \Lambda(t, x);$$

such that the state set Q is the smallest set obtained by application of the above rules. The machine $(Q, q_0, X, Y \cup \{fail\}, D_{A \oplus B}, \psi, \phi)$, where $D_{A \oplus B} = \{st, x \mid st \in Q \wedge (s, x) \in D_A\}$, is called the *distinguishing machine* (DM) of A and B , and is denoted by $A \oplus B$.

Clearly, the set Q includes only states reachable from the initial state. Note that similar notions of FSM products are used in other work, see, e.g., [ACY95], [FJJ96], [PYB96], [KPY99].

If FSM B is distinguishable from A , then any input sequence such that, applied at the initial state of DM $A \oplus B$, covers a fail-transition (one with the output *fail*), distinguishes B from A and can be used as a test case to expose faults modeled by B . The DM enjoys several other properties useful for test generation, some of them are collected in the following proposition.

Proposition 1. Given two FSMs, $A = (S, s_0, X, Y, D_A, \delta, \lambda)$ with n states and $B = (T, t_0, X, Y', \Delta, \lambda)$ with m states, let $A \oplus B = (Q, q_0, X, Y \cup \{fail\}, D_{A \oplus B}, \psi, \varphi)$ be DM of A and B . Then

(a) $X_{A \oplus B}^*(s, t) = X_A^*(s)$, for all $s \in S$ and $t \in T$ such that $(s, t) \in Q$.

(b) $|Q| \leq nm$. □

Proposition 1(a) says that any input sequence acceptable in DM is also acceptable in the specification machine A and can be used as a test. The following proposition, which is an immediate corollary to Proposition 1(b), establishes an upper bound on tests for partial FSMs.

Corollary 1. For any FSM A with n states and the set X_A^* of acceptable input sequences, there exists a subset of the set $\{\alpha \in X_A^* \mid |\alpha| = nm\} = X_A^{nm}$ that is an m -complete test suite. □

Example

Consider the specification FSM A with n states and implementation FSM B with m states shown in Figure 1 [YePe89]. DM has the only fail-transition out of state (s_n, t_m) . The shortest input sequence that covers this transition from the initial state is $(1^{n-1}2)^m$, its length is nm . Therefore, the bound nm is indeed tight for partial machines.

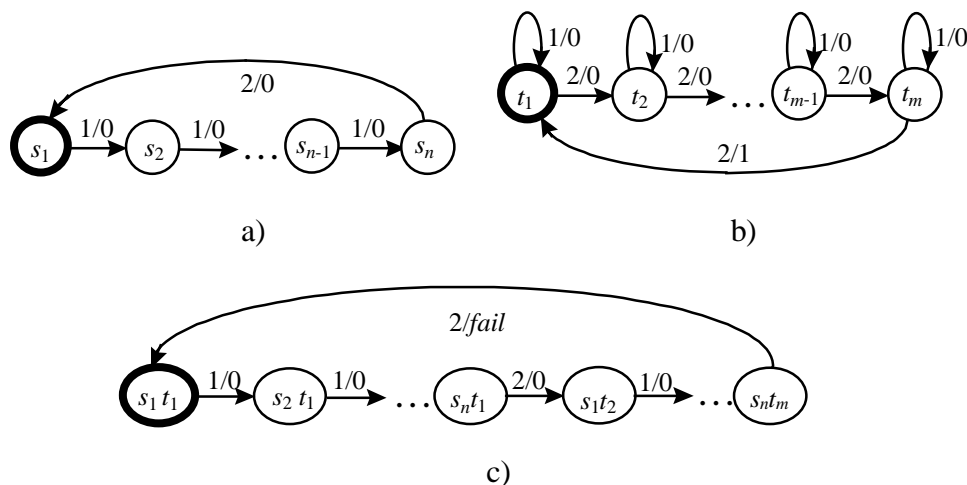


Figure 1. An example of machines distinguished by a sequence of length nm :
a) FSM A , b) FSM B , c) DM $A \oplus B$.

If the set of acceptable sequences is finite (when, for example, the machine has no cycling behavior) then it is a complete test suite regardless of the bound on the number of states in implementation machines. A systematic method is required to determine complete test suites

for FSMs with cyclic behavior. We first develop a method for several important classes of unreduced FSMs and then extend it to an arbitrary deterministic FSM, defined partially or completely.

3.2 Deriving Tests for Special Classes of Unreduced FSMs

We will describe a method for test derivation that accepts any deterministic I/O FSM, completely or partially specified, which may have compatible, equivalent, quasi-equivalent and distinguishable states. By combination, one can define many FSM classes, which differ in relations between the states of any machine from the class. Here we consider the following classes:

- FSMs such that each two states are compatible, but neither quasi-equivalent nor distinguishable.
- FSMs such that each two states are either compatible or quasi-equivalent, but not distinguishable.
- A general type of FSMs such that each two states are compatible, quasi-equivalent, or distinguishable. This class includes reduced machines and completely defined machines.

3.2.1 FSMs Without Distinguishable or Quasi-Equivalent States

Given an implementation FSM $B \in \mathcal{J}_m(X)$ such that $B \not\cong A$, a test suite to be m -complete should have a sequence that covers a fail-transition in DM $A \oplus B$. It is known that the length of a shortest input sequence that covers a transition of any FSM does not exceed the number of its states. On the other hand, states of distinguishing machines consist of states of FSMs A and B , therefore any state of A constitutes at most m states of DM. It means that a transfer sequence traversing the initial state of A m times is cyclic in DM and can be replaced by a shorter (acyclic) one. A transfer sequence is said to be *acyclic* if any two prefixes of it take FSM into different states. These considerations lead us to the following statements.

Lemma 1. Given a state cover V of FSM A and state q of DM $A \oplus B$, there exist a sequence $\alpha \in V$ and a transfer sequence $\alpha\beta \in X_{A \oplus B}^*$ to state q such that β , applied at state $\delta_{s_0}(\alpha)$, traverses any state of FSM A at most $m - 1$ times.

Proof. Let $\alpha\beta \in X_{A \oplus B}^*$ be a transfer sequence to state q in DM $A \oplus B$ such that β , applied at state $\psi(q_0, \alpha)$ is an acyclic transfer sequence. Then each state of FSM A is traversed by β at most m times. Suppose there exists state s of A traversed by β , applied at state $\delta_{s_0}(\alpha)$, exactly m times. Let q_1, \dots, q_m be the sequence of states $A \oplus B$ with the component state s traversed by β , applied at state $\psi(s_0, \alpha)$. For β to be acyclic in $A \oplus B$ it is required that all component states of FSM B in q_1, \dots, q_m be distinct states, denoted t_1, \dots, t_m , where $t_m = t$. Consider a transfer sequence $\omega \in V$ to state s in FSM A . Then the corresponding state $\Delta(t_0, \omega)$ of FSM B must be one of the states t_1, \dots, t_m , since FSM B has at most m states, i.e., there exists a transfer sequence $\omega\beta' \in X_{A \oplus B}^*$ to state q in DM $A \oplus B$, such that $\omega \in V$ and β' , applied at state $\delta_{s_0}(\omega)$, traverses state s of FSM A at most $m-1$ times. □

The lemma allows us to determine the maximal length of input sequences covering each transition in arbitrary DM $A \oplus B$ to obtain a complete test suite.

Proposition 2. Let V be a state cover of A and $E = \bigcup_{\alpha_i \in V} \alpha_i L_i \subseteq X_A^*$ be the set of input sequences such that $\beta \in L_i$ iff $\beta \in X_A^*(\delta_{s_0}, \alpha_i)$ and state $\delta_{s_0}, \alpha_i \beta$ is traversed by β , applied at state δ_{s_0}, α_i , m times while any other state of A is traversed by β at most $m - 1$ times. The set E is an m -complete test suite of FSM A .

Proof. Let $B \in \mathcal{I}_m(X)$ and $B \not\cong A$. If $B \not\cong A$ then in DM $A \oplus B$, there exists state q with an emanating fail-transition. Let x be the input that labels the fail-transition. By Lemma 1, there exists a transfer sequence $\alpha_i \beta \in X_{A \oplus B}^*$ to state q such that $\alpha_i \in V$, $\delta_{s_0}, \alpha_i = s_i$, and β traverses any state of FSM A at most $m - 1$ times. Then the sequence β is a proper prefix of some sequence in the set L_i . Hence, the sequence βx is a prefix of some sequence in the set L_i . Thus, $B \not\cong_E A$. □

The use of a prefix-closed state cover facilitates merging sequences in sets $\alpha_i L_i$ and may thus reduce tests. However, there might be even no need to consider each transfer sequence from a state cover, as follows from the proof of Lemma 1.

Corollary 2. Given an m -complete test suite E for A from Proposition 2, let $s_j \in S$ be state of A , different from the initial state, such that $s_j = \delta_{s_0}, \alpha_j$ and for all $\alpha_i \in V$, s_j is traversed less than m times by any input sequence $\beta \in L_i$, then the set $E \setminus \{\alpha_j L_j\}$ is an m -complete test suite for A . □

In a general case, the length of sequences in the resulting test suite cannot be further reduced if states of the given machine are neither distinguishable nor quasi-equivalent. FSM A in Figure 1 serves as an example of such machines. For other classes of machines, which have quasi-equivalent or distinguishable states, shorter (and fewer) sequences than in X_A^{nm} may constitute a complete test suite. In the next section, we consider how quasi-equivalent states can be taken into account to reduce a complete test suite.

3.2.2 FSMs without Distinguishable States

Assume now that a given machine has quasi-equivalent states, but no two states are distinguishable. The existence of quasi-equivalent states may shorten transfer sequences in distinguishing machines, compared to Lemma 1. In fact, let (s', t) and (s, t) be states of DM $A \oplus B$ such that $s' \sqsupseteq s$, i.e., $(s', t) \sqsupseteq (s, t)$. Any transfer sequence to state q that first traverses (s', t) and then (s, t) can be replaced by a shorter sequence that takes the machine to a state that is quasi-equivalent to state q , but avoids state (s, t) . Indeed, successors of quasi-equivalent states for any acceptable input sequence are also quasi-equivalent states. This observation motivates the following definitions.

A transfer sequence $\nu \in X_{A \oplus B}^*$ is said to be *acyclic* in DM $A \oplus B$ if, for any two non-empty prefixes α and β of ν , $|\alpha| < |\beta|$, it holds that if $\psi(q_0, \alpha) = (s, t)$, $\psi(q_0, \beta) = (s', t')$, and $s' \sqsupseteq s$, then $t \neq t'$.

We introduce a designated function, denoted $\#$, which for given states $s, p \in S$ and input sequence $\alpha \in X_A^*(s)$, returns a maximal length of a sequence of states, traversed by α , applied at state s , each of which is quasi-equivalent to all subsequent states in the sequence, including state p . Formally, $\#_s(p, \alpha) = l$, where l is the maximal number of prefixes $\alpha_1, \dots, \alpha_l$ of α such that $0 < |\alpha_1| < \dots < |\alpha_l|$, and $\delta(s, \alpha_1) \sqsupseteq \dots \sqsupseteq \delta(s, \alpha_l) \sqsupseteq p$. $|\alpha| \geq \#_s(p, \alpha) \geq 0$ and $\#_s(p, \alpha) = 0$ if α is an empty sequence or none of the states traversed by a non-empty sequence α is quasi-equivalent to p .

We also introduce the concept of a core of an FSM. Given FSM A , we determine a minimal by inclusion set of states of A such that contains the initial state and, for each state $s \in S$, a state quasi-equivalent to s , called a *core* of FSM A . In the case, when the machine A has no quasi-equivalent states, the core coincides with the state set S . A subset $K \subseteq X_A^*$ is said to be a *core cover* of A if, for each state in the core of A , it has exactly one transfer sequence.

Example

Consider the partial FSM A shown in Figure 2. States 1 and 3 are compatible states. These states have no common acceptable input sequence. We also have $2 \sqsupseteq 3$. The core of FSM A includes states 1 and 2. The machine has the core cover $K = \{\varepsilon, 1\}$, where ε is an empty sequence.

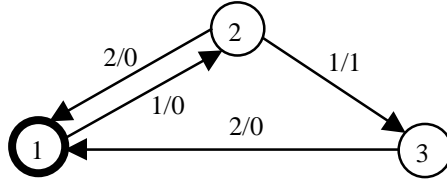


Figure 2. The partial FSM A .

Using the above notions, Lemma 1 can be generalized to the case of quasi-equivalent states as follows.

Lemma 2. Given a core cover K of FSM A and state q of DM $A \oplus B$, there exist a sequence $\alpha \in K$ and a transfer sequence $\alpha\beta \in X_{A \oplus B}^*$ to state $q' \sqsupseteq q$ in DM $A \oplus B$, such that $\#_{\delta(s_0, \alpha)}(s, \beta) \leq m - 1$ for all $s \in S$.

Proof. Let $\nu\beta \in X_{A \oplus B}^*$ be a transfer sequence to state q in DM $A \oplus B$ such that β , applied at state $\psi(q_0, \nu)$ is an acyclic transfer sequence in DM $A \oplus B$, i.e., $\#_{\delta(s_0, \nu)}(s, \beta) \leq m$ for all $s \in S$. Assume that there exists state $s \in S$ such that $\#_{\delta(s_0, \nu)}(s, \beta) = m$, while $\#_{\delta(s_0, \nu)}(s', \beta) < m$ for all $s' \in S$, $s' \neq s$. We have m prefixes β_1, \dots, β_m of β such that $0 < |\beta_1| < \dots < |\beta_m|$ and $\delta(s_0, \nu\beta_1) \sqsupseteq \dots$

$\sqsupseteq \delta_{s_0, v\beta_m} \sqsupseteq s$. The sequence β is acyclic, i.e., the corresponding m states of FSM B are distinct. Thus, we have m states of DM $A \oplus B$ $(\delta_{s_0, v\beta_1}, \Delta(t_0, v\beta_1)), \dots, (\delta_{s_0, v\beta_m}, \Delta(t_0, v\beta_m))$, such that $\delta_{s_0, v\beta_1} \sqsupseteq \dots \sqsupseteq (\delta_{s_0, v\beta_m} \sqsupseteq s$ and $\Delta(t_0, v\beta_i) = \Delta(t_0, v\beta_j)$ implies $i = j$. By definition of a core cover, there exists $\alpha_i \in K$ such that $\delta_{s_0, \alpha_i} = s_i$ and $s_i \sqsupseteq \delta_{s_0, v\beta_1} \sqsupseteq \dots \sqsupseteq (\delta_{s_0, v\beta_m} \sqsupseteq s$. The sequence α_i takes DM $A \oplus B$ into state $\psi(q_0, \alpha_i) = (s_i, \Delta(t_0, \alpha_i))$ such that $\Delta(t_0, \alpha_i) = \Delta(t_0, v\beta_j)$ and $\delta_{s_0, \alpha_i} \sqsupseteq \delta_{s_0, v\beta_j}$ for some non-empty prefix β_j of β , as FSM B has at most m states. Then $\psi(q_0, \alpha_i) \sqsupseteq \psi(q_0, v\beta_j)$ and the transfer sequence $v\beta$ to state q can be represented as a concatenation $v\beta_j\beta'$, such that $\#_{\delta_{s_0, v\beta_j}}(s, \beta') \leq m - 1$ for all $s \in S$. $\psi(q_0, \alpha_i) \sqsupseteq \psi(q_0, v\beta_j)$ implies that $\psi(q_0, \alpha_i\beta') \sqsupseteq \psi(q_0, v\beta_j\beta') = q$. Thus, there exists a transfer sequence $\alpha_i\beta'$ to a state quasi-equivalent to the given state q , such that $\#_{\delta_{s_0, \alpha_i}}(s, \beta') \leq m - 1$ for all $s \in S$. \square

The lemma leads us to the following statement that includes Proposition 2 as a special case.

Proposition 3. Let K be a core cover of A and $F = \bigcup_{\alpha_i \in K} \alpha_i M_i \subseteq X_A^*$ be the set of input sequences such that $\beta \in M_i$ iff $\beta \in X_A^*(\delta_{s_0, \alpha_i})$, $\#_{\delta_{s_0, \alpha_i}}(\delta_{s_0, \alpha_i}\beta, \beta) = m$ and $\#_{\delta_{s_0, \alpha_i}}(s, \beta) < m$ for all $s \neq \delta_{s_0, \alpha_i}\beta$. The set F is an m -complete test suite of FSM A .

Proof. Let $B \in \mathcal{I}_m(X)$ and $B \not\cong A$. If $B \not\cong A$ then in DM $A \oplus B$, there exists state q with an emanating fail-transition. Let x be the input that labels the fail-transition. By Lemma 2, there exists a transfer sequence $\alpha_i\beta \in X_{A \oplus B}^*$ to state $q' \sqsupseteq q$, such that $\alpha_i \in K$ and $\#_{\delta_{s_0, \alpha_i}}(s, \beta) \leq m - 1$ for all $s \in S$. Then the sequence β is a proper prefix of some sequence in the set M_i . $q' \sqsupseteq q$ implies that $X_{A \oplus B}^*(q') \supseteq X_{A \oplus B}^*(q)$ and x induces a fail-transition from the state q' . Hence, the input x is also acceptable at state q' and sequence βx is a prefix of some sequence in the set M_i that covers a fail-transition. Thus, $B \not\cong_F A$. \square

Similar to Corollary 2, we have the following.

Corollary 3. Given an m -complete test suite F from Proposition 3, let $\alpha_j \in K$ be a transfer sequence from a core cover of A such that $s_j = \delta_{s_0, \alpha_j}$, $s_j \neq s_0$ and for any $\alpha_i \in V$, $i \neq j$, any $p \in S$, $s_j \sqsupseteq p$, it holds that $\#_{\delta_{s_0, \alpha_j}}(p, \beta) < m$ for all $\beta \in M_i$, then the set $F \setminus \{\alpha_j M_j\}$ is an m -complete test suite.

3.3 Deriving Tests for General Type of FSMs

Now we allow for the specification FSM A to have any type of states, compatible, quasi-equivalent or distinguishable.

Given FSM A , we determine distinguishable states of FSM A , using standard algorithms for state minimization of partial FSMs [Gil62], [Koh70], [GrL65], [LeY96]. We will use maximal by inclusion sets of mutually distinguishable states. These sets can be determined as

cliques of the so-called state distinguishability graph, where two nodes representing states have an edge if these states are distinguishable.

Given a transfer sequence $\alpha\beta \in X_A^*$, let $D_{\beta\text{after}\alpha}$ denote a maximal by inclusion set of mutually distinguishable states of A , such that at least one of them is traversed by β , applied at state $\delta_{(s_0, \alpha)}$. Let n_{\max} denote the maximal number of mutually distinguishable states of A . Hereinafter, we assume that m , the number of states in an implementation machine, is not less than n_{\max} . Distinguishable states of FSM A may create distinguishable states in DM $A \oplus B$. In particular, states of DM $A \oplus B$ (s, t) and (s', t) are said to be *conflicting* if $s \not\cong s'$ [PYB96].

To generalize Proposition 3 for the case of conflicting states in DM $A \oplus B$, we need the following notation. Given state $p \in S$, an input sequence $\beta \in X_A^*(p)$ and a set of mutually distinguishable states $D \subseteq S$, let $\#_p(D, \beta)$ denote $\sum_{s \in D} \#_p(s, \beta)$. Let $\#_p(D, \beta) = l$. If $l > 0$ then by definition of the function $\#$, β has exactly l prefixes β_1, \dots, β_l , $0 < |\beta_1| < \dots < |\beta_l|$ with the following property. For each $s \in D$, such that $\#_p(D \setminus \{s\}, \beta) < l$ and for each i , $i \leq l$, if $\delta(s, \beta_i) \cong s$ and $\delta(s, \beta_j) \cong s$ for some $j < i$, then $\delta(s, \beta_j) \cong \delta(s, \beta_i)$. We use $P(\beta, D)$ to denote the set $\{\beta_1, \dots, \beta_l\}$.

Proposition 4. Let K be a core cover of FSM A and $G = \bigcup_{\alpha_i \in K} \alpha_i N_i \subseteq X_A^*$ be a set of input sequences, where N_i comprises each shortest sequence $\beta \in X_A^*(\delta_{(s_0, \alpha_i)})$ such that $\#_{\delta_{(s_0, \alpha_i)}}(D, \beta) = m - |D| + 1$ holds for an appropriate set D of mutually distinguishable states of A . The following statements hold.

1. If FSM A has no distinguishable states then the set G covers a fail-transition in each DM $A \oplus B$, such that $B \in \mathcal{I}_m(X)$ and $B \not\cong A$.
2. Let $B \in \mathcal{I}_m(X)$, $B \not\cong A$ and $B \cong_G A$. Then there exist $\alpha_i \in K$, $\beta \in N_i$ and $D_{\beta\text{after}\alpha_i}$ such that $|D_{\beta\text{after}\alpha_i}| > 1$ and $\#_{\delta_{(s_0, \alpha_i)}}(D_{\beta\text{after}\alpha_i}, \beta) = m - |D_{\beta\text{after}\alpha_i}| + 1$, moreover in DM $A \oplus B$ the set of states $\{\psi(q_0, \alpha) \mid (\alpha \in K \wedge \exists s \in D_{\beta\text{after}\alpha_i}: \delta_{(s_0, \alpha)} \cong s) \vee \exists \gamma \in P(\beta, D_{\beta\text{after}\alpha_i}) : \alpha = \alpha_i \gamma\}$ contains two conflicting states.

Proof. 1. If FSM A has no distinguishable states then the set G coincides with the set F in Proposition 3, i.e., G is an m -complete test suite.

2. We have to demonstrate that for an FSM B that has passed the test suite G , but is distinguishable from A , the set of states in FSM $A \oplus B$ traversed by an input sequence from some N_i , applied at the state $\psi(s_0 t_0, \alpha_i)$, together with the set of states where sequences of the core cover K take the $A \oplus B$ from the initial state, has two conflicting states. Moreover, in these two conflicting states, states of FSM A are quasi-equivalent to states of the set D that bounds the input sequence.

Let $B \not\cong A$ and β be a shortest sequence such that the sequence $\alpha_i \beta \in X_{A \oplus B}^*$, $\alpha_i \in K$, is a transfer sequence in DM $A \oplus B$ to state q with a fail-transition. By definition of the set G , there exists a set $D_{\beta\text{after}\alpha_i}$ such that $\#_{\delta_{(s_0, \alpha_i)}}(D_{\beta\text{after}\alpha_i}, \beta) \geq m - |D_{\beta\text{after}\alpha_i}| + 1$, since $B \cong_G A$. Let ν be the shortest prefix of β such that:

1. $\#_{\delta_{(s_0, \alpha_i)}}(D_\nu, \nu) = m - |D_\nu| + 1$ for some set D_ν of mutually distinguishable states of A ,

2. ν is an acyclic sequence from the state $\psi(q_0, \alpha_i)$, moreover, for any state (s, t) traversed by ν and any state $(s', t') = \psi(q_0, \mu)$, $\mu \in K$, it holds that $s' \supseteq s \Rightarrow t' \neq t$.

We have the set $P(\nu, D_\nu) = \{\nu_1, \dots, \nu_{m-|D_\nu|+1}\}$ of $m - |D_\nu| + 1$ prefixes of ν such that $0 < |\nu_1| < \dots < |\nu_{m-|D_\nu|+1}|$. If the corresponding $m - |D_\nu| + 1$ states of FSM B , $\Delta(t_0, \alpha_i \nu_1), \dots, \Delta(t_0, \alpha_i \nu_{m-|D_\nu|+1})$, are not distinct then, by definition of an acyclic sequence, the sequence ν traverses two conflicting states (s, t) and (s', t) , where s, s' are quasi-equivalent to two different states of the set D_ν , i.e., $|D_\nu| > 1$. Assume, therefore, that the states $\Delta(t_0, \alpha_i \nu_1), \dots, \Delta(t_0, \alpha_i \nu_{m-|D_\nu|+1})$ are distinct. Let K_ν be a subset of the core cover K of A that has exactly $|D_\nu|$ transfer sequences such that for any $j = 1, \dots, m - |D_\nu| + 1$ there exists $\gamma \in K_\nu$ for which $\tilde{\alpha}_{s_0, \gamma} \supseteq \tilde{\alpha}_{s_0, \alpha_i \nu_j}$ holds. The corresponding $|D_\nu|$ states of FSM B are distinct, if the core cover K does not traverse conflicting states. Then the two subsets of states of FSM B , $\{\Delta(t_0, \alpha_i \nu_1), \dots, \Delta(t_0, \alpha_i \nu_{m-|D_\nu|+1})\}$ and $\{\Delta(t_0, \gamma) \mid \gamma \in K_\nu\}$ intersect, as, together, they have $m - |D_\nu| + 1 + |D_\nu| = m + 1$ states, while B has only m states. Thus, there exist a sequence $\mu \in K_\nu$ and non-empty prefix ν_j of ν such that $\Delta(t_0, \mu) = \Delta(t_0, \alpha_i \nu_j)$, while $\tilde{\alpha}_{s_0, \mu}$ and $\Delta(t_0, \alpha_i \nu_j)$ are quasi-equivalent to some states in the set D_ν . If they are quasi-equivalent to the same state then $\tilde{\alpha}_{s_0, \mu} \supseteq \tilde{\alpha}_{s_0, \alpha_i \nu_j}$ by definition of the set K_ν . The case $\tilde{\alpha}_{s_0, \mu} \supseteq \tilde{\alpha}_{s_0, \alpha_i \nu_j}$ contradicts to property (2) of the sequence ν . Thus, $\tilde{\alpha}_{s_0, \mu}$ and $\tilde{\alpha}_{s_0, \alpha_i \nu_j}$ are quasi-equivalent to two different states of the set D_ν . Hence, $|D_\nu| > 1$ and the set of states $\{\psi(q_0, \alpha) \mid \alpha \in K \wedge \exists s \in D_\nu : \tilde{\alpha}_{s_0, \alpha} \supseteq s \vee \exists \gamma \in P(\nu, D_\nu) : \alpha = \alpha_i \gamma\}$ contains two conflicting states $[\tilde{\alpha}_{s_0, \mu}, \Delta(t_0, \mu)]$ and $[\tilde{\alpha}_{s_0, \alpha_i \nu_j}, \Delta(t_0, \alpha_i \nu_j)]$, where $\tilde{\alpha}_{s_0, \mu}$ and $\tilde{\alpha}_{s_0, \alpha_i \nu_j}$ are quasi-equivalent to two different states of the set D_ν . \square

The set G from Proposition 4 can easily be extended to cover fail-transitions missed by G , using the following property of distinguishing machines.

Lemma 3. Let (s', t) and (s, t) be conflicting states of $A \oplus B$ such that $s' \not\cong_\alpha s$ for some input sequence α , then the input sequence α covers a fail-transition in $DM A \oplus B$ from (s', t) or (s, t) . \square

The presence of distinguishable states in FSM A allows us to decrease the length (and therefore, the number) of sequences needed to traverse states (cf. Proposition 3 and 4) at a price of including additional input sequences distinguishing states in A . The gain increases with the number of distinguishable states. Informally speaking, the more distinguishable states FSM A has, the more the structure of complete test suites resembles that implied by existing test derivation methods for reduced machines. To demonstrate the point more formally, we determine the set G of Proposition 4 for the case of completely defined reduced FSM A .

The core cover K of such a machine coincides with a state cover V of the given machine. All n states of A are mutually distinguishable, thus $|D_{\beta \text{ after } \alpha}| = n$ for any $\alpha \beta \in X_A^*$ with non-empty β . Then each set N_i is nothing more but the set of all input sequences of the length up to $m - n + 1$ acceptable in state $\tilde{\alpha}_{s_0, \alpha_i}$, i.e. $N_i = X^{m-n+1} \cap X_A^*(\tilde{\alpha}_{s_0, \alpha_i})$. If A is completely defined then $N_i = X^{m-n+1}$ for all $\alpha_i \in V$. Thus, $G = VX^{m-n+1}$. Let W denote a characterization set of FSM A . For any two distinguishable states, it contains a sequence that distinguishes

them. According to Lemma 3, the sequences of W , concatenated to each sequence in the set G , cover any remaining fail-transition. The set $VX^{m-n+1}W$ is exactly the test suite derived by the W-method [Vas73], [Cho78], which is m -complete. A similar observation applies to the test derivation method developed for reduced partial FSMs in [YaL95]. This indicates that the method we are developing includes the existing methods as special cases.

We are now ready to present the detailed description of the method.

Input. FSM A ; the list of pairs of distinguishable states and for each pair (s, s') , an input sequence $\gamma(s, s')$ that distinguishes s and s' ; all maximal by inclusion sets of mutually distinguishable states; and an integer m equal or greater than the maximal number of mutually distinguishable states.

Output. An m -complete test suite for FSM A .

The method includes the following steps.

1. Determine a core cover K of A .
2. For each $\alpha_i \in K$, determine the set of all shortest input sequences $N_i \subset X_A^*(\delta_{s_0}, \alpha_i)$ such that $\#\delta_{s_0, \alpha_i}(D_{\beta \text{ after } \alpha_i}, \beta) = m - |D_{\beta \text{ after } \alpha_i}| + 1$ and $\delta_{s_0}, \alpha_i \beta \in D_{\beta \text{ after } \alpha_i}$ for each sequence $\beta \in N_i$. Let $D(N_i)$ denote the union of sets $D_{\beta \text{ after } \alpha_i}$ over all $\beta \in N_i$.
3. $\alpha_1 := \varepsilon$. $R := \{\alpha_1\}$.
4. For each N_i , where $\alpha_i \in R$, and for each state $s \in D(N_i)$, determine $\alpha \in K$ such that $\delta_{s_0}, \alpha \sqsupseteq s$ and include α in R . Repeat this step until $R = K$ or no more sequences from K are included into R .
5. Determine the set $\{\alpha_i \gamma(\delta_{s_0}, \alpha_i), \delta_{s_0}, \alpha_j) \mid \alpha_i, \alpha_j \in R\} = TS_1$.
6. Determine the set $\{\alpha_i \kappa \gamma(s, s') \mid \alpha_i \in R \wedge \beta \in N_i \wedge \kappa \in P(\beta, D_{\beta \text{ after } \alpha_i}) \wedge s, s' \in D_{\beta \text{ after } \alpha_i}\} = TS_2$.
7. Determine $TS_1 \cup TS_2 = TS$, that is an m -complete test suite. □

We provide a few comments on the steps of this method. Determining a core of a given FSM in Step 1 is a typical combinatorial, set cover problem, i. e., find a minimal number of states that “cover” all the states of a given machine. We say that one state “covers” another state if the former is quasi-equivalent to the latter. Once a core is determined, a core cover can be obtained from a spanning tree similar to the one used to find a state cover, see e.g. [Chow78]. Step 2 is a key part of the method. It corresponds to constructing the set of traversal sequences VX^{m-n+1} in the W-, Wp-methods and other methods for reduced machines. The step can be performed in two different ways. The first option is to start from all accepted sequences of the length of nm and to prune each of them until the condition $\#\delta_{s_0, \alpha_i}(D_{\beta \text{ after } \alpha_i}, \beta) = m - |D_{\beta \text{ after } \alpha_i}| + 1$ is satisfied for a sequence β at hand. Another option is to start with an empty input sequence and to grow it in all the possible ways (obtaining acceptable sequences) to satisfy the condition. Step 4 attempts to minimize the number of sets of traversal sequences applied after transfer sequences from the core cover (see Corollaries 2 and 3), and thus, can always be omitted, by assuming $R = K$.

The set TS_1 ensures that if sequences in the core cover (or the set R , to be more precise) traverse conflicting states in a particular DM $A \oplus B$ then a fail-transition is covered by a

sequence from TS_1 . Similarly, the set TS_2 guarantees that if conflicting states are traversed by a sequence in N_i then a fail-transition is covered. Note that if FSM A has no distinguishable states then $TS_1 = R$ and $TS_2 = G = \bigcup_{\alpha_i \in R} \alpha_i N_i \supseteq R$, i.e., Steps 5-7 can be skipped (see Proposition 3).

Example

We derive a complete test suite for the partial FSM A shown in Figure 2, assuming that the number of states in any implementation machine does not exceed 3, i.e., $m \leq 3$. The core cover $K = \{\varepsilon, 1\}$. We have $1 \not\cong 2$ and $\gamma(1, 2) = 1$, as the machine in response to input 1 produce output 0 in state 1 and 1 in state 2. $(1, 2)$ is the only pair of distinguishable states, hence, we have two maximal by inclusion sets of mutually distinguishable states - $D_1 = \{1, 2\}$ and $D_2 = \{3\}$.

We determine traversal sequences. In Step 2, let $\alpha_1 = \varepsilon$. The process of determining the set of input sequences N_1 is illustrated in Figure 3. A sequence β in N_1 has to be trimmed as soon as $\#_1(D_\beta, \beta) = m - |D_\beta| + 1$, this means when either $\#_1(D_1, \beta) = m - |D_1| + 1 = 3 - 2 + 1 = 2$ or $\#_1(D_2, \beta) = m - |D_2| + 1 = 3 - 1 + 1 = 3$. As an example, the sequence 11 cannot be terminated for neither of the two conditions is satisfied. However, for the sequence 112, we have $\#_1(D_1, 112) = 2$, so it is not extended any farther. We obtain $N_1 = \{112, 12\}$.

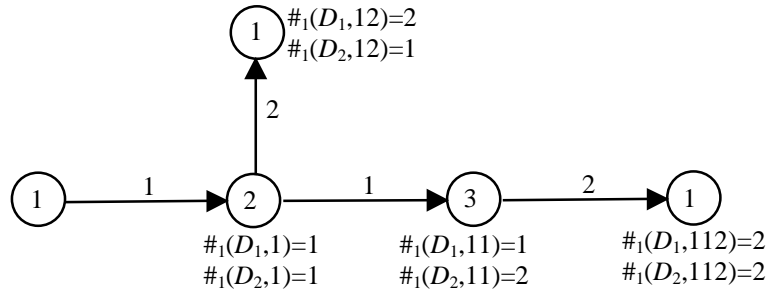


Figure 3. Constructing the set N_1 of traversal sequences.

Similarly, the set $N_2 = \{121, 21\}$ is determined for $\alpha_2 = 1$. Figure 4 illustrates the process.

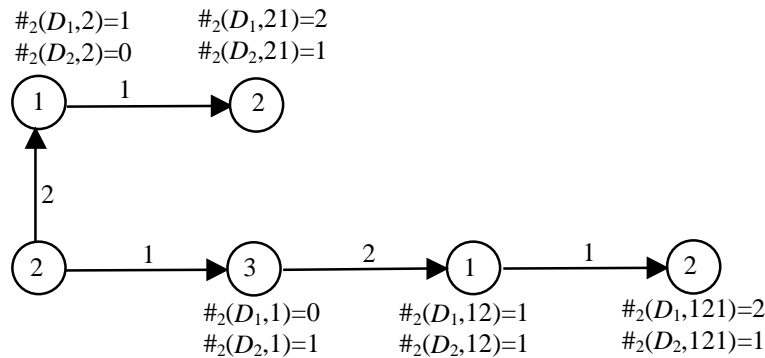


Figure 3. Constructing the set N_2 of traversal sequences.

In Step 3 and 4, we have $R = K = \{\varepsilon, 1\}$. Executing Step 5 yields $TS_1 = \{\alpha_i \chi(\delta_{s_0}, \alpha_i), \delta_{s_0}, \alpha_j) \mid \alpha_i, \alpha_j \in R\} = \{1, 11\}$. Step 6 - the set $TS_2 = \{\alpha_i \kappa \chi(s, s') \mid \alpha_i \in R \wedge \beta \in N_i \wedge \kappa \in P(\beta, D_{\beta \text{ after } \alpha_i})\} \wedge s, s' \in D_{\beta \text{ after } \alpha_i} = \{11, 1121, 11211, 121, 1211\} = \{11211, 1211\}$. The test suite is $TS = TS_1 \cup TS_2 = \{11211, 1211\}$, it is m -complete for $m \leq 3$.

Proposition 5. The set of input sequences obtained with the above method is an m -complete test suite for a given FSM A . □

The statement comes as a corollary to Proposition 4 and Lemma 3.

Finally, we discuss the complexity of test suites obtained with the proposed method. As Figure 1 illustrates, there exist pathological FSMs for which the smallest m -complete test suite may coincide with the set X_A^{nm} . On the other hand, in the class of completely defined reduced FSMs, the resulting test suite coincides with the set $VX^{m-n+1}W$ (the W-method). The size of tests for FSMs with distinguishable states depends on the choice of distinguishing sequences $\chi(s, s')$ (constituting a characterization set W), for these sequences are the input to our method. "Optimal" distinguishing sequences should yield a test suite of a minimal complexity. The problem of determining such sequences remains open for all existing test derivation methods with fault coverage, our method included. Treating distinguishing sequences as the input to our method, we separate this combinatorial problem from all the other issues of deriving tests for partial machines, for which we offer the solutions in this paper.

4 CONCLUSION

We revisited in the paper the problem of test derivation with fault coverage from partially defined deterministic FSMs and further elaborated the so-called state-counting approach outlined in our earlier work.

The contributions of this paper can be summarized as follows. We investigated the role quasi-equivalent states play in the structure of tests with complete fault coverage. We demonstrated that in presence of quasi-equivalent states, the use of a state cover is definitely redundant. A subset of it is sufficient to obtain a complete test suite. This result allowed us to extend the notion of a state cover to a core cover for a given FSM. The use of this new characteristic of partial machines significantly reduces the number of traversal sequences necessary to cover transitions in implementation machines. The resulting save is indeed essential, since each transfer sequence in a state cover is concatenated with several traversal sequences which, in turn, are concatenated with state identification sequences (if they exist).

The rules for trimming traversal sequences were further refined, enforcing their earlier termination in the presence of quasi-equivalent states. We showed that the length of traversal sequences could be defined by counting the number of times quasi-equivalent states and not necessarily the same state (as was previously suggested) are traversed. We also demonstrated that there is no need to apply distinguishing sequences after each prefix of traversal sequences.

These new features of our method can significantly shorten the resulting test suite in the presence of quasi-equivalent and distinguishable states. Due to these non-trivial

improvements, our method subsumes the existing methods; moreover, there are FSM classes where our method is superior to them, as it yields shorter tests.

Several research directions can be pursued from this work. The ideas presented in this paper may help improve the results of [PYL93], [LPB94], [PYB96] for the nondeterministic case. Based on this approach it is also interesting to develop methods for deriving complete tests from partial specifications in the form of partially defined LTSs and I/O automata. Another potential direction is to try to construct a single checking sequence from the resulting complete test suite for a partial machine, similar to what was done in [ReU95] for complete machines. Last but not least, in spite of the fact that the proposed method provides a significant reduction in length of tests, the problem of finding a complete test suite of a minimal overall length for a given I/O FSM remains open and requires further research.

Acknowledgments. This work was in part supported by the NSERC grant OGP0194381.

5 REFERENCES

- [ACY95] R. Alur, C. Courcoubetis, and M. Yannakakis, *Distinguishing tests for nondeterministic and probabilistic machines*, Proceedings of the 27th ACM Symposium on Theory of Computing, 1995, pp. 363-372.
- [Cho78] T. S. Chow, *Test software design modeled by finite state machines*, IEEE Transactions, SE-4, 3, 1978, pp. 178-187.
- [FBK91] S. Fujiwara, G. v. Bochmann, F. Khendek, M. Amalou, and A. Ghedamsi, *Test selection based on finite state models*, IEEE Transactions, SE-17, 6, 1991, pp. 591-603.
- [FJJ96] J.-C. Fernandez, C. Jard, T. Jérón, and C. Viho, *An experiment in automatic generation of test suites for protocols with verification technology*, Science of Computer Programming, 1996.
- [Gil62] A. Gill, *Introduction to the theory of finite-state machines*, Mc Graw-Hill, New York, 1962.
- [GrL65] A. Grasselli and F. Luccio, *A method for minimizing the number of internal states in incompletely specified sequential networks*, IEEE Transactions, EC-14, 3, 1965, pp. 350-359.
- [Hen64] F. C. Hennie, *Fault detecting experiments for sequential circuits*, Proceedings of the IEEE 5th Ann. Symp. On Switching Circuit Theory and Logical Design, 1964, pp. 95-110.
- [Koh70] Z. Kohavi, *Switching and finite automata theory*, Mc Graw-Hill, New York, 1970.
- [KPY99] I. Koufareva, A. Petrenko, and N. Yevtushenko, *Test generation driven by user-defined fault models*, Proceedings of the Twelfth International Workshop on Testing of Communicating Systems, Hungary, 1999, pp. 215-233.
- [LeY96] D. Lee and M. Yannakakis, *Principles and methods of testing finite state machines, a survey*, Proceedings of the IEEE, vol. 84, 8, 1996, pp. 1090-1123.
- [LBP94] G. Luo, G. v. Bochmann, and A. Petrenko, *Test selection based on communicating nondeterministic finite state machines using a generalized Wp-method*, IEEE Transactions, SE-20, 2, 1994, pp. 149-162.
- [LPB94] G. Luo, A. Petrenko, and G. v. Bochmann, *Selecting test sequences for partially specified nondeterministic finite state machines*, Proceedings of the IFIP Seventh International Workshop on Protocol Test Systems, Japan, 1994, pp. 95-110.

- [Pet91] A. Petrenko, *Checking experiments with protocol machines*, Proceedings of the IFIP Fourth International Workshop on Protocol Test Systems, the Netherlands, 1991, pp. 83-94.
- [PBD93] A. Petrenko, G. v. Bochmann, and R. Dssouli, *Conformance relations and test derivation*, Invited Paper, Proceedings of the IFIP Sixth International Workshop on Protocol Test Systems, France, 1993, pp. 157-178.
- [PBY96] A. Petrenko, G. v. Bochmann, and M. Yao, *On fault coverage of tests for finite state specifications*, Computer Networks and ISDN Systems, 29, December 1996, pp. 81-106.
- [PYB96] A. Petrenko, N. Yevtushenko, and G. v. Bochmann, *Testing deterministic implementations from their nondeterministic specifications*, Proceedings of the IFIP Ninth International Workshop on Testing of Communicating Systems, Germany, 1996, pp. 125-140.
- [PYL93] A. Petrenko, N. Yevtushenko, A. Lebedev, and A. Das, *Nondeterministic state machines in protocol conformance testing*, Proceedings of the IFIP Sixth International Workshop on Protocol Test Systems, France, 1993, pp. 363-378.
- [ReU95] A. Rezaki and H. Ural, *Construction of checking sequences based on characterization sets*, Computer Communications, 18, December 1995, pp. 911-920.
- [Vas73] M. P. Vasilevskii, *Failure diagnosis of automata*, Cybernetics, Plenum Publishing Corporation, NY, No. 4, 1973, pp. 653-665.
- [YaL95] M. Yannakakis and D. Lee, *Testing finite state machines: fault detection*, Journal of Computer and System Sciences, 50, 1995, pp. 209-227.
- [YePe89] N. Yevtushenko and A. Petrenko, *Checking capabilities of multiple experiments*, Automatic Control and Computer Sciences, Allerton Press, Inc., USA, #3, 1989.
- [YePe90] N. Yevtushenko and A. Petrenko, *Test derivation method for an arbitrary deterministic automaton*, Automatic Control and Computer Sciences, Allerton Press, Inc., USA, #5, 1990.