Overcoming observability problems in distributed test architectures✩

J. Chen a,∗, R.M. Hierons b, H. Ural c

a School of Computer Science, University of Windsor, Windsor, ON, Canada
b School of Information Systems, Computing and Mathematics, Brunel University, Uxbridge, Middlesex, UK
c School of Information Technology and Engineering, University of Ottawa, Ottawa, ON, Canada

Received 24 January 2005; received in revised form 10 February 2006; accepted 10 February 2006
Available online 9 March 2006
Communicated by J.L. Fiadeiro

Keywords: Finite state machine; Testing; Distributed systems; Observability; Controllability

1. Introduction

In distributed testing, a distributed test architecture is used where a tester is placed at each port of the system under test (SUT) N and an input sequence is applied. When N is a state based system specified as a finite state machine (FSM) M an input sequence to be applied to N can be constructed from M; the input sequence is then called a test sequence or a checking sequence. The application of a test/checking sequence [5] in the distributed test architecture introduces the possibility of controllability and observability problems. These problems occur if a tester cannot determine either when to apply a particular input to N, or whether a particular output from N has been generated in response to a specific input, respectively [6].

✩ This work was supported in part by Natural Sciences and Engineering Research Council (NSERC) of Canada under grant RGPIN 976 and 209774, Leverhulme Trust grant number F/00275/D, Testing State Based Systems, and Engineering and Physical Sciences Research Council grant number GR/R43150, Formal Methods and Testing (FORTEST).
* Corresponding author.
E-mail address: xjchen@uwindsor.ca (J. Chen).

Clearly, the testers in a distributed test architecture should coordinate with each other in order to overcome controllability and observability problems. This coordination can be achieved by exchanging external coordination messages among the testers over a dedicated channel during the application of the test or checking sequence. However, sometimes we want to avoid the use of external coordination messages whenever possible since they require us to set up an additional communications network and this makes testing more expensive. In addition, coordination messages introduce delays and these delays can cause problems if we have timing issues in our testing. Let us suppose, for example, that in testing we wish to follow the input of $x_1$ at port $p_1$ with the input of $x_2$ at port $p_2$ ($p_1 \neq p_2$) and in order to achieve this we sent a coordination message from the tester at $p_1$ to the tester at $p_2$ after $x_1$ has been input. If we require that the time between $x_1$ and $x_2$ being sent is at most $t$ and the process of sending coordination messages takes time $t' > t$ then this approach is not appropriate. The timing issues can be particularly problematic if the SUT responds rapidly to inputs, relative to the network used for coordina-
tion messages. See [4] for a discussion of some of the timing issues that arise in using external coordination messages.

As an alternative solution to overcome controllability and observability problems, we can construct a test or checking sequence where the coordination among testers is achieved indirectly via the testers’ interactions with \( N \) [6,7], and thus no external coordination message exchange is necessary. There exist specifications however, where such a test or checking sequence does not exist [1,2,8].

This paper investigates conditions that must be satisfied by an FSM for the existence of test or checking sequences that can be applied in a distributed test architecture without encountering controllability and observability problems and without using external coordination messages. Such conditions have two potential values. First, they can be used to determine whether we require coordination messages and thus a network that connects the testers. Second, if we wish to avoid the use of coordination messages in testing then these conditions can be seen as testability conditions that can inform the design process. Results given in this paper differ from those in [3] in the following ways. First, the conditions are strictly weaker than those in [3] since we are less restrictive in the ways we achieve our goals. Second, [3] only considered observability problems; we consider both controllability and observability problems. In addition, [3] only considered a particular type of observability problem and we generalize this. Finally, we investigate the situation in which we need only add input sequences to complement a given test/checking sequence and prove that the conditions for this problem are equivalent to those for the original one.

2. Preliminaries

An \( n \)-port Finite State Machine \( M \) (simply called an FSM \( M \)) is defined as \( M = (S, I, O, \delta, \lambda, s_0) \) where \( S \) is a finite set of states; \( s_0 \in S \) is the initial state; \( I = \bigcup_{i=1}^{n} I_i \), where \( I_i \) is the input alphabet of port \( i \), and \( I_i \cap I_j = \emptyset \) for \( i, j \in [1, n], i \neq j \); \( O = \prod_{i=1}^{n} (O_i \cup \{-\}) \), where \( O_i \) is the output alphabet of port \( i \), and \(-\) means null output; \( \delta : S \times I \rightarrow S \) is the transition function; and \( \lambda : S \times I \rightarrow O \) is the output function. Each \( y \in O \) is a vector of outputs \( \langle o_1, o_2, \ldots, o_n \rangle \) where \( o_i \in O_i \cup \{-\} \) for \( i \in [1, n] \). In the following, \( p \in [1, n] \) is a port, \( x \in I \) is a general input, and \( x_p \in I_p \) is an input at \( p \). We use \( y \mid_p \) to denote the output at \( p \) in \( y \). A transition of \( M \) is a triple \( t = (s_1, s_2, x/y) \), where \( s_1, s_2 \in S, x \in I, \) and \( y \in O \) such that \( \delta(s_1, x) = s_2, \lambda(s_1, x) = y, s_1 \) and \( s_2 \) are called the starting state and the ending state of \( t \), respectively. The input/output pair \( x/y \) is the label of \( t \). \( T \) denotes the set of all transitions in \( M \).

A path \( \rho = t_1, t_2, \ldots, t_k \) \((k \geq 0)\) is a finite sequence of transitions such that for \( k \geq 2 \), the ending state of \( t_i \) is the starting state of \( t_{i+1} \) for all \( i \in [1, k-1] \). When the ending state of the last transition of path \( \rho_1 \) is the starting state of the first transition of path \( \rho_2 \), we use \( \rho_1 \rho_2 \) to denote the concatenation of \( \rho_1 \) and \( \rho_2 \). The label of a path \( (s_1, s_2, x_1/y_1) (s_2, s_3, x_2/y_2) \ldots (s_k, s_{k+1}, x_k/y_k) \) \((k \geq 1)\) is the sequence of input/output pairs \( x_1/y_1 x_2/y_2 \ldots x_k/y_k \) which is an input/output sequence. The input portion of a path \( (s_1, s_2, x_1/y_1) \ldots (s_k, s_{k+1}, x_k/y_k) \) \((k \geq 1)\) is the input sequence \( x_1 x_2 \ldots x_k \).

Given an FSM \( M \) and a sequence \( tt' \) of consecutive transitions, \( t = (s_1, s_2, x/y) \) and \( t' = (s_3, s_4, x'/y') \), a controllability problem occurs if the port \( p \) at which \( x' \) is input is not involved in \( t : x \notin I_p \) and \( y \mid_p = -\). If this problem occurs then the tester at \( p \) does not know when to send \( x' \) and so \( tt' \) cannot be applied in testing.

Fig. 1 shows a 2-port FSM where ports \( U \) and \( L \) stand for its upper interface and the lower interface respectively. An output vector \( y = \langle o_{out1}, o_{out2} \rangle \) on the label of a transition of a 2-port FSM is a pair of outputs with \( o_{out1} \in O_1 \cup \{-\} \) at \( U \) and \( o_{out2} \in O_2 \cup \{-\} \) at \( L \). * denotes any non-empty output. In this FSM, controllability problem occurs in sequence \( t_2 t_3 \) because tester \( U \) neither gives the input of \( t_2 \) nor receives the output of \( t_2 \) and thus does not know when it should provide input \( i_3 \) of \( t_3 \). Therefore, we should avoid attempting to apply \( t_2 t_3 \) in testing.

Consecutive transitions \( t \) and \( t' \) form a synchronizable pair of transitions if \( t' \) can follow \( t \) without causing a controllability problem. A path in which every pair of consecutive transitions is synchronizable is called a synchronizable path. An input/output sequence is synchronizable if it is the input portion of a synchronizable path.

A specification FSM is called intrinsically synchronizable if for any pair of transitions \((t, t')\) there is a synchronizable path that starts with \( t \) and ends with \( t' \). In order to construct synchronizable test or checking sequences it is necessary that the specification FSM is intrinsically synchronizable [1], for reasons that are similar to the requirement when testing from a single-port FSM \( M' \) that \( M' \) is strongly connected. As reflected in our results formalized in Theorems 1, 2 and 4, here
we only consider specification FSMs that are intrinsically synchronizable.

A same-port-output-cycle in an FSM is a synchronizable path \((s_1, s_2, x_1/y_1)\) \((s_2, s_3, x_2/y_2)\) \(\ldots\) \((s_k, s_{k+1}, x_k/y_k)\) \((k \geq 2)\) such that \(s_1 = s_{k+1}, s_i \neq s_{i+1}\) for \(i \in [1, k]\), and there exists a port \(p\) with \(y_i|p \neq -\) and \(x_i \notin I_p\) for all \(i \in [1, k]\). If such a cycle exists then there is no bound on the number of outputs the tester at port \(p\) can see without providing an input. This is a situation not too dissimilar to a livelock. As stated in Theorem 1, here we assume that the specification FSMs contain no same-port-output-cycles.

In Fig. 1, we have mentioned that we should not use \(t_2t_3\) in the application of test or checking sequence in order to avoid controllability problems. Instead, we can apply \(t_2t_4\) or \(t_3t_4\) since \((t_2, t_4)\) and \((t_3, t_3)\) are synchronizable pairs of transitions. Now, suppose in \(N\), output \(o_4\) at \(U\) is shifted to transition \(t_2\) in \(N\), as the dashed arrow shows. When using \(t_2t_4\) in testing, tester \(U\) will observe the expected output sequence \(o_4\) in response to input sequence \(i_2i_4\). It will not notice a failure because the two output faults have masked each other. Since this can occur, we say that \(t_2\) and \(t_4\) are involved in a potentially undetectable output shift fault in the sense that it is possible that the SUT has an output shifted to/from another transition and this fault cannot be detected without additional subsequences or external coordination message exchanges.

Similarly, if \(o_4\) at \(U\) is shifted to transition \(t_1\) in \(N\), by using \(t_1t_2t_4\), tester \(U\) will not be able to detect the fault. So \(t_1, t_2\) and \(t_4\) are all involved in potentially undetectable output shift faults.

Formally, a transition \(t\) is involved in a potentially undetectable output shift fault at \(p\) if and only if there exists a transition \(t'\) and a path \(\rho\) such that at least one of the following holds.

1. \(t\rho t'\) is a synchronizable path, no transition in \(\rho t'\) contains input at \(p\), the outputs at \(p\) in all transitions contained in \(\rho\) are empty, \(t|p = -\) and \(t'|p \neq -\). In this case an undetectable output shift fault can occur between \(t\) and \(t'\) in \(t\rho t'\) and we call this a backward output shift fault.
2. \(t'\rho t\) is a synchronizable path, no transition in \(\rho t\) contains input at \(p\), the outputs at \(p\) in all transitions contained in \(\rho\) are empty, \(t|p = -\) and \(t'|p \neq -\). In this case an undetectable output shift fault can occur between \(t'\) and \(t\) in \(t'\rho t\) and we call this a forward output shift fault.

When \(p\) is empty, we also say that \(t\) is involved in a potentially undetectable 1-shift output fault.

The observability problem occurs when we have potentially undetectable output shift faults in the specification FSM and this is a problem we wish to avoid in test or checking sequence generation. Note that [3] only considers potentially undetectable 1-shift output faults.

### 3. Definitions of leading and trailing paths

We want to produce a test or checking sequence from \(M\) without using external coordination messages among remote testers so that when it is applied to \(N\), if each tester’s observation is correct, then we are able to conclude that the output of each transition at each port that \(N\) produces in response to the input is correct. Due to the observability problems, if a transition \(t\) is involved in a potentially undetectable output shift fault in the path of a test or checking sequence with another transition \(t'\) at a port \(p\), we require that this test or checking sequence contains a subsequence so that we can identify its output sequence from the overall output sequence produced by \(N\) and that we can conclude the correct output of \(t\) or \(t'\) at \(p\) from the correct output of this subsequence. The following concept characterizes the properties of such subsequences.

**Definition 1.** Given transition \(t = (s_1, s_2, x/y)\), a synchronizable path \(\rho_1\rho_2\) is said to be a verifying path for...
(t, p) if the following holds: for every synchronizable path \( \rho = \rho_1 t \rho_2 \rho_3 \) of \( M \) with starting state \( s_0 \), if the tester at \( p \) sees the expected sequence of inputs and outputs when the input portion of \( \rho \) is applied to the SUT then we can deduce that when input portion of \( \rho \) was applied the SUT must have produced output \( y \) at \( p \) in response to the input of \( x \) after the input portion of \( \rho_1 \). We call \( \rho_1 \) a leading path for \((t, p)\), and \( \rho_2 \) a trailing path for \((t, p)\). When \((t, p)\) has a verifying path, we also say that \((t, p)\) is verifiable.

If we have a verifying path \( \rho_1 t \rho_2 \) for \((t, p)\) then we can embed its input portion within any test/checking sequence and we know that if no failure is observed when the test/checking sequence is applied to the SUT then the SUT must have produced the expected output at \( p \) in response to the input that was intended to trigger \( t \). This allows us to check the output of \( t \) at \( p \) but relies on us knowing that the corresponding transition of \( N \) is executed when expected. This is the case if either it is known that every transition of \( N \) has the required final state or if the final state of each transition is verified in another part of the test/checking sequence. This paper concerns the issue of overcoming observability problems and so we assume that the final state of each transition is either known to be correct or is verified through some other means. In this paper, we consider the existence of absolute verifying paths for \((t, p)\) where \( t \) has non-empty output.

**Definition 2.** Given transition \( t = (s_1, s_2, x/y) \) where \( y \) is not \( - \), \( \rho_1 \) is an absolute leading path for \((t, p)\) if either \( \rho_1 = \epsilon \) and \( x \in I_p \) or \( \rho_1 \neq \epsilon \) and: \( \rho_1 t \) is a synchronizable path; all transitions in \( \rho_1 \) have non-empty output at port \( p \); and the first transition, and only the first transition in \( \rho_1 \) has input at \( p \). Path \( \rho_2 \) is an absolute trailing path for \((t, p)\) if \( t \rho_2 \) is a synchronizable path; all transitions in \( \rho_2 \), possibly except the last, have non-empty output at port \( p \); and the last transition, and only the last transition in \( \rho_2 \) has input at \( p \). \( \rho = \rho_1 t \rho_2 \) is an absolute verifying path for \((t, p)\) if \( \rho_1 \) and \( \rho_2 \) are absolute leading path and absolute trailing path for \((t, p)\), respectively.

No matter how \( \rho = \rho_1 t \rho_2 \) is concatenated with other sequences, we can determine the output sequence at \( p \) in response to the first \(|\rho| - 1\) inputs of \( \rho \) as this is immediately preceded and followed by input at \( p \). Further, since we expect \(|\rho| - 1\) outputs at \( p \) within this output sequence, and there are \(|\rho| - 1\) corresponding inputs, the output of \( t \) at \( p \) must have been correct if the correct sequence of observations was seen at \( p \). Thus, absolute verifying paths are verifying paths. Note that the conditions ensure that \( \rho_1 \) and \( \rho_2 \) cannot be shortened without violating the required properties.

In Fig. 1, \( t_4 \) is involved in potentially undetectable output shift faults at port \( U \), and \( \alpha_4 \neq - \). \( t_5 t_4 t_5 \) and \( t_3 t_4 t_5 \) are all absolute verifying paths for \((t_4, U)\).

### 4. The goals

Let \( T_p \) denote the transitions of \( M \) that can be involved in potentially undetectable output fault shifts at \( p \). Thus \( t \in T_p \) if there exists a transition \( t' \) and a synchronizable path \( \rho t' \) or \( t' \rho t \) of \( M \) such that there is a potentially undetectable output fault involving \( t \) and \( t' \) at \( p \). If transition \( t \) has output \( y \) then \( y \) denotes \( y \) at \( p \). Let \( T'_p = T_p \cap \{ t \mid |t|_p \neq - \} \) denote the set of transitions involved in potentially undetectable output shift faults at \( p \) whose output at \( p \) are non-empty.

The first goal is to determine if \((t, p)\) is verifiable for every \( p \in [1, n] \) and \( t \in T_p \). If this is the case then we can produce a verifying path for each \((t, p)\) and include its input portion in a test or checking sequence to check the output of every transition of the SUT at every port without suffering from controllability or observability problems.

The second goal is: given a test/checking sequence which is the input portion of path \( \rho \), determine if \((t, p)\) is verifiable for every \( p \) and \( t \) such that \( t \) is the first or last transition in \( \rho \) or \( t \) is involved in a potentially undetectable output shift fault in \( \rho \). This appears to weaken the requirements since we are simply verifying that there is no potentially undetectable output shift faults within a given \( \rho \) or at its first/last transition.

Below, we present necessary and sufficient condition for \((t, p)\) to have an absolute verifying path for every \( p \) and \( t \in T_p \) and show that this achieves the first goal. Then, we prove that the condition is the same for the second goal.

**Theorem 1.** Let \( M \) be a given FSM which is intrinsically synchronizable and has no same-port-outputcycles. Let \( p \) be any port of \( M \).

(i) \((t_0, p)\) has an absolute leading path for every \( t_0 \in T'_p \), if and only if for every \( t = (s_1, s_2, x/y) \in T'_p \), \( x \notin I_p \) implies \( \exists (s_3, s_1, x'/y') \in T \) synchronizable with \( t \) such that \( y' \notin - \).

(ii) \((t_0, p)\) has an absolute trailing path for every \( t_0 \in T'_p \), if and only if for every \( t = (s_1, s_2, x/y) \in T'_p \),

\(^2\) We include the first and last transitions of \( \rho \) since we will combine \( \rho \) with other sequences to form a single test/checking sequence.
\((x_2, s_4, x'/y') \in T \) synchronizable with \( t \) such that 
\( x' \in I_p \lor y' | p \neq - \).

**Proof.** We prove part (i); part (ii) follows in a similar way.

\((\Leftarrow)\) Consider some \( t_0 \in T_p^f \); we prove that there is an absolute leading path \( \sigma_0 \). If the input of \( t_0 \) is at \( p \), \( \sigma_0 = \epsilon \). Suppose that the input of \( t_0 \) is not at \( p \). We use proof by contradiction: suppose \( t_0 \) has no absolute leading path and let \( \sigma \) denote a longest path such that \( \sigma t_0 \) is synchronizable, every transition in \( \sigma \) has non-empty output at \( p \) and no transition in \( \sigma \) has input at \( p \). Since \( M \) has no same-port-output-cycles and has a finite number of states there must exist such a (finite) \( \sigma \). Let \( t_2 = (r_3, r_4, x_2/y_2) \) be the first transition of \( \sigma \) and thus \( x_2 \notin I_p \).

Suppose \( t_2 \in T_p^f \). Since \( x_2 \notin I_p \), according to the condition, there exists a transition \( t_3 = (r_5, r_3, x_3/y_3) \), such that \( t_3 t_2 \) is synchronizable and \( y_3 | p \neq - \). Suppose instead that \( t_2 \notin T_p^f \). Since \( M \) is synchronizable and there exists a transition \( t_3 = (r_5, r_3, x_3/y_3) \) such that \( t_3 t_2 \) is synchronizable. As \( t_2 \notin T_p^f \), we know that \( y_3 | p \neq - \). In each case, since \( t_0 \) has no absolute leading path, \( x_2 \notin I_p \) and so by considering \( t_3 \sigma \) we contradict the maximality of \( \sigma \) as required.

\((\Rightarrow)\) Consider a transition \( t = (r_1, r_2, x/y) \in T_p^f \) where \( x \notin I_p \), \( y | p \neq - \). Let \( \sigma \) denote an absolute leading path for \( t \). Since \( x \notin I_p \), \( \sigma \notin \epsilon \). By definition, the last transition of \( \sigma \) must have non-empty output at \( p \) and must be synchronizable with \( t \) and so the result follows. \( \square \)

We now consider the problem of checking the output of transition \( t \) at \( p \) where \( t | p = - \). We prove that if we can verify the output of every transition at \( p \) such that \( t | p \neq - \) then we can verify the output of every transition at \( p \).

**Definition 3.** Let \( R \) be a set of transitions in \( M \). The synchronizable path \( \rho \) is an absolute verifying path for \((t, p)\) upon \( R \) if we know that the output of \( t \) at \( p \) must be correct whenever the following hold:

1. The output at \( p \) of every transition in \( R \) is correct in the SUT \( N \); and
2. There exists a synchronizable path \( \rho' \rho'' \) in \( M \) that starts at \( s_0 \) such that the tester at \( p \) sees the expected sequence of observations when the input portion of \( \rho' \rho'' \) is applied to \( N \).

This says that if we have an absolute verifying path \( \rho \) for \((t, p)\) upon \( R \) and we know that the transitions in \( R \) are correct then we can use any synchronizable path that contains \( \rho \) in order to check the output of \( t \) at \( p \). The following shows that if we can produce absolute verifying paths for each \( t \in T_p^f \) then we can also check the output at \( p \) of any \( t \notin T_p^f \).

**Theorem 2.** Given any FSM \( M \) that is intrinsically synchronizable and port \( p \), every transition \( t \notin T_p^f \) has an absolute verifying path upon \( T_p^f \).

**Proof.** Consider transition \( t \) with empty output at \( p \). Find a synchronizable path \( \rho = \rho_1 t_m = t_1 \ldots t_m \) in \( M \) such that \( t = t_j \) for some \( j \in [1, m - 1] \) and both \( t_1 \) and \( t_m \) have input at \( p \). The existence of such a path is guaranteed since \( M \) is intrinsically synchronizable. Suppose \( t_1 \) and \( t_m \) have input at \( p \), if we embed \( \rho \) within a path \( \rho' \rho'' \) we can determine, from the observations at \( p \), the output produced at \( p \) in response to the input portion of \( \rho_1 \). If the output of \( t' \) at \( p \) is correct for all \( t' \in T_p^f \), then when the input portion of \( \rho \) is applied we know that the correct output is produced by every transition \( t' \in T_p^f \) from \( \rho_1 \). Thus, if the expected number of outputs are observed at \( p \) when the input portion of \( \rho \) is applied then the output of \( t \) at \( p \) must be empty and so is correct. Thus \( \rho \) is an absolutely verifying path for \((t, p)\) upon \( T_p^f \). \( \square \)

This allows us to use weaker hypotheses than in [3]: the result in [3] included conditions that deal with transitions in \( T_p \setminus T_p^f \). In addition, [3] does not consider the controllability problem and considered only \( n \)-shift output faults.

The second goal concerns the problem of verifying the outputs of those transitions that could be involved in a potentially undetectable output shift fault in a path \( \rho \) plus the first and last transitions. When the input portion of \( \rho \) is a test/checking sequence, we assume that \( \rho \) contains every transition of \( M \). We prove that the conditions given above cannot be weakened when \( \rho \) contains every transition of \( M \). Observe that this problem was not considered in [3].

For simplicity, we consider below only potentially undetectable \( n \)-shift output faults in \( \rho \). Let \( T_{\rho, p} \) denote the set of transitions involved in potentially undetectable \( n \)-shift output fault at \( p \) in \( \rho \): \( t \in T_{\rho, p} \) if there exists a transition \( t' \) such that \( t' \) or \( t' \) is a synchronizable path in \( \rho \) in which there is a potentially undetectable \( n \)-shift output fault at \( p \). We prove that the above conditions cannot be weakened even to determine if \((t, p)\) is verifiable for any \( p \) and any \( t \) such that \( t \) is the first or last transition in \( \rho \) or \( t \in T_{\rho, p} \). Apparently, this implies that we cannot weaken the conditions to guarantee that for
any $p$ and any $t$ such that $t$ is the first or last transition in $\rho$ or $t$ is involved in a potentially undetectable output shift fault in $\rho$, $(t, p)$ is verifiable. Again, we first consider pairs $(t, p)$ such that $t_1|p \neq -$. Below, $T'_{\rho, p} = T_{\rho, p} \cap \{ t | t_1|p \neq - \}$ denotes the set of transitions that are involved in potentially undetectable 1-shift output faults at $p$ in $\rho$ and have non-empty output at $p$.

**Lemma 1.** Given an FSM $M$ and a port $p$, let $t_1t_2$ be a synchronizable transition sequence such that $t_1|p \neq -$ and $t_2|p \neq -$. Then

- $(t_1, p)$ has an absolute leading path $\Rightarrow (t_2, p)$ has an absolute leading path.
- $(t_2, p)$ has an absolute trailing path $\Rightarrow (t_1, p)$ has an absolute trailing path.

**Proof.** We prove the first part (the proof of the second part is similar). If the input of $t_2$ is at $p$, then $\epsilon$ is a leading path of $(t_2, p)$. If the input of $t_2$ is not at $p$, since the outputs of $t_1$ and $t_2$ at $p$ are non-empty, $p$ is an absolute leading path of $(t_1, p)$ implies $\rho t_1$ is an absolute leading path of $(t_2, p)$. □

**Theorem 3.** Given FSM $M$, port $p$, and a synchronizable path $\rho = t_1 \ldots t_m$ whose input portion is a test or checking sequence, if for every $t' \in T'_{\rho, p} \cup \{ t_1, t_m \}$ there is an absolute verifying path of $(t', p)$, then there is an absolute verifying path of $(t, p)$ for every $t \in T'_p$.

**Proof.** Consider the leading path only (the part for trailing paths is similar). Let $t^*$ be any transition in $T'_p - T'_{\rho, p} \cup \{ t_1, t_m \}$. We prove there is an absolute verifying path of $(t^*, p)$. As we assume any path whose input portion is a test/checking sequence contains all transitions in $M$, $t^*$ is a transition in $\rho$. Let $\rho' = t_1' \ldots t_k'$ ($k \geq 2$) be a subsequence of $\rho$ such that $t_k' = t^*$; $t_1' = t_1$ or $t_1'$ has empty output at $p$; and $\forall i \in [2, k - 1]$, $t_i'$ has non-empty output at $p$. Since $t^* \notin T'_{\rho, p} \cup \{ t_1, t_m \}$ we know there is such a subsequence with $k \geq 2$.

If $t_1'$ has empty output at $p$, since $t_2'$ has non-empty output at $p$, we know that $t_2'$ has an absolute leading path. This is because if the input of $t_2'$ is at $p$ then $\epsilon$ can be used as an absolute leading path; if the input of $t_2'$ is not at $p$, then $t_2' \in T'_{\rho, p}$, so according to the condition $t_2'$ has an absolute leading path $\rho'$. If $t_1'$ has non-empty output at $p$, then $t_1' = t_1$ and so $t_1'$ has an absolute leading path. Since both $t_1'$ and $t_2'$ have non-empty output at $p$, by Lemma 1, $t_2'$ has an absolute leading path $\rho'$. Thus, in both cases, $t_2'$ has an absolute leading path $\rho'$. Clearly, $\rho' t_2' \ldots t_k'$ is an absolute leading path for $t^*$ as required. □

The proof of the following is equivalent to the proof of Theorem 2.

**Theorem 4.** Given FSM $M$ that is intrinsically synchronizable and port $p$, for any transition $t$ with empty output at $p$, $(t, p)$ has a verifying path upon $T'_{\rho, p}$.

Thus, there exist transition sequences to overcome all potentially undetectable 1-shift output faults in the path $\rho$ of a test/checking sequence, if and only if we can overcome all possible observability problems in $M$.

5. Conclusions

This paper investigated conditions that must be satisfied by a specification in order for us to be able to produce a test/checking sequence that is free from controllability and observability problems. This problem is represented in the following way. For each transition $t$ and port $p$ we wish to produce a path $\rho t p_2$ that checks the output of $t$ at $p$. The effectiveness of $\rho t p_2$, at checking the output of $t$ at $p$, must not be affected by controllability and observability problems. This paper gives conditions for the existence of such a path for each transition $t$ and port $p$ for a class of FSMs. This class of FSMs is strictly larger than that considered in [3] and the conditions produced are strictly weaker than those given in [3]. Interestingly, we also proved that these conditions are not weakened if we only wish to overcome potentially undetectable output shift faults in the path of a given test/checking sequence.

**References**