Analysis and Research on Distributed Network Protocol Testing
Controllability Problem

Xiao Wang\textsuperscript{1, a}, Jin Hua Zhang\textsuperscript{2, b}

\textsuperscript{1}Software College of Yunnan University, KunMing, 650091, China
\textsuperscript{2}Information College of Yunnan University, KunMing, 650091, China

\textsuperscript{a}email: 136868000@qq.com, \textsuperscript{b}email: jhzhang@ynu.edu.cn

Keywords: Conformance testing; Distributed test system; Multi port finite state machine; Controllability problems Introduction

Abstract. When conformance testing of distributed network protocol, there exit controllability problem and observability problem in NP-FSM model described distributed test system. Based on the detailed research of the distributed test system, this paper proposes a single input / single output, single input / multi output, multi input / single output and multi input / multi output 4 distributed testing model according to the characteristics of the input and output ports distributed test system, put forward to 4 kinds of algorithm to solve distributed test model controllability problems according to the characteristics of 4 kinds of distributed test model.

Overview

With the development and application of distributed network systems, the test of conformance of the distributed network protocol becomes a hot research topic.

Figure 1 shows the structure of a universal distributed test system for network protocols. In this structure, Implementation Under Test (IUT) has multiple ports and the test system consists of multiple testers with one for each port. Testers and IUT communicate with each other through ports and testers use external channels to exchange auxiliary information.

Fig.1. Distributed test system structure

Usually, a multi-port finite state machine with n ports (NP-FSM) is used for formalized description of the distributed test system as in Figure 1 and to generate the test sequence required by the test process. To ensure the normal test execution in the process of testing, the test sequence shall be input into all test ports of the test system in a specific chronological order and the output at the ports generated by the IUT shall be read in a specific chronological order. However, due to the concurrency of distributed systems and absence of global clock, there is no way to ensure that the test sequence can be input in the order as required. As a result, controllability problem between testers in IUT occur [2].Similarly, testers will experience observability problem when reading output of IUT [2]. Controllability is the capability of the test system to transmit input to the IUT in a given order. Observability is the capability of the test system to determine the relationship between input and output..

To solve controllability problem and observability problem in distributed test system, the most common solution is the addition of synchronous messages and observation messages to the test sequence. These messages enable the test system to input test sequence and read output in a given
Researchers who research controllability and observability issues in the distributed test system propose various specific solutions.

In reference 2, the test sequence is generated based on test methods, for example, method T and method U. Then, the possible controllability and observability problem in continuous transitions of the test sequence are considered. Finally, synchronous messages and observation messages are added to the test sequence.

In references 3 to 10, during the generation of the test sequence, the addition of synchronous messages and observation messages is considered to generate the optimum test sequence.

In references 11 and 12, the binding time method is used to address controllability and observability issues.

Based on researches by predecessors, this paper further analyzes controllability issues in distributed test systems. The paper concludes that, to effectively address controllability issues, different methods shall be used to generate synchronous messages for different types of distributed test systems.

In later articles, distributed test systems are divided into single input/single output systems and single input/multi-output systems to discuss different methods to generate synchronous messages.

### Analysis of Controllability PROBLEM in Distributed Test Systems

A multi-port FSM with n ports (NP-FSM) is defined as $M=(S, \Sigma, \Gamma, \delta, \lambda, s_0)$, where:

- $S$ is the finite set of states of $M$; $s_0 \in S$ is the initial state of $M$;
- $n$-tuple $\Sigma=(\Sigma_1, \Sigma_2, \cdots, \Sigma_n)$, where $\Sigma_k$ is the input character of port $k$; and $\Sigma_i \cap \Sigma_j = \emptyset$, $i \neq j$, $i,j=1,2,3,\ldots,n$;
- $I$ stands for the input character set: $I=\Sigma_1 \cup \Sigma_2 \cup \cdots \cup \Sigma_n$;
- $n$-tuple $\Gamma=(\Gamma_1, \Gamma_2, \Gamma_3, \cdots, \Gamma_n)$, where $\Gamma_k$ is the output character of port $k$; and $\Gamma_i \cap \Gamma_j = \emptyset$, $i \neq j$, $I$, $j=1,2,3,\ldots,n$;
- $O$ stands for the output character set: $O=(\Gamma_1 \cup \{-\}) \times (\Gamma_2 \cup \{-\}) \times \cdots \times (\Gamma_n \cup \{-\}),\{-\}$ indicates empty output;
- $\delta$ is the transition function $\delta : S \times I \rightarrow S$;
- $\lambda$ is Output function $\lambda : S \times I \rightarrow O$;

![Fig.2. A model of np-FSM with 2 ports](image)

Figure 2 gives an example of two-part FSM. States are represented by circles, and transitions are represented by arrows. Symbols above arrows are input and output corresponding to transitions. For example:

Transition $t_1$ indicates that if the current state of the system is $S_0$ and port 1 receives input $a$, the next state of the system is $S_1$, –(empty) and 1 are outputted at ports 1 and 2 respectively.

For NP-FSM in figure 2, $S=\{S_0, S_1, S_2\}$, $\Sigma_1=\{a\}$, $\Sigma_2=\{b\}$, $\Gamma_1=\{0\}$, $\Gamma_2=\{1,2\}$; $S_0$ is the initial state.

With NP-FSM in figure 2 we can generate the following global test sequence for the test system: $GTS=[t_1, t_3, t_5, t_1, t_2, t_6, t_4, t_5]$

Namely:

$$GTS=!a?-1!b?-1!a?-1!a?-1!a?-1!a?0,-!b?-2!a?0,-!a?-1$$

(1)

In express 1:

!a indicates to send the input a to port 1;
?{-,1} indicates that ports 1 and 2 receive output – and 1 respectively. Symbol – indicate a null output.

Mapping the global test sequence (1) column to ports 1 and 2 generates the local test sequences for ports 1 and 2:

Port 1: LTS1= !a !a !a?0!a?0!a

Port 2: LTS2=?!b?!b?!b?2?1

In the distributed test system, the two testers connected to ports 1 and 2 respectively execute LTS1 and LTS2. During the execution, if the two testers fail to cooperate, controllability problem may occur, thereby affecting the final test results.

In a test system, each tester shall send input to the IUT in a given order. If tester A does not send input or receive the output IUT in the previous transition and test A needs to send input in the current transition, test A cannot determine when to send the input to the IUT for the current transition. In this context, controllability problem occur, namely, input synchronization problem.

To be specific, in two continuous transitions ti=(si, sj, xi/yi) and tj=(sj, sk, xj/yj), if port p meets the following conditions:

\[ p=\text{port}(x_j) \text{ and } p \notin \text{port}(x_i) \cup \{\text{port}(y_i)\} \]

Where:
- port(x) indicates the port corresponding to input x;
- port(y) denotes the port set corresponding to the – components in output y.
- Controllability problem occur in ti and tj [3,4].

In the above GTS, controllability problem occur in continuous transitions [t3, t5].

If two continuous transitions do not experience controllability problem, they are self-synchronous continuous transitions. In the above GTS, [t1, t3] and [t5, t1] are self-synchronous continuous transitions.

The common method to address controllability problem is exchange synchronous messages between continuous transitions experiencing controllability problem.

For [t3, t5] with controllability problem, if tester 2 sends test 1 a control message C after receiving input 1 for transition t3 and tester 1 sends input a for transition t5 after receiving the control message C, testers send input to IUT in a given order.

Local test sequences, LTS1 and LTS2, are added control messages for continuous transition [t3, t5]:

\[ \text{LTS1}= !a \text{ ?C}2!a !a!a?0!a?0!a \]
\[ \text{LTS2}=?!b?!b?!b?!b?2?1 \]

Where, !Cj indicates control messages are sent to port j; ?Ci indicates reception of control messages from port i.

The above analysis of controllability problem is based on an ideal context where there is no transmission delay between testers and IUT. In practice, there are different delays between different testers and IUT.

In addition, the above analysis does not consider the number of ports executing input and output for each transition. Therefore, the above solution is not sound.

In the testing process in figure 3, theoretically there are no controllability problem for input xi and xi+1 for transitions ti and ti+1. As shown in (a), after port 2 receives output b for transition ti, input xi+1 for transition ti+1 can be input at port 2. However, errors in (b) may occur due to delay.
in the execution process, namely, port 1 does not receive output a for transition ti before port 2 executes input xi+1.

To solve the above shortage in the algorithm, the following proposes a sound solution by analyzing the number of ports executing input and output for each transition.

**Algorithm for Solution to Controllability Problem**

The section categorizes test systems into SISO system (single input/single output), SIMO system (single input/multiple output), MISO system (multiple input/single output) and MISO system (multiple input/multiple output). Different solutions are proposed for different system categories.

The following variables are defined for easy description:
- test(p): tester connected to port p;
- ip(ti): one input port for transition ti;
- ips(ti): set of all input ports for transition ti, \( \{\text{ip}(t_i)\} \subseteq \text{ips}(t_i) \);
- op(ti): one output port for transition ti;
- ops(ti): set of all output ports for transition ti, \( \{\text{op}(t_i)\} \subseteq \text{ops}(t_i) \);

**Algorithm for Solution to Controllability Problem in SISO Test System**

SISO test system refers to the test system in which each execution of transition requires input at only one port and output is generated at only one port.

For such test systems, correct test needs to meet the following condition:

For two continuous transitions \( t_i \) and \( t_{i+1} \), the input for transition \( t_{i+1} \) shall occur after the output of transition \( t_i \).

Therefore, the design algorithm is as follows:

1. If \( \text{op}(t_i) = \text{ip}(t_{i+1}) \), namely, output port for transition \( t_i \) and input port for transition \( t_{i+1} \) use the same port, synchronization is not considered for these two transitions and synchronization control message C is not required.

   ![Figure 4(a)](image)

   Fig.4. SISI test system

   (a) No control message

   ![Figure 4(b)](image)

   (b) A control message

2. If \( \text{op}(t_i) \neq \text{ip}(t_{i+1}) \), namely, output port for transition \( t_i \) and input port for transition \( t_{i+1} \) use different ports, synchronization is considered for these two transitions and synchronization control message is required. After \( \text{op}(t_i) \) receives the output from transition \( t_i \), \( \text{test}(\text{op}(t_i)) \) sends synchronization control message C to \( \text{test}(\text{ip}(t_{i+1})) \). After receiving the synchronization control message C, \( \text{test}(\text{op}(t_i)) \) sends the output of transition \( t_{i+1} \) to \( \text{ip}(t_{i+1}) \).

See Figure 4(b). For two continuous transitions \( t_i \) and \( t_{i+1} \), if the input port for transition \( t_i \) is port 1 and input is \( x_i \), and its output port is port 2 and output is \( b \); the input port for transition \( t_{i+1} \) is port 2 and the input is \( x_{i+1} \), Synchronization control message is not required because output \( b \) and input \( x_{i+1} \) use the same port.
Algorithm for Solution to Controllability Problem in SIMO test System

SIMO test system refers to the test system in which each execution of transition requires input at only one port and output is generated at multiple ports.

For such test systems, correct test needs to meet the following condition:

For two continuous transitions $t_i$ and $t_{i+1}$, the input for transition $t_{i+1}$ shall occur after the all outputs of transition $t_i$.

Therefore, the design algorithm is as follows:

For transition $t_i$, output port set $A = \text{ops}(t_i)$, $(a = \text{op}(t_i)) \in A$ indicates any port in set $A$. For transition $t_{i+1}$, input port $b = \text{ip}(t_{i+1})$. If:

1. If $a = b$, namely, output port $a$ for transition $t_i$ and input port $b$ for transition $t_{i+1}$ use the same port, port $a$ does not need to send synchronization control message $C$ to port $b$ after it receives the output of transition $t_i$.

2. If $a \neq b$, namely, output port $a$ for transition $t_i$ and input port $b$ for transition $t_{i+1}$ use different ports, port $a$ needs to send synchronization control message $C$ to port $b$ after it receives the output of transition $t_i$.

3. After receiving the synchronization control messages $C$ of $a \neq b$ ports, port $b$ sends the input of transition $t_{i+1}$.

See Figure 5. It is a single input/3 output test system. For two continuous transitions $t_i$ and $t_{i+1}$, if the input port for transition $t_i$ is port 1 and input is $x_i$, and its output ports are ports 1, 2, and 3 and outputs are $a_1$, $a_2$, and $a_3$; the input port for transition $t_{i+1}$ is port 2 and the input is $x_{i+1}$, Synchronization control messages $C_{12}$ and $C_{32}$ are required because outputs $a_1$ and $a_3$ of transition $t_i$ and input $x_{i+1}$ use different ports. After test(2) receives synchronization control messages $C_{12}$ and $C_{32}$, $x_{i+1}$ is input at port 2.

Algorithm for Solution to Controllability Problem in MISO test System

MISO test system refers to the test system in which each execution of transition requires input at multiple ports and output is generated at only one port.

For such test systems, correct test needs to meet the following condition:

For two continuous transitions $t_i$ and $t_{i+1}$, the input for transition $t_{i+1}$ shall occur after the output of transition $t_i$.

Therefore, the design algorithm is as follows:

For transition $t_i$, output port set $a = \text{ops}(t_i)$, and for transition $t_{i+1}$, input port set $B = \text{ips}(t_{i+1})$. $(b = \text{ip}(t_{i+1})) \in B$ indicates any port in set $B$. If:

1. If $a = b$, namely, output port $a$ for transition $t_i$ and input port $b$ for transition $t_{i+1}$ use the same port, port $a$ does not need to send synchronization control message $C$ to port $b$ after it receives the output of transition $t_i$.

2. If $a \neq b$, namely, output port $a$ for transition $t_i$ and input port $b$ for transition $t_{i+1}$ use different ports, port $a$ needs to send synchronization control message $C$ to port $b$ after it receives the output of
transition $t_i$.

(3) After all $a \neq b$ ports in set B receive synchronization control message C successfully, they send input of transition $t_{i+1}$. After all $a = b$ ports in set B send synchronization control message C, they send input of transition $t_{i+1}$.

See Figure 6. 3-input/single output test system. For two continuous transitions $t_i$ and $t_{i+1}$, if the input ports for transition $t_i$ are ports 1, 2, and 3 and input are $x_{1i}$, $x_{2i}$, and $x_{3i}$, and its output port is port 2 and output is $a_2$, the input ports for transition $t_{i+1}$ are ports 2, 3, and 4 and the inputs are $x_{2i+1}$, $x_{3i+1}$, and $x_{4i+1}$. Synchronization control message $C_{23}$ is required because output $a_2$ and input $x_{3i+1}$ and $x_{4i+1}$ use different ports. After test(3) and receive test(4) synchronization control messages $C_{23}$ and $C_{24}$, $x_{3i+1}$ and $x_{4i+1}$ are input at ports 3 and 4. After test(2) sends synchronization control messages $C_{23}$ and $C_{24}$, $x_{2i+1}$ is input at port 2.

Algorithm for Solution to Controllability Problem in MIMO test System

MIMO test system refers to the test system in which each execution of transition requires input at multiple ports and output is generated at multiple ports.

For such test systems, correct test needs to meet the following condition:

For two continuous transitions $t_i$ and $t_{i+1}$, the input for transition $t_{i+1}$ shall occur after the output of transition $t_i$.

Therefore, the design algorithm is as follows:

For transition $t_i$, output port set $A = \text{ops}(t_i)$. A port in set A, for example port x, is used as the main output port. For set $A1 \cup \{x\} = A$, A1 indicates the set of all output ports of $t_i$ except for the main port in set A. For transition $t_{i+1}$, input port set $B = \text{ips}(t_{i+1})$. (b=ip(t_{i+1})) $\in$ B indicates any port in set B.

1. All ports in set A successfully receive output of transition $t_i$;
2. All ports in set A1 send control message $C_1$ to main port x;
3. The main port x successfully receives the control message $C_1$ sent by all ports in set A1;
4. If $x \neq b$, namely, main output port for transition $t_i$ and input port b for transition $t_{i+1}$ use different ports, port x sends synchronization control message $C_2$ to port b.
5. After all $b \neq x$ ports in set B receive synchronization control message $C_2$ successfully, they send input of transition $t_{i+1}$. After all $b = x$ ports in set B send synchronization control message $C_2$, they send input of transition $t_{i+1}$.

Fig.7. MIMO test system

See Figure 7. 3-input/3-output test system. Port 2 is used as the main port.

For two continuous transitions $t_i$ and $t_{i+1}$, if the input ports for transition $t_i$ are ports 1, 2, and 3 and inputs are $x_{1i}$, $x_{2i}$, and $x_{3i}$, and its output ports are ports 1, 2, and 4 and output are $a_{1i}$, $a_{2i}$, and $a_{4i}$, the input ports for transition $t_{i+1}$ are ports 2, 3, and 4 and the inputs are $x_{2i+1}$, $x_{3i+1}$, and $x_{4i+1}$.

Because output $a_{1i}$ and $a_{4i}$ correspond to ports 1 and 4 other than the main port 2, ports 1 and 4 send control messages $c_{112}$ and $c_{142}$ to the main port 2 after receiving $a_{1i}$ and $a_{4i}$. After receiving
control messages c1_{12} and c1_{42}, the main port 2 sends control messages c2_{23} and c2_{24} to input ports 3 and 4 of transition t_{i+1} respectively.

After input ports 3 and 4 of transition t_{i+1} receive the control messages, x3_{i+1} and x4_{i+1} are input at ports 3 and 4. After test(2) send C2_{23} and C2_{24}, x2_{i+1} is input at port 2.

Summary

This paper studies controllability problem in test systems for conformance testing of distributed network protocols. Based on researches by predecessors, this paper proposes four types of distributed test models, SISO, SIMO, MISO and MIMO. It then gives four algorithms for solutions to controllability problem in the four test system models. Future researches shall further verify and optimize the four algorithms.

Acknowledgement

In this paper, the research was sponsored by Information Security Professional Education System Constructions from YUNNAN University, and YUNNAN Education Department. Sponsor by Yunnan Education Department (NO.2012c107).

References


