동기화된 시험순서를 생성하기 위한 개선된 시험 생성 방법

정 윤 희† · 홍 범 기†

요 약

프로토콜 구현물들은 호환성 및 상호 연동성을 증가시키기 위하여 표준에 정의된 규격과 일치하는지를 시험하여야 한다. 일반적으로 프로토콜 구현물들은 일련의 입력을 적용하여 이로 부터 얻은 출력들이 규격에서 기대하는(expected) 출력들과 일치하는지를 비교하는 적합성 시험을 수행한다. 본 논문에서는 최소 길이의 동기화된 시험 순서를 생성하는 새로운 방법을 제안한다. 시험의 길이는 대칭테스트그래프(symmetric test graph)의 간선(edge)의 갯수와 같다는 사실로 부터, MUIO와 최단경로를 이용하는 기존의 방법과 FSM으로 부터 직접 대칭테스트그래프를 구할 수 있다는 사실로 부터 MUSS(Multiple Unique State Signature)를 이용하는 기존의 방법들을 수정하여 동기화된 시험 순서를 생성하는 새로운 방법을 제안한다. 제안한 방법들은 MUIO를 이용하는 기존의 방법 보다 7~29%와 7~42% 개선된 시험 순서를 구할 수 있다.

Techniques Using MUIO and Shortest Path(MUSP) and Multiple Unique State Signature(MUSS) for Synchronizable Test Sequence Generation

Yoon Hee Jung† · Beum Gee Hong†

ABSTRACT

A procedure presented in this paper generates test sequences to check the conformity of an implementation with a protocol specification, which is modeled as a deterministic finite state machine(FSM). We proposed a technique to determine a minimum-cost tour of the transition graph of the FSM. The technique using Multiple UIO and Shortest Path(MUSP) saves the cost 1~9% over MUIO and directly, derive a symmetric test graph from an FSM graph. From this fact, we proposed a technique using Multiple Unique State Signature(MUSS) to solve an open issue that the multiple UIO assignment may not minimize the length of the tour. In this paper, the proposed technique is, also, applied to generate a synchronizable test sequence. And the result shows that the technique using MUSP and MUSS saves the cost 7~29% and 7~42% over the previous approach using MUIO, respectively.

1. Introduction

Test sequence generation for conformance testing has been widely advocated for ensuring that protocol implementations are consistent with their specifications[3]. Several approaches have been developed for protocol conformance testing. The concept of state

† 정 회 원:한국전자통신연구원
논문접수:1996년 9월 17일, 심사완료:1997년 1월 7일

signature called the Unique Input/Output sequence is proposed in [2]. A UIO sequence verifies that the protocol implementation is in an expected state. The UIO sequence approach was combined with the Rural Chinese Postman(RCP) algorithm to provide a robust and compact test sequence. In [8] and [9], we propose two approaches for automatically generating conformance test sequences of communication protocols by means of the MUIO and Shortest Path(MUSP) and the Multiple Unique State Signature(MUSS). The result is that the techniques using MUSP and MUSS save the length of the test sequence 1~9% and 0.4~28% over the MUIO technique.

In this paper, we apply these techniques to generate a synchronizable test sequence and can reduce the length of the synchronizable test sequence required for protocol conformance testing. This paper is organized as follows: Section 2 reviews the RCP/MUIO, the MUSP and the MUSS technique. Section 3 presents these techniques are applied to generate a synchronizable test sequence. Section 4 describes the simulation results of these techniques. Finally, in Section 5, conclusions are given.

2. Test Sequence Generation Techniques

2.1 Preliminaries

A protocol can be specified as a deterministic finite state machine(FSM). An FSM represents a directed graph G=(V, E), where V is a finite, non-empty set of vertices and E is a set of edges. The vertices of G are the states of an FSM. Each edge is labeled as a/o, where a and o respectively, represent an input and an output operation and represented by a tuple (V_i, V_j, a/o). The cost associated with each edge is the time taken to realize the corresponding transition in the FSM.

There are three basic steps to test an edge of a given vertex. Suppose that an edge (V_i, V_j:a/o) is to be tested. First, the FSM has to be put into the state corresponding to vertex V_i. Then, the input operation a required for that edge is applied to the FSM and the output o is checked. The test fails if the output generated by the FSM does not conform to the specification. If the output is correct, the new state of the FSM is identified in the third step. The protocol conformance testing is reduced to traversing every edge of G, provided that the output message of the protocol contain a message that uniquely identifies each state.

Aho and Dahbura[2] propose a UIO sequence to verify that the protocol implementation is in a expected state. The technique based on RCP/UIO, as briefly outlined here.

1) Construct a test graph G' from a graph G, where
$G' \equiv \{V, E_C\}$,
$E_C = \{(V_i, V_k; a_l/a_m \cdot UIO_j) : (V_i, V_j; a_l/a_m) \in E,$ and $tail(UIO_j) = V_k\}$
2) Construct a symmetric test graph G* from the test graph G' by duplicating some edge of G, such that the total cost of edges in G* is minimum and the in-degree of each vertex $V_i \in G^*$ is equal to its out-degree.
3) Find an Euler tour of the resulting symmetric test graph G*.

In [4], the use of the MUIO sequences is proposed for the purpose of checking a new state. A key observation is that there may exist several minimum-length UIO sequences for a given state and a judicious choice of UIO sequences for each edge could reduce the length of the overall test sequence. So, it was proposed in [4] that assignment of UIO sequences to the edges of G is such that degree $\Delta(G') = \sum_{i=1}^{n} |d_{in}^{Ec}(V_i) - d_{out}^{Ec}(V_i)|$ is minimized, where $d_{in}^{Ec}(V_i)$ and $d_{out}^{Ec}(V_i)$, respectively, denote the in-degree and out-degree of vertex V_i in graph G'. And a minimum-cost maximum flow method is used.

2.2 Techniques Using MUSP and MUSS

This paper proposes the use of MUSP to minimize the total length of the test sequence. A key obser-

vation is as follows: The vertices V_K and V_l, respectively, need some outgoing edges and some incoming edges for producing a symmetric test graph from a test graph. If there is no edges $(V_k, V_l) \in E$, then G* is made by duplicated the edges constructed the shortest path from V_K to V_l.

Given G, let $G_N = \{V_N, E_N\}$ be a directed graph such that $V_N \equiv \{S, T\} \cup V_X \cup V_Y$, where $V_X = \{X_0, X_1, \cdots, X_{n-1}\}$ and $V_Y = \{Y_0, Y_1, \cdots, Y_{n-1}\}$ and $E_N \equiv E_S \cup E_T \cup E^{\wedge} \cup E_{SH}$, where $E_S = \{(S, X_i); X_i \in V_X\}$, $E_T = \{(Y_j, T); Y_j \in V_Y\}$, $E^{\wedge} = \{(X_i, Y_j)$: there exists $UIO_i^j\}$, where UIO_i^j is a minimum-length UIO sequence of vertex V_i and tail of last edge of the UIO sequence is V_j and $E_{SH} = \{(Y_i, Y_j); Y_i, Y_j \in V_Y\}$. Let each edge $(Y_j, T) \in E_T$ have cost zero and capability $d_{out}^E(V_j)$; let edge $(X_i, Y_j) \in E^{\wedge}$ have cost β_j and infinite capability; and let each edge $(Y_i, Y_j) \in E_{SH}$ have cost α_j and infinite capability, where β_j is the length of UIO_i^j and α_j is the length of the shortest path from V_i to V_j in G. A flow of F_N on G_N is a function satisfying the following conditions;

$$\text{For } X_i \in V_X,\ d_{in}^E(V_i) = \sum_{(X_i, Y_j) \in E^{\wedge}} F_N(X_i, Y_j) \quad (1)$$

$$\text{For } Y_j \in V_Y,\ F_N(Y_j, T) + \sum_i F_N(Y_j, Y_i) = \sum_k F_N(X_k, Y_j) + \sum_k F_N(Y_k, Y_j) \quad (2)$$

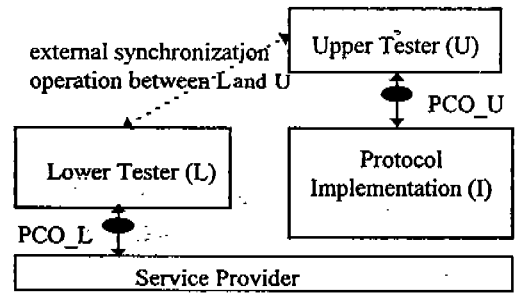
$$\text{For } Y_j \in V_Y,\ F_N(Y_j, T) = d_{out}^E(V_j) \quad (3)$$

$$\text{The cost of the flow FN is: } C(F_N) = \sum_i \sum_j \beta_j F_N(X_i, Y_j) + \sum_i \sum_j \alpha_j F_N(Y_i, Y_j)$$

Note that flow f_N on G_N, directly, derives a symmetric test graph from an FSM graph. The multiple UIO sequence assignment procedure only derive a test graph from an FSM graph. Several open issues remain open. One open issue is that one MUIO sequence assignment may lead to two edges of unit cost being replicated to form symmetric test graph while an "optimal" assignment may lead to the replication of a single edge of high cost[4]. This open issue can be solved if we use Multiple Unique State Signature(MUSS) instead of MUIO, where $MUSS = \{(US_0^0, \cdots, US_0^{n-1}\}, \cdots \{US_{n-1}^0, \cdots US_{n-1}^{n-1}\})$ and US_i^j represents minimum-length state signature among those with the tail state V_j for a given state V_i. And we do not use shortest paths any more.

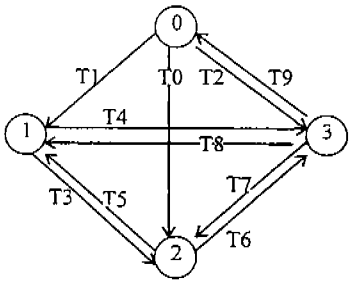
3. Synchronizable Test Sequence Generation Using MUSP

A common feature of various test architecture proposed for protocol conformance testing is that Implementation(I) is tested as a block box. One such architecture is shown in (Fig. 1).



(Fig. 1) A protocol testing system

During the application of a pre-determined test sequence, the synchronization between inputs from Upper Tester (U) and Lower Tester (L) becomes a problem. For example, L(or U) is expected to send an input to I after I sends an output to U(or L) but L(or U) is unable to determine whether I sent that output. Synchronization between U and L can be achieved by using external synchronization operations. ISO-9646 proposes two external synchronization operation LtoU and UtoL as L(U) informs U(L) that it is right time to send the next message[7]. A test sequence which contains a minimum number of external synchronization operation and no synchronization problem considered as the synchronizable test sequence.



T0:U2/LU0 T5:L2/U2
T1:L0/U2 T6:U2/L0
T2:L2/L1 T7:L1/LU2
T3:U0/LU2 T8:U2/L0
T4:L1/LU0 T9:U1/L1

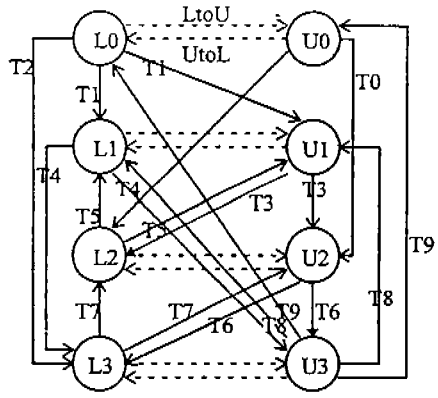
Remarks: U2/LU0: input 2 from U to I and output 0 from I to both U and L

(Fig. 2) The transition digraph G(V, E) of FSM.

[1] has been proposed a technique using duplexU digraph obtained from a duplexE digraph for FSM G. The procedure consists of four steps. Detail procedure is shown in [1].

1) A duplexE digraph is constructed from FSM. As an example, a duplexE digraph constructed from G(V, E) in (Fig. 2) is shown (Fig. 3).
2) A set of L- and U-synchronizable UIO sequences for each state of FSM is constructed. A set L- and U-synchronizable UIO sequences for each state of the FSM in (Fig. 2) is shown in ⟨Table 1⟩.
3) The duplexU digraph is constructed from the duplexE digraph.
4) Find an rural Chinese postman tour(RCPT) over the bold edges of the duplexU digraph.

In this paper, a minimum-cost maximum flow method is used to generate minimum-cost synchronizable test sequence. That is, this paper substitute a minimum-cost maximum flow method for duplexU



(Fig. 3) The duplexE digraph obtained from the G(V, E) in (Fig. 2).

⟨Table 1⟩ A set of synchronizable UIO sequences in (Fig. 2).

Beginning State	Synchronizable UIO Sequence	Ending State
0	L-UIO : T1	1
	L-UIO : T2	3
	L-UIO : LtoU, T0	2
	U-UIO : T0,	2
	U-UIO : UtoL, T1	1
	U-UIO : UtoL, T2	3
1	L-UIO : T4	3
	L-UIO : LtoU, T3	2
	U-UIO : T3	2
	U-UIO : UtoL, T4	3
2	L-UIO : T5	1
	U-UIO : UtoL, T5	1
3	L-UIO : T7,	2
	L-UIO : LtoU,T9	0
	U-UIO : T9,	0
	U-UIO : UtoL, T7	2

graph and automatically generating synchronizable test sequences. This technique use the MUSP.

Let a edge be L(U)-related edge if input and output of the edge are controlled and observed by only PCO_L(PCO_U). And let a edge be LU-related edge if the edge is neither L-related nor U-related edge.

Given G, let $G_L=\{V_L, E_L\}$ be a directed graph, such that $V_L=\{S, T\}\cup V_{X_s}\cup V_{Y_s}$, where $V_{X_s}=\{X_0^L, X_0^U, \cdots,$

X^L_{n-1}, X^U_{n-1}} and V_{Y_S}={Y^L_0, Y^U_0, ···, Y^L_{n-1}, Y^U_{n-1}}, and $E_L \equiv E_{S_S} \cup E_{T_S} \cup E^* \cup E_{SH_S}$, where E_{S_S}={$(S, X^{p_i}_i)$; $X^{p_i}_i \in V_{X_S}$}, E_{T_S}={$(Y^{p_i}_i, T)$; $Y^{p_i}_i \in V_{Y_S}$}, E_{SH_S}={$(Y^{p_i}_i, Y^{p_j}_j)$; $Y^{p_i}_i$, $Y^{p_j}_j \in V_{Y_S}$} and E* is made according to the following steps,

1) $E^* = \phi$

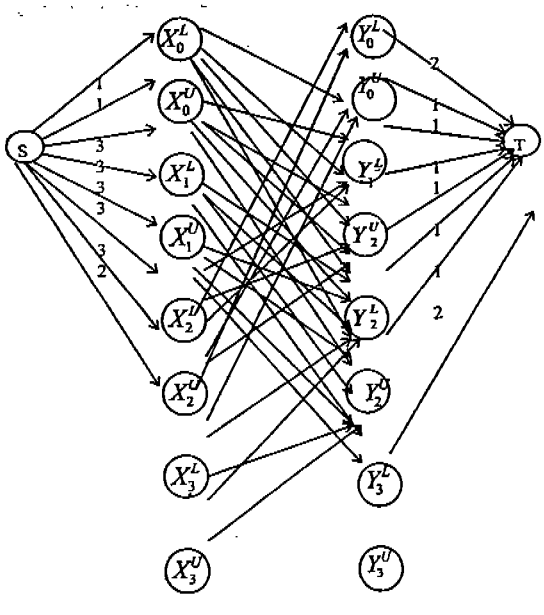
2) for each node $V_i \in V$, if V_i has a p_i-UIO(p_i is either U or L)

a) add $(X^{p_i}_i, X^L_k)$ to E^*, if last edge of the p_i-UIO is a L-related edge,

b) add $(X^{p_i}_i, X^U_k)$ to E^*, if last edge of the p_i-UIO is a U-related edge, and

c) add $(X^{p_i}_i, X^L_k)$ and $(X^{p_i}_i, X^U_k)$ to E^*, if last edge of the p_i-UIO is a LU-related edge.

Let each edge $(S, X^{p_i}_i) \in E_{S_S}$ have zero cost and capability $s(p_i)^E_{in}(V_i)$, where $s(p_i)^E_{in}(V_i)$ is the number of edges which is incoming edge of vertex V_i and p_i-related edge. Let each edge $(Y^{p_j}_j, T) \in E_{T_S}$ have zero cost and capability $t(p_j)^E_{out}(V_j)$, where $t(p_j)^E_{out}(V_j)$ is the number of edges which is outgoing edge of vertex V_j and the p_j tester send a input to I. Let each edge $(X^{p_i}_i, Y^{p_j}_j) \in E^*$ have cost δ_j and infinite capability, where δ_j is the length of synchronizable p_i-UIO(tail(p_i-UIO)=p_jj). Finally, let each edge $(Y^{p_i}_i, Y^{p_j}_j) \in E_{SH_S}$ have cost χ_j and infinite capability, where χ_j is the length of the shortest path from $V^{p_i}_i$ to $V^{p_j}_j$ in duplexE graph. The graph G_L for the FSM of (Fig. 2) is given (Fig. 4).



(Fig. 4) The graph GL for the FSM of (Fig. 2).

A flow of F_L on G_L is a function satisfying the following conditions;

$$\text{For } X^{p_i}_i \in V_{X_S}, \quad \sum_{(X^{p_i}_i, Y^{p_j}_j) \in E^*} F_L(X^{p_i}_i, Y^{p_j}_j) \le s(p_i)^E_{in}(V_i), \quad (4)$$

$$\text{For } X^U_i, X^L_i \in V_{X_S}, \quad \sum_{(X^L_i, Y^{p_j}_j) \in E^*} F_L(X^L_i, Y^{p_j}_j) \sum_{(X^U_i, Y^{p_j}_j) \in E^*} F_L(X^U_i, Y^{p_j}_j) = d^E_{in}(V_i), \quad (5)$$

$$\text{For } Y^{p_j}_j \in V_{Y_S}, \; F_L(Y^{p_j}_j, T) + \sum_{Y^{p_i}_i \in V_{Y_S}} F_L(Y^{p_j}_j, Y^{p_i}_i) = \sum_{X^{p_i}_i \in V_{X_S}} F_L(X^{p_i}_i, Y^{p_j}_j) + \sum_{Y^{p_i}_i \in V_{Y_S}} F_L(Y^{p_i}_i, Y^{p_j}_j) \quad (6)$$

$$\text{For } (Y^{p_j}_j, T) \in E_T, \quad F_L(Y_j, T) \le t(p_j)^E_{out}(V_j) \quad (7)$$

The cost of the flow F_L is:

$$C(F_L) = \sum_{(X^{p_i}_i, Y^{p_j}_j) \in E^*} \sum \delta_j F(X^{p_i}_i, Y^{p_j}_j) + \sum_{(Y^{p_i}_i, Y^{p_j}_j) \in E_{SH_S}} \sum \chi_j F(Y^{p_i}_i, Y^{p_j}_j)$$

Theorem 1:

If f_L is a minimum-cost maximum flow on G_L, then the corresponding assignment of UIO sequence to the edges of G is that assuming that we use the multiple L-or U-synchronizable UIO sequences as a state signature for each state, the number of augmented edges to make the symmetric test graph from a duplexE diagraph is minimized.

Proof: Since there exist edges of infinite capacity from each $X^{p_i}_i \in V_{X_S}$ to some $Y^{p_j}_j \in V_{Y_S}$, equation (5) and (6) guarantees that each edge in G is assigned a UIO sequence. And since there exist edges of infinite capacity from each $Y^{p_i}_i \in V_{Y_S}$ to each $Y^{p_j}_j \in V_{Y_S}$, $f_L(Y^{p_j}_j, T) = t(p_j)^E_{out}(V_j)$ for each $Y^{p_j}_j \in V_{Y_S}$. Note that $t(p_j)^{Ec}_{out}(V_j) = t(p_j)^E_{out}(V_j)$ and $t(p_j)^{Ec}_{out}(V_j) = \sum_{X^{p_i}_i \in V_{X_S}} F_L(X^{p_i}_i, Y^{p_j}_j)$. From the previous definition, the equation (6) can be rewritten as $t(p_j)^{Ec}_{out}(V_j) + \sum_{Y^{p_i}_i \in V_{Y_S}} F_L(Y^{p_j}_j, Y^{p_i}_i) = t(p_j)^{Ec}_{in}(V_j) +$

$\sum_{Y_j^{p_j} \in V_{Y_S}} F_L(Y_i^{p_i}, Y_j^{p_j})$. Since the cost of flow through edges in E_{SH_S} is positive, any maximum flow of minimum cost will minimize the flow if possible. The value of flow for edge in E_{SH_S} is as follows: If $t(p_j)_{out}^{E_C}(V_j) > \sum_{X_i^{p_i} \in V_{X_S}} F_L(X_i^{p_i}, Y_j^{p_j})$, then $F_L(Y_k^{p_k}, Y_j^{p_j}) > 0$, for any k and p_k, and $F_L(Y_j^{p_j}, Y_l^{p_l}) = 0$ for all l and p_l. Otherwise, $F_L(Y_k^{p_k}, Y_j^{p_j}) = 0$ for all k and p_k, and $F_L(Y_j^{p_j}, Y_l^{p_l}) > 0$ for any l and p_l. So, the value $\sum_{(Y_j^{p_j}, Y_i^{p_i}) \in E'} \chi_j F_L(Y_j^{p_j}, Y_i^{p_i})$ mean the total number of augmented edges to make the test-graph symmetric. The value $\sum_{(X_i^{p_i}, Y_j^{p_j}) \in E'} \sum \delta_j F(X_i^{p_i}, Y_j^{p_j})$ mean the total number of augmented edges to construct a test graph from a duplexE diagraph, assuming that we use the multiple L-or U-synchronizable UIO sequences as a state signature for each state. Therefore G_L is produced with minimum number of augmented edges to make the symmetric test graphs from a duplexE diagraph, assuming that we use the multiple L-or U-synchronizable UIO sequences as a state signature for each state.

A minimum-cost maximum flows on G_L are as follows: $f_L(X_0^U, Y_2^U) = 1$, $f_L(X_1^U, Y_3^L) = 1$, $f_L(X_1^U, Y_3^U) = 2$, $f_L(X_2^L, Y_1^U) = 1$, $f_L(X_2^L, Y_1^U) = 1$, $f_L(X_3^L, Y_0^U) = 1$, $f_L(X_3^U, Y_0^L) = 1$, and $f_L(Y_1^U, Y_2^L) = 1$, where, solution $f_L(X_0^U, Y_2^U) = 1$ means that a U-UIO of vertex U0 with tail(U-UIO) = U2 is assigned to an edge incoming to vertex U0 and $f_L(Y_1^U, Y_2^L) = 1$ means that a shortest path from vertex U1 to vertex L2 is needed to make test-graph symmetric. The resulting synchronizable test sequence is T0, T5, T4, T9, T1, T4, T7, T5, T3, T5, T3, T5, T4, T8, T4, T9, T0, T6, T9, and T2, LtoU, T9 with 1 external synchronizable operation. The underline presents a test segments.

Let we assign the cost of LtoU to 1, then the L or U-synchronizable UIO is changed as shown in 〈Table 2〉 and the resulting synchronizable test sequence is converted in a test sequence T0, T5, T3, T5, T4, T9, T1, T4, T7, T6, T9, T2, T7, T5, T4, T8, T4, T9, T0 and T6, T9 with zero additional external synchronization operation. It is reduced to the number of the external synchronization operation.

〈Table 2〉 A set of synchronizable UIO sequences in (Fig. 2).

Beginning State	Synchronizable UIO Sequence	Ending State
0	L-UIO : T1	1
	L-UIO : T2	3
	U-UIO : T0	2
1	L-UIO : T4	3
	U-UIO : T3	2
2	L-UIO : T5	1
	U-UIO : UtoL, T5	1
	U-UIO : T6,T7	2
	U-UIO : T6,T8	1
	U-UIO : T6,T9	0
3	L-UIO : T7	2
	U-UIO : T9	0

4. Simulation

This section describes some results of computer simulation. First, we generated 5 FSM's randomly, which each FSM has 5 nodes. And then we generated MUIO/MUSS sequences per state, and constructed the graph G_N/G_L. Finally we computed minimum-cost maximum flow on G_N/G_L using "LINDO(Linear, Interactive, Discrete Optimizer)" software, which sloves linear, integer, quadratic program. The procedure of above mentioned was performed on a number of FSM specifications and yielded favorable results, as shown in 〈Table 4〉. 〈Table 4〉 represents the total length of synchronizable test sequence and the number of external synchronization operation on the following case:

Case 1) we use the MUIO and assign the length of external synchronization operation to 0.

Case 2) we use the MUIO and assign the length of external synchronization operation to 1.

Case 3) we use the MUSP and assign the length of external synchronization operation to 0.

Case 4) we use the MUSP and assign the length of external synchronization operation to 1.

〈Table 3〉 The length of tours(the number of external synchronization operation) generated using the MUIO, the MUSP, the MUSS approach.

FSM	|V|	|E|	|I|	|O|	Case 1		Case 2		Case 3		Case 4		Case 5
					$\|E_C\|$	Total	$\|E_C\|$	Total	$\|E_C\|$	Total	$\|E_C\|$	Total	Total
FSM.1	5	11	3	3	21(7)	41(9)	18(4)	54(12)	16(3)	41(10)	15(1)	42(9)	38(7)
FSM.2	5	14	3	3	22(9)	46(13)	21(6)	44(3)	19(4)	38(5)	18(3)	37(4)	36(5)
FSM.3	5	13	3	3	23(8)	37(9)	16(3)	30(3)	17(3)	32(3)	14(1)	28(1)	28(1)
FSM.4	5	11	3	3	22(6)	41(10	21(2)	40(7)	20(4)	41(7)	17(1)	33(3)	33(1)
FSM.5	5	9	3	3	16(7)	36(11)	15(4)	30(5)	12(1)	26(1)	12(1)	24(1)	24(1)

Case 5)we use the MUSS and assign the length of external synchronization operation to 1.

Total length include the number of external synchronization operation. 〈Table 4〉 indicates that the number of external synchronization operation of the Case 2 and 4 are less than the Case 1and 3, the total length of the test sequence can be reduced , if we use the technique using MUSS instead of MUSP or MUIO.

5. Conclusions

A procedure presented in this paper generates test sequences to check the conformity of an implementation with a protocol specification, which is modeled as a deterministic FSM. This paper introduce a technique using MUIO and Shortest Path to determine minimum-cost tour of the transition graph of the FSM. The technique using MUSP saves the cost 1~9% over MUIO and directly, derive a symmetric test graph from an FSM graph. From this fact, we propose a technique using MUSS to solve an open issue that the multiple UIO assignment may not minimize the length of the tour. In this paper, the proposed technique is, also, applied to generate a synchronizable test sequence. We know that if we assign the length of the external synchronization operation to 1 instead of 0, we can reduce the number of external synchronization operation included in the synchronizable-test sequence. And applying the technique using MUSS instead of using MUSP, the result shows that the technique using MUSP and MUSS saves the cost 7~29% and 7~42% over the previous approach using MUIO, respectively.

References

[1] W.H. Chen and H.Ural, "Synchronizable Test Sequences Based on Multiple UIO Sequence", IEEE/ACM Transaction On Networking, vol. 3, no. 2, pp. 152-157, 1995.

[2] A.Aho, A. T. Dahbura, D. Lee, and U. Uyar, "An Optimization Technique for Protocol Conformance Test Generation Based on UIO sequence and Rural Postman Tour", IEEE Transaction on Communications, vol. 39, no. 11, pp. 1604-1615, 1991.

[3] K. K. Sabnani, and A. T. Dahbura, "A Protocol Test Generation Procedure", Computer Networks and ISDN Systems, vol. 15, no. 4, pp. 285-297, 1988.

[4] Y. N. Shen, F. Lombardi, and A. T. Dahbura, "Protocol Conformance Testing Using Multiple UIO Sequence", IEEE Transaction on Communications, vol. 8, pp. 1282-1287, 1992.

[5] M. U. Uyar, and A. T. Dahbura, "Optimal Test Sequence Generation Protocols:The Chinese Postman Algorithm Applied to Q.931", Proceedings of the IEEE Global Communication Conference, vol.

1, pp. 68-72, 1986.

[6] D. Sidhu and T. K. Leung, "Formal methods for protocol testing: A detailed study", IEEE Transaction Software Engineering, vol. 14, no. 4, pp. 413-426, 1989.

[7] OSI Conformance Testing Methodology and Framework-Part 1-5

[8] Y. H. Jung, and B. K. Hong, "Test Sequence Generation Using MUIO and Shortest Path", The Journal of the Korean Institute of Communication Science, vol. 21, no. 5, pp. 1193-1200., 1996.

[9] Y. H. Jung, B. K. Hong, and J. W. Lee, "Test Sequence Generation Using Multiple State Signature", submitted for publication, The Journal of the Korean Institute of Communication Science.

정 윤 희

1990년 한양대학교 산업공학과 졸업(학사)
1992년 한국과학기술원 산업공학과 졸업(석사)
1992년~현재 한국전자통신연구원 연구원
관심분야: 프로토콜 적합성 시험

홍 범 기

1982년 홍익대학교 전자계산학과 졸업(학사)
1984년 홍익대학교 전자계산학과 졸업(석사)
1997년 충북대학교 컴퓨터공학과(공학박사)
1984년~현재 한국전자통신연구원 책임연구원
관심분야: 초고속정보통신기반, 멀티미디어 서비스, 상호연동성시험