

SDL을 사용한 INAP 프로토콜 시험

도현숙[†] · 조준모^{**} · 김성운^{***}

요 약

본 논문은 형식기술기법으로 기술된 차세대지능망 응용프로토콜 (INAP: Intelligent Network Application Protocol)로부터 추상시험스위트 (Abstract Test Suite) 자동생성에 대한 연구 결과를 기술한다. 시험 자동생성을 위해 RCP tour (Rural Chinese Postman tour) 개념과 UIO시퀀스 (UIO: Unique Input Output Sequence) 개념을 응용하였다. SDL (System Description Language) 형식기술기법으로 명세화된 INAP명세로부터 생성한 I/O FSM (Input/Output Finite State Machine)을 중간모델로 하여 UIO 시퀀스를 정의하고, 정의된 UIO 시퀀스를 RCP tour개념과 결합하여 최적의 시험 시퀀스를 생성하였다. 본 논문에서는 또, 생성된 시험스위트의 오류커버영역 예측방법을 제시하고, 마지막으로 표준화된 시험 표기법인 TTCN (Tree and Tabular Combined Notation)으로 생성된 시험스위트의 변환방법도 제시한다.

Protocol Conformance Testing of INAP Protocol in SDL

Hyun Suk Do[†], Jun Mo Jo^{**} and Sung Un Kim^{***}

ABSTRACT

This paper describes a research result on automatic generation of Abstract Test Suite from INAP protocol in formal specifications by applying many existing related algorithms such as Rural Chinese Postman Tour and UIO sequence concepts. We use the I/O FSM generated from SDL specifications and a characterizing sequence, called UIO sequence, is defined for the I/O FSM. The UIO sequence is combined with the concept of Rural Chinese Postman tour to obtain an optimal test sequence. It also proposes an estimation methodology of the fault coverage for the Test Suite obtained by our method and their translation into the standardized test notation TTCN.

1. 서 론

최근 대부분의 프로토콜 명세들이 국제 표준화 기구들에 의해 표준화되고 있는 디지털 통신분야 프로토콜 측면에서, 구현이 표준 또는 명세에 얼마나 적합하게 구현되었는가를 시험하는 적합성 시험이, 소프트웨어 공학 분야의 중요한 이슈로 떠올랐다[1]. 프로토콜 표준으로부터 관련 제품 구현에 있어, 적합성 시험은 구현 제품의 프로토콜 표준에 대한 구현의 올바름을 검증하는 시험으로 프로토콜 표준으로부터 시험계열 (Test case)을 생성하여 구현에 적용함으

로써 적합성 여부를 판단하는 일련의 과정이다 [2].

적합성 시험 자체에 대한 표준화를 위해 국제표준화 기구인 ITU (International Telecommunication Union)와 ISO (International Standard Organization)에서는 각각 X.290 시리즈[3] 또는 IS9646[1]형태로 자연어로 기술된 프로토콜명세와 구현간 적합여부 판단을 위한 적합성시험체계 CTMF (Conformance Testing Methodology and Framework)를 표준화하여 적용하고있다. 그러나 자연어에 기초한 명세 및 이러한 명세로부터 시험 생성은 자연어 명세의 표현에 관련된 단점들 즉, 애매모호함 (ambiguity), 일관성결여 (inconsistency), 불완전성 (incompleteness) 등으로 인해 형식기술기법의 사용이 활발해지고, 이러한 형식기술기법에 의한 명세로부터 출발한 적합

[†] 한국전자통신연구원 선임연구원

^{**} 동명대학 사무자동화과 전임강사

^{***} 부경대학교 정보통신공학과 조교수

성시험체계 표준화가 FMCT (Formal Methods on Conformance Testing)로 완성되었다[4].

본 논문에서는 차세대 지능망 응용 프로토콜 중 하나인 SRF를 형식기술기법중의 하나인 SDL[5]로 모델링하여, 이 모델에 최적화 기술을 적용하여 추상시험스위트의 자동 생성과 얻어진 시험 계열의 성능을 예측하고, 또 표준화된 TTCN표기법에 적용하는 방법에 대해 기술한다.

이를 위해서 2장에서는 적합성 시험에 대해 기술하고, 3장에서는 차세대지능망 응용 프로토콜 중 하나인 SRF의 구조 및 응용실체절차에 대해 기술한다. 4장에서는 SRF를 SDL로 명세화하여 최적화 기술에 의해 시험 계열을 생성하는 방법에 대해 기술하며, 5장은 4장의 알고리즘을 통해 얻어진 시험 계열의 오류 커버 영역을 검정하는 알고리즘에 대해 기술한다. 6장은 추상시험스위트의 명세를 표준화된 방법으로 기술하는 TTCN을 기술하고, 7장은 최적화 기술에 의해 자동적으로 생성된 시험 계열을 TTCN표현에 적용하는 방법론에 대해 기술한다. 8장은 7장에서 기술된 TTCN변환 방법론의 적용에 대해 기술하고, 마지막으로 9장은 본 연구의 결론과 향후 연구 추진 사항에 대해 정리한다.

2. 적합성 시험

프로토콜 적합성 시험의 목적은 어떤 프로토콜의 구현 I (Implementation)가 원래 명세 (혹은 표준) S (Specification)에 합당하게 구현되었는지를 시험하는 것으로 통신 프로토콜 제품 구현 과정에서 중요한 역할을 한다. 일반적으로 적합성시험의 정의는 주어진 명세 S를 기초로 하여 생성된 시험 계열로서 구현 I가 명세 S에 대해 프로토콜의 행위 (Behavior)와 능력 (Capacity)이 일치하는지를 시험하는 것이다[2].

시험계열생성을 위한 일반적인 많은 방법들은 명세 S가 I/O FSM형태로 표현된 프로토콜로부터 출발하는데, 일반적으로 I/O FSM은 통신프로토콜의 행위제어 부분을 모델화하는데 많이 사용되어져 왔다. I/O FSM의 정의는 다음과 같다[6].

2.1 I/O FSM

프로토콜 명세의 행위제어 부분을 나타내는 I/O FSM은 다음의 다섯 가지 요소로 구성된다.

$M = \langle S, s_0, I, O, t_r \rangle$ 여기서

- S : 한정된 스테이트들의 집합 (a set of finite states)
- s_0 : 초기 스테이트 (initial state)
- I : 한정된 입력 알파벳 (finite input alphabet)
- O : 한정된 출력 알파벳 (finite output alphabet)
- t_r : 천이함수 (transition function), $t_r \subseteq \{s-i/o \rightarrow s' | s, s' \in S \wedge i \in I \wedge o \in O\}$

다음의 4개 요소 (p, a, b, q)는 I/O FSM M의 천이 (transition)로 불려지는데 아래와 같이 정의된다.

$$(p, a, b, q) \in S \times I \times O \times t_r$$

위의 정의에서 각 천이는 하나의 입력과 하나의 출력에 의해 일어난다.

I/O FSM $M = \langle S, s_0, I, O, t_r \rangle$ 은 방향화된 그래프 (Oriented graph), $G=(V, E)$ 로 표기되어질 수 있는데 여기서

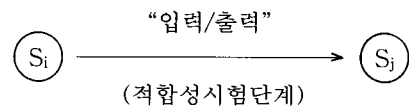
V = S의 스테이트에 해당하는 노드의 집합

E = 천이함수 t_r 에 해당하는 아크의 집합

일반적으로 I/O FSM으로 기술된 명세는 통신 프로토콜의 기능과 동작에 기반을 두고 만들어지게 된다.

2.2 시험의 정의

프로토콜의 명세를 나타내는 I/O FSM 모델이 다음과 같이 주어질 때 적합성 시험의 정의는 아래의 3단계로 이루어진다.



- (1) 명세 I/O FSM의 스테이트 S_i 에 해당하는 스테이트로 구현 I/O FSM을 위치시킨다.
- (2) 명세 I/O FSM에서 얻어진 시험 계열 "입력"을 구현 I/O FSM에 적용시킨 후 생성되는 "출력"을 판단한다.
- (3) 구현 I/O FSM에서 생성된 출력이 명세 I/O FSM에서 기술된 "출력"과 같은지를 확인하고 도착한 스테이트가 명세 I/O FSM의 것과 같은지를 검정한다.

출력 오류와 천이 오류를 걸러내기 위해, (1)에서 (3)의 적합성 시험 단계에서 일반적으로 두 가지 문

제가 야기된다. 단계 (1)에서 일반적으로 구현 I/O FSM을 원하는 스테이트로 위치시킬 수 없다는 것이고, 두 번째로 단계 (3)에서 시험 계열의 "입력"을 적용한 후에 원하는 "출력"을 구현 I/O FSM으로부터 얻은 경우라도 구현 I/O FSM이 해당 스테이트에 위치하였는지 알 수 없다는 것이다. 첫 번째 문제를 controllability limit이라 하고, 두 번째 문제를 observability limit이라 한다.

일반적으로 controllability limit 문제를 해결하기 위해서 초기 상태 s_0 로부터 시작하여 원하는 스테이트까지 가장 짧은 천이패스 (shortest transition path)를 이용하여 스테이트에 도착한 후 해당 천이를 시험하게 되고, observability limit 문제를 해결하기 위해 시험하는 천이 후에 도착한 스테이트의 유일한 입력/출력 시퀀스 (UIO sequence)를 시험 계열에 포함시켜 적용한 후 구현 I/O FSM의 결과 스테이트 확인하는 방법을 사용한다. 본 논문에서는 UIO 시퀀스를 사용하여 observability limit 문제를 해결하며, 또 controllability limit 문제는 RCP tour로 해결한다.

UIO 시퀀스를 사용하여 I/O FSM으로 표현된 프로토콜 명세로부터 적합성 시험 경우 생성은 명세에 나타나 있는 각 천이에 도착 스테이트의 UIO 시퀀스를 concatenation하여 생성하는데 다음과 같은 식 " $T_{ij}@UIO(S_j)$ "으로 나타낼 수 있다. 위 식에서 T_{ij} 는 시험되어야 할, 즉 명세 I/O FSM의 스테이트 S_i 에서 S_j 로 보내는 천이를 나타내고, $UIO(S_j)$ 는 도착 스테이트의 UIO 시퀀스를 나타낸다. 또 @는 concatenation 심볼이다. 여기서 UIO 시퀀스는 시험되는 천이에 의해 도착된 스테이트가 명세에서 원하는 올바른 S_j 인가를 시험하는데 사용된다.

UIO 시퀀스를 사용하여 I/O FSM 모델로부터의 시험 계열 자동 생성을 위한 여러가지 방법들이 제안되었다[7]. 본 논문에서는 구현 I/O FSM 내에 명세 I/O FSM에 명세된 각 천이가 존재하는 지를 시험하기 위해 명세 I/O FSM으로부터 단일 시험 계열을 생성하는 방법인 최적화 기술을 사용한다[7]. 3장에서는 차세대 지능망 응용 프로토콜 중 하나인 SRF의 구조 및 응용실체절차에 대해 기술한다.

3. SRF 구조 및 응용 실체 절차

INAP 프로토콜은 차세대 지능망의 물리실체간의

상호작용을 지원하기 위한 응용계층 프로토콜로 SSF (Service Switching Function), SCF (Service Control Function), SRF (Service Resource Function)으로 구성되고, 이를 지원하는 하부 프로토콜이 공통선 신호방식(CCS No.7)이다. 여기서 SRF의 구조는 FEAM (Functional Entity Access Manager), SRME (SRF Management Entity), SRSM (SRF Call State Model)로 구성되며 SRF FSM의 구조는 그림 1과 같다[8].

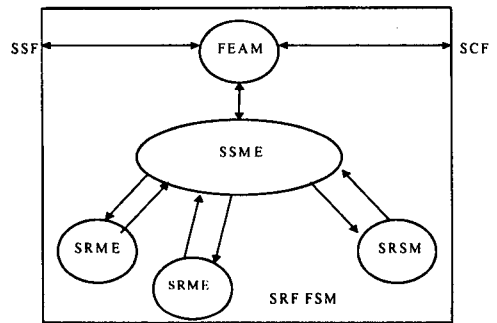


그림 1. SRF FSM 구조

FEAM은 하위레벨 인터페이스를 유지 보수하는 기능을 제공하고 SCF 및 SSF간 인터페이스 설정 및 유지와 SCF(Service Control Function) 및 SSF (Service Switching Function)로부터 수신한 메시지를 SRME로 전달하거나 큐에 저장하고, SRME로부터 수신한 메시지를 포매팅하여 SCF 및 SSF로 전송 및 큐에 저장한다.

SRME는 SCF로부터 수신한 오퍼레이션의 수행에 관련된 관리기능과 SCSM의 모든 인터페이스를 위해 SSF와 SCF와의 다이얼로그를 유지하고, SRSM의 생성, 개시, 유지보수의 기능과 수신한 입력 메시지를 해석하여 해당 SRSM사건으로 변환하며, SRSM의 출력 메시지를 다른 기능 실체에 해당하는 메시지로 변환한다.

SRSM은 수신 오퍼레이션에 오류가 있을 때 유지 보수기능에 알리고 SCF에게 보고하며 오류를 처리한다. 일련의 연속된 오퍼레이션을 조사하여 "안내 방송" 또는 "사용자 상호작용" 오퍼레이션에 대한 취소 오퍼레이션이 있을 시 즉시 수행을 하며 수신된 오퍼레이션은, 큐에 넣고 차례로 처리한다. 임의의 상태에서 SSF와 다이얼로그를 종료하면 SRSM은

그 다이얼로그에 할당된 자원을 해제한다. 지나치게 호가 중단되는 것을 막기 위해 타이머 T_{srf}를 사용한다. 4장에서는 SRF의 시험계열을 최적화 기술에 의해 생성하는 방법을 기술한다.

4. 시험 계열 생성

본 논문에서는 프로토콜 규격화 언어로 SDL을 선택해서 사용하였다. 이는 SDL을 이용한 명세화 및 행위 시뮬레이션 기능을 지원하는 도구들이 많이 상용화 되어 있기 때문이다.

그림 2, 그림 3 및 그림4는 프랑스 Verilog사의 Object-GEODE 도구를 사용한, 3장에서 설명한 SRF의 행위를 SDL로 system, block, process형태로 명

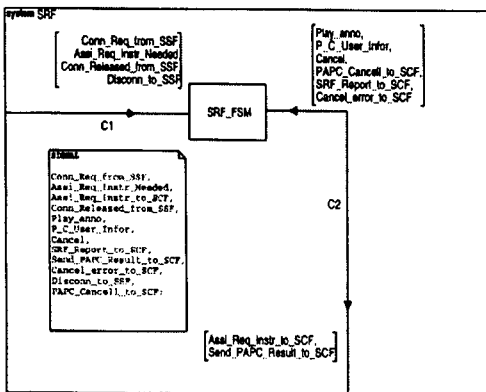


그림 2. SRF의 System

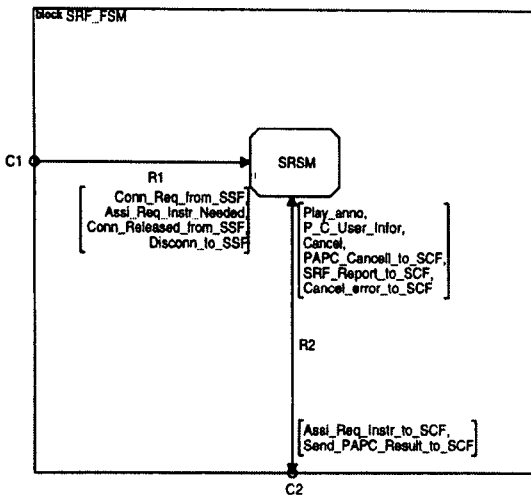


그림 3. SRF Block

세화 한 것이다[9].

아래 그림 5는 SRF에 대한 SDL명세로부터 Object-GEODE도구의 시뮬레이션 기능을 사용하여 생성된 I/O FSM이며, 각 이벤트의 입력/출력을 간략하게 표현하기 위해, 아래와 같이 알파벳 A에서 K까지 대체한다.

그림 6은 그림 5에 주어진 I/O FSM의 각 스테이트에 대한 UIO시퀀스이고, 각 스테이트는 S₀, S₁, S₂로 나타낸다[5][6].

최적화 기술에 의한 시험 계열 생성 방법에 기초한 기본 개념은 다음과 같다.

- (1) 명세 I/O FSM내에서 명세된 각 천이를 위해 시험 subsequence를 생성하는데, 각 시험 subsequence들은 TUT (Transition Under Test)의 입력 심볼과 이 천이에 의해 도착한 스테이트의 UIO시퀀스의 concatenation으로 구성된다.
- (2) (1)에서 구성된 모든 시험 subsequence들을 적어도 한번 방문하는 단일 최적 시험 계열을 Rural Chinese Postman (RCP) tour개념을 사용하여 생성한다.

위의 최적화 방법을 구현하기 위해 그림6에서 구해진 모든 스테이트들의 가장 짧은 길이의 UIO시퀀스를 결정한 후 주어진 I/O FSM내의 모든 천이에 대해 시험 subsequence를 다음과 같은 형태를 생성한다.

$$TSS_{ij} = TR_{ij} @ UIO(s_j)$$

위의 식에서 TSS_{ij}는 스테이트 i에서 스테이트 j로의 천이를 위한 시험 subsequence를 나타내고, TR_{ij}는 앞에서 언급한 TUT (Transition Under Test)이며, @는 concatenation심볼을 나타내고, UIO(s_j)는 TR_{ij}에 의해 도착하는 천이의 UIO시퀀스를 나타낸다.

모든 구해진 시험 subsequence를 원래의 I/O FSM G(V, E)에 새로 추가하면 새로운 그래프 G'=(V', E')가 생성되는데 G'는 다음과 같다.

$$G'=(V',E'), V'=V \text{ 와 } E'=E \cup E_c$$

여기서 E_c={ (v_i, v_k; e_iUIO(v_j)); (v_i, v_j) ∈ E } 그리고 tail (UIO(v_j)=v_k).

위의 식에서 tail(UIO(v_j))는 UIO(v_j)의 수행 후에 도착한 스테이트이고 E_c는 시험 subsequence TSS_{ij}

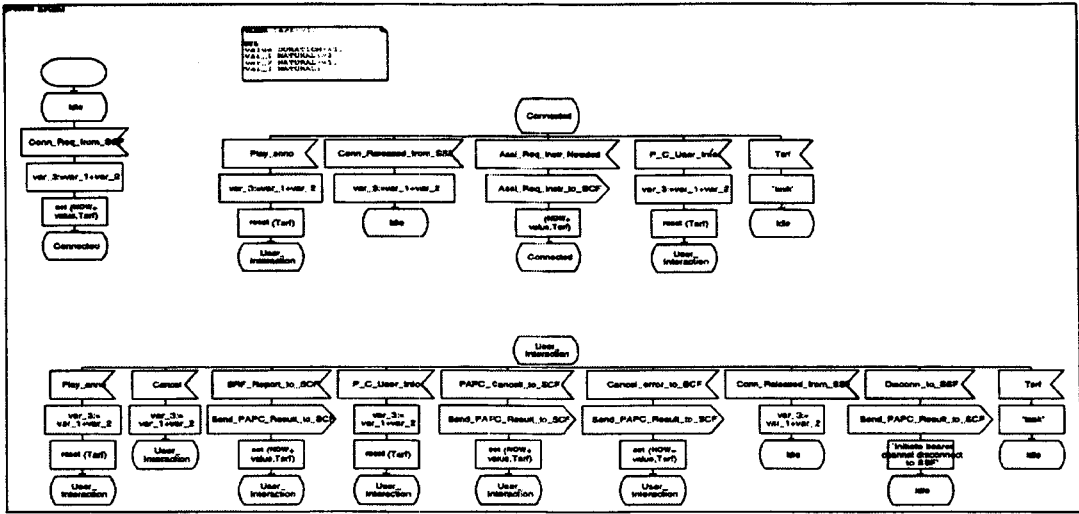
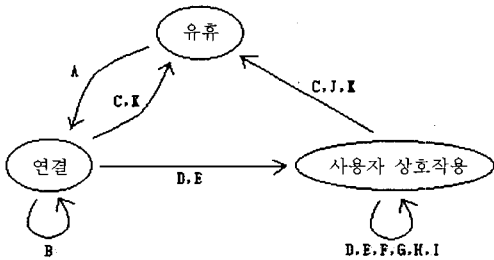


그림 4. SRF Process



- * Connect_Request_from_SSF/ε → A
- * Assist_Request_Instructions_needed/Assist_Request_Instruction_to_SCF → B
- * Connect_Released_from_SSF/ε → C
- * Play_Announcement/ε → D
- * Prompt_and_Collect_user_Information/ → E
- * Cancel/ε → F
- * SRF_Report_to_SCF/Send_PAPC_Result_to_SCF → G
- * PAPC_Cancelled_to_SCF/Send_PAPC_Result_to_SCF → H
- * Cancel_Error_to_SCF/Send_PAPC_Result_to_SCF → I
- * Disconnect_to_SSF/Send_PAPC_Result_to_SCF → J
- * Tsr/ε → K

그림 5. SRF의 I/O FSM

상태	UIO
유휴 (S ₀)	A
연결 (S ₁)	B
사용자상호작용 (S ₂)	J

그림 6. 그림 5의 I/O FSM에 대한 UIO 시퀀스

= TR_i@UIO(S_j)를 나타낸다. 그림 7은 위와 같은 방법으로 구해진 그래프 G'를 나타낸다.

그림 7의 새롭게 얻어진 그래프 G'=(V', E')에서 초기 스테이트에서 출발하여 모든 TSS_i들을 1번 포함하고 다시 초기 스테이트로 돌아오는 tour가 최적의 완전 시험 스위트에 해당한다. 이러한 tour를 Euler tour라고 이야기 하는데, Euler tour가 존재하는 조건은 얻어진 그래프 G'=(V', E')가 symmetric 해야 한다. 즉 모든 vertex에 들어오는 edge수와 나가는 edge수가 같아야 한다는 것이다. 이러한 과정이 symmetric augmentation 과정이다.

그래프 G'=(V', E')가 모든 시험 subsequence를 1번 포함하는 Euler tour를 계산하는데 있어 G'를 symmetric하게 하기 위해서는 E_c를 중심으로 "flow maximum cost minimum"[10] 개념으로 E_c는 한 번

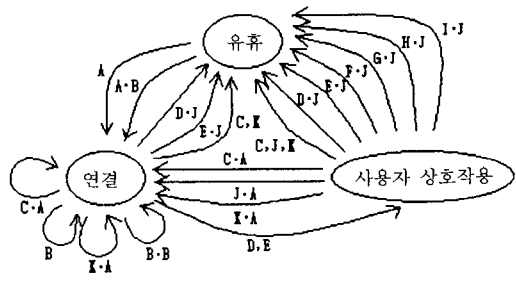


그림 7. Subsequence를 포함한 그래프 G'

포함하고 원래 기존의 천이 E를 부분적으로 포함하여 새로운 형태의 symmetric한 그래프 $G^*=[V^*, E^*]$ 를 구해야 한다. 이러한 개념으로 구해진 그래프 G^* 이 그림 8에 나타나 있다.

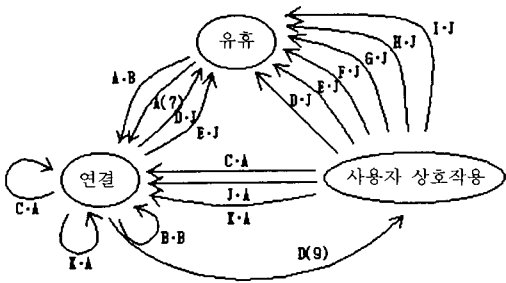


그림 8. Symmetric augmented 그래프 G^*

Symmetric한 그래프 $G^*=[V^*, E^*]$ 을 얻기 위해서 $G'=[V', E']$ 의 모든 E_c 는 한번만 포함하고, 기존의 E (유휴 상태에서 연결 상태로의 천이 A를 7번 반복, 연결 상태에서 사용자 상호 작용 상태로의 천이 D를 9번 반복)를 부분적으로 포함시켜 생성하는데 이렇게 하여 구해진 $G^*=[V^*, E^*]$ 에서의 RCP tour 계산은, 다음의 알고리즘으로 수행된다.

- 1) 포인터 ptr을 초기 스테이트에 두고, 각 스테이트에 관계된 하나의 벡터를 생성한다.
- 2) ptr에 의한 포인터 된 스테이트로 아직까지 탐색되지 않은 천이를 생성한 후, 이 천이의 시작 스테이트 S의 벡터에 ptr에 의해 포인터 된 스테이트를 저장한다.
- 3) 스테이트 S에 들어오는 탐색되지 않은 천이가 있으면 step 2)를 수행하고 ptr을 스테이트 S에 둔다. 그렇지 않으면 step 4)를 수행한다.
- 4) ptr에 의해 포인터 된 스테이트에 들어오는 천이에 의해 연결된 스테이트를 선택하고, 이들 중에서 이들로 들어오는 탐색되지 않은 천이를 가진 또 그들의 벡터값이 있는 스테이트 S가 존재하면 step 3)을 수행하고 그렇지 않을 경우에는 step 5)를 수행한다.
- 5) 초기 스테이트에 관계된 벡터의 마지막 요소로부터 각 천이를 탐색한다. 위에서 벡터의 요소의 값도 다음에 탐색되어질 스테이트를 나타내고 이와 같은 방법으로 모든 벡터들의 값이 탐

색될 때까지 계속한다.

이러한 방법으로 탐색된 천이의 연속이 Euler tour에 해당하며, 궁극적으로 RCP tour에 해당되고 최적의 완전 시험 스위트에 해당한다.

아래 표 1에 나타낸 시험 계열은 위에서 언급된 최적화 기술에 의해 생성된 완전 시험 스위트이다.

표 1. 생성한 완전시험스위트

A	D	J	A	B	B
(S ₀ →S ₁)	(S ₁ →S ₂)	UIO _{S2}	(S ₀ →S ₁)	UIO _{S1}	(S ₁ →S ₁)
B	K	A	C	A	E
UIO _{S1}	(S ₁ →S ₀)	UIO _{S0}	(S ₁ →S ₀)	UIO _{S1}	(S ₁ →S ₂)
J	A	D	C	A	D
UIO _{S3}	(S ₀ →S ₁)	(S ₁ →S ₂)	(S ₁ →S ₂)	UIO _{S0}	(S ₁ →S ₂)
J	A	D	K	A	D
(S ₂ →S ₀)	UIO _{S0}	(S ₁ →S ₂)	(S ₂ →S ₀)	UIO _{S0}	(S ₂ →S ₂)
J	A	D	E	J	A
UIO _{S2}	(S ₀ →S ₁)	(S ₁ →S ₂)	(S ₂ →S ₂)	UIO _{S2}	(S ₀ →S ₁)
D	F	J	A	D	G
(S ₁ →S ₂)	(S ₂ →S ₂)	UIO _{S2}	(S ₀ →S ₁)	(S ₁ →S ₂)	(S ₂ →S ₂)
J	A	D	H	J	A
UIO _{S2}	(S ₀ →S ₁)	(S ₁ →S ₂)	(S ₂ →S ₂)	UIO _{S2}	(S ₀ →S ₁)
D	I	J			
(S ₁ →S ₂)	(S ₂ →S ₂)	UIO _{S2}			

위의 표의 의미는 초기상태에서 시작하여 모든 천이를 한번씩 방문하고 다시 초기 상태로 돌아갈 RCP tour이다.

5. 오류 판단 예측 알고리즘

시험대상구현(IUT: Implementation Under Test)이 명세에 적합하게 구현되었는지 여부를 결정하는 시험 계열의 능력은 시험 계열이 찾아내는 구현에 대한 오류 판단의 영역에 의존한다. 생성된 시험 계열의 오류 판단 영역을 평가하기 위해서는 주어진 I/O FSM 명세와 같지는 않지만 주어진 시험 계열을 사용하는 유사한 I/O FSM의 집합을 우선 생성하여야 한다. 이상적으로 오류 판단 영역은 I/O FSM 명세로부터 시뮬레이션을 통하여 생성된 모든 가능한 유사한 I/O FSM에 얻어진 시험 계열을 적용하여, 시험에 통과한 I/O FMS 수와 트레이스 동치 관계에 있지 않은 I/OFSM 수의 비율이다.

그러나, 시험 계열의 오류판단 영역 예측은 시험해야될 유사한 I/O FSM의 수가 매우 많기 때문에 어려운 작업이다. 즉, 이러한 유사 I/O FSM구현들을 모

두 시험하는 것은 불가능하다. 대신에 I/O FSM명세와는 조금 다른 유사 I/O FSM의 집합을 샘플 한다. 이들 유사 I/O FSM은 I/O FSM명세에서 하나 또는 그 이상의 천이의 도착상태를 바꾸거나, 하나 또는 그 이상의 천이의 출력부분을 변환하여 생성된다. 위에서 정의된 유사 I/O FSM의 오류모델을 다음과 같이 분류된다.

- Class 1 : I/O FSM 명세에서 두 개의 random하게 선택된 천이의 도착상태를 변환하여 이 class의 유사 I/O FSM생성
- Class 2 : I/O FSM 명세에서 두 개의 random하게 선택된 천이의 출력부분을 변환하여 이 class의 유사 I/O FSM 생성
- Class 3 : 하나의 random하게 선택된 천이의 출력 부분과 I/O FSM명세에서의 다른 한 개의 random하게 선택된 천이의 도착 상태를 변환하여 이 class의 유사 I/O FSM 생성
- Class 4 : 두개의 random하게 선택된천이의 출력 부분과 도착 상태를 변환하여 이 class의 유사 I/O FSM 생성

위의 오류 모델 생성 과정에서, 새로운 값들은 고정성이 확인된 독립적인 유사 random sequence generation 알고리즘을 사용한다.

위에서 주어진 알고리즘은 시험 계열 시험을 통과한 유사 I/O FSM이 실제 명세 I/O FSM 명세에 트레이스 동치관계에 있는지를 결정하기 위해 쓰여진다. 즉 전체적인 오류판단 영역예측 알고리즘은 그림9와 같다.

다음의 오류판단영역 예측 알고리즘은 시험 계열을 적용하기 전에 UIO검정 (명세 I/O FSM의 각 상태들의 UIO시퀀스를 시뮬레이션된 유사 I/O FSM에서도 유일성이 존재하는지 검정)을 step 4에서 먼저 수행하고 나머지 부분은 위에서 언급한 내용과 같다.

위에 언급된 알고리즘을 사용한 시뮬레이션 결과로는, 최적화 시험 계열 생성 방법에 의해 얻어진 시험 계열은 도착 상태 및 출력 부분이 변환된 유사 I/O FSM 모델에서 하나 또는 그 이상의 오류를 UIO 검증에 의해 완벽하게 찾아내었다.

- 1) 명세 I/O FSM을 읽어 들인다.
- 2) 완전 시험 계열 생성기로부터 시험 계열을 생성한다.
- 3) 명세 I/O FSM으로부터 조금 다른 유사 I/O FSM을 위에 기술된 4가지 경우에 의해서 생성한다.
- 4) 각 UIO 시퀀스의 유일성은 step 3에서 생성된 각각의 유사 I/O FSM에서 검정하고, 각 UIO 시퀀스가 유일하다면, step 5를 수행한다.
- 5) step 3에서 만들어진 각각의 유사 I/O FSM들이 step 2에서 생성된 시험계열을 수용하는지 검정하기 위해 적용한다.
- 6) step 5의 시험을 통과한 유사 I/O FSM은 실제로 명세 I/O FSM과 트레이스 동치관계에 있는지 검정한다

그림 9. UIO에 의한 오류판단영역예측 알고리즘

시뮬레이션에서 얻어진 결과를 근거로 하여, UIO 검증을 통한 최적의 완전 시험 계열 생성 방법에 의한 시험 계열의 오류탐지능력은 IUT에서 도착상태이트 변환으로 생성된 유사 I/O FSM내의 하나 또는 그 이상의 오류를 완벽하게 탐지할 수 있다고 결론이 내려진다.

6. 표준 시험 표기법 TTCN

IS9646의 part3에서는 적합성 ATS시험스위트의 명세를 표준화된 방법으로 기술하는 TTCN에 대해 명시되어 있다. 이러한 TTCN 표기법의 목적은 ATS에 대한 전반적인 해석을 용이하게 하고, 시험기기에 의한 시험들의 자동적인 실행의 기초를 제공한다[1].

TTCN은 GR (graphical representation) 형태로 알려진 graphical form과 MP (machine-processable) form이 있다. 위 두 형태의 문법은 대부분 공통이며, GR형태는 table내에 구조를 포함하는 반면, MP형태는 기계가 읽을 수 있는 순차적인 열로 구성되어 있다. 대부분의 경우 GR형태를 더 많이 사용한다.

TTCN은 Type Definition table, Structured Type definition table, Timer Declarations table, PDU Type definition table, PCO Declaration table, 시험스위트 Constant Declaration table, 시험스위트 Operation Definition table, Dynamic Behavior table 등으로 구성되어 있다. 위에 언급된 모든 table중에서 ATS를 실제로 시험하기 위해 나타내는 가장 중요한 table인 Dynamic Behavior table에 대해 아래에 기술한다.

아래 그림 10은 Dynamic Behavior table의 보기이다. 이 요소에는 event tree들이 놓여지게 되는데 TTCN에서 보내는 event들은 느낌표(!) 다음에 놓여지게 되며, 받기를 기대하는 event들은 물음표(?) 다음에 놓여지게 된다. 즉 !X는 X라는 메시지를 보낸다는 것을 의미하고, ?Y는 Y라는 메시지를 받을 것을 기대한다는 의미이다. 또한 event가 일어난 PCO(Point of Control and Observation)를 나타낼 수 있는데, L!N_Connect_Request는 L로 정의된 PCO를 경유하여 N_Connect_Request를 보낸다는 것을 의미한다. 그림 10에서 Behavior Description 중 첫 항목인 여러event의 시간적 순서는 event를 indentation 함으로써 나타내는데, X다음에 Y, Z가 일어날 수 있고 Y가 일어날 경우, Test_body가 다음으로 일어날 수 있다는 것을 의미한다. 또한 다른 tree를 부를 수 있는데 이때는 +뒤에 명명된 subtree의 이름을 쓴다. 즉, 그림10에서 +Postamble의 의미는 이 자리에 Postamble에 대한event가 놓여지게 됨을 알 수 있다.

Test Case Dynamic Behavior			
Test Case Name : P30006			
Group : Transport_suite/Connection_Tests/Class_Negotiation			
Purpose : Verify that the IUT can accept a Transport Connection			
Comments :			
Label	Behavior Description	Constraints Reference	Verdict
	!X ?Y +Test_body +Postamble ?Z Test_body !A ?B ?C ?D Postamble !P !Q GOTO L !R ?S ?OTHERWISE		Fail Pass Fail Inconc Fail Inconc Fail
Extended Comments :			

그림 10. Dynamic behavior table

7. 변환 방법론

이 장에서는 간단한 예에 대해 최적화 방법을 사용하여 생성된 시험 계열을, 6장에서 기술된 TTCN표

현에 적용하는 방법론에 대해 기술한다.

최적화 기술에 의해서 생성된 연속된 시험 계열은 IUT를 시험하기 위한 시험 그룹에 해당하고, 시험을 위한 IUT가 이러한 시험 계열의 입력부분을 받아 들이고, 출력을 보내주면 표준에 적합한 구현으로 해석된다. 시험기는 IUT와 직접 통신하므로, 이 시험 계열들은 주어진 순서에 따라 연속적으로 시험기와 IUT 사이의 PCO에 전달된다.

본 연구에서 최적화 기술로 생성한 시험 스위트는 다음과 같은 모델로 나타낼 수 있다.

"a₁/b₁input₁/output₁@UIO₁a₂/b₂input₂/output₂@UIO₂"

이 모델에서는 두개의 시험 경우, 즉

a₁/b₁input₁/output₁@UIO₁과 a₂/b₂input₂/output₂@UIO₂로 나눌 수 있으며, input₁/output₁, input₂/output₂는 앞에서 언급한 TUT (Transition Under Test)로서 Test body이며, @는 concatenation심볼이고, UIO₁, UIO₂는 input₁/output₁, input₂/output₂에 의해 도착한 스테이트의 UIO시퀀스를 나타내며 postamble로 취급된다. a₁/b₁과 a₂/b₂는 TUT가 아닌 기존의 천이 중 RCP tour에서 최적 시험 계열 생성을 위해 반복적으로 사용되어진 천이이며, preamble로 취급되어진다.

변형된 TTCN은 Test case Dynamic Behavior table들과 preamble 또는 postamble을 나타내는 Test Step Dynamic Behavior table들로 구성되어진다. 위 모델 중에서 첫번째 시험경우 "a₁/b₁input₁/output₁@UIO₁"에서 test body(input₁/output₁)가 상태S1에서 상태 S2로의 천이를 나타내고, UIO₁이UIO_{Input}UIO_{Output}으로 되어있다면, 시험스텝과 시험경우는 아래와 같이 구성되어진다.

그림 11의 preamble은 TUT가 아닌 기존의 천이 중 RCP tour에서 반복적으로 사용되어진 천이이며,

Test Step Dynamic Behavior			
Test Step Name : S1_PREAMBLE			
Group :			
Purpose : transition of state1			
Comments :			
Label	Behavior Description	Constraints Reference	Verdict
	!a ₁ ?b ₁ ?OTHERWISE		Pass Fail
Extended Comments :			

그림 11. Preamble

시험기가 a₁ 을 IUT에 보내고, b₁을 받아 IUT의 상태를 상태 S1으로 옮기는 것이다.

그림 12의 postamble은 test body에 의해 도착한 상태 즉, 상태S2의 증명을 위해 사용되어진 상태 S2의 UIO시퀀스이다.

Test Step Dynamic Behavior			
Test Step Name : S2_POSTAMBLE			
Group :			
Purpose : UIO sequence of state2			
Comments :			
Label	Behavior Description	Constraints Reference	Verdict
	!UIO _{Input} ?UIO _{Output} ?OTHERWISE		Pass Fail
Extended Comments :			

그림 12. Postamble

그림 13의 Test case에서 Test Case Name에서 S1_S2_001은 최적화 기술에 의해 생성된 시험 계열 중에서 여러 번 반복될 수 있는 상태S1에서 S2로의 천이 가운데 첫번째 것을 의미하고, Behavior Description부분에서는 앞에서 설명한 preamble과 pos

Test Case Dynamic Behavior			
Test Case Name : S1_S2_001			
Group :			
Purpose :			
Comments :			
Label	Behavior Description	Constraints Reference	Verdict
	+S1_PREAMBLE LT_PCO! _{input1} LT_PCO? _{output1} +S2_POSTAMBLE		
Extended Comments :			

그림 13. Test case

tamble이 접속하게 된다. Test body부분인 LT_PCO!_{input1}와 LT_PCO?_{output1}는 상태 S1에서 상태 S2로의 천이를 나타내고, LT_PCO는 하위 시험기에서 IUT를 제어 관찰점을 의미한다.

8. TTCN 변환

이 장에서는 최적화 기술에 의해 생성된 차세대 지능망 응용 프로토콜 SRF의 시험 계열을 5장에서 설명한 TTCN변환 방법론을 적용하여 변환한다.

상태 S₀(유휴), S₁(연결), S₂(사용자 상호작용)의 UIO시퀀스, 즉 A, H, J는 각각 S₀_Postamble, S₁_Postamble, S₂_Postamble로 변환하며, RCP tour에서 반복적으로 사용되어진 기존의 천이 A, D는 각각 S₁_Preamble, S₂_Preamble로 변환한다.

그림 14, 그림 15는 각각 S1_PREAMBLE과 S2_POSTAMBLE을 나타내며, 그림 16은 S2_POSTAMBLE과 S1_PREAMBLE를 사용하여 4장에서 생

Test Step Dynamic Behavior			
Test Step Name : S1_PREAMBLE			
Group : SRF/SRF_SIG/PREAMBLE			
Purpose : Transition of state S1			
Default :			
Comments :			
Label	Behavior Description	Constraint Reference	Verdict
LB1	LT_PCO!Connect_Request_from_SSF LT_PCO?OTHERWISE GOTO LB1	SR(CO_V1!:'0'B, CO_V2:::'00'O)	(P)
Extended Comments :			

그림 14. S1_Preamble

Test Step Dynamic Behavior			
Test Step Name : S2_POSTAMBLE			
Group : SRF/SRF_SIG/POSTAMBLE			
Purpose : UIO sequence of state S2			
Default :			
Comments :			
Label	Behavior Description	Constraints Reference	Verdict
LB1	LT_PCO!Disconnect_to_SSF LT_PCO?Send_PAPC_Result_to_SCF LT_PCO?OTHERWISE GOTO LB1	SR(DS_P:'0'B,DS_S:'01'O)	(P)
Extended Comments :			

그림 15. S2_Postamble

Test Case Dynamic Behavior			
Test Case Name : S1_S2_001			
Group : SRF/SRF_SIG/			
Purpose : Test of Transition from S1 to S1			
Default :			
Comments :			
Label	Behavior Description	Constraints Reference	Verdict
	+ S1_PREAMBLE + S2_PREAMBLE + S2_POSTAMBLE		(P)
Extended Comments :			

그림 16. TUT를 포함한 Test case

성된 시험 계열 중 첫번째 (Connect_Request_from_SSF/ε · Play_Announcement/ε · Disconnect_to_SSF/Send_PAPC_to_SCF), 즉 연결상태에서 사용자 상호 작용 상태로의 천이 Play_Announcement/ε에 대한 시험 표기를 나타낸다. 또한, 나머지 시험 계열에 대해서는 동일한 방법을 적용한다.

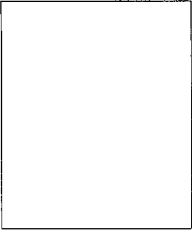
9. 결론 및 향후 연구 추진 사항

본 논문에서는 적합성 시험 정의 및 최적화 기술 기법에 의한 ATS생성 방법을 기술하고 생성된 시험 계열의 오류 커버 영역을 검정하는 알고리즘에 대해 기술하였다. 또, 제시된 TTCN변환 방법을 사용하여 차세대 응용 프로토콜 중 하나인 SRF을 위해 생성된 시험 계열에 적용하였다.

TTCN으로 표현된 ATS는 TTCN 컴파일러에 의해 수행 가능한 ETS (Executable Test Suite)로 변환하여, 지능망에 구현된 SRF 및 SSF (Service Switching function) 와 SCF (Service Control Function) 프로토콜들을 대상으로 한국형 INAP적합성 시험에 적용되었다. 향후 한국통신과 한국전자통신연구원을 중심으로 구현되고있는 차세대 지능망을 위한 CS-2 (Capability Set 2) 의 INAP에 본연구결과의 적용에 대해 연구 추진중에 있다.

참 고 문 헌

- [1] ISO/IEC 9646, "Information Technology - open Systems Interconnection - Conformance testing methodology and framework", 1994
- [2] R.J.Linn. "Conformance Testing for OSI Protocols", *Computer Network and ISDN Systems* 18, 1989.
- [3] B. Sarikaya, A. Wiles, "Standard Conformance Test Specification Language TTCN", *Computer Standards and Interfaces*, pp. 117-144, 1992.
- [4] Ana R. Cavalli, J. P. Favreau, and M. Philippou, "Formal methods for conformance testing: results and perspectives", in *Protocol Test Systems VI*, North Holland: Elsevier Science Publishers B. V., pp. 3-19, Sep. 1992.
- [5] 김상기, 정재운, 김성운, 김태균, "SDL을 사용한 INAP프로토콜 모델링", 정보처리학회 '96 추계 학술 발표 논문집, 제3권 2호 10월, 1996.
- [6] Z. Kohavi, "Switching and finite Automata Theory", New York, McGraw-Hill, 1978.
- [7] A. V. Aho *et al.*, "An optimization technique for protocol conformance test generation based on UIO sequences and rural Chinese postman tours", *IEEE Trans. on Communications*, vol. 39, no. 11, pp. 1604-1615, Nov. 1989.
- [8] 한국전자통신연구소, "차세대지능망 응용 프로토콜 규격", 8월, 1996.
- [9] Samuel T. Chanson *et al.*, "On tools supporting the use of formal description techniques in protocol development", *Computer Network and ISDN Systems* vol. 25, pp. 723-739, 1993.
- [10] J. Edmons and E. L. Johnson, "Matching, Euler tours and the Chinese postman's tour", *Mathematical Programming*, vol. 5, pp. 88-124, 1973.



도 현 속

194년에 동경공업대학교에서 석사학위 취득후 95년 이후 한국전자통신연구원 선임연구원으로 근무하고있음. 관심분야는 차세대 지능망 및 INAP 프로토콜 관련 시험임.



조 준 모

95년과 97년에 경북대학교에서 석사학위 및 박사과정을 수료한 후, 98년이후 동명대학에 교수로 근무하고 있음. 관심분야는 ATM 교환기, VHDL임.



김 성 운

90년과 93년에 프랑스 파리 7대학 석사 및 박사학위를 취득하였다. 82년부터 85년까지 한국전자통신연구원에서 근무하였으며, 86년부터 95년까지 한국통신연구개발원에서 근무하였음. 96년 이후 부경대학교 정보통신공학과

조교수로 재직중임.

관심분야 : 초고속망 및 프로토콜공학분야임.