

## LTS 간접시험 생성기법에 의한 프로토콜 적합성 시험에 관한 연구

正會員 許 其 澤\*, 正會員 金 光 鉉\*\*, 正會員 李 銅 鎬\*\*\*

### A Study on Protocol Conformance Testing Using LTS Indirect Test Case Generation Method

Gi Taek Hur\*, Gwang Hyun Kim\*\*, Dong Ho Lee\*\*\* Regular Members

이 논문은 1993 년도 한국학술진흥재단의 자유공모과제 연구비에 의하여 연구되었음

#### 要 約

통신 프로토콜과 서비스를 서술할때 일반적으로 비형식적인 방법들이 사용됨으로 해서 의미의 모호성을 내포하게 된다. 이로 인하여 발생하는 문제점을 해결하고, 프로토콜 구현시 자동화를 쉽게하기 위해서는 프로토콜을 설계, 구현 및 시험시 동일한 형태의 형식명세기법(Formal Description Technique :FDT)을 사용하여야 한다. 그리고 구현된 프로토콜이 이기종 시스템에서 연동하기 위해서는 구현된 프로토콜의 동작이 표준에 맞도록 구현되었는가를 확인하는 적합성 시험이 필요하다. 따라서 본 논문에서는 프로토콜설계, 구현 및 시험할때 발생하는 문제점들을 지적하고 이중 비결정성을 제거하기 위해서 중간 모델인 LTS를 이용함으로써 효율적인 시험항목을 생성하기 위한 방법을 제시하였다.

#### ABSTRACT

Using the informal method in specification of communication protocol and service, ambiguity is contained in a protocol. To remove the ambiguity contained in a protocol and automatize easily in protocol design, FDT(Formal Description Technique) must be used in protocol specification, implementation and testing. To interoperate an implemented protocol in heterogeneous systems, conformance testing must be performed about the capability and behaviors of a protocol. In this paper, we propose the method is to generate efficient test case by using intermediate model LTS to remove the problem generated in protocol design, implementation and testing.

\*동신대학교 전자계산학과

Dept. of Computer Science, Dong Shin Univ.

\*\*기전여자 전문대학 사무자동화과

Dept. of Office Automation, Kijeon Women's Junior College

\*\*\*광운대학교 전자계산학과

Dept. of Computer Science, Kwang Woon Univ.

論文番號 : 94307

接受日字 : 1994年 11月 3日

## I. 서론

컴퓨터 통신망의 증대와 통신프로토콜의 복잡성의 증가로 인하여 통신프로토콜을 정확하게 기술하고자 하는 요구가 대두됨에 따라 자연언어를 사용한 기존의 명세작성보다는 형식명세기법을 사용하는 경향이 나타나게 되었다(1, 8, 10). 이를 위해서 ISO에서는 두개의 형식명세기법인 LOTOS와 ESTELLE 사용을 제안하였고, ITU-T에서는 SDL을 제안하였다(7, 10). 하나의 프로토콜이 개발되어서 실제로 사용되기 위한 과정을 총체적으로 프로토콜 공학이라 하고 이의 구조는 그림 1과 같다(6).

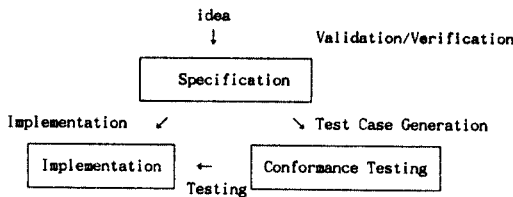


그림 1. 프로토콜공학  
Fig. 1. Protocol Engineering

그림 1에서 처럼 프로토콜 요구사항 서술과정, 구현및 적합성 시험들간의 유기적인 관계는 프로토콜 공학에서 중요한 위치를 차지한다. 이들을 형식명세기법을 사용하여 서술함으로써 자연언어를 사용한 비형식적인 형태로 서술할때 보다 시스템을 모호성 없이 정확하게 기술할 수 있을뿐만 아니라, 검증및 구현등을 더욱 더 쉽게 자동화 할 수 있다. 따라서 이러한 각 과정의 결합은 각 프로토콜 개발 도구를 통합하여 통합환경을 구성할 수 있으므로 비형식적인 형태로 서술함으로써 발생하는 문제점들을 해결할 수 있다.

## II. 적합성 시험

본 절에서는 프로토콜을 설계 및 구현시 비형식적인 방법들이 사용됨으로서 발생하는 많은 문제점들을 해결하기 위해서 사용되는 표준 형식명세기법과 적합성 시험 방법등에 대해서 살펴보았다.

### 2.1 표준 형식명세기법

ISO에서 정의한 ESTELLE은 모듈간의 상호작용의 파라

미터, 상태변수 및 이와 관련되는 계산처리를 위해서 파스칼의 자료구조, 수식, 명령문을 추가하여 확장한 유한상태기계(Extended Finite State Machine : EFSM)모델에 기초한 구현 중심 서술 언어(Implementation Oriented Specification Language)이다(1, 13). LOTOS는 자료를 묘사하는 부분과 행위를 묘사하는 두 개의 부분으로 구성된다. 자료부분은 추상 데이터 유형으로 서술되고, 행위부분은 CCS(Calculus of Communication Systems)로 묘사한다(8). ITU-T에서는 확장한 유한상태기계를 기초로 한 SDL를 제안하였는데 이는 ESTELLE과 유사하게 프로그램 변수, 데이터 구조와 함께 추상적 데이터 유형의 개념을 사용하였다. SDL의 표현방식은 순서도와 유사한 그래픽 다이어그램으로 전체 시스템을 표현하는 SDL/GR(Graphical Representation)과 프로그래밍 언어와 유사한 SDL/PR(textual Phrase Representation)이 있다. 그런데 일반적으로 이해하기 더 쉬운 SDL/GR이 시스템을 설계하는데 주로 이용된다(4, 10).

ESTELLE의 구문은 프로토콜의 기능을 표시하는 모듈과 모듈간의 통신을 제공하는 채널이라는 개념을 이용하여 구성되므로 통신 프로토콜을 기술할때 다른 형식명세기법보다 이해하기 더 쉽고 효율성이 높아 널리 이용되고 있다. LOTOS명세는 자료와 행위묘사 부분으로 구성된다. 자료는 추상적인 데이터 유형으로 묘사되고 행위는 Milner의 CCS를 기초로 한 프로세스 대수이론을 사용하여 묘사된다. 행위묘사는 게이트를 사용해서 상호작용하는 증명된 프로세스들의 계층구조 형태로 이루어진다. LOTOS의 자료부분은 추상적인 데이터 유형으로 묘사되고 행위부분은 CCS로 묘사되므로 다른 형식명세기법에 비해서 복잡하고 이해하기 어려운 단점이 있지만 구조자체가 간단하여 복잡한 시스템을 쉽게 나타낼 수 있다. 특히 LOTOS에서는 병렬성에 대한 표현이 다른 형식명세기법보다 강력하여 분산 시스템을 표현하기에 가장 적합한 언어이다. 최근에는 분산 시스템이나 프로토콜들을 좀 더 실제적으로 명세하기 위해서 객체지향의 개념이 LOTOS에 도입되고 있다. 그리고 LOTOS에서는 추상 데이터 유형과 Hiding의 개념을 지원하므로 객체지향화가 용이하므로 본 논문에서도 프로토콜을 LOTOS로 기술하였다.

### 2.2 적합성 시험

프로토콜의 적합성 시험은 그림 2와 같이 구현된 프로토콜의 동작이 프로토콜 표준으로 정의된 요구사항에 적합하게 동작하는가를 검증하는 것으로 구현된 OSI 프로토콜들 간의 상호운용성을 높이기 위한 것이다 [4,5,10]. 그림 2에서 관찰은 일련의 입출력에 대해서 시스템이 어떻게 동작하는가를 보여주는 것이고, 요구사항은 프로토콜 명세시 동작하도록 가정된 행위를 의미하는 것으로 관찰된 행위와 예측된 행위가 일치하는가를 검증하는 과정을 적합성 시험이라고 한다.

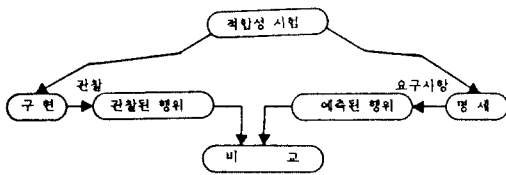
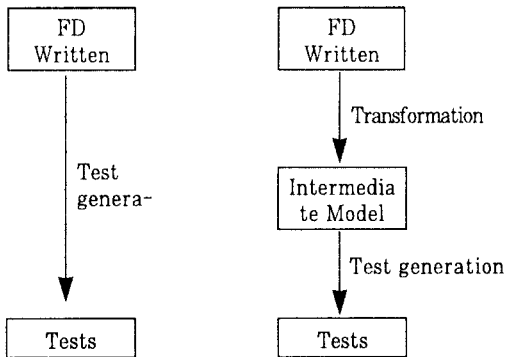


그림 2. 구현과 명세 사이의 관계  
Fig. 2. Relationship between implementation and specification

### Ⅲ. 시험항목 생성방법

형식명세기법을 기초로한 시험항목 생성방법은 두가지로 분류될 수 있다. 하나는 형식명세로부터 직접 시험항목을 생성하는 경우와 형식명세를 중간모델로 변환하고 중간모델로부터 시험항목을 생성하는 방법이 있다 [8].



(a) Direct Test Generation (b) Indirect Test Generation

그림 3. 테스트 생성방법  
Fig. 3. Test Generation Method

### 3.1 FSM-based 시험항목 생성

FSM모델에 관한 이론들이 광범위하게 연구되었고 프로토콜의 제어흐름을 쉽게 모델화할 수 있으므로 많은 적합성 시험 방법들이 FSM를 기초로 하여 이루어지고 있다. 프로토콜을 명세하기 위해서는 프로토콜이 무엇을 수행하고 서비스 프리미티브나 타임 아웃과 같은 사건에 대해서 어떻게 대응해야 되는가를 기술해야 한다. 그런데 프로토콜은 일반적으로 매우 복잡하므로 프로토콜 설계단계에서 부터 완전하고, 정확하며, 간결하면서도 모호성을 포함하지 않게 명확하게 기술되어야만 한다. 이를 위해서 형식명세기법이 사용된다. 프로토콜을 명세하기 위해서 형식명세기법을 사용했다고 할지라도 구현자가 프로토콜을 잘못 구현하면 구현된 제품들 간의 상호운용이 이루어지지 않게 된다. 그러므로 이를 방지하기 위해서는 적합성 시험이 시험순서를 사용하여 수행되어야만 한다.

### 3.2 UIO 순서

본절에서는 지금까지 연구된 시험순서 생성방법들이 주로 제어흐름을 쉽게 나타낼 수 있는 결정적 유한상태기계를 기초로 하여 이루어진 Transition Tour, W-method, Distinguishing 순서, UIO(Unique Input Output) 순서 등과 같은 방법중 가장 많이 사용되는 UIO 순서에 대해서만 설명한다.

UIO 순서는 프로토콜이 최소의 유한상태기계로 명세되었다는 가정을 기초로 한다. 이 방법은 어떤 다른 상태에서는 발생되지 않아서 특정 상태를 유일하게 구분할 수 있는 입출력 순서의 쌍인 UIO 순서를 이용하여 시험 항목을 생성하는 방법이다[12,13,14]. 예를 들면 그림 4의 각 상태의 UIO 순서가 표 1에 있다. 그림 4에서 상태 A로부터 길이가 1인 순서는 a/y 와 b/y가 있다. 그런데 a/y는 상태 B에서 C로 천이를 발생시키는 순서이므로 상태 A에만 유일한 것이 아니므로 UIO 순서가 아니다. 순서 b/y는 어떤 다른 상태에서 볼 수 없는 유일한 값이므로 b/y가 상태 A의 UIO 순서가 된다. 상태 B에서는 길이가 1인 유일한 순서가 없다. 그러므로 길이가 2인 순서를 찾으면 순서 a/y a/x 가 다른 상태에서는 발생되지 않으므로 상태 2의 UIO 순서가 된다. 상태 C에서는 입출력 순서 a/x에 의해서 상태 A로 천이가 발생하고 a/x는 상태 C에만 유일하므로 a/x가 상태 C의 UIO 순서가 된다.

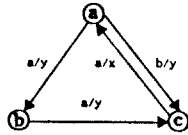


그림 4. 결정적 유한상태기계  
Fig. 4. Deterministic Finite State Machine

표 1. UIO 순서  
Table 1. UIO 순서

| 상 태 | UIO 순서  |
|-----|---------|
| A   | b/y     |
| B   | a/y a/x |
| C   | a/x     |

### 3.3 중간모델을 이용한 시험항목 생성

명세 S로 부터 직접 수행가능한 테스트로부터 표현하기가 어렵기 때문에 중간모델을 정의하여 먼저 중간모델로 변환한 후에 중간모델로부터 시험항목을 생성한다. 본 절에서는 중간모델로 정의된 EFSM, ASM, CHART, LTS중 LTS에 대해서만 정의하였다[8].

#### 정의 1. LTS(Labelled Transition System)

레이블된 천이시스템은 4개의 쌍인  $L = \langle G, A, -\mu \rightarrow, g_0 \rangle$ 로 표현된다.

여기서  $G$ : 상태들의 집합

$A$ : 외부적으로 관찰가능한 사건들의 집합

$-\mu \rightarrow$ :  $G$ 에서 정의된 2항 관계집합

(여기서  $\mu \in AU(\tau)$ 이고,  $\tau$ : 내부사건이다.)

$g_0$ : 초기상태 ( $g_0 \in G$ )

관계  $-\mu \rightarrow$ 은 사건  $\mu$ 의 수행의 효과를 나타내고,  $g \rightarrow g'(g, g' \in G)$ 은 사건  $\mu$ 를 수행함으로써 상태  $g$ 에서  $g'$ 로 천이가 발생함을 의미한다.  $g \rightarrow g'$ 은 외부적으로 관찰이 불가능한 내부사건에 의해서 상태  $g$ 에서  $g'$ 로 천이가 발생함을 의미한다[2].

## IV. 비결정성 제거

유한상태기계에 관한 이론들이 광범위하게 연구되었고 프로토콜의 제어흐름을 쉽게 나타낼 수 있어서 시험항목 생성시 결정적 유한상태기계가 주로 사용되었다. 그러나 프로토콜이 상태 천이 시스템 모델로 표현될때 많은 프로토콜들은 한 상태에서 한 개의 똑같은 입력에 의해서 하나 이상의 상태로 천이가 발생할 수 있는 비결정성 문제들을 포함할 수 있다[1,2,3,13]. 비결정성은 다음과 같은 두가지 유형으로 된다.

#### ① 구분가능 비결정성(Distinguishable Non-determinism)

똑같은 입력에 의해서 개시된 모든 천이가 각각 서로 다른 출력을 생성하는 것으로 시험기는 발생될 천이가 어떤것 인가를 정확하게 탐지할 수 있는 비결정성

#### ② 구분 불가능 비결정성(Indistinguishable Non-determinism)

똑같은 입력에 의해서 개시된 모든 천이가 똑같은 출력을 생성하므로 시험기는 어떤 천이가 발생될 것인가를 탐지할 수 없는 비결정성

비결정성으로 인하여 천이에 대한 예측이 어렵기 때문에 프로토콜의 제어흐름을 나타내기가 어려울 뿐 만 아니라 지금까지 잘 알려졌던 결정적 유한상태기계에 적용되었던 시험항목 생성방법들을 그대로 적용하여 시험항목을 생성할 수 없다. 따라서 이 장에서는 천이에 대한 예측이 가능토록 하기 위해서 비결정성이 제거되어야 하고, 비결정성이 제거된 후의 결과에 대한 검증이 필요하므로 이를 위한 절차가 그림 5에 제시되었다.

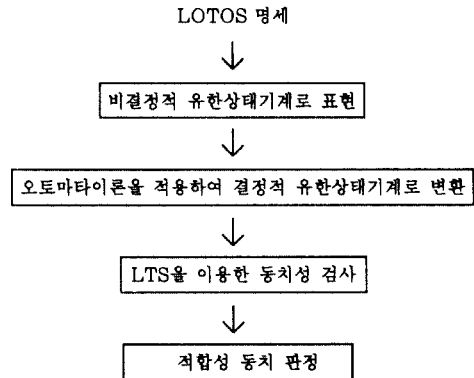


그림 5. 비결정성 제거 과정  
Fig. 5. The Nondeterminism Removal Step

4.1 비결정성 제거

프로토콜이 비결정성을 포함하면 비결정적 유한상태기계로 표현되고, 비결정성을 제거한 후의 결과는 결정적 유한상태기계로 나타낸다. 그런데 시험기는 관찰 가능한 입출력값만을 처리할 수 있으므로 비결정성은 하나의 동일한 입력값에 의해서 발생하는 천이가 하나 이상 존재할때, 어떤 천이가 먼저 처리되어야 할 것인지를 결정하지 못했을때 발생하게 된다. 따라서 이와 같은 문제를 해결하기 위해서 오토마타론을 LOTOS로 표현된 트랜스포트 프로토콜에 적용한 결과가 그림 7에 제시되었다[3].

```

specification T_Protocol(U,L) no exit
behavior TP_Simple(U,L)
where
process TP_Simple(U,L) : noexit :=
    U?TCOnreq;L!CR(L?CC:L!DR:U!TDisSind:L?DC:TP_simple(ts,ns)
        ()
        L?CC:U!TCOnconf:Open(ts,ns)
    )
    ()
    L?CR(L!DR:TP_simple(U,L)
        ()
        U!TCOnNind(Uclose(U,L)
            ()
            U?TCOnresp:L!CC:Open(U,L)
        )
    )
endproc
process Open(U,L) : noexit :=
    (U?TDATAreq:L!DT:Open(U,L)
    () L?DT:U!DT:Open(U,L)
    () L?AK:Open(U,L)
    () Uclose(U,L)
    () Rclose(U,L)
    () !:U!TDisSind:L!DR:L?DC:TP_Simple(U,L)
    () !:L!AK:Open(U,L)
    )
endproc
process Uclose(U,L) : noexit :=
    U?TDisSreq:L!DR:L?DC:TP_Simple(U,L)
endproc
process Rclose(U,L) : noexit :=
    L?DR:U!TDisSind:L!DC:TP_Simple(U,L)
endproc
endspec
    
```

그림 6. LOTOS로 명세된 트랜스포트 프로토콜  
Fig. 6. A Transport Protocol Specification in Basic LOTOS

5. LTS를 이용한 적합성 동치

지금까지 비결정성 문제를 처리하기 위해서 먼저 LOTOS에 포함된 비결정성의 유형을 정의하고 이를 오토마타론을 적용하여 제거하였다. 그런데 오토마타이

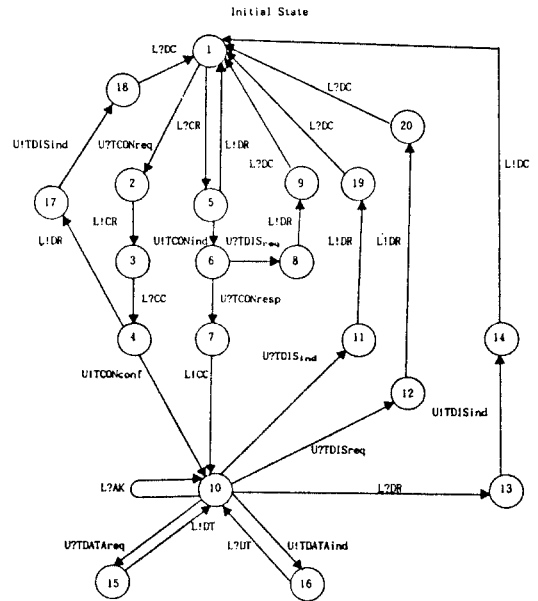


그림 7. 유한상태기계로 표현된 트랜스포트 프로토콜  
Fig. 7. The Transport Protocol Specified with FSM

론을 적용하여 비결정성이 제거되었다고 할지라도 LOTOS에서 비결정성이 제거 되기전에 인식했던 모든 천이들을 인식할 수 있는가의 여부를 검증하기 위한 과정이 필요하다. 이를 위해서 Tr(S)를 S가 받아들일 수 있는 모든 입출력 순서들의 집합이라고 하면, 다음과 같은 정의 2가 성립될 수 있다[2,10,13].

정의 2( 순서 추적 동치관계)

적합성 시험시 두 개의 유한상태기계 N과 D에 대해서 Tr(N) 과 Tr(D)가 같으면 순서 추적 동치(Trace Equivalence)관계가 성립한다고 한다.

그런데 순서 추적 동치관계는 교착상태나 라이브 리등을 포함하는 시스템에는 부적합하므로 이를 처리할 수 있는 방법들이 필요하다[2,9]. 정의1에서 정의된 LTS와 유한상태기계 모델은 유사하지만 단 하나의 차이점은 LTS에서는 사건 a(a∈AU{τ})에 의해서 상태 g에서 g'로 천이가 발생하면 g-a→g'로 나타내지만, 유한상태기계 모델에서는 상태 g에서 i가 입력되면 o를 출력하고 g'로 천이가 발생하면 g<sup>i/o</sup>→g'로 나타낸다. 두 개의 천이시스템은 LTS의 사건을 유한상태기계의 입/출력 쌍으로 대응시키거나, 역으로 비결정적 유한상태기계의 입

/출력 쌍을 LTS의 사건으로 대응시키면 쉽게 서로간의 변환이 가능하므로 비결정적 유한상태기계에서 정의되었던 기호들이 LTS 모델에서도 그대로 사용될 수 있다. 예를 들면 비결정적 유한상태기계의  $\sigma=i1/o1, \dots, i_n/o_n$  은 LTS의  $\sigma=a_1, \dots, a_n$  (여기서  $a_k=ik/ok, 1 \leq k \leq n$ ) 으로 표현될 수 있다. 다음과 같은 LOTOS 표현을 LTS 으로 표현하면 그림 8과 같다.

```

process A_Sample[a] : noexit =
  i: A_Sample[a]
  []
  a: stop
endproc

process B_Sample[a] : noexit =
  i: a: stop
endproc
    
```



그림 8. 유도트리  
Fig. 8. The Derivation Tree

그림 8에서 관찰 가능한 사건에 대한 동치성을 고려할 경우에는 (a) 와 (b) 둘다 사건 a만을 인식할 수 있지만 A\_Sample 프로세스에서 비결정적인 선택에 의해서 내부전이만이 계속적으로 수행될 경우에는 라이브 렉이 발생할 수 있다. 이와같이 비록 두개의 레이블된 천이 시스템이 똑같은 입출력 순서의 쌍을 처리할 수 있다고 할지라도 라이브 렉이 발생된 경우에는 순서 추적 동치관계만으로는 두개의 레이블된 시스템에 대한 동치성을 보일 수 없을 뿐만 아니라, 교착상태가 발생하는 경우에는 모든 가능한 경우에 대한 입출력 순서의 쌍을 고려하는 것은 불가능하므로 이들을 처리할 수 있는 약 Bisimulation을 정의하면 다음과 같다(2.11).

정의 3(약 Bisimulation)

두 개의 LTS  $L_1=(G_1, a_1, -\mu \rightarrow, g_1)$ 와  $L_2=(G_2, a_2, -\mu \rightarrow, g_2)$ 에 대해서  $G=G_1 \cup G_2$ 이고 모든 사건을  $\alpha(\alpha \in a_1 \cup a_2)$ 라 할때, 2항 관계  $R(R \in G \times G)$ 이  $(P, Q) \in R$ 에 대해서 다음 조건을 만족하면 약 Bisimulation이라 한다.  
(i) 만약  $P \xrightarrow{\alpha} P'$ 인  $P'(P' \in G)$ 가 존재하면,

$Q \xrightarrow{\alpha} Q'$ 인  $Q'(P', Q') \in R$ 이 존재한다. (단  $\alpha' \in \alpha \setminus \{\tau\}$ ,  $\tau$ 은 내부전이 이다.)

(ii) 만약  $Q \xrightarrow{\alpha} Q'$ 인  $Q'(Q' \in G)$ 가 존재하면,  $P \xrightarrow{\alpha} P'$ 인  $P'$ 와  $(P', Q') \in R$ 이 존재한다.

정의 4 (약 관계)

- (i) 두 상태 P, Q에 대한 약 Bisimulation R이 존재하면 P와 Q를 약 관계(Weak Relation)에 있다고 한다.
- (ii) 두 개의 천이시스템  $L_1$ 과  $L_2$ 의 초기상태에 대한 약 Bisimulation 관계가 성립되면 두 개의 천이시스템  $L_1$ 과  $L_2$ 가 약 관계에 있다고 한다.

따라서 본 연구에서도 약 관계를 이용하여 비결정적 유한상태기계와 결정적 유한상태기계의 동치관계를 검증하였다. 약 관계를 이용하여 동치관계를 검증하는 알고리즘은 다음과 같다.

Algorithm1(Search the weak relation between two labeled transition systems)

/\* 두 개의 LTS  $L_1=(G_1, a_1, -\mu \rightarrow, g_1)$ 과  $L_2=(G_2, a_2, -\mu \rightarrow, g_2)$ 에 대한 약 관계 조사과정 \*/

- step (1). 초기 약 Bisimulation S구성  
For  $\forall \alpha (\alpha \in a_1 \cup a_2)$ , If  $\exists p \xrightarrow{\alpha} p'$  and  $q \xrightarrow{\alpha} q'$   
(단,  $p, p', q, q' \in G_1 \cup G_2$ ) Then add  $(p, q)$  to S
- step (2). 새로운 약 Bisimulation S'를 구성한다.  
(i) For  $\forall \beta$ , if  $p \xrightarrow{\beta} p'$  and  $q \xrightarrow{\beta} q'$  then check  $(p', q') \in S$   
(ii) If  $(p', q') \in S$  then add  $(p, q)$  and  $(p', q')$  to S'
- step (3). 새로운 약 Bisimulation이 생성되지 않을 때까지 2를 반복 수행한다.

정리1. 약관계를 찾는 algorithm1의 time complexity는  $O(n(1,1,1_2+t))$  이다.

증명) step1) 초기 약 bisimulation S 구성시간 사건수를 n이라 하고 LTS  $L_1$ 과  $L_2$

에서 각 사건에 대해서 발생하는 천이 수를 각  $l_1, l_2$  라 하자. 그러면 step1에서는 모든 사건에 대해서  $L_1$  과  $L_2$ 에서 발생하는 천이에 대한 약 bisimulation S를 찾아야 하므로  $n \times l_1 \times l_2$  이다.

step2) 새롭게 구성된 관계가 약 bisimulation S에 포함되는지의 여부를 판별하는 시간을 t라 하면 각 사건에 대해서 새로운 약 bisimulation S'를 time complexity는  $L_1$ 시스템에서 각 사건에서 발생하는  $l_1$ 에 대해서  $L_2$ 시스템에서 각 사건에 대해서 발생하는 천이수  $l_2$ 의 모든 경우이다 새롭게 구성된 관계가 초기 약 bisimulation S에 포함되는지의 여부를 판별하여야 하므로  $n(l_1 l_2 + t) = n l_1 l_2 + n t$  이므로 전체 time complexity는  $O(n(l_1 l_2 + n t))$ 가 된다.

예를 들면 다음과 같은 LOTOS의 표현식을 LTS으로 표현한 것이 그림 9(a)에 있고 비결정성이 제거된 결과가 그림 9(b)에 있다. 일반적으로 적합성 시험은 black box 형태로 수행되므로 어떤 외부의 입력이 없어도 천이가 발생하는 내부천이는 시험기에 의해서는 제어될 수 없다. 따라서 내부상황에 관한 상세한 정의는 구현자에 의해서 결정되므로 시험자의 관점에서는 어떤 천이가 선택되어도 합법적인 것으로 취급되어야만 한다. 즉, 외부시험기에 적용될 수 있는 시험항목들은 제어 가능한 입력들과 관찰가능한 출력들로만 표현되어야 하므로 내부천이가 제거되어야 한다. 일반적으로 두개의 시스템이 동치성을 갖기 위해서는 외부적으로 관찰가능한 동작들로 두개의 시스템을 구분할 수 없을까 두개의 시스템이 동치성을 갖는다. 그러므로 내부천이는 chun[11]에서 제시한 비결정성 제거알고리즘을 적용하여 제거하고 그 결과가 그림 9(b)에 제시되었다. 그리고 일단 비결정성이 제거되면 구분가능 비결정성을 제거하기 위해서 [3]에서 제시한 알고리즘을 적용하고 그 결과는 그림 9(c)에 제시하였다.

그림 9에 있는 두 개의 시스템에 대한 동치관계를 조사하기 위해서 위에서 제시된 알고리즘을 적용한 결과가 다음에 제시되었다.

$a:(b:a:stop()|i:c:stop)()|a:c:stop$

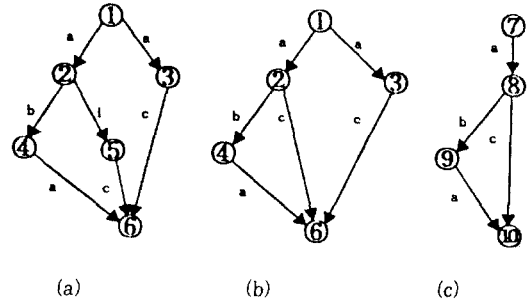


그림 9 LTS  
Fig. 9. Labelled Transition Systems

1. 초기 약 Bisimulation S 구성.

사건 a에 대한 관계: (1,7), (1,9), (4,7), (4,9)

사건 b에 대한 관계: (2,8)

사건 c에 대한 관계: (3,8), (5,8)

stop : (6,10)

$S = \{(1,7), (1,9), (4,7), (4,9), (2,8), (3,8), (5,8), (6,10)\}$

2. 새로운 약 Bisimulation 구성.

사건 a에 대한 관계 :

- $(1-a \rightarrow 2) \in L_1$ 일때  $(7-a \rightarrow 8 \text{ and } 9-a \rightarrow 10) \in L_2$  이므로

$(2, 8) \in S$ 이고  $(2, 10) \notin S$  이므로  $S' = \{(1,7), (2,8)\}$

- $(1-a \rightarrow 3) \in L_1$ 일때  $(7-a \rightarrow 8 \text{ and } 9-a \rightarrow 10) \in L_2$  이므로

$(3,8) \in S$ 이고  $(3,10) \notin S$ 이므로  $S' = \{(1,7), (2,8), (3,8)\}$

- $(4-a \rightarrow 6) \in L_1$  일때  $(7-a \rightarrow 8 \text{ and } 9-a \rightarrow 10) \in L_2$  이므로

$(6, 8) \in S$ 이고  $(6,10) \in S$  이므로  $S' = \{(1,7), (2,8), (3,8), (6,10)\}$

사건 b에 대한 관계 :

- $(2-b \rightarrow 4) \in L_1$ 일때  $(8-b \rightarrow 9) \in L_2$  이므로

$(2, 8) \in S$ 이고  $(4, 9) \in S$  이므로  $S' = \{(1,7), (2,8), (3,8), (6,10), (4,9)\}$

사건 c에 대한 관계 :

- $(3-c \rightarrow 6) \in L_1$  이고  $(8-c \rightarrow 10) \in L_2$  일때

$(3,8) \in S$  이고  $(6,10) \in S$  이므로  $S' = \{(1,7), (2,8), (3,8), (6,10), (4,9)\}$

- $(5-c \rightarrow 6) \in L_1$  이고  $(8-c \rightarrow 10) \in L_2$  일때
- $(5, 8) \in S$  이고  $(6, 10) \in S$  이므로
- $S' = \{(1, 7), (2, 8), (3, 8), (6, 10), (4, 9), (5, 8)\}$

따라서 그림 9의 (a)와(b)의 초기 상태인 1과 7이 약 Bisimulation관계가 성립되므로 정의 4에 의해서 약 관계가 성립되고, (a)에서 발생하는 천이들인 aba와 ac가 (b)에서도 그대로 생성될 수 있으므로 동치관계가 성립된다.

먼저 두개의 유한상태기계인 N과D가 받아들일 수 있는 모든 입출력순서들의 집합인  $Tr(N)$ 과  $Tr(D)$ 가 같으면 N과 D는 순서 추적 동치관계가 성립된다. 그러나 그림8과 같이 라이브러리가 발생하면 순서 추적 동치관계만으로는 동치성을 검증하는것이 불가능 하다. 예를들면 그림8의 a와 b에서 생성한  $Tr$ 는  $Tr(8.a) = i^* a = a$  이고  $Tr(8.b) = i+ a = a$  로 같다. 그러나 그림8.a에서 i가 비결정적으로 계속선정되면 라이브러리가 발생할 수 있으므로 적합성 동치관계를 판별할 수 없는 경우가 발생할 수 있다. 이때 모든 사건에 대한 약 관계를 조사하면 그림 8.a는 ①<sup>a</sup>→②가 되고 그림8.b는 ④<sup>a</sup>→⑤가 되어 동치관계가 성립됨을 알 수 있으므로 다음과 같은 정의5가 성립될 수 있다.

정의 5(적합성 동치관계)

두 개의 유한상태기계 N과 D에 대해서 순서 추적 동치 관계와 약 관계가 만족하면 N과 D는 적합성 동치(Conformance Equivalence) 관계에 있다고 한다.

일반적으로 두개의 천이 시스템에 대한 동치성을 검증하기 위해서 순서 추적 동치관계가 주로 사용된다. 그런데 순서 추적 동치관계만을 이용하여 동치성을 검증할 경우에는 라이브러리가 발생하면 검증이 불가능하므로, 본 논문에서는 라이브러리가 발생한 경우까지도 동치성을 조사할 수 있는 약 Bisimulation을 적용한 약 관계를 이용하여 적합성 동치관계를 검증하였다.

Ⅶ. 결 론

현재 FMCT에서는 비형식적인 형태로 프로토콜을 서술할 경우에 제기되는 애매성을 제거하고, 자동화를 쉽

게 하기 위해서 형식명세기법을 사용한 프로토콜서술, 구현및 시험하는 통합환경을 구축하기 위한 연구가 진행 중이다. 이와같은 통합환경을 구성할때에 가장 중요한 시험항목 생성방법들은 주로 FSM을 기초로 하여 구성된다. 그런데 이를 실제로 프로토콜에 적용할 경우에 FSM의 제한성 때문에 많은 문제점이 발생한다. 이중 특히 포함된 비결정성으로 인하여 프로토콜에 대한 예측이 불가능하므로 비결정성은 제거되어야만 한다. 이를 위해서 본 논문에서는 프로토콜을 중간 모델인 LTS를 적용하여 비결정성을 제거하였고, 비결정성이 제거된 결과를 약 관계를 이용하여 적합성 동치관계를 검증할 수 있는 알고리즘을 제시하였다.

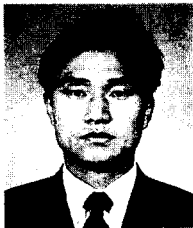
앞으로 더 연구되어야 할 방향은 제어흐름 뿐만 아니라 자료의 흐름까지를 함께 고려하여 처리할 수 있는 방법과 FSM상에 포함된 오류 모델을 처리할 수 있는 방법에 관한 연구가 더 진행되어야 할 것이다.

참고문헌

1. 이도영, "상호통신하는 확장된 FSM모델에 근거한 프로토콜의 형식적 적합성시험에 관한 연구", 포항공대 박사학위논문, 1991.
2. 허기택, "비결정성 제거에 근거한 통신 프로토콜의 효율적인 시험항목 생성", 광운대 박사학위논문, 1994
3. 허기택, 이동호, "효율적인 통신프로토콜 시험을 위한 비결정성 제거 알고리즘", 한국통신학회논문지, 제18권10호, pp.1572-1581, 1993, 10.
4. G. T. Hur, H. Y. Choi, D. H. Lee, "An Efficient Test Case Generation of Communication Protocols Using Multiple UIO/UIIO Sequences", ICCT94, June, Shanghai, China, 1994.
5. B.Sarikaya and G. V. Bachman, "A Test Design Methodology for Protocol Testing", IEEE 1987.
6. D.Rayner, "OSI Conformance Testing", Computer Networks&ISDN Systems, 14 : 79-87, 1987.
7. D.Shidu and T.Leung, "Formal Method for Protocol Testing: A Detailed Study", IEEE Trans. On. SE., Vol.15, No.4, April, 1989.



8. J.P.Favreau, D.Hogrefe, and J.Kroom, "Formal Methods in Conformance Testing : Status and Expectation", working draft reports,1992.
9. K.Sabani and A.Bahbura,"A Protocol Test Generation Procedure", Computer Networks and ISDN systems, 15 : pp285-297, 1988.
10. R. J. Linn Jr, " Conformance Testing for OSI Protocols", Computer Network & ISDN Systems, 18 : 203-219, 1990.
11. R.Milner, Communication and Concurrency, Prentice Hall,1989.
12. T.S.Chow, " Testing Software Design Modeled by Finite State Machines ", IEEE Trans. On. SE., Vol.SE-4, May, 1978.
13. W.J. Chun, "Test Case Generation for Protocols Specified in ESTELLE ", Ph.D. Thesis Delaware, May, 1992.
14. W. Y. L. Chan, S. T. Vuoung, and M.R. Ito, "An Improved Protocol Test Generation Procedure Based on UIO's", SIGCOMM'88 Symposium : Communication Architecture and Protocols in Computer Comm. Review 19(4), September 1989, PP.283-294.



許 其 澤 (Gi Taek Hur)    정회원  
 1960년 8월 9일생  
 1984년 2월 : 전남대학교 계산통계학과 졸업(이학사)  
 1986년 2월 : 전남대학교 대학원 계산통계학과 졸업(이학석사)  
 1994년 2월 : 광운대학교 대학원 전자계산학과 졸업(이학박사)

1989년 3월~현재 : 동신대학교 전산통계학과 조교수  
 1990년 3월~현재 : 동신대학교 전자계산소 소장  
 ※ 주관심분야 : 프로토콜 공학, 초고속 통신



金 光 鉉 (Kim Gwang Hyun) 정회원  
 1962년 2월 10일생  
 1989년 2월 : 광운대학교 전자계산학과 졸업(이학사)  
 1991년 2월 : 광운대학교 대학원 전자계산학과 졸업(이학석사)  
 1992년 3월~현재 : 광운대학교 대학원 전자계산학과 박사과정

1991년 3월~현재 : 전주기전여자전문대학 전자계산과 전임강사  
 ※ 주관심분야 : 컴퓨터 통신, 프로토콜 공학, 컴퓨터네트워크



李 銅 鎬 (Dong Ho Lee)    정회원  
 1957년 1월 15일생  
 1979년 : 서울대학교 전자공학 학사  
 1983년 : 서울대학교 컴퓨터공학 석사  
 1988년 : 서울대학교 컴퓨터공학 박사  
 1984년 9월~현재 : 광운대학교 전자계산학과 부교수

※ 주관심분야 : 컴퓨터 구조, 운영체제론, 컴퓨터 네트워크