

# 적합성 시험에서 그래프 재표기 시스템을 활용한 강한 연결 판단 방법

이 준 원<sup>†</sup> · 김 성 운<sup>††</sup> · 구 연 설<sup>†††</sup>

## 요 약

I/O FSM 모델로 표현된 프로토콜들로부터의 시험 계열 생성은 I/OFSM 명세 자체가 강하게 접속되어야 하며 (strongly connected), 상태(state) 수가 최소(minimal)여야 하며, 그리고 또 결정형(deterministic)이어야 한다는 가정에서 출발한다. 본 논문에서는 프로토콜을 나타내는 명세 I/OFSM (또는 Graph)이 이러한 가정으로 출발되는 이유를 객관화시키고, 또 그래프 재표기 시스템(Graph Rewriting System)을 정의하고, 이를 명세 그래프(명세 I/OFSM)에 적용시켜 기존의 알고리즘보다 훨씬 빨리 강한 접속 여부를 판단하는 알고리즘을 제시한다.

## Strong Connectivity Decision Method using Graph Rewriting System in Conformance Testing

Joon Won Lee<sup>†</sup> · Sung Un Kim<sup>††</sup> · Yeon Seol Koo<sup>†††</sup>

## ABSTRACT

Test generation from the communication protocol specified in I/OFSM protocol is based on the assumption that the specification S and implementation I are strongly connected, minimal and deterministic. In this paper, we identify why these assumptions are necessary for minimal test cases generation from I/OFSM protocol specification, and we propose a Graph Rewriting System and its application to the specification I/OFSM for verifying its strong connectivity. We prove that the suggested algorithm is more efficient than the traditional strongly connected component find algorithm.

### 1. 서 론

프로토콜 적합성 시험의 목적은 어떤 프로토콜 구현 I(implementation)가 원래 명세(혹은 표준) S(Specification)에 합당하게 구현되어 있는지를 시험하는 것으로 통신 프로토콜 제품 구현 과정에서 중요한 역할

을 한다. 적합성은 명세와 구현의 관계로 구현이 명세에 대하여 올바른지의 여부이다. 또한 적합성 여부의 판단 대상은 임의의 명세 또는 주로 표준화된 명세이다[1].

간단한 명세의 경우 구현의 적합 여부는 직접 계산함으로써 결론지어질 수 있다. 그러나, 복잡한 국제 표준 명세의 경우 직관적 계산 또는 어떤 연역적 증명에 의하여 적합성 여부를 판단하는 것은 매우 어려운 일이다. 이러한 경우 시험이라는 실험적, 경험적 방법을 동원하여 판단하게 되는데 이러한 실험적 방

† 정 회 원: 한국전자통신연구원 초고속망연구실

†† 정 회 원: 부경대학교 정보통신공학과 교수

††† 정 회 원: 충북대학교 컴퓨터과학과 교수

논문접수: 1996년 7월 31일, 심사완료: 1997년 4월 23일

법을 적합성 시험이라고 부른다.

명세는 명세 언어를 통하여 기술되는데 명세 언어 자체는 자연어와 형식 언어(FDL: Formal Description Language)로 나뉘고, 현재 국제 표준들은 자연어를 기본으로 하여, 자연어가 가지고 있는 애매성(ambiguity), 모호성(vagueness)을 극복하기 위하여, 형식 언어로 된 기술로 보충하고 있으며, 점차 형식 언어의 비중이 커져가는 추세이다.

형식 언어들은 대체로 확장된 개념의 유한 상태기계(Extended Finite State Machine)에 기초로 한 ESTELLE 또 SDL들이 있고 CCS, CSP같은 프로세서들간의 개념 모델에 기초로 한 LOTOS 등이 있는데, 많은 적합성 시험 계열 생성 방법들이 이들 형식 언어로부터 직접 시험 계열을 생성하기 위해 중간 모델(intermediate model)로 I/OFSM을 사용하여 변환한 후 I/OFSM으로부터의 시험 계열 생성에 초점을 맞추어 왔다[2]. 다른 한편으로는 I/OFSM모델로 통신 프로토콜을 표현하여 자체를 형식 언어로 사용하여 시험 계열 방법 연구에 많이 몰두하여 왔다[3].

I/OFSM으로 표현된 명세에 대한 프로토콜 적합성 시험을 위한 접근 방법은 명세에 정의된 I/OFSM의 각 천이(transition)에 대하여 구현의 천이 동작을 확인하는 방법인데, 이 방법을 경험적 확인 방법(checking experiment method)이라 한다. 이 방법은 명세의 각 상태(state)에서 각각의 시험 대상 천이에 대한 입력과 이 입력에 대한 출력이 구현에서도 올바르게 이루어졌는지를 확인하고, 구현에서 결과적으로 도착한 상태에 대해서도 검증하는 과정으로 구성된다. 즉 이러한 각 천이를 시험하기 위한 시험 계열(test case)은 입력/출력 시퀀스로 이루어진다[4].

현재까지 경험적 확인 방법에 근거하여 여러 종류의 시험 계열 생성 방법이 연구되어져 왔는데, 시험 대상 천이에 대한 시험 후 구현의 결과 상태 확인 방법에 따라 UIO(Unique Input Output) 방법, DS(Distinguishing Sequence) 방법, 그리고 CS(Characterization Set) 방법 등으로 나뉘어 진다. 이러한 방법들은 명세의 각 상태에서 시험 대상 천이 "입력/출력" 시퀀스에 이 시퀀스에 의해 도착한 상태의 유일한 I/O 시퀀스를 연결하여 시험하므로 구현에서 도착 상태를 검증하는 방법이다.

이러한 방법들은 자세히 살펴보면, I/OFSM으로

표현된 프로토콜 명세 S와 구현 I가 강한 연결형이고 결정형이며 최소 상태 수이고, 또 구현 I의 상태 수가 명세 S의 그것과 일치하는 상태에서 트레이스 동치(trace equivalence) 또는 관찰 동치(observational equivalence) 관계로 요약된다[5]. 본 논문에서는 이러한 동치 관계 확인을 위한 중요한 조건인 강한 연결성 여부 가정을 객관화시키고, I/OFSM에서 강한 연결 상태를 확인하기 위해서 기존의 그래프 관련 알고리즘 "strongly connected component find" 보다 효과적인 알고리즘을 생성하기 위해 그래프 재표기 시스템을 정의하고, 이를 명세 또는 구현 I/OFSM(또는 graph)에 적용하여 더욱 더 효과적임을 제안한다.

이를 위하여, 본 논문의 2장에서 프로토콜 모델링에 관계된 I/OFSM에 대한 정의 및 관련 특징들을 기술하고, 또 그래프 재표기 시스템에 대한 정의와 관련 규칙들에 대해 설명한다. 3장에서는 I/OFSM으로 표현된 프로토콜 명세의 적합성 개념과 여러 가지 위에 언급된 가정들을 수학적으로 객관화하고, 강한 연결 특성이 요구되어지는 이유를 정의하고, I/OFSM 프로토콜 명세의 강한 연결 여부 판단을 위하여 기존의 "strongly connected component find" 계산 알고리즘보다 그래프 재표기 시스템을 이용한 알고리즘이 더욱 더 효율적임을 제안한다. 5장에서 결론을 맺는다.

## 2. I/OFSM과 그래프 재표기 시스템(Graph Rewriting System)

### 2.1 I/OFSM

일반적으로 I/OFSM은 통신 프로토콜의 제어 부분을 모델화하는데 많이 사용되어져 왔다. I/OFSM의 정의는 다음과 같다[2].

[정의 1] 입출력 유한 상태 기계(I/OFSM)

프로토콜 명세의 제어 부분을 나타내는 I/OFSM은 다음의 다섯 가지 요소로 구성된다.

$$M = \langle S, s_0, I, O, tr \rangle$$

여기서

- S: 유한 상태들의 집합(a set of finite states)
- s<sub>0</sub>: 초기 상태(initial state)
- I: 유한 입력 알파벳(finite input alphabet)

- O: 유한 출력 알파벳(finite output alphabet)
- tr: 천이 함수(transition function),  $tr \subseteq (\{s-i/o \rightarrow s' \mid s, s' \in S \wedge i \in I \wedge o \in O\})$

다음의 4개 요소(p, a, b, q)는 I/OFSM M의 천이로 불려지는데 아래와 같이 정의된다.

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

위의 정의에서 각 천이는 하나의 입력과 하나의 출력에 의해 일어난다.

I/OFSM  $M = \langle S, s_0, I, O, tr \rangle$ 은 방향화 된 그래프(oriented graph)  $G = (V, E)$ 로[6] 표기되어질 수 있는데, 여기서

- $V = S$ 의 상태에 해당하는 노드의 집합(a set of nodes)
- $E =$  추이함수 tr에 해당하는 아크의 집합(a set of arcs)

따라서 본 논문에서는 그래프와 I/OFSM, 오토마타는 동의어로 사용하고 각각에 적용되는 용어들을 구별 없이 이용한다[4][5]. ■

[정의 2] 결정성(determinism)

I/OFSM M은 모든 상태  $q \in S$ , 또 모든 입력  $a \in I$ 를 위해 만약  $(q, a, b, q')$ 가 M에서 하나의 천이일 때  $q'$ 가 유일하게 하나의 상태로 결정될 경우로 정의한다. ■

[정의 3] 최소 I/OFSM(minimal I/OFSM)

I/OFSM M을 인식하는 언어(Language) L을 가지고 같은 언어 L을 인식하면서 가장 작은 수의 상태로 M이 구성되었을 시 최소 I/OFSM이라 정의한다. ■

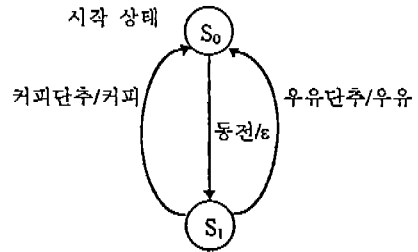
[정의 4] 강한 연결(strongly connected)

I/OFSM(혹은 방향화 된 그래프)의 어떤 상태(또는 그래프의 노드)에서도 어떠한 다른 상태(또는 그래프의 노드)로도 패스(path)가 존재할 때 강한 연결 특성을 가졌다고 하며, 이 특성은 주어진 프로토콜 명세(I/OFSM 혹은 그래프)가 deadlock이나 livelock 성질이 없음을 의미한다. ■

[예 1] 본 예에서는 위에서 논의된 정의들을 설명하기 위해 단순한 기능을 가진 자판기에 대해서 설명한다. 자판기 V의 명세는 다음 네 개의 자연어로 된 요구 사항( $r_1, r_2, r_3, r_4$ )으로 이루어져 있다.

- $r_1$ : V는 하나의 동전을 받아들인다.
- $r_2$ : V가 동전을 하나 받아들인 뒤에 커피단추와 우유단추 중 하나가 눌러진다.
- $r_3$ : 커피단추가 눌러진 후에, V는 커피를 내보낸다.
- $r_4$ : 우유단추가 눌러진 후에, V는 우유를 내보낸다.

이와 같이 자연어로 표현된 자판기 명세 V를 I/OFSM 명세로 표기하면 다음 (그림 1)과 같이 표현된다.



(그림 1) 자판기 V를 나타내는 I/OFSM (Fig. 1) I/OFSM for specification V

(그림 1)은 자판기 V를 I/OFSM 형태로 나타낸 명세인데, 여기서 동전에 대한 출력 "ε"은 빈 행위(null action)를 나타낸다. 이 I/OFSM 명세는 정의 2, 3 그리고 4에 의하여 결정형이며, 최소 I/OFSM이며, 또 강한 연결 특성을 가지고 있다.

2.2 그래프재표기 시스템(Graph Rewriting System)

2.2.1 기본 개념

단순 그래프(simple graph)  $G = (V, E)$ 에서 V는 한정된 노드의 집합이며, E는 노드 쌍의 집합, 즉  $E \subseteq \{\{v, v'\} \mid v, v' \in V\}$ 로 나타내어진다. 이러한 그래프 정의와 관련하여 다음의 관련 정의들을 그래프 재표기 시스템을 정의하기 위해 내린다[7].

[정의 5] 페 루프 없음(no self-loop)

모든  $e = \{v, v'\}$ 에 대하여  $v \neq v'$ 이면, 이 그래프는

폐 루프를 포함하지 않는다고 정의한다. ■

[정의 6] 이웃(neighbor)

$e = \{v, v'\}$ 에 대하여  $e$ 는  $v$ 에 접해 있다고 하며,  $v$ 는  $v'$ 의 이웃이라고 정의한다. ■

[정의 7] 인접 에지(adjacent edge)

두 에지는 그들이 공통의 노드들을 가질 때 인접하다고 정의한다. ■

[정의 8] 정도(degree)

노드  $v$ 의 정도는  $d(v)$ 로 나타내는데  $v$ 에 인접한 에지의 수로 정의한다. ■

[정의 9] 패스(path) 및 사이클(cycle)

$p = v_1, e_1, v_2, e_2, \dots, e_{i-1}, v_i$ , 여기서  $1 \leq j < i$ ,  $e_j$ 는  $v_j$ 의  $v_{j+1}$ 과 인접해 있을 경우  $p$ 를 길이  $i-1$ 의 패스라 정의한다. 만약  $p$ 내에서  $v_i = v_1$ 이면  $p$ 는 사이클이라 정의한다. ■

[정의 10] 단순 패스(simple path)

하나의 패스 내에서 각 노드가 단지 한 번씩 나타날 때 단순 패스라 정의한다. ■

[정의 11] 연결(connected)

단순 그래프에서 어느 두 노드도 서로 연결되어 있을 경우 그래프가 연결되었다고 정의한다. ■

### 2.2.2 그래프 재표기 시스템

앞 절에서 정의된 연결된 단순 그래프 상에서 그래프 재표기 시스템이 정의되는데, 이 시스템은 기본 그래프를 변경하는 것이 아니라 노드와 아크 상에 레이블을 추가하고 재표기 법칙(rewriting rule)에 의해 레이블을 변경하여 더 이상 재표기 법칙이 적용될 수 없는 상태로 천이하면서 과정 자체가 하나의 원하는 알고리즘으로 정의되는 것이다.

[정의 12] 그래프 isomorphic

두 그래프  $G_1(V_1, E_1)$ ,  $G_2(V_2, E_2)$ 는  $V_1$ 과  $V_2$  사이,  $E_1$ 과  $E_2$  사이에 일대일 대응의 짝  $\phi = (\phi_V, \phi_E)$  즉,

$$\forall \{v, v'\} \in E_1, \phi E(\{v, v'\}) = \{\phi v(v), \phi v(v')\}$$

을 만족할 때, 서로 isomorphic 하다고 정의한다. ■

그래프 재표기 시스템에서 사용되는 레이블화 된 그래프(labeled graph)는 단순 그래프에 레이블이 첨가된 그래프인데, 두 개의 유한 집합  $C^{(V)}$ 와  $C^{(E)}$ 를 그래프  $G(V, E)$ 에 레이블화 시킨 레이블화 된 그래프를  $G(V, E, \lambda)$ 로 나타내면, 레이블화 함수  $\lambda = (\lambda^{(V)}, \lambda^{(E)})$ 로 나타내어진다. 여기서  $\lambda^{(V)}$ (각기  $\lambda^{(E)}$ )는  $V$ (각기  $E$ )로부터  $C^{(V)}$ (각기  $C^{(E)}$ )로의 매핑이다.

[정의 13] 레이블화 된 그래프 isomorphic (Labelled Graph Isomorphic)

두 레이블화 된 그래프  $G_1(V_1, E_1, \lambda_1)$ ,  $G_2(V_2, E_2, \lambda_2)$ 는  $G_1(V_1, E_1)$ 과  $G_2(V_2, E_2)$  사이에 isomorphism  $\phi = (\phi_V, \phi_E)$  즉,

$$\lambda_2^{(V)} \cdot \phi_V = \lambda_1^{(V)}, \quad \lambda_2^{(E)} \cdot \phi_E = \lambda_1^{(E)}$$

을 만족할 때 서로 isomorphic 하다고 정의하고,  $G_1(V_1, E_1, \lambda_1) \cong (G_2(V_2, E_2, \lambda_2))$ 로 표기한다. ■

[정의 14] 그래프재표기 법칙(graph rewriting rule)

레이블화 된 연결형 그래프(labelled connected graph)의 레이블의 짝  $R = (G_R(V_R, E_R, \lambda_R), G'_R(V_R, E_R, \lambda'_R))$ 을 그래프 재표기 법칙이라 정의하며, 재표기 관계를  $\vec{R}$ 로써, 즉

$$G(V, E, \lambda) \vec{R} G'(V, E, \lambda')$$

로 표기한다. ■

레이블화 된 연결형 기본 그래프에서 더 이상  $G(V, E, \lambda) \rightarrow G'(V, E, \lambda')$ 로  $R$ 을 적용할  $G'(V, E, \lambda')$ 가 존재하지 않을 때, 그래프를 irreducible 하다고 하며, 더 이상의 재표기 법칙이 적용되지 않는 그래프를 의미한다. 명백하게, 주어진 레이블화 된 연결형 기본 그래프  $G(V, E, \lambda)$ 에서 여러 번의 원하는 재표기 법칙을 적용한 후에 유일한 irreducible 그래프  $I_n(G)$ 를 얻는데, 여기서 재표기 법칙 적용 각 스텝들이 우리가 원

하는 알고리즘을 나타낸다.

정의 13의 레이블화 된 그래프와 정의 14의 그래프 재표기 법칙을 이용하여 적합성 시험을 위한 I/OFSM을 레이블화 된 연결형 그래프로 모델링하여 강하게 연결되었는지의 여부(strongly connected characteristic)를 판별하는 알고리즘을 다음 장에서 자세하게 기술한다.

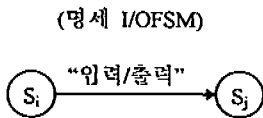
### 3. I/OFSM 적합성 개념과 그래프 재표기 시스템 응용

#### 3.1 I/OFSM 모델의 적합성 개념

일반적으로 I/OFSM으로 기술된 명세는 통신 프로토콜의 기능과 동작에 기반을 두고 만들어지게 되는데, 기술된 명세 S에서 시작되는 적합성 시험은 명세 I/OFSM의 모든 천이에 대해 black box로 여겨지는 구현 I가 같은 천이를 보이는지를 경험적 확인 방법에 근거하여 시험하는 방법이다. 다음 정의는 I/OFSM으로 기술된 명세 S에서 시작되는 적합성 시험 과정을 나타낸다.

#### [정의 15] 적합성 시험의 정의

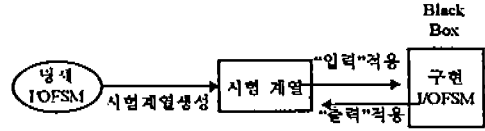
프로토콜의 명세를 나타내는 I/OFSM 모델이 다음과 같이 주어질 때, 적합성 시험의 정의는 아래의 3단계로 이루어진다.



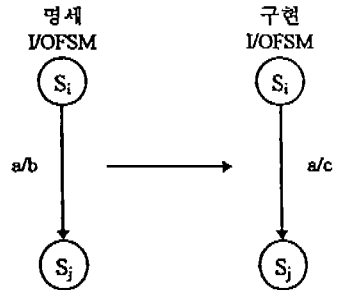
#### (적합성 시험 단계)

- (1) 명세 I/OFSM의 상태 Si에 해당하는 상태로 구현 I/OFSM을 위치시킨다.
- (2) 명세 I/OFSM에서 일어난 시험 계열 "입력"을 구현 I/OFSM에 적용시킨 후 생성되는 "출력"을 판단한다.
- (3) 구현 I/OFSM에서 생성된 출력이 명세 I/OFSM에서 기술된 "출력"과 같은지를 확인하고 도착한 상태가 명세 I/OFSM의 그것과 같은지를 검증한다. ■

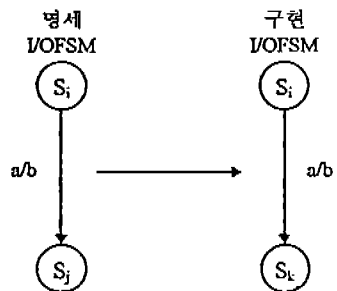
정의 15의 적합성 시험 과정을 도식화하면 다음과 같이 표현된다.



위의 적합성 시험 과정에서 확인되는 오류들은 일반적으로 "출력 오류"와 "천이 오류"로 구분되는데, 출력 오류는 주어진 "입력"에 대한 예상되지 않은 "출력"이 구현 I/OFSM으로부터 다음과 같이 도출될 때이다.



천이 오류는 주어진 "입력"에 대해 예기치 않은 상태로 천이가 일어나는 다음과 같은 경우이다.



출력 오류와 천이 오류를 걸러내기 위한 정의 15의 적합성 시험 단계에서 일반적으로 두 가지 문제가 야기된다. 단계 1에서 일반적으로 구현 I/OFSM은 black box로 고려되므로, 다른 추가의 행위(action)없이 구현 I/OFSM을 원하는 상태로 위치시킬 수 없다는 것이고, 두 번째로 단계 3에서 시험 계열의 "입력"을 적

용한 후에 원하는 “출력”을 구현 I/OFSM으로부터 얻은 경우라도 구현 I/OFSM이 해당 상태에 위치하였는지 알 수 없다는 것이다. 첫 번째 문제를 제어성 한계(controllability limit)라 하고, 두 번째 문제를 관찰성 한계(observability limit)라 한다.

일반적으로 제어성 한계 문제를 해결하기 위해서 초기 상태  $s_0$ 로부터 시작하여 원하는 상태까지 가장 짧은 패스(shortest path)를 이용하여 상태에 도착한 후 해당 천이를 시험하게 되고, 관찰성 한계 문제를 해결하기 위해 시험하는 천이 후에 도착한 상태의 유일한 입력/출력 시퀀스를 시험 계열에 포함시켜 적용한 후 구현 I/OFSM의 결과 상태를 확인하는 방법을 사용한다. 이러한 유일한 입력/출력 시퀀스로 UIO 시퀀스, DS 시퀀스, CS 시퀀스 등을 사용한다.

I/OFSM 모델로 표현된 프로토콜들로부터의 시험 계열 생성은 위의 모든 문제를 해결하고 구현에 대해서도 100% 시험 영역 커버(test coverage)를 달성하는 가장 짧은 길이의 시험 시퀀스 생성 문제로 귀결된다. 이러한 여러 문제들을 I/OFSM 명세 자체가 강하게 접속되어야 하며, 상태의 수가 최소어야 하고, 또 결정형이어야 한다는 가정과 서로 밀접한 관계를 가진다. 다음 소절에서는 이들의 서로 연관성에 대해서 논의한다.

3.2 강한 접속, 최소 상태 수, 결정형 I/OFSM

정의 2의 결정성 정의를 다음과 같이 다른 방법으로 정의를 내릴 수 있다.

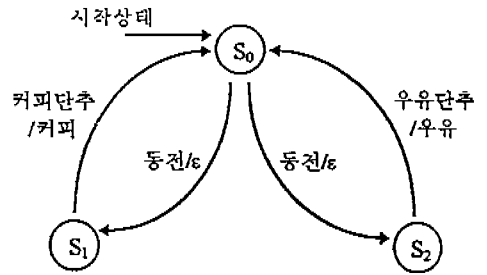
[정의 16] 결정성의 또 다른 정의

정의 1을 만족하는 I/OFSM의 천이 집합  $T_i$ 내에서, 어느 한 상태에 대해 같은 입력  $i \in I$ 에 대해 오직 하나의 천이만 존재하고,  $T_i$ 은 다음 조건을 만족할 때 I/OFSM은 결정형이다.

$$\forall s, s_1, s_2 \in S, i \in I, o_1, o_2 \in O : (s - i/o_1 \rightarrow s_1 \wedge s - i/o_2 \rightarrow s_2) \rightarrow o_1 = o_2 \wedge s_1 = s_2 \quad \blacksquare$$

[예 2]

만약 프로토콜을 표현한 명세 I/OFSM이나 구현 I/OFSM이 결정형이 아닐 경우에 대해서 살펴보자. (그림 2)는 (그림 1)에 나타난 자판기 I/OFSM에 대한



(그림 2) (그림 1)에 나타난 자판기 I/OFSM의 또 다른 표현 (Fig. 2) Another I/OFSM for specification V shown in (Fig. 1)

또 다른 가능한 표현이다.

위의 (그림 2)에서  $s_0$ 에서  $s_1$ ,  $s_0$ 에서  $s_2$ 로 같은 천이 “동전/ε”를 나타내는 비결정형이다.

이런 경우 “동전/ε”에 대한 시험이 어느 것을 의미하는지를 알 수 없고, 특히 위의 그림이 구현 I/OFSM인 경우 시험 계열 “동전/ε” 시험 시, 어느 것을 의미하는지 전혀 알 수 없으므로 시험에 실패할 가능성이 높다. 결과적으로 위와 같은 경우를 위해 비결정형 명세 I/OFSM과 비결정형 구현 I/OFSM에 대한 시험 방법에 대해 많이 연구되고 있지만, 본 논문에서는 결정형에 제한하여 다룬다.

정의 16의 결정형의 가정에 의해 I/OFSM M의 상태  $S_i$ 는 유일한 다음 상태와 유일한 출력을 생성하는 함수이다. 즉,  $v = i_0 i_1 \dots i_k - 1$ 이라 두면

$T_i(S_i, v)$  = 다음 상태(next state)

$S_i(v)$  = 출력 시퀀스

로 표기되고, 입력 시퀀스  $v$ 에 대한 I/OFSM M의 트레이스(trace)는  $S_i(v)$ 로써 정의되고, 이는 즉 I/OFSM의 초기 상태에서 시작된 모든 트레이스를 의미한다.

결정형의 두 I/OFSM의  $(M_1, M_2)$  isomorphic 정의는 초기 상태에서 시작된 모든 트레이스  $V$ 에 대한  $M_1$ 과  $M_2$  사이의 트레이스 동치를 의미하고, 이는 관찰 동치와도 같은 것인데 두 I/OFSM의 적합성 정의를 만족시키는 관계이다.

[정의 17] 강한 연결의 또 다른 정의

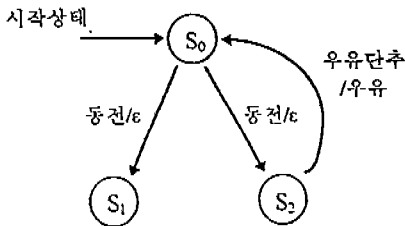
정의 1을 만족하는 I/OFSM M이 강한 연결 상태

이면, 각 상태  $S_i$ 에 대해서 입력/출력 시퀀스  $p(S_i)$ 가 존재해서 이 시퀀스를 적용하면,  $M$ 을 다른 모든 상태  $S_k (S_k \neq S_i)$ 로부터  $S_i$ 에 도착하게 하는 다음을 만족한다.

$$T_i(S_k, P(s_i)) = S_i \quad \blacksquare$$

[예 3]

다음 I/OFSM은 강한 연결이 아니다.



위의 예는 정의 17의 강한 연결 정의에서  $T_i(S_1, P(s_0)) = s_0$ 을 만족시키지 못하므로 강한 연결이 아니고, 이런 경우  $s_0$ 에서  $s_1$ 으로의 천이 “동전/ε”에 대한 도착 상태 확인 시퀀스가 존재하지 않고, 프로토콜 명세 자체가 deadlock을 의미하며 적합성을 위한 트레이스 동치 관계도 의미가 없어서 적합성 관계에 대한 정의를 내릴 수가 없다.

다음 소절은 위에 언급된 적합성 관계 확립에 가장 중요한 명세 혹은 구현 I/OFSM의 강한 연결 여부를 그래프 재표기 시스템에 의해 판단하는 효과적인 새로운 알고리즘에 대해서 설명한다.

### 3.3 그래프 재표기 시스템 응용

I/OFSM  $M$ 의 강한 연결 상태를 확인하기 위해 그래프 재표기 시스템을 적용한다. 우선 I/OFSM  $M = \langle S, s_0, I, O, t, \rangle$ 을 방향화 된 그래프(oriented graph)  $G = (V, E)$ 로 표기할 수 있는데, 여기서

$V = S$ 의 상태에 해당하는 노드의 집합  
 $E =$  천이 함수  $t_e$ 에 해당하는 방향화 된 아크의 집합

으로 두고, 이 기본 방향화 된 그래프의 노드에 레이블 첨가하여 그래프 재표기 시스템을 정의한다[8].

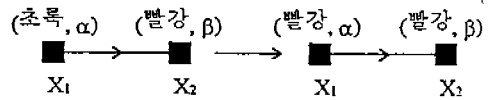
먼저 초기 상태  $s_0$ 에 해당하는 노드에 색깔 레이블

작(빨강, 파랑)을 부여하고, 나머지 노드에는 색깔 레이블 작(초록, 초록)을 부여하면, 색깔로 레이블화 된 그래프가 생성된다. 노드 색깔 집합은  $C(V) = \{\text{초록, 빨강}\} \times \{\text{초록, 파랑}\}$ 이다. 이와 같이 얻어진 색깔로 레이블화 된 그래프에 다음의 그래프 재표기 법칙  $R_1$ 과  $R_2$ 를 적용한다:

#### • 재표기 법칙 $R_1$

$\alpha, \beta \in \{\text{초록, 파랑, 빨강}\}$ 이라 하고, 노드  $X_1$ , 즉 색깔 (초록,  $\alpha$ )인 것이 아크  $(X_1, X_2)$ 에 의해 노드  $X_2$ , 즉 색깔 (빨강,  $\beta$ )에 연결되어 있으면  $X_1$ 은 새로운 색깔 레이블 (빨강,  $\alpha$ )을 부여한다.

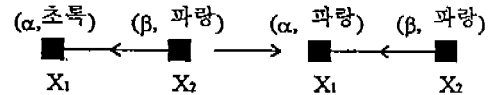
이 법칙은 다음의 그래프 재표기 법칙으로 도식화 된다.



#### • 재표기 법칙 $R_2$

만약 노드  $X_1$ , 즉 색깔 ( $\alpha$ , 초록)이 아크  $(X_2, X_1)$ 에 의해 노드  $X_2$ , 즉 색깔 ( $\beta$ , 파랑)에 연결되어 있으면,  $X_1$ 은 새로운 색깔 레이블 ( $\alpha$ , 파랑)을 부여한다.

이 법칙은 다음의 그래프 재표기 법칙으로 도식화 된다.



위에서 제시된 그래프 재표기 법칙  $R_1$ 과  $R_2$ 를 적용 하면서 그래프 재표기 시스템 응용 과정은 다음의 5 단계로 이루어진다.

#### [강한 연결 여부 검증 단계]

- (1) 프로토콜 명세 그래프  $G(V, E)$ 의 초기 노드에는 (빨강, 파랑) 색깔 짝을 부여하고, 나머지 노드들은 (초록, 초록) 색깔 짝을 부여하여 레이블화 된 그래프를 생성한 후 (2)를 수행한다.
- (2) 레이블화 된 프로토콜 명세 그래프 상에서 ( $\beta$ , 파

랑)→(α, 초록) 형태로 된 아크가 존재하면 (3)을 수행하고, (초록, α)→(빨강, β) 형태로 된 아크가 존재하면 (4)를 수행한다. 만약 위에 언급된 형태와 아크가 존재하지 않으면(5)를 수행한다.

(3)(β, 파랑)→(α, 초록) 형태의 레이블화 된 아크의 해당 노드들에 (β, 파랑)→(α, 파랑) 형태로 레이블을 변형하고 (2)를 수행한다.

(4)(초록, α)→(빨강, β) 형태의 레이블화 된 아크의 해당 노드들에 (빨강, α)→(빨강, β) 형태로 레이블을 변형하고 (2)를 수행한다.

(5) 레이블화 된 결과 그래프의 모든 노드들이 (빨강, 파랑) 레이블 형태인 색깔 짝으로 구성되어 있으면 프로토콜 명세 그래프 G(V, E)는 강한 연결 형태이고 그렇지 않으면 강한 연결 형태가 아니다. ■

위에서 정의한 강한 연결 여부 판정을 위한 검증 단계는 재표기 법칙 R<sub>1</sub>과 R<sub>2</sub>를 필요한 수만큼 적용한 후에 유일한 irreducible 그래프 I<sub>n</sub>(G)가 얻어지는데, 여기서 재표기 법칙을 적용하여 각 스텝들이 우리가 원하는 알고리즘을 나타낸다.

[정리 1]

재표기 법칙이 더 이상 적용될 수 없는 색깔로 레이블화 된 그래프 G를 irreducible하다고 이야기하며, 명확하게, 주어진 그래프 G로부터 위의 재표기 법칙들을 적용하므로써 단지 하나의 irreducible 그래프 I<sub>n</sub>(G)가 얻어진다.

증명) 매번 위의 재표기 법칙 R<sub>1</sub> 혹은 R<sub>2</sub>를 적용할 때마다, 초록 색깔이 하나씩 사라진다. 그래서 색깔로 레이블화 된 그래프 G(V, E)에 재표기 법칙을 적용하는 수는 2(n-1)번 인데, 여기서 n은 노드의 개수이다. 정확하게 2(n-1)번의 재표기 법칙 후 Irr(G)가 얻어진다. 더욱 상세한 증명은 예 4로 가름한다. ■

[정리 2]

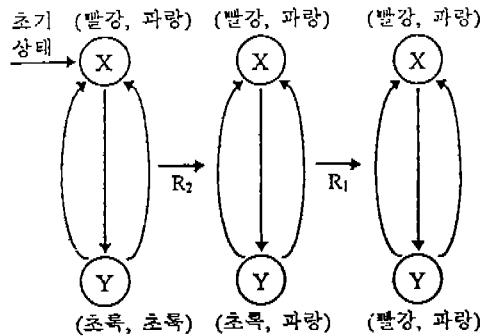
초기 상태 s<sub>0</sub>에 해당하는 노드의 강하게 연결된 요소는 I<sub>n</sub>(G) 그래프의 (빨강, 파랑) 색깔 레이블을 가진 노드의 집합이다.

증명) I<sub>n</sub>(G)의 색깔 (α, 파랑)을 가진 어느 노드 y도 초기 상태에서부터 y에 이르는 직접 패스가 존재한다.

또 I<sub>n</sub>(G)의 색깔 (빨강, β)을 가진 어느 노드 y도 이 노드로부터 초기 상태에 이르는 직접 패스가 존재한다. 따라서, I<sub>n</sub>(G)의 (빨강, 파랑) 색깔을 가지는 노드 Y는 초기 상태에서부터 y 혹은 y로부터 초기 상태로 직접 패스가 존재하며, 이것은 y가 초기 상태의 강하게 연결된 요소임을 의미한다. ■

[예 4]

(그림 1)의 자판기 I/OFSM을 색깔로 레이블화 된 그래프 G=(V, E, λ)로 표현하면, 다음의 첫 그래프와 같이 되고 재표기 법칙 R<sub>1</sub>과 R<sub>2</sub>를 적용하면 결과적으로 다음과 같이 나타내며 강하게 연결된 component는 I<sub>n</sub>(G) 그래프의 (빨강, 파랑) 색깔 레이블을 가진 노드의 집합이다.



위에 소개된 그래프 재표기 시스템에 의한 강하게 연결된 요소를 찾는 알고리즘은 고전적인 그래프 이론에 의거한 강하게 연결된 요소를 찾는 알고리즘에 비해 더욱 효과적이다. 왜냐하면, 고전적인 "strongly connected component find" 계산 알고리즘은 O(Max(n, e))(여기서 n은 노드의 수이고, e는 아크의 수)의 시간 복잡도를 요구하는 반면[8], 위에 언급된 알고리즘은 O(2(n-1))를 요구한다. 일반적으로 통신 프로토콜은 I/OFSM(혹은 그래프)로 많이 명세화되어 사용되고 구현되어진다. 프로토콜을 명세한 I/OFSM(혹은 그래프)는 프로토콜의 행위 특성에 따라 e의 수가 n의 수보다 훨씬(일반적으로 수배 이상) 크기 때문에 그래프 재표기 시스템을 사용한 알고리즘을 적용하면 O(2(n-1)) 시간 복잡도로 간단하게 구현할 수 있어 더욱 효과적이다. 예 4에서도 간단히 검증할 수



가 있다.

#### 4. 결 론

I/OFSM으로 표현된 프로토콜로부터의 적합성 시험 계열 생성은 black box로 취급되는 구현에 대한 제어성 한계와 관찰성 한계를 해결하고, 100% 시험 영역 커버(test coverage)를 달성하는 가장 짧은 길이의 시험 시퀀스 생성 문제로 귀결된다. 본 논문에서는 이러한 여러 문제들은 I/OFSM 명세 자체가 강하게 절속되어야 하며, 상태의 수가 최소여야 하고, 또 결정형이어야 한다는 가정과 밀접한 관계를 가짐을 설명함으로써 객관화하였다. 특히, I/OFSM 자체가 강하게 연결되어야 하는 이유를 정의하였고, I/OFSM의 강한 연결 여부를 판단하기 위하여 기존의 그래프 이론에 바탕을 둔 "strongly connected component find" 계산 알고리즘보다 더욱 더 효율적인 그래프 재표기 시스템을 이용한 알고리즘을 제안하였다. 제시된 알고리즘이 기존의 알고리즘보다 더욱 효율적임이 두 가지 정리를 증명하므로 밝혀졌다. 이러한 그래프 재표기 시스템에 대해서 재표기 법칙간의 우선 순위를 적용하는 우선 순위를 적용한 그래프 재표기 시스템에 대한 연구가 향후 요구된다.

#### 참 고 문 헌

- [1] ISO, Information Processing Systems-Open Systems Interconnection-LOTOS-A Formal Description Technique Based on the Temporal Ordering of Observational Behaviour, International Standard IOS 8807, Genhve, January 1989.
- [2] W. Chun and P. D. Amer, Improvements on UIO Sequence Generation and Partial UIO Sequences, 12th International Symposium on Protocol Specification, Testing and Verification, Lake Buena, Florida, USA, 1992.
- [3] M. S. Chen, Y. Choi, A. Kershenbaum, Approaches Utilizing Segment Overlap to Minimize Test Sequence, 10th International Symposium on Protocol Specification, Testing and Verification Ottawa, Canada, June 1990.
- [4] K. Sabnani and A. Dahbura, A Protocol Test Generation Procedure, Computer Networks and ISDN Systems, Vol. 15, No 4, 285-297, 1988.
- [5] A. Gibbons, Algorithmic Graph Theory, Cambridge; Cambridge University Press, 1985.
- [6] Berge, C., Graphes et Hypergraphes, Dunod, Paris, 1970.
- [7] Huet, G., Confluent Reductions: Abstract Properties and Applications to Term Rewriting Systems, J.ACM 27 (4), 1980, pp. 93-112.
- [8] Aho, A. O., Hopcroft, J. E., and Ullman, J. D., The Design and Analysis of Computer Algorithms, Addison-Wesley, Reading, Mass. 1974.



#### 이 준 원

1976년 서울대학교 전자공학과(공학사)  
 1992년 충북대학교 대학원 전산통계학과(이학석사)  
 1995년 충북대학교 대학원 전자계산학과 박사과정수료

1977년~1979년 삼성전기 기술개발실  
 1980년~현재 한국전자통신연구원 초고속망 연구실장  
 관심분야: 초고속정보통신시스템, 소프트웨어공학, 통신프로토콜 등



#### 김 성 운

1982년 경북대학교 전자공학과  
 1990년 프랑스 국립파리 7대학 정보공학과(공학 석사)  
 1993년 프랑스 국립파리 7대학 정보공학과(공학 박사)  
 1982년~1985년 한국전자통신연구소 데이터통신연구소(연구원)

1986년~1995년 한국통신 연구개발원(선임연구원)  
 1990년~1993년 프랑스 전기통신기술연구소(초빙연구원)  
 1995년~현재 부경대학교 정보통신공학과(교수)  
 관심분야: 프로토콜 엔지니어링, 데이터 통신 통신프로토콜시험, 컴퓨터 네트워크



### 구 연 설

- 1964년 청주대 상학과
- 1975년 성균관대 대학원 전자  
자료처리학과
- 1981년 동국대 대학원 통계학  
과(이학석사)
- 1988년 광운대학교 대학원 전  
자계산학과(이학박사)

1979년 이후 충북대전자계산소장, 한국정보과학회 부  
회장 역임

현재 충북대 컴퓨터과학과 교수

관심분야: 소프트웨어공학, 정보통신 등