□기술해설□

차세대 지능망 서비스 상호작용

한국전자통신연구소 최정훈* · 김현수**
한국과학기술원 권용래*

목 차

1. 서 론
2. 상호작용의 개념
 2.1 상호작용의 정의
 2.2 상호작용의 사례
 2.3 상호작용의 복잡성
3. 상호작용의 연구 방향
4. 국내외 연구 현황
 4.1 ITU-T
 4.2 ATR의 방법
 4.3 ROSA
 4.4 페트리 네트 기반 접근 방법 (Petri-Nets based approaches)
 4.5 기 타
5. 결 론

1. 서 론

통신 기술의 발달에 따라 다양한 통신 서비스가 제공되고 동일한 가입자가 여러개의 통신 서비스에 동시에 가입하는 상황이 일반화됨에 따라, 서비스 상호작용이 발생할 가능성 또한 높아 지게 되었다.

이러한 상호작용은 서로 연동하는 서비스의 종류에 따라 긍정적인 결과가 나올 수 있으나 통상적으로 사용자에게 부정적인 결과를 나올 가능성이 많으며, 우리가 예상하지 못한 심각한 결과를 야기할 수도 있다. 부정적인 결과의 예로써 서비스 상호작용의 영향으로 하나의 서비스가 다른 서비스를 오버라이드함으로써 예상하지 못한 서비스가 수행되거나, 과금에 오류가 발생할 수 있는 결과를 낳을 수도 있다.

서비스 상호작용 문제는 통신 서비스의 신속한 도입에 심각한 걸림돌로 등장하고 있다. 특히 지능망에서의 서비스 상호작용은 새로이 도입될 지능망 서비스가 기존의 지능망 서비스에만 영향을 미치는 것이 아니라 기존의 교환기 기반 서비스에도 문제를 발생시키므로 지능망 서비스 개발 및 도입에 있어서 중요한 문제이다.

최근의 통신 산업계의 변화는 상호작용 문제를 더욱 복잡하게 만들고 있는 데, 중요한 변화들을 살펴보면 다음과 같다. 첫째, 서비스 생성 과정은 더 이상 단일 기관이나 조직에 의해 지배를 받지 않는다. 둘째, 통신망은 이기종 장치나 다양한 공급자에 의한 소프트웨어를 지원해야 한다. 세째, 지능망 플랫폼은 운용 회사, 관련 공급자, 정보 제공자들에 의해 고객을 위한 맞춤형 서비스들을 신속하게 도입할 수 있는 구조로 진화하고 있다.

상호작용을 해결하기 위해서 기존에는 수작업에 의한 서비스-서비스간 분석 방법을 주로 이용해 왔으나, 수많은 서비스와 다양한 장치들이 연동해야 하는 환경에서는 더이상 적합하지 않다. 서비스 상호작용을 관리할 수 있는 체계적이고 자동화할 수 있는 새로운 기술들이 필

*종신회원
**정 회 원

요하다. 따라서 다양한 방법들이 제안되고 있으나 아직은 초기 단계이며, 제안된 연구 결과들에 대한 체계적인 분류 및 비교 분석의 필요성이 제시되고 있는 실정이다.

본 논문에서는 최근 문제의 심각성과 중요성이 인식되어 연구가 활발히 진행되기 시작한 서비스 상호작용에 대한 개념 및 연구 방향, 그리고 국내외 연구 동향 등을 살펴본다.

2. 상호작용의 개념

2.1 상호작용의 정의

서비스 특징(feature)은 해당 서비스의 성격을 결정지울 수 있는 중요한 기본 단위가 되는 것으로서, 하나의 서비스는 한개 이상의 서비스 특징들의 조합으로 이루어진다. 그런데 동일한 가입자가 동시에 여러개의 서비스를 제공받을 수 있으므로 하나 이상의 서비스/서비스 특징들이 동시에 제공하는 상황이 발생한다. 또한 지능망에서는 여러 명의 사용자들이 공존하므로 실제 망 내에서 동시에 활성화되는 서비스 특징들은 매우 다양하다.

이와 같이 동시에 활성화되는 서비스 간에는 기대하지 않았던 현상이 발생할 수 있으며 이를 서비스 상호작용이라 한다. 서비스는 서비스 특징들로 구성되어 있으므로 상호작용은 결국 특정 서비스특징이 다른 특징에 의해 예측하지 못하거나 요구하지 않은 방향으로 수행되는 경우를 의미하게 된다. 따라서 서비스 상호작용은 문헌에 따라서 서비스특징 상호작용(feature interaction)으로 불리워지고 있다[1, 2, 3]. 본 논문에서는 이들을 구별하지 않고 사용하기로 한다.

2.2 상호작용의 사례

서비스 상호작용은 매우 광범위한 상황에서 발생할 수 있다. 먼저 하나의 서비스에 연관되는 서로 다른 특징들간에 상호작용이 일어날 수 있다. 다음은 동일한 사용자가 두개이상의 서비스를 요청하는 경우, 그 서비스들 사이에서 발생할 수 있다. 또한 동일한 사용자의 서비스에만 국한하지 않고, 망내의 여러 서비스가 활성화되었을 때 이용하게 되는 발신선이나 터미널 등의 망 자원과 연관하여 상호작용이 발생할 수 있다. 이 절에서는 서비스간의 상호작용이 발생하는 몇가지 예를 살펴본다[3, 4].

가. OCS(Originating Call Screening) 서비스와 CF(Call Forwarding) 서비스

OCS 서비스는 지정한 전화번호와 일치하는 호를 차단하는 서비스이며, CF 서비스는 다이얼한 번호와는 다른 라인으로 호를 전환하는 서비스이다. 가입자 A가 OCS 서비스에 가입하여 C란 번호를 차단하고자 한다. A가 B에게 전화를 걸었을 때, 만약 가입자 B가 CF 서비스에 가입하여 C로 전환해 놓았다면 문제가 발생한다.

나. TCS(Terminating Call Screening) 서비스와 ACB(Automatic Call Back) 서비스

TCS 서비스는 차단 리스트에 있는 인입호(incoming call)를 차단하는 서비스이며, ACB는 최근 인입호로 회신해 주는 서비스이다. 만약 두 서비스에 동시에 가입해 있는 가입자에게 어떤 호가 들어왔다고 하자. 그런데 이 호는 TCS 서비스의 차단 리스트에도 들어 있으며, ACB도 적용가능한 상황이라면 문제가 발생한다.

다. AA(Automatic Alarm) 서비스와 CFN(Call Forwarding, No answer) 서비스

두 가지 서비스의 명세(specification)는 분명하다. AA 서비스는 아침에 원하는 시간에 일어나기 위해 링을 울리도록 하는 서비스이다. 사용자는 자신의 전화기를 이용하여 일어나기 원하는 시간을 다이얼하면 지정한 시간에 링이 울린다. CFN 서비스는 누군가에게서 전화가 걸려 왔을 때, 특정 시간동안 응답이 없으면 사용자가 서비스 구동시 지정해놓은 전화번호로 호가 전환된다. 그런데 이 두가지 서비스는 망에서 제공하는 데에 기술적인 문제는 없으나 사회적인(social) 문제가 있다. 만약 두가지 서비스를 사용하는 데에 아무런 제약이 없다면 자동 알람은 호 전환되어 엉뚱한 사람을 깨우게 되는 문제가 발생한다(서비스 사용자의 고의적인, 혹은 무의식적으로). 따라서 망 운용자는 원치않는 사회적 문제를 야기시키지 않기

위해 이 두 서비스의 조합은 금지시킨다. 혹은 AA 서비스는 항상 신청 전화에서만 동작되고 다른 번호로는 호 전환이 되지 않게 제한해버리는 방법도 있을 수 있다.

라. 연속적으로 CF(call forwarding) 서비스가 적용되는 경우

CF 서비스는 문제없이 동작할 수 있으나 사용자가 주의하지 않으면 무한 루프를 발생할 수 있다. 예를 들어 A가 B에게 호 전환하고, B는 C에게 호 전환 하고, C는 다시 A에게 호 전환하는 경우에 무한 루프가 발생한다. 이것이 망 운용자가 이러한 서비스의 사용을 제한해야 하는 이유이다. 가장 일반적인 제한 방법은 상황을 복잡하게 할 위험이 있는 어떠한 서비스의 조합도 금지시키는 것이다. 망에서 루프를 피하기 위하여 통상적으로 사용하고 있는 제한 조치는 한 호에 대해서 연속적인 호 전환이 두번만 허용되도록 제한해 버리는 것이다.

그러나 이러한 제한 조치는 서비스의 유용성을 떨어뜨리고 결과적으로 서비스를 사용하는 가입자에게 불편을 주는 방법이므로 궁극적인 해결책이 아니다. 따라서 이러한 서비스의 제공은 교환기에서 제공되는 서비스로 구현하기 보다는 지능망에서 SCP(Service Control Point)를 이용하여 서비스를 제공해야 하는 이유를 설명하기에 좋은 예가 되고 있다. 즉 지능망을 기반으로 서비스를 제공하면, SCP가 망에서 활성화되는 서비스의 전체 상황을 알 수 있으므로 사용자에 의해 유발되는 무한 루프의 경우를 미리 피할 수 있으며, 사용자는 무한 루프가 되지 않는 한 횟수에 제한없이 호 전환 서비스를 사용할 수 있게 할 수 있다. 물론 이러한 경우에 무한 루프를 검사하는 상호작용 메카니즘이 SCP에 존재해야 한다.

2.3 상호작용의 복잡성

지능망의 발전과 더불어 서비스 및 서비스 변이(service variation)[1]에 대한 수가 증가하면서 서비스 상호작용에 대한 관심도 증가하고 있다. 지능망이 도입되기 이전의 상황인, 보통 하나의 망이 20개보다 적은 서비스를 보유하고 있을 때에도 서비스 사이의 연동에 대한 가능한 조합을 하나씩 시험해 보는 일은 매우 복잡한 작업이었다. 왜냐하면, 모든 서비스가 모든 나머지 서비스에 대해 연동 시험을 해야 하므로, 예를 들어 20개 서비스에 대하여 두 서비스 사이를 고려하면 $(20\times 20)/2 = 200$개 정도의 연동 시험을 해야 하기 때문이다. 물론 완벽한 시험을 위해서는 두개보다 많은 서비스들이 함께 연동하는 상황을 고려해야 한다.

지능망이 도입된다면 고려해야 할 서비스 수가 100에서 500개 정도 이상이 될 것으로 예상된다. 여기에다 서비스 변이들도 시험 대상이 되어야 한다. 따라서 전통적인 수작업의 방법으로 서비스 연동 시험을 하기에는 시험해야 할 서비스 조합 수가 너무 많아지게 된다.

상황을 더 복잡하게 만드는 것은, 지금까지는 모든 서비스가 하나의 망에 존재하는 상황만을 고려한 것이다. 가입자에 대해서 서비스 연동의 완벽한 제어를 보장하기 위해서는 서로 다른 망으로부터 가입자들을 연결해주는 서비스들을 고려해야 한다. 또 다른 차원의 문제로서 동일한 서비스에 대해서 구현 방법이 서로 다른 서비스인 서비스 다이얼렉트(service dialect)에 대한 문제도 남아 있다. 두개의 동일한 서비스가 서로 다른 방법으로 구현되었다면 가입자에 대해서는 정확히 동일한 식으로 동작하지만, 다른 서비스와의 연동시에는 예측하지 않은 방향으로 동작할 수 있다.

이상에서 살펴본 바와 같이, 가입자가 망에서 경험하게 될 수도 있는 모든 상상할 수 있는 서비스 연동 시나리오를 시험한다는 것은 불가능하다. 왜냐하면 엄청난 수의 조합을 시험해야 하기 때문이다. 따라서 우리는 이러한 문제를 해결하기 위한 좀 더 체계적인 방법을 찾아야 할 것이다[4].

3. 상호작용의 연구 방향

서비스 상호작용 연구의 궁극적인 목표는 상호작용이 더이상 문제가 되지 않게 하는 것이

[1] 일반적으로 제공되는 서비스에 고객의 요구에 따라 부가가치 특징(value added feature)들을 추가하여 맞춤형 서비스로 제공하는 형태.

다. 즉 이용자는 그들의 서비스가 기대하지 않거나 바라지 않는 방향으로 진행되는 것을 경험하지 않는 것이며, 서비스나 망 제공자는 이것을 합당한 비용으로 보장할 수 있게 하는 것이다.

그러나 서비스 상호작용은 문제 영역이 광범위하고 매우 복잡하므로 당분간은 어떤 하나의 해답만으로 문제를 전부 해결하기는 어려울 것으로 보인다. 따라서 현재 제시되고 있는 방법들을 분석하여 해당 플랫폼에 맞게 적절히 조합하는 것이 보다 현실적인 대안이다. 서비스 상호작용은 이 문제에 어떻게 접근하느냐에 따라 "회피(avoidance)", "발견(detection)", "해결(resolution)"의 세 가지 부류로 나눌 수 있다[5].

가. 회 피

상호작용을 회피하려는 접근방법은 구현되는 서비스들의 상호작용이 내부적으로 적게 발생하도록 유도하는 서비스 플랫폼과 서비스개발환경을 개발하는 것이다. 예로서는 통신 시스템에 개방형 분산 컴퓨팅 플랫폼을 도입하여 상호작용을 "분산시스템의 내부적인 문제"로 취급하거나, 상호작용을 방지하기위하여 현재의 신호 프로토콜을 확장한 새로운 프로토콜을 정의하는 방법 등이다. 한편 서비스 상호작용을 예방하기 위하여 서비스 생성이나 서비스 관리시의 가이드라인을 제시해주는 방법등도 이 부류에 속한다.

나. 발 견

현재의 통신 시스템이나 서비스 생성 과정에서 상호작용 회피 방법만으로는 현실적으로 모든 상호작용을 예방하기는 어려우므로 잠재하는 상호작용을 발견하는 것이 필요하다. 상호작용의 발견은 오프라인과 온라인의 경우로 나눌 수 있다. 오프라인의 경우는 서비스 생성 과정의 모든 시점에서 해당 서비스의 상호작용의 원인이 될 수 있는 서비스 특징들을 상세히 기술함으로써, 서비스 생성 과정의 단계마다 상호작용을 검사하는 것이다. 오프라인의 경우는 서비스가 망에 도입되어 운용되기 전에 상호작용을 발견해낼 수 있으므로 서비스 상호작용 연구에 있어서 우선적이고 중요한 분야가 된다.

한편, 수많은 서비스들이 개입되는 경우에는 상세히 기술된 서비스 특징들이 매우 방대하고 복잡해 지므로 모든 상호작용을 발견해내는 것도 역시 어려운 문제가 된다. 이러한 경우에는 망에서의 서비스 제공 도중의 특정한 상황에서 관련된 서비스 특징들의 인스턴스들을 대상으로 상호작용을 검사하는 온라인 발견 방법이 이용된다.

다. 해 결

일단 상호작용이 발견되면 해당 상호작용이 결과적으로 나쁜 영향을 미치는 것인지, 어떤 방법으로 해결할 것인지 등을 결정해야 한다. "발견" 방법과 같이 "해결" 방법도 오프라인과 온라인으로 나눌 수 있다. 오프라인 해결 방법은 수작업을 통하여 재설계하고 재구현하는 것이다. 온라인 해결방법은 운용중 상호작용이 발생하게 되었을 때 상호작용을 해결하는 것이다. 온라인 해결 방법의 예로서는 지능망의 상호작용 관리자(feature interaction manager) 혹은 이벤트-기반 해결 메카니즘(event-based resolution mechanism) 등이 있다.

위와 같이 상호작용의 세가지 부류를 정리해 보면 그림 1과 같다.

현재 상호작용의 오프라인 발견 방법에 관한 연구가 가장 활발히 이루어지고 있으나, 오프라인 해결 방법에 관한 체계적인 연구는 거의 없는 상황이다. 또한 온라인 발견 방법에 관한 연구는 드문 반면에, 온라인 해결 방법에 관한 연구가 활발히 진행중이다.

상호 작용의 연구가 이러한 방향으로 진행되는 이유는 다음과 같이 분석해볼 수 있다. 가장 연구가 활발한 오프라인 발견 방법은 새로운 서비스 설계시부터 정보 모델링, 인공지능, FDT (Formal Description Techniques)등의 기술을 응용해 체계적 혹은 자동적으로 상호작용을 미리 발견해낼 수 있기 때문이다. 또한 온라인 해결 방법은 지능망 시스템, 특히 SSP(Service Switching Point) 및 SCP(Service Control Point)가 자동적으로 실시간에 상호작용을 해결해주어야 하는 요구에 의해 연구 및 개발이 필수적인 분야이기 때문이다.

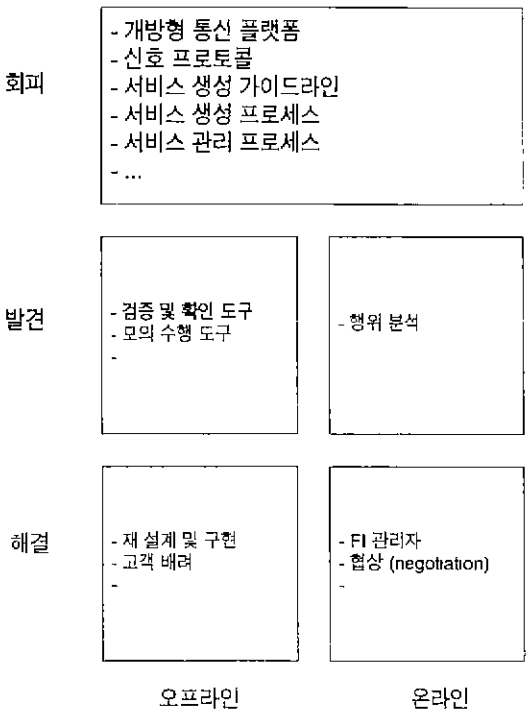


그림 1 상호작용 연구에 관한 접근 방법의 분류

반면 오프라인 해결 방법은 일반적으로 상당한 지식을 가진 전문가에 의존하여 이루어지는 과정으로, 해결 방법을 체계화 내지는 자동화하기가 어려운 분야이다. 온라인 발견부분은 주로 특정 상황에서의 데이타에 관련한 상호 작용을 다루고 있으므로 비교적 단순한 연구 분야로 볼 수 있다. 따라서 온라인 발견 방법은 통상적으로 온라인 해결 방법 연구의 일부분으로 간주된다.

4. 국내외 연구 현황

4.1 ITU-T

ITU-T의 IN CS(Intelligent Network Capability Set)에서는 정적(static) 및 동적(dynamic) 메카니즘의 두가지로 나누어 상호작용을 취급한다. 정적 메카니즘이란 서비스의 설계 단계와 제공 단계에서 상호작용을 식별하는 것이다. 이러한 메카니즘은 서비스 제공자가 특정 규칙들을 사용하여 서비스 특징 사이에 발생할 수 있는 상호작용을 예견할 것을 요구한다.

동적 메카니즘은 서비스 상호작용에 대한 결정이 수행시 결정되는 것으로 미리 예측된 상황과 예측되지 않은 상황으로 나누어진다. 예측된 상황에서는 서비스 설계자나 제공자에의해 예측된 상호작용에 대한 수행중의 해결책을 포함한다. 예측되지 않은 상황은 서비스 설계자나 제공자가 미처 예측하지 못했거나 불가능했던 상호작용에 관한 것으로써, 원칙적이고 강제적인 해결책이 필요하다.

ITU-T의 상호작용 연구는 IN CS-1에서는 거의 연구가 이루어지지 않았으며, IN CS-2에서 서비스 평면(service plane)을 중심으로 하여 연구가 진행중이나 아직까지 구체적인 결과가 발표되지는 않고 있다[6, 7].

IN CS-2에서는 상호작용을 서비스 처리중의 상호작용과 서비스 관리상의 상호작용으로 나누고 있다. 서비스 처리중의 상호작용은 주로 실시간 서비스 수행중의 상호작용에 촛점이 맞추어져 있으며 호의 제어와 관계가 있다. 비지능망 서비스/서비스 특징과 지능망 서비스/서비스 특징과의 상호작용을 위해 IN-SSM(IN-Switching State Model)에 매핑시키는 연구가 언급되고 있다. 서비스 관리에 관한 상호작용 연구는 서비스의 생성, 도입, 가입 단계로 나누어 각 단계에서의 상호작용을 연구한다.

4.2 ATR의 방법

일본의 ATR에서는 STR(State Transition Rule)을 이용하여 서비스 특징을 규칙의 형태 즉, 전상태(previous state), 사건(event), 후상태(next state)로 기술하고 STR로 기술된 서비스들 간에 상호작용을 검사하는 방법을 제안하였다[8]. 각각의 상태는 전이 기본 요소(transition primitives)와 메모리 기본 요소(memory primitives)들로 구성된다. 전이 기본 요소는 시간에 따라 변하는 요소인데 반해 메모리 기본 요소는 서비스 특징의 성질을 나타내는 요소로 서비스 실행 동안 변하지 않는다. CFV(Call Forwarding Variable)와 CW(Call Waiting)를 STR로 명세한 예는 표 1과 같다.

여기서 굵게 표시된 부분은 사건을 나타내며, 사건이 일어나기 전의 조건들이 전상태, 사건이

표 1 CFV와 CW의 STR 명세 예

서비스 이름	STR 기술	의 미
CFV	dialtone(A), path(B, D), m-cfv(B, C), idle(C) dial(A, B) . ringback(A, C), ringing(C, A) path(B, D), m-cfv(B, C)	B에 도달하는 호는 C로 호전이가 일어나고(m-cfv(B, C)), B와 D는 서로 통화중(path(B, D))이고 C는 유휴상태이고 A가 전화를 걸기위해 수화기를 든 상태일 때, 만약 A가 B에게 전화를 걸면, A와 C사이의 호 연결이 이루어진다.
CW	dialtone(A), path(B, D), m-cw(B) dial(A, B) ringback(A, B), cw-ringing(B, A) path(B, D), m-cw(B)	B가 CW서비스에 가입되어 있고(m-cw(B)) 현재 D와 통화중(path(B, D))일 경우, 만약 A가 B에게 전화하면(dial(A, B)), A와 B사이에 호 시도가 이루어지고 B의 경우 cw-ringing이 나타난다.

일어난 후의 조건들이 후상태이다. 또한 dialtone(A)와 같은 형태는 전이 기본 요소이고 m-cfv(A, B)와 같은 형태는 메모리 기본 요소이다.

상호작용의 유형은 비결정적 상태 전이(nondeterministic state transitions)와 적당한 상태 전이의 부족(lack of appropriate state transitions)으로 크게 나누어진다. 비결정적 상태전이는 하나의 입력 사건에 대해서 여러 개의 상태 전이가 존재하는 경우를 나타내고 적당한 상태 전이의 부족은 하나의 입력 사건에 대해 적당한 상태 전이가 존재하지 않는 경우를 일컫는다.

앞서 기술한 CFV와 CW간에는 비결정적 상태 전이 상호작용이 발생한다. 만일 CFV와 CW의 가입자 B가 D와 연결되어 있고, B의 CFV 서비스가 C로 호전이 되어있을 때, A가 B에게 전화를 걸면, B의 CW 규칙과 CFV 규칙이 동시에 만족되므로 비결정적인 상황이 발생하게 된다.

상호작용은 서비스 규칙들 사이의 관계를 파악함으로써 알 수 있다. 하나의 입력 사건에 의해서 동시에 활성화되는 두개의 서비스 I와 J에 대해서 R_i와 R_j를 서로 각각 두서비스에 관련된 규칙이라 하고, P_i와 P_j를 각각 R_i와 R_j의 규칙들의 전조건들이라 하자. 이때 두 전조건 P_i, P_j 사이에는 다음과 같은 관계가 존재한다.

(1) $(P_i \subset P_j \wedge P_i \neq P_j)$ 또는 $(P_j \subset P_i \wedge P_j \neq P_j)$
(2) $(P_i \not\subset P_j \wedge P_j \not\subset P_i \wedge P_i \cap P_j \neq \phi)$
또는 $(P_i \cap P_j = \phi)$

(1)의 경우, P_i가 P_j의 부분집합일 때, 하나의 입력 사건이 발생하면 P_i가 P_j의 부분집합이므로 R_j를 항상 우선적으로 실행시킬 수 있다. 그러면 R_i는 R_j가 적용되지 않을 때 적용되므로 R_i와 R_j 사이에는 아무런 충돌이 발생하지 않는다. P_j가 P_i의 부분집합일 때도 동일한 결과가 된다. (2)의 경우, 두 서비스 간에 상호작용이 일어날 가능성이 존재한다. 다시 말해, 상호작용 검사 방법은 하나의 입력 사건에 의해 동시에 활성화되는 서비스들의 전조건들이 (2)의 관계를 만족하는 지를 검사하는 것이다.

이 방법의 장점은 서비스를 기술하기 용이하다는 것이다. 즉 간단한 기본 요소들의 조합으로 하나의 서비스를 생성할 수 있다. 또한 메모리 기본 요소를 통하여 서비스의 성질을 쉽게 표현할 수 있으므로 표현력이 뛰어나다는 것이다. 그렇지만 한 서비스의 완전성 검사를 위한 정형화된 방법이 존재하지 않는다. 또한 상호작용의 검사에 있어서도 비결정성을 검사하는 데는 유리하지만 실행 순서, 무한 반복처럼 비결정성에 의하지 않은 상호작용은 검사하기 쉽지 않은 한계가 있다. ATR에서의 상호작용 검사 방법은 기본적으로 하나의 사건(event)에 의해 두 서비스가 동시에 활성화되는 경우에만 그 두 서비스간에 상호작용이 발생하는 지를 검사한다. 따라서 두 서비스가 활성화되는 시점이 다른 경우 ATR의 방법에서는 이들 간의 상호작용을 검사하지 못한다.

4.3 ROSA

ROSA는 유럽 공동체(EC)로 부터 자금 지원을 받은 프로젝트이다[9]. 여기에 참여한 기업체로는 British Telecom, France Telecom, Ericsson, PTT 등이 있다. ROSA는 개방형 서비스 구조(open services architecture)로서 통합형 광대역 통신을 위하여 개발되었다.

ROSA는 객체 지향 시스템을 기반으로 서비스를 객체(object)로 모델링한다. 즉 서비스 명세는 시스템의 실제 위치와 구현에 상관없이 사용될 수 있는 구성 요소들을 서비스 설계자에게 제공한다. 이러한 구성 요소들은 공통되는 통신 기능을 추상화한 객체이다. ROSA에서 서비스 환경은 하나의 거래자(trader)와 다수의 사용자, 사용자 중개인(user agent), 사용자 개요(user profile), 그리고 호 객체(call object) 등의 객체로 구성된다. 각각의 서비스는 객체로 모델링되고 그것들을 사용자 개요에 등록함으로써 한 사용자가 사용 가능한 서비스는 정의된다. 예를 들어 사용자 A가 CW 서비스를 가입하였을 때 이를 모델링하기 위해서는 사용자는 condition = rejected-busy라는 속성으로 사용자 개요에 CW 서비스를 등록하면 된다. 이와같이 사용자 개요에 여러 서비스가 등록되었을 때 만일 사용자 B가 A를 호출하면 A의 사용자 중개인은 A의 사용자 개요를 참조하여 호가 일어난 상황에 적절한 서비스를 활성화시킨다. 이때 서비스의 활성화는 등록된 서비스 객체의 실체화(instantiation)를 통하여 이루어 진다.

ROSA에서 서비스 상호작용은 어느 순간 활성화될 수 있는 서비스의 수가 두개 이상으로 그들 간에 충돌이 일어나는 경우로 정의된다. 이러한 충돌은 사용자 중개인 또는 사용자 개요 객체에 의해서 어떤 서비스를 활성화 시킬 것인지를 결정함으로써 해결한다. ROSA에서 각 서비스의 완전성 검사는 서비스 명세를 검사하는 정적 분석 방법으로 가능하나 서비스 객체 간의 상호작용은 정적 분석 방법보다는 시뮬레이션 등을 통한 동적 분석 방법으로 파악한다. 즉 상호작용을 서비스들이 실제 활성화되는 순간 서비스 객체들의 실체화 과정에서 파악한다. 결과적으로 상호작용의 분석은 그만큼 어려운 것이다.

4.4 페트리 네트 기반 접근 방법(Petri-Nets based approaches)

국내에서 서비스 상호작용에 관한 연구는 한국과학기술원과 한국전자통신연구소가 공동으로 연구를 진행해왔다. 이 방법은 페트리 네트를 이용하여 서비스를 모델링하고 페트리네트의 여러 분석 기법을 이용하여 상호작용을 검사한다[10, 11].

지능망 서비스는 기본 호처리 기능에서 개시되는 부가적인 기능이므로, 먼저 기본 호처리(BCP : Basic Call Process) 기능을 페트리 네트로 모델링하고 그것을 기반으로 부가적인 기능을 모델링하여 하나의 서비스를 나타낸다. BCP를 페트리 네트로 보다 정밀하게 모델링하기 위하여 BCP를 BCSM(Basic Call State Model)으로 부터 파악하고 모든 서비스 특징의 모델링은 이 BCSM을 기반으로 행하였다. 서비스 특징을 페트리 네트를 이용하여 명세한 예는 그림 2와 같다.

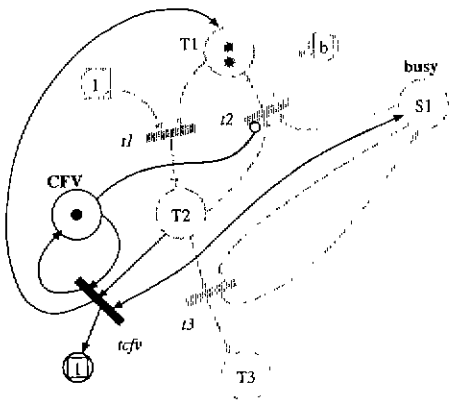


그림 2 서비스 특징을 페트리 네트를 이용하여 명세한 예

그림 2는 CFV 서비스 특징을 나타내고 있다. CFV는 서비스 가입자가 통화중일 때 다른 착신 호가 도달할 경우, 이 호를 미리 정해놓은 다른 착신측으로 전환시키는 서비스이다. 그림에서 볼 수 있듯이 만일 서비스 가입자가 통화중일 때, 즉 S1에 토큰이 있을 때, 다른 착신 호

가 도달하면, 결국 T2에 토큰이 흘러 들어오고, 따라서 T2, S1, CFV에 토큰이 존재하므로 tcfv만이 활성화(firing)될 수 있으며, 전이 tcfv의 활성화에 의해 발신상태(원 내부에 사각형이 있는 상태) 1을 통해 다른 가입자로 호가 전달된다. 이때에 T1에는 여러 개의 토큰이 존재하므로 언제든지 새로운 착신호가 도달 가능하다. 그림 2에서 연하게 나타난 부분은 BCP를 나타내며 진하게 표현된 부분은 서비스의 고유 특성을 나타낸다. 다른 서비스들도 이런 방법으로 기술된다.

서비스 상호작용을 한 서비스 가입자 내부에서의 서비스 특징들의 상호작용과 여러 서비스 가입자가 관련된 경우의 상호작용으로 파악하고 이들 각각의 경우에 대한 검사 방법을 제안하였다. 하나의 서비스가 여러 서비스 특징으로 구성되어 있거나, 또는 한 가입자가 여러 서비스를 가입하였을 경우 서비스를 구성하는 서비스 특징 간의 충돌이 발생할 수 있다. 한 서비스 가입자 내부에서의 상호작용의 검사는 두 개의 서비스 특징 네트를 하나의 BCP 모델위에 통합하여 분석한다. 여러 서비스 가입자가 관련된 경우의 상호작용 검사는 여러 서비스 특징 네트를 한 네트의 발신 상태와 같은 번호를 갖는 다른 네트의 착신 상태를 연결하는 형태로 서로 연결하여 연결된 모델을 기반으로 상호작용 검사가 이루어진다.

상호작용의 유형은 무력화, 비결정성, 무한반복 등이 있다. 무력화는 다른 특정 서비스와의 상호작용에 의해 자신의 서비스가 항상 실행되지 못하는 상황을 말한다. 즉 두 기능 중 한 가지 기능이 항상 수행되지 못하는 상황이다. 비결정성은 주어진 조건에서 두개 이상의 다른 서비스 특징이 동시에 수행될 수 있는 상황을 말한다. 이 상황은 하나의 서비스 특징이 수행되면 다른 서비스 특징이 수행될 수 없다는 점에서 무력화와 유사하지만 무력화와 달리 어느 서비스 특징이 수행될 지 알 수 없다. 무한 반복은 CF(Call Forwarding) 서비스 특징을 가입한 두 가입자들이 서로 호를 상대방에게 전이시켜 놓은 경우처럼 가입자 간에 호가 비정상적으로 무한 반복되는 경우를 말한다.

페트리 네트의 분석 방법을 이용한 상호작용 검사의 전반적인 과정은 그림 3과 같다. 우선 페트리 네트의 통합/연결 방법에 의해 하나의 네트로 나타낸다. 분석하는 네트를 간단히 하기 위해 6가지 네트의 축약 방법을 적용하여 네트를 단순화시킨다. 축약된 네트에서 페트리 네트의 여러 분석 방법을 이용하여 여러 부류의 서비스 상호작용을 검사하게 된다. 연결성 검사는 기본적인 네트의 모델링 오류뿐만 아니라 서비스 특징 간의 신호의 불일치 등을 검사한다. 도달성 트리는 비결정성 검사뿐만 아니라 무력화 검사 즉, 실행되지 않는 전이에 대한 분석에 이용된다. 네트의 교착상태, 트랩, 상태불변을 이용한 분석은 네트에서 발생하는 교착상태의 검사와 무한반복 같은 모델의 이상현상 검사에 이용되며, 또한 네트가 어떤 특성을 만족하는지 여부의 검사에도 이용된다. 여러 상호작용 검사 방법에 의해 검사를 거친 통합된 네트는 하나의 서비스 특징과 동일하게 취급되어 새로운 입력 서비스 특징으로 사용되어 진다. 이와같은 반복적인 검사 과정을 통해 여러 서비스 특징 간의 상호작용 검사가 이루어진다.

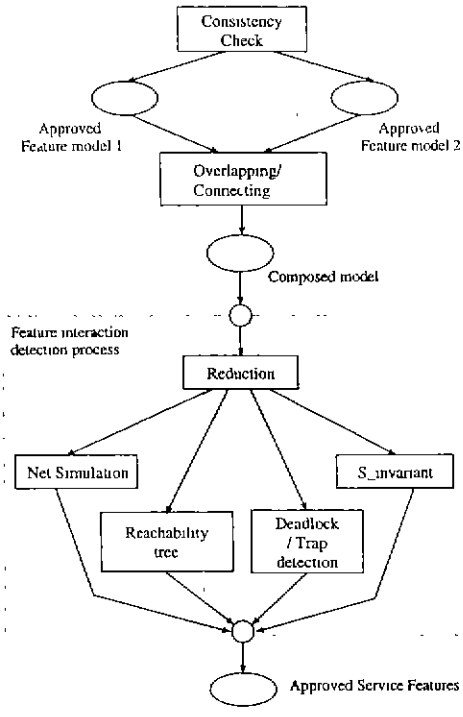


그림 3 서비스 상호작용의 발견 과정

페트리 네트는 그래픽한 특성을 가지므로 모델을 쉽게 이해할 수 있으며 또한 서비스에 나타나는 동시성을 쉽게 표현할 수 있다. 페트리 네트가 지닌 가장 큰 장점은 다양한 분석기법이 존재한다는 것이다. 따라서 이를 기반으로 서비스 자체의 완전성 검사뿐만 아니라 서비스 간의 상호작용 검사도 강력하게 수행할 수 있다. 그러나 페트리 네트가 지닌 단점으로 모델링하므로 이와같은 방식을 지능망 시스템 모델링에 그대로 사용하기에는 많은 어려움이 있다. 우리의 방법은 그런 어려움을 줄이기 위하여 각각의 서비스 특징들을 독립적으로 모델링하는 방식을 취하고 있으며 이와같이 모델링된 서비스 특징을 효과적으로 통합하기 위한 framework을 제시하고 있다.

4.5 기 타

이밖에도 상호작용에 관한 연구는 여러 사람, 여러 연구 기관에서 이루어지고 있다. Cameron[3]은 서비스 특징 간의 상호작용을 보다 구체적으로 정의하여 상호작용을 정적인 측면, 동적인 측면, 망의 고유한 측면, 분산 시스템 내부적인 측면 등으로 분류하고 각각의 경우마다 상호작용을 해결하기 위해 필요한 지식, 노력 등을 기술하여 서비스를 개발하는 여러 개발자들 간의 역할 분담, 관리 범위 등을 밝히는 구조적인 접근을 시도하였다.

Wakahara[12]는 상호작용을 찾아내기 위하여 필요한 지식을 중심으로 상호작용을 고찰하였다. 상호작용을 검사하기 위해 필요한 지식들로 각각의 서비스 특징을 명백히 기술하기 위한 지식, 서비스 특징들 사이에 내포된 관계를 나타내기 위한 지식, 이들 두가지 지식으로부터 유추된 상호작용을 검사하기 위한 지식 등을 제시하였다.

5. 결 론

통신 서비스 발달에 따라 상호작용 연구가 활발히 이루어지고 있지만 다양한 연구 결과들이 발표되고 있을 뿐, 아직까지 표준화되거나 널리 인정된 방법은 없다. 또한 제안된 방법들의 능력 비교나 통합 방법에 대한 연구도 미진한 실정이다.

서비스 상호작용 문제를 복잡하게 만드는 것은 맞춤형 서비스, 서비스 다이얼렉트, 더구나 다른 망에 있는 발 착신측 서비스의 가능한 조합과 같이 경우의 수가 너무 방대하다는 데에 있다. 따라서 이러한 방대한 상호작용 문제를 모두 해결할 수 있는 하나의 답이 존재한다고 보기는 매우 어렵다. 더구나 아직까지 단일망에서의 서비스들과 망들을 거치는 서비스들간의 상호작용들중에 발견되지 않고 숨겨져 있는 상호작용도 많이 있다.

따라서 상호작용에 관한 연구는 현재 좀 더 일반적인 서비스 상호작용 문제를 해결하려고 노력하고 있지만, 그럼에도 불구하고 여전히 유용한 해결책들이 개발되려면 오랜 시간이 필요할 것으로 예측된다.

서비스 상호작용에 관한 연구는 아직 우리나라에서는 그 중요도에 비추어 연구가 널리 이루어 지지 않고 있다. 하지만 앞으로 통신 서비스가 지능화, 다양화, 개인화되는 것은 필수적인 발전 방향이라고 볼 때, 서비스 상호작용에 대한 관심과 집중적인 연구가 반드시 필요하다고 본다.

참 고 문 헌

[1] E. jane Cameron, Nancy D. Griffeth, Yow-Jian Lin, Margaret E. Nilson, William K. Schnure, and Hugo Velthuisen, "A Feature Interaction Benchmark for IN and Beyond," *Feature Interactions in Telecommunications Systems,* L. G. Bouma and H. Velthijsen(Eds.), IOS Press, pp. 1-13, 1994.

[2] Imene Aggoun, Nathalie Caudy, Jacques Muller, and Philippe Razol, "Discussing the Complementarity of Creation Based and Execution Based Solutions to the Feature Interaction Problem." *Intelligent Network '94 Workshop,* Heidelberg, Germany, May 24-26, Section 221.4, 1994.

[3] E. J. Cameron et.al., "A Feature Interaction Benchmark for IN and Beyond," *IEEE Communications Magazine,* Vol. 31, Mar., pp. 64-69, 1993.

[4] Jan Thorner, Intelligent Networks, Artech House Inc., pp. 127-133, 1994.

[5] Wiet Bouma, Hugo Velthuijsen, "Report on the Second International Workshop on Feature Interactions in Telecommunications Software Systems," *Inatelligent Network '94 Workshop,* Heidelberg, Germany, May 24-26, Section 221.1, 1994.

[6] ITU-T Recommendation Q.1214, Geneva, Switzerland, April, pp. 36-77, 1995.

[7] ITU-T Recommendation Q.1222, Geneva, Switzerland, April, pp. 1-7, 1995.

[8] Y. Inoue, K. Takami, and T. Ohta, "Method for Supporting Detection and Elimination of Feature Interaction in a Telecommunication System," *Proc. Int'l Workshop on Feature Interactions in Telecommunications Software Systems,* pp. 61-81, 1992.

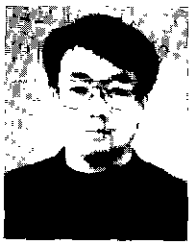
[9] J. Mierop, S. Tax, and R. Janmaat, "Service Interaction in an Object Oriented Environment," *Proc. Int'l Workshop on Feature Interactions in Telecommunications Software Systems,* pp. 133-152, 1992.

[10] 김현수, 이우진, 최정훈, 권용래, "차세대 지능망에서의 서비스 정의 및 검증," *한국정보과학회논문지,* 제 22 권, 제 2 호, pp. 326-336, 1995.

[11] J. H. Choi, H. S. Kim, W. J. Lee, and Y. R. Kwon, "A Petri-Nets based Approach for Detecting Feature Interactions in Telecommunications Services," (accepted to) *12th Int'l Conf. on Computer Communication,* 1995.

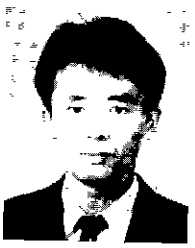
[12] Y. Wakahara et.al., "A Method for Detecting Interactions Among Services and Service Features," *Proc. Int'l Workshop on Feature Interactions in Telecommunications Software Systems,* pp. 82-94, 1992.

최 정 훈



1985 경북대학교 전자공학과 졸업(학사)
1987 한국과학기술원 전산학과 졸업(석사)
1987~현재 한국전자통신연구소 지능망서비스연구실 선임연구원
1992~현재 한국과학기술원 전산학과 박사과정
관심분야 : 차세대지능망 기술(서비스개발환경 및 서비스 상호작용), CASE, Formal Description Techniques, 분산처리환경 임

김 현 수



1988 서울대학교 계산통계학과 졸업(학사)
1991 한국과학기술원 전산학과 졸업(석사)
1995 한국과학기술원 전산학과 졸업(박사)
1995. 3~현재 한국전자통신연구소 지능망연구부 (Post Doc.)
관심분야 : Software maintenance, Reverse Engineering, Intelligent Networks (Feature Interactions), Groupware(Concurrency Control)

권 용 래

1969. 2 서울대학교 문리과대학 이학사(물리학)
1971. 2 서울대학교 대학원 이학석사(물리학)
1971. 9~1974. 9 육군사관학교 전임강사
1978. 12 미국 피츠버그대학 이학박사(물리학)
1978. 8~1983. 7 미국 Computer Sciencess Corporation 연구원
1983. 8~현재 한국과학기술원 전산학과 교수
관심분야 : Software Specification, Development Tools, Programming Environment, Human-Computer Interface, Computer Supported Cooperative Works, CASE