

# SPN에 의한 이산사건 시뮬레이션 결과 예측에 관한 연구

## A Study on the Outputs Prediction of Discrete Event Simulation with SPN

정 영 식\*

Young-Sik Jeong

### Abstract

In general, simulation and analytic method are used for real system analysis. However, to date, there has been only the theoretical works on each approach. Therefore it is required that we study on the relationship between each approaches to obtain more reliable and correct system analysis results.

In this paper, using SPN(Stochastic Petri Net) formalism, we propose the method of output prediction of the DEVS(Discrete Event system Specification) simulation. For this, we suggest a transformation algorithm which transform SPN form DEVS formalism based on the event scheduling world view and a verification algorithm for it. we then show an example to apply it to the real system, such that the Grocery Store System.

## 1. 서 론

이산사건 시뮬레이션(Discrete Event Simulation : DES)은 연속적인 시간상에서 이산적으로 발생하는 사건에 따라 실 시스템의 상태를 분석하는 과학적인 문제 해결방법이다[1,4,5,10]. DES의 중요한 장점은 동적인 실 시스템 분석에 효과적이며, 다양한 설계방법의 비교가 가능하고 의사결정을 위한 근거를 제시한다는 것 등이 있다[4,5]. 반면에 DES의 가장 큰 단점은 대규모의 자료수집을 통한 많은 횟수의 실험을 요구하여 수행시간의 낭비를 초래한다는 것이다[4,5]. 이 문제점을 해결하기 위한 방안은 첫째, 병렬 시뮬레이션 환경 구축을 통한 수행시간의 단축

으로 해결하는 방법이다. 그러나, 병렬 시뮬레이션 환경은 현재 병렬 컴퓨터상에서 구현되는 현실이므로 일반 사용자들이 접근하기에는 상당한 어려움이 있다. 두번째는 DES 결과를 예측할 수 있어 모델의 타당성을 어느 정도 보증받고 수행하는 것이다. 이 방법은 DES 모델을 구축한 후 그 모델을 기반으로 동일한 데이터에 대해 여러번 수행되는 단점을 해결하는 관점이다. 그러므로 DES 모델이 어떤 형식론(formalism)을 사용하여 구축되었는가에 따라 타당성 검사방법에 약간의 차이가 있다. 그러나, 일반적으로 모델의 타당성 검사를 위한 완전한 방법이 개발되어 있지 못한 실정이다[1,2,4,5,12].

본 연구에서는 두번째 해결책에 근본을 두는 것으로 해

\* 원광대학교 컴퓨터공학과

석적 방법을 이용하여 DES 결과 예측에 대한 방법을 제안하여 보다 신뢰할 수 있고 정확한 실 시스템 분석결과를 사용자에게 제공하는 이론적 기반을 마련하고자 한다.

지금까지 시물레이션과 해석적 방법은 그 자신의 분석 방법에 관한 연구만이 진행되었다. 이는 두 분석방법의 모델링 과정에서 이용되는 형식론 사이의 상호 관련성 연구가 부족한 실정이기 때문이다. 두 방법 사이의 상호 관련성에 대한 기존 연구는 Haas[3]에 의해 DES를 위하여 지수분포에 의한 확률변수를 갖는 SPN(Stochastic Petri Net) 모델로 표현하여 SPN 모델의 병렬성, 병행성, 동기화 특성을 이용하였다. 그러나 이는 모델링 과정에서 GSMP (Generalized Semi-Markov Process) 형식론을 이용하여 SPN으로 표현함으로써 모델링 과정이 수학적 이론에 치우쳐 모델링이 어렵다는 단점을 갖고 있다. 또한, 시물레이션 모델을 구축하는데 있어서 단지 해석적 형식모델인 GSMP와 SPN을 마치 시물레이션 형식론처럼 이용하여 완전한 형식론 사이의 연구가 아니라는 점이다. 손진곤[12]은 컴퓨터 통신 프로토콜 설계에 있어서 해석적 방법으로부터 접근하여 시물레이션 형식론을 이용하는 것으로 TPN(Timed Petri Net) 모델을 사용하여 개념적 프로토콜을 설계하고 동적표현 능력을 부가하여 DES를 위한 형식론인 DEVS(Discrete Event system Specification)로 변환하는 통합 설계 방법론을 제시하였다. 이는 해석적 방법에서 접근한 연구로써 컴퓨터 통신 프로토콜 설계에 제한하고 있다.

각 분석방법 사이의 관련성에 관한 연구부족 때문에 한 가지 방법을 선택할 경우 실 시스템 분석결과에 대한 오류가 있을 가능성이 많고, 두가지 방법 모두를 선택할 경우 동일한 실 시스템에 대해 두가지 모델을 개발하여야 하는 비경제적인 단점을 갖는다. 이와같은 이원적 성질의 분석방법은 그 결과가 개별적인 분석에 지나지 않고 상호 보완적이지 못하다는 치명적인 단점을 갖는다. 그러므로 정확하고 신뢰할 수 있는 실 시스템 분석결과를 지원할 수 있게 하고 두 접근방법이 상호 보완적인 관계를 유지하는 이론정립의 필요성이 요구된다.

본 연구에서, 이산사건 시물레이션 모델은 DEVS (Discrete Event system Specification) 형식론을 기반으로 하고 해석적 방법의 형식론은 transition 점화에 관련하는 시간변수들이 일반적인 확률분포를 따르는 SPN(Stochastic Petri Net)을 기반으로 한다. 먼저, 제 2장에서는 DEVS와

SPN 형식론에 대해 살펴보고 각각의 장단점을 파악한다. 제 3장에서는 SPN을 이용하여 DEVS 모델에 의한 시물레이션 결과를 예측하는 전체적인 관점에서의 방법을 소개한다. 제 4장에서는 DEVS에서 SPN으로의 구조적 모델 변환 알고리즘과 모델변환에 대한 검증 알고리즘을 제안한다. 제 5장에서는 Grocery Store System을 적용하여 실제적으로 SPN으로 DEVS 시물레이션 결과를 예측한다. 제 6장은 본 연구의 결론 및 추후 연구방향을 제시한다.

## 2. DEVS와 SPN 형식론

### 2.1 DEVS

DEVS 형식론은 연속적인 시간상에서 이산적으로 발생하는 사건들에 대하여 시스템의 행위를 측정하는 것으로 다음과 같다[10].

#### [정의 1] DEVS 형식론

DEVS는 입력집합  $X$ , 출력집합  $Y$ , 상태집합  $S$ , 시간진행함수  $t_a$ , 외부상태전이함수  $\delta_{ext}$ , 내부상태전이함수  $\delta_{int}$ , 출력함수  $\lambda$ 로 구성된다.

$$COM = \langle X, Y, S, t_a, \delta_{ext}, \delta_{int}, \lambda \rangle$$

여기서,  $t_a : S \rightarrow R^+_{0,\infty}$

$$s \mapsto t_a(s)$$

단,  $R^+_{0,\infty}$ 은 음수를 제외한 실수집합

$$\delta_{ext} : S \times X \rightarrow S$$

$$(s, x) \mapsto \delta_{ext}(s, x)$$

$$\delta_{int} : S \rightarrow S$$

$$s \mapsto \delta_{int}(s)$$

$$\lambda : S \rightarrow Y$$

$$s \mapsto \lambda(s)$$

입력집합은 시스템 외부에서 발생하는 사건들의 집합을 의미하고 출력집합은 출력변수들의 집합을 나타낸다. 상태집합은 상태변수들의 각 정의구역들의 곱집합을 의미하며 상태  $s(s \in S)$ 는 시간진행에 따른 시스템의 순차적인 스냅샷(sequential snapshot) 상태를 의미한다. 시간진행함수는 시스템이 외부사건을 입력받지 않는 한, 상태  $s$ 에 머물

수 있도록 허용한 시간으로 정의한다. 외부상태전이함수는 시스템 외부에서 발생한 사건에 의한 모델의 상태변화를 나타내는 함수로 정의하고, 내부상태전이함수는 외부 사건이 없는 경우 시간진행에 따라 모델의 상태변화를 설명해 주는 함수로 정의한다. 출력함수는 상태  $s$ 에서 시스템의 출력으로 정의한다.

2.2 SPN

상태/사건 모델로 간주되는 Petri Net은 순서도나 블록 다이어그램과 같이 시각적으로 의사소통을 가능하게 해주는 도식적 도구일 뿐만 아니라 견고한 이론을 갖는 수학적 모델링 도구이다[6,7,9,11]. Petri Net은 여러가지 목적에 따라 다양하게 수정, 확장되어 왔다. 본 연구에서는 transition 점화가 일반적인 확률분포함수에 의해 생성된 시간이 경과한 후 점화되고 inhibitor arc를 갖는 SPN (Stochastic Petri Net)에 대한 구조적 정의와 그래프를 소개하고 이용한다[9].

【정의 2】 SPN 형식론

inhibitor arc를 갖는 SPN은 place의 유한집합  $P$ , transition 유한집합  $T$ , 입력함수  $I$ , 출력함수  $O$ , 억제입력함수  $f$ , 점화대기시간함수  $\rho$ , 초기마킹  $\mu_0$ 로 구성된다.

$$N = \langle P, T, I, O, f, \rho, \mu_0 \rangle$$

- 여기서,  $I : T \rightarrow P$   
 $t_i \mapsto I(t_i)$  단,  $I(t_i):t_i$ 의 입력 place 집합
- $O : T \rightarrow P$   
 $t_i \mapsto O(t_i)$  단,  $O(t_i):t_i$ 의 출력 place 집합
- $f : T \rightarrow P$   
 $t_i \mapsto f(t_i)$  단,  $f(t_i):t_i$ 의 inhibitor arc에 연결된 입력 place 집합
- $\rho : T \rightarrow R^+$   
 $t_i \mapsto \rho(t_i)$  단,  $\rho(t_i):t_i$ 의 점화대기시간  
 $R^+$ :음수를 제외한 실수집합
- $\mu_0$  : 초기마킹

SPN 그래프는 place, transition, directed arc 등으로 구성되는데 각 place(원으로 표시)는 상태를 나타내고 각

transition (선분으로 표시)은 사건을 의미한다. 입력함수  $I(t_i) = \emptyset$ 가 되는  $t_i$ 를 source transition이라 정의하고 출력함수  $O(t_i) = \emptyset$ 인  $t_i$ 는 sink transition으로 정의하고 이들은 집합  $T$ 에 포함된다. place에 토큰을 위치시킴으로써 상태가 유지됨을 표현하고 place와 transition, transition과 place의 연결은 directed arc에 의해 구성된다. transition은 자신에게 입력되는 모든 place가 토큰을 가지면 점화가능한 transition으로 분류한다. transition 자신에게 입력되는 place가 inhibitor arc에 의해 연결될 경우 입력되는 place가 토큰을 가지고 있지 않을 때 점화가능한 것으로 분류한다. transition이 점화되면 각 출력 place에 토큰을 하나씩 첨가한다.

점화대기시간함수  $\rho$ 에 의해 transition  $t_i$ 에 부여된 실수 값은  $t_i$ 가 점화가능한 시점에서 점화될 때까지의 대기시간으로 정의하고 transition 점화는 원자적 운영(atomic operation)이다.

SPN에서 transition  $t_i (\in T)$ 가 실제 점화되는 조건은 점화가능한 transition이어야 하고 점화대기시간이 경과해야 한다. SPN에서 place에 분산된 토큰의 분포를 마킹  $\mu$ 라고 하고 마킹  $\mu$ 로부터 도달가능한 모든 마킹들을 고려할 때, SPN 구조  $N$ 과 마킹  $\mu$ 에 대한 도달성 집합  $R(N)$ 은 다음과 같이 정의한다[9].

【정의 3】 도달성 집합

마킹  $\mu$ 를 가진 SPN에서 도달성 집합(reachability set)은 다음과 같다.

- ①  $\mu \in R(N)$
- ② if  $\mu' \in R(N)$  & 임의의 transition  $t_i$ 에 대하여  $\mu'' = \delta(\mu', t_i)$  이면  $\mu'' \in R(N)$

여기서,  $\mu'' = \delta(\mu', t_i)$ :마킹  $\mu''$ 는 마킹  $\mu$ 로부터 즉시 도달가능

【정의 4】 도달성 문제

도달성 문제(reachability problem)는 마킹  $\mu, \mu'$ 를 가진 SPN 구조  $N$ 이 주어지면  $\mu' \in R(N)$ 임을 검사하는 것이다.

단,  $R(N)$ 은 SPN 구조  $N$ 에 대한 마킹  $\mu$ 로부터의 도달성 집합

【정의 5】 생존성

R(N)에 속하는 모든 마킹에 대하여 transition  $t_i$ 가 잠재적으로 점화가능하면 SPN 구조 N에 포함되는 transition 이 마킹  $\mu$ 에서 live하다.

도달성, 생존성[7,9]등의 SPN 속성과 도달성 집합 R(N)에 의해 개별적인 사건발생에 따른 시스템의 상태변화 및 도달가능한 상태가 어떤 것들이 존재하는지를 파악한다.

### 2.3 장단점

DEVS는 전체상태집합  $Q((s,e) | s \in S, 0 \leq e \leq ta(s))$ (e: 경과시간)와 시간진행함수를 이용하여 시스템에 대한 동적인 표현능력을 갖고 연속적인 시간상에서 이산적으로 발생하는 사건에 따라 시스템의 행위를 추경함으로써 보다 현실적인 모델링을 가능하게 하는 장점을 갖는다. 또한, DEVS 형식론에 의한 모델은 모듈적, 계층적 특성을 가지며, 집합론에 근거한 수학적 형식구조를 제공하는 것이 장점이다[10,13,14].

한편, DEVS는 그 구조의 수학적 특성 때문에 모델설계자가 이해하기 어렵고, 모델구축을 위해서는 상당한 지식과 경험의 축척을 필요로 한다. 또한 DEVS 모델은 해석적 분석능력을 가지지 못해 개별적인 사건발생에 따른 시스템 상태파악을 손쉽게 할 수 없고 그 모델의 타당성 검사를 위한 일반적인 도구가 개발되지 못하여 모델의 타당성 보장에 문제를 야기하며, DEVS의 보편적 사용을 제약하는 단점을 갖는다[10,12,14].

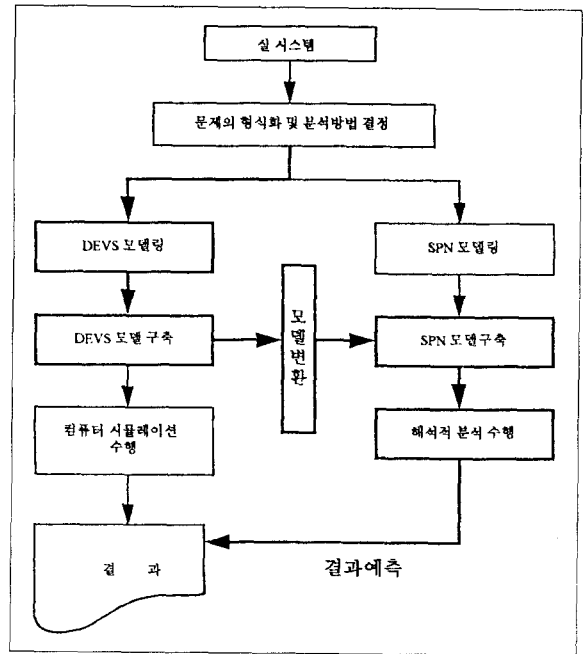
SPN의 장점은 시스템을 표현하는 도식적 모델링 도구로서 수학적인 구조를 함께 제공하여 모델의 모호성을 배제한다는 것과 시스템의 병행적, 비결정적, 병렬적, 분산적, 동기화 특성을 모두 표현한다. 해석적 방법에 적용하기 위한 이론적인 정립이 견고하여 시스템 특성에 따라 도달성 트리, 영향행렬, 도달성 그래프 및 Reduction 방법 등의 여러가지 분석방법을 지원하고 있다[6,7,8,9].

어떠한 시스템에 대해서도 시스템의 행위를 체계적으로 관찰하기 위해서는 계층적(hierarchical)인 관점을 제공해야 한다. 이는 정보의익과 추상화 장점을 부가하고 시스템에 모듈단위의 추가, 갱신, 삭제 등의 연산에 의해 시스템 전체에 파악시키는 경우를 배제하고 하향식 모델 설계에 유익한 점을 제공한다. 그러나 SPN 형식론에 의한 모델은 시스템 내부 상태변화의 명세를 위한 좋은 도구이지

만, 외부의 또 다른 시스템과의 연결 등이 표현되기에는 난해하고 아직까지는 일반화된 방법이 개발되지 못한 실정이다[6,7,8,9,12].

### 3. DEVS 시물레이션 결과 예측 방법

DEVS 형식론에 의한 시물레이션 결과를 SPN 형식론을 이용하여 예측하기 위한 전체적인 제어 흐름도는 <그림 1>과 같다.



<그림 1> SPN에 의한 DEVS 시물레이션 결과 예측 제어 흐름도

<그림 1>의 제어 흐름도를 기본으로 DEVS 시물레이션 결과 예측을 위한 방법은 다음과 같다.

- 【단계 1】 모델 구축 : 실 시스템에 대해 DEVS 형식론을 이용하여 분석목적에 부합하는 모델을 구축한다.
- 【단계 2】 구조적 모델변환 : 구조적 모델 변환 알고리즘에 의해 DEVS 모델을 SPN 모델로 변환한다. 4.1절에서 자세히 소개한다.
- 【단계 3】 검증 : 모델의 행위적 동등성 개념과 검증 알고

리즘에 의해 모델의 변환이 제대로 되었는가를 검사한다. 자세한 내용은 4.2절에서 논의한다.

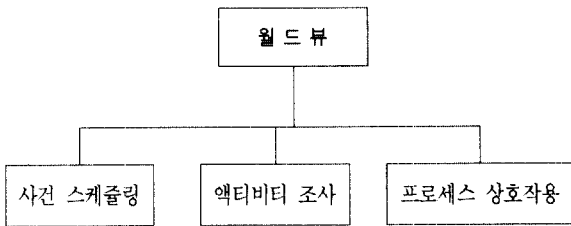
**[단계 4] 분석 :** 변환된 SPN과 DEVS 모델의 행위가 동등한가를 판단하여 2.2절에서 논의한 도달성 집합, 도달성 문제, 생존성등을 통하여 SPN 모델을 분석한다.

**[단계 5] 예측 :** DEVS와 SPN이 행위적으로 동등함을 보장하게 되므로써 SPN을 분석한 결과가 시뮬레이션 결과로 예측되어 불필요한 시뮬레이션 수행을 방지한다.

### 4. 구조적 모델변환 및 검증 알고리즘

#### 4.1 변환 알고리즘

DEVS의 상태집합  $s$ 를  $\langle s_1, s_2, \dots, s_n \rangle$ 이라 하면, DEVS는 입출력 개념을 명시하는 형식론임으로 DEVS가 SPN보다 추상적이고 포괄적인 모델을 구축하는 형식론이다. 또한, 시뮬레이션을 이산사건 관점에서 접근할 때, 모델링 월드뷰(world view)는 모델 구축방법에 따라 다음 <그림 2>와 같이 분류된다[4,10].



(그림 2) 이산사건 월드 뷰

이에 본 연구는 가장 현실적인 사건 스케줄링 월드 뷰를 고려하여 다음과 같은 다섯가지 상태변수 타입을 정의한다.

- i) R 타입 :  $s_i \in \{r_j \mid j \in N\}$
- ii) Q 타입 :  $s_i \in \{(q_1, q_2, \dots, q_m) \mid m \in N\}$
- iii) V 타입 :  $s_i \in \{v_j \mid v_j \in R^+, j \in N\}$
- iv) E 타입 :  $s_i \in \{(e_1, t_1), (e_2, t_2), \dots, (e_n, t_n) \mid n \in N\}$
- v) T 타입 :  $s_i \in \{t_j \mid t_j \in R^+, j \in N\}$

R 타입은 확률형 상태변수로서 시스템에서 발생하는 외부, 내부사건 발생을 나타내는 상태변수이다. Q 타입은 서술형 상태변수로서 엔티티가 구성요소에서 대기하는 상태를 리스트로 표현한 것이다. 엔티티의 갯수에 의미를 두고 있으며, 이를 V 타입의 실수값으로 나타내도 같은 의미를 갖고 표현 형태만 다른 것이다. V 타입은 시스템의 구성요소에 대한 특별한 의미를 실수값으로 할당하여 그 구성요소의 상태를 나타내는 상태변수이다. 또한, 시스템 구성요소의 상태에 대한 의미를 실수값으로 부여하는 것이 아니라 실수가 표현상의 의미를 가지는 상태변수도 포함한다. E 타입은 엔티티에 부여된 시간값을 나타내는 리스트 형태의 상태변수로서 시간값  $t_i$ 들은 일반적인 확률분포 함수에 의해 생성된다. T 타입은 잔여시간(time left) 상태변수로서 시스템 구성요소의 역할 수행을 위한 시간 표현 상태변수이고 일반적인 확률분포 함수에 의해 생성되는 값을 갖는다.

DEVS에서 SPN으로의 모델변환은 모델을 구조적인 관점에서 변환을 시도하고 구조적 모델변환을 위한 이론정립 및 증명은 참고문헌 [14]에 소개하고 있다.

구조적 모델변환 과정은 SPN의 모델구조 N에 대한 구성요소를 DEVS의 COM으로 부터 생성하는 데 모두 여섯단계를 걸쳐 수행된다.

#### Algorithm 1 : Transformation Algorithm

- Step 1 : Determine a place set of SPN from a sequential state set ;
- Step 2 : Compose an influence diagram for state variables ;
- Step 3 : Determine a transition set from internal, external events ;
- Step 4 : Determine input place sets, output place sets, and a place set by connecting inhibitor arcs from Step 2, & Step 3 ;
- Step 5 : Produce a probability distribution function for each transition ;
- Step 6 : Specify an initial marking  $\mu_0$  (it is equalized to the meaning of states under DEVS at transformation time  $t$ );

먼저, SPN의 place 집합 구성을 위한 과정으로 상태집

합  $s$ 의 상태변수들중  $R$  타입의 외부사건 발생을 의미하는 것을 제외한 모든 상태변수의 갯수들은  $place$ 로 정의하고 각  $place$ 에 대한 의미를 명세한다.

상태변수 영향도는 시스템의 전체 운영과정을 통하여 외부사건 발생에 대한 피영향자가 되는 상태변수 타입을 선택하고 내부사건별 영향자와 피영향자에 해당되는 상태변수 타입을 선택한다. 시스템을 떠나는 내부사건이 발생하였을 경우 영향자가 되는 상태변수 타입과 피영향자가 되는 상태변수 타입을 결정한다.

SPN의 transition 집합 구성을 위한 과정은 상태집합  $s$ 의 구성요소중  $R$  타입의 외부사건 발생을 의미하는 것과  $E$  타입,  $T$  타입의 내부사건 발생을 의미하는 것으로 부터 구성한다. 외부사건 발생을 의미하는  $R$  타입 상태변수를 하나의 source transition으로 명세하고  $E$ ,  $T$  타입에 의해 명세되는 내부사건들은 입력  $place$ 와 출력  $place$ 을 갖는 transition으로 명세한다. sink transition은 시스템의 전체 운영과정으로 부터 시스템을 벗어나는 엔티티가 존재하면 명세하고 그렇지 않으면 명세하지 않는다.

각 transition에 대한 입력  $place$  집합을 상태변수 영향도의 영향자에 해당되는 상태변수를 입력  $place$  집합의 원소로 결정하고, 출력  $place$  집합은 피영향자에 해당되는 상태변수를 출력  $place$  집합의 원소로 결정한다. 상태변수 영향도에서  $R$  타입과  $T$  타입의 쌍이 피영향자가 되는 경우  $R$  타입의 상태변수를 inhibitor arc에 의해 연결되는  $place$ 로 구성하여 억제입력함수  $f$ 의 치역으로 생성한다.

각 transition이 점화가능한 상태에서 점화되기 위해서 대기하는 시간값을 생성하는 확률분포 함수를 결정한다. DEVS의 각 사건발생을 나타내는 확률분포 함수를 SPN으로 사상(mapping)되는 transition의 점화대기시간을 생성하는 확률분포 함수로 구성한다. sink transition에 대한 확률분포 함수는 sink transition의 입력  $place$   $p_i$ 에 대하여,  $place$   $p_i$ 를 출력  $place$ 로 간주하는 transition  $t_i$ 의 확률분포 함수와 동일하게 구성한다.

SPN의 초기마킹을 토큰의 분포에 의해 생성하는데 변환하고자 하는 시점에서 상태집합  $s$ 에 의해 나타나는 시스템 전체상태에 대한 의미를 SPN의 초기마킹으로 생성한다. 초기마킹은 일반적으로 시물레이션 초기상태이거나 임의의 시간에서 시스템에 대해 해석적 분석을 위한 초기상태가 된다.

SPN 그래프로 부터 시스템 내부 상태변화의 명세를 통

해 도식적으로 의사소통이 되고 각 transition 점화에 의한 시스템 상태변화를 도달성 집합  $R(N)$ 으로 표현하여 시스템을 분석함으로써 시물레이션 구현에 있어서 수행정도(degree)를 제시하는데 효과적이다.

## 4.2 검증 알고리즘

DEVS에서 SPN으로의 모델변환은 모델의 구조적인 관점에서 변환을 시도한 것이다. 그러므로 모델 변환시 잘못된 가능성은 1) 모델 설계자의 DEVS에 대한 잘못된 이해, 2) 1)을 바탕으로 구조적 모델변환 개념의 잘못 사용하는 경우, 3) 알고리즘 적용시 오류를 포함하는 경우 등이 있다. 그러므로 이와같은 오류들을 제거하고 타당하게 모델이 변환되었음을 지원하기 위해서는 모델변환 자체에 대한 검증 메카니즘이 요구된다. 따라서, 모델의 행위적 동등성이란 개념을 이용하여 각 사건에 대한 DEVS의 행위와 동일한 사건에 대해 DEVS에서 구조적으로 모델변환된 SPN의 행위가 동등함으로써 타당한 모델변환이 이루어짐을 알 수 있다.

SPN으로의 변환 알고리즘에 대한 타당성 검사를 위하여 다음의 모델에 대한 행위적 동등성(behaviorally equivalence)을 정의하고 이를 이용한다[13,14,15].

### [정의 6] 모델의 행위적 동등성

동일한 시스템을 모델링한 모델  $M$ 과 모델  $M'$ 의 상대표현을 각  $s_1, s_2$ 라 가정하고 이들은 의미가 같은 상태를 나타내고 있다고 하자( $s_1 \equiv s_2$ 로 표기). 이때, 의미가 같은 사건이 발생하여 시스템에 대한 상태를 모델  $M$ 과 모델  $M'$ 가 각  $s_1', s_2'$ 로 변화된 후  $s_1' \equiv s_2'$ 가 만족하면 모델  $M, M'$ 는 행위적으로 동등( $M_B \approx M'_B$ )하다.

DEVS와 SPN에서 동일한 의미의 사건 발생에 의한 시스템 상태변화를 분석하여, DEVS의 행위와 SPN의 행위가 동일할가를 파악하기 위하여 모델의 행위적 동등성을 이용한다.

### Algorithm 2 : Verification Algorithm

Step 1: Set up a state  $s(\in S)$  at the transformation time ;  
Step 2: Select event(internal or external) by time advance

function  $t_a$ (let the selected event be  $e$ ) ;

Step 3 : Specify the state  $s'(\in S)$ , after the event  $e$  occurs ;

Step 4 : Determine the marking  $\mu$  for a state  $s$  using Transformation Algorithm ;

Step 5 : Determine the marking  $\mu'$  for a state  $s'$  using Transformation Algorithm ;

Step 6 : The next marking  $\mu''$  can be found by the marking  $\mu$  for each  $p_1 \in P$ , when  $t_3$  is enable under  $\mu$  (the meaning of  $t_3$  equals to the meaning of the event  $e$ ) ;

Step 7 : if  $\mu = \mu''$  then  $DEVS_B \approx SPN_B$   
 else not( $DEVS_B \approx SPN_B$ ) ;

Step 8 : if not( $DEVS_B \approx SPN_B$ ) then  
 print a message "invalid transformation" ;

Step 9 : Stop ;

알고리즘 2는 DEVS의 시뮬레이션 초기 상태 혹은 변환하고자 하는 시점에서의 상태(s1)와 s1 상태를 기준으로 구조적 모델변환을 시도한 후, SPN의 상태(s2)를 중심으로 사건 발생에 따른 상태변화들이 동등해야 함을 보이는 것이다. 기준이 되는 상태, 즉 모델변환을 적용하기 위한 초기상태에서 모든 도달 가능한 상태(reachable state)에 반복 적용한다.

### 5. 적용사례

ENTRANCE, SHOP.AREA, CHECKOUT, EXIT를 시스템 구성요소로 갖는 Grocery Store System에 DEVS 형식론을 이용하여 모델링한 후, SPN에 의해 시뮬레이션 결과 예측 방법을 적용한 사례를 보인다. 계산대 구성요소에서는 소비자 대기기를 위한 부-구성요소(sub-component) LINE를 갖는다.

1) The Grocery Store System의 DEVS 형식론 적용

1. 외부사건 집합 :  $\{\emptyset, n_1\}$ ,  $n_1$ : 소비자 도착사건

2. 상태집합 :  $s = \langle r_1, \begin{pmatrix} (x_1, t_1) \\ (x_2, t_2) \\ \dots \\ (x_n, t_n) \end{pmatrix}, \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_m \end{pmatrix}, r_2, \sigma \rangle$

3. 시간진행함수 :  $t_a(s) = \min\{t_1, t_2, \dots, t_n, \sigma\}$   
 $t$

4. 외부상태전이함수

$$\delta_{ext}(s) = \langle \Gamma_1(r_1), \begin{pmatrix} (x_1, t_1-e) \\ (x_2, t_2-e) \\ \dots \\ (x_n, t_n-e) \\ (x, SHOPPING.TIME(x)) \end{pmatrix}, \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_m \end{pmatrix}, r_2, \sigma-e \rangle$$

단,  $e < \min\{t_1, t_2, \dots, t_n, \sigma\}$

5. 내부상태전이함수

① 서비스 시작 사건이 발생할 경우

$$\delta_{int}(s) = \langle r_1, \begin{pmatrix} (x_1, t_1-\sigma) \\ (x_2, t_2-\sigma) \\ \dots \\ (x_n, t_n-\sigma) \end{pmatrix}, \begin{pmatrix} y_2 \\ y_3 \\ \dots \\ y_m \end{pmatrix}, \Gamma_2(r_2), SERVICE.TIME(t_2) \rangle, \text{ if } m \geq 2$$

$$\delta_{int}(s) = \langle r_1, \begin{pmatrix} (x_1, t_1-\sigma) \\ (x_2, t_2-\sigma) \\ \dots \\ (x_n, t_n-\sigma) \end{pmatrix}, \emptyset, r_2, \infty \rangle, \text{ if } m \leq 1$$

② SHOP.AREA 떠남 사건이 발생할 경우

$$\delta_{int}(s) = \langle r_1, \begin{pmatrix} (x_1, t_1-t_j) \\ (x_2, t_2-t_j) \\ \dots \\ (x_{j-1}, t_{j-1}-t_j) \\ (x_{j+1}, t_{j+1}-t_j) \\ \dots \\ (x_n, t_n-t_j) \end{pmatrix}, \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_m \\ x_j \end{pmatrix}, r_2, \sigma-t_j \rangle$$

단,  $t_j = \min\{t_1, t_2, \dots, t_n\}$

③ 사건 ①과 사건 ②가 동시발생할 경우

$$\delta_{int}(s) = \langle r_1, \begin{pmatrix} (x_1, t_1-t_{\sigma})_1 \\ (x_2, t_2-t_{\sigma})_1 \\ \dots \\ (x_n, t_n-t_{\sigma})_1 \end{pmatrix}, \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_m \\ (x_1, t_1-t_{\sigma})_2 \\ (x_2, t_2-t_{\sigma})_2 \\ \dots \\ (x_n, t_n-t_{\sigma})_2 \end{pmatrix}, r_2', \sigma' \rangle$$

단,	$(x_1, e)_1$	$(x_1, e)_2$
$e > 0$	$(x_1, e)$	$\phi$
$e = 0$	$\phi$	$x_1$

$$t_* = \sigma = \min\{t_1, t_2, \dots, t_n\}$$

$$r_2' = \begin{cases} \Gamma_2(r_2) & \text{if } m \geq 2 \\ r_2 & \text{otherwise} \end{cases}$$

$$\sigma' = \begin{cases} \text{SERVICE.TIME}(r_2) & \text{if } m \geq 2 \\ \infty & \text{otherwise} \end{cases}$$

6. 출력집합 및 출력함수

$$Y = \begin{cases} \{\theta, a, b, \dots\} & \text{if OUTSIDE VIEW=BYE,BYE} \\ \{\theta, a, b, \dots\}\{a, b, \dots\} & \text{if CUSTOMER SATISFACTION} \\ & \text{=BYE,BYE,LINE} \\ \text{YES, NO} & \text{if OUT UTILIZATION=BUSY} \end{cases}$$

$$\textcircled{1} \lambda_1(s) = \begin{cases} y_1 & \text{if } \sigma = 0 \\ \phi & \text{otherwise} \end{cases} \quad \textcircled{2} \lambda_2(s) = (\lambda_1(s), y_1, \dots, y_m)$$

$$\textcircled{3} \lambda_3(s) = \begin{cases} \text{YES} & \text{if } y_1, \dots, y_m \neq \phi \\ \text{NO} & \text{if } y_1, \dots, y_m = \phi \end{cases}$$

여기서,  $r_1$ 지수분포함수( $\Gamma_1$ )에 의해 생성

$t_1$ 지수분포함수(SHOPPING.TIME)에 의해 생성  
 $\sigma$ 와 서비스시작사건:지수분포함수(SERVICE.TIME,  
 $\Gamma_2$ )에 의해 생성

2) 구조적 모델변환

place 구성

상태변수  $[(x_1, t_1), (x_2, t_2), \dots, (x_n, t_n)]$ 은 E 타입으로 place  $p_1$ 으로 명세하고 쇼핑지역에 소비자가 쇼핑시간을 확률분포함수에 의해 할당받아 쇼핑하는 소비자에 대한 리스트를 나타낸다. 상태변수  $[(y_1, y_2, \dots, y_m)]$ 은 Q 타입으로 place  $p_2$ 로 명세하고 CHECKOUT의 서비스를 받기 위하여 대기중인 소비자들의 리스트를 표현하고 있다. 상태변수  $r_2, \sigma$ 는 각 R 타입과 T 타입으로 CHECKOUT의 상태를 쌍으로 나타내고 있는데 이를 각  $p_3, p_4$ 로 명세한다.

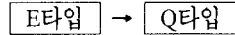
상태변수 영향도 구성

- 외부사건에 대한 영향도:SHOP.AREA 도착사건



단, E 타입 :  $[(x_1, t_1), (x_2, t_2), \dots, (x_n, t_n)]$

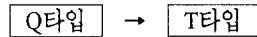
- 내부사건에 대한 영향도(SHOP.AREA 떠남사건):E 타입이 영향자인 경우



단, E 타입 :  $[(x_1, t_1), (x_2, t_2), \dots, (x_n, t_n)]$

Q 타입 :  $[(y_1, y_2, \dots, y_m)]$

- 내부사건에 대한 영향도(서비스 시작사건):Q 타입이 영향자인 경우



단, Q 타입 :  $[(y_1, y_2, \dots, y_m)]$

R 타입 :  $r_2, T$  타입 :  $\sigma$

- 내부사건에 대한 영향도(CHECKOUT 떠남사건):R,T 타입이 영향자인 경우 피영향자는 존재하지 않는다.

transition 구성

상태변수  $r_1$ 은 외부사건 발생을 나타내는 R 타입으로 transition  $t_1$ 으로 명세한다. 시간진행함수  $t_a(s) = \min\{t_1, t_2, \dots, t_n, \sigma\}$ 로부터 임의의 시간  $t_1$ 에서 최소값이 될 때, E 타입에서 소비자  $x_1$ 가 쇼핑지역을 떠나는 사건이 발생하는데 이 사건을 transition  $t_2$ 로 명세한다. 상태변수  $\sigma$ 의 값이 최소일때, 소비자  $y_1$ 이 대기상태에서 서비스를 받기 위해 LINE를 떠나 CHECKOUT에서 서비스가 시작되는 사건이 발생하는데 이 사건을 transition  $t_3$ 으로 명세한다. 서비스를 완료한 소비자는 The Grocery Store System을 떠나는데 이 사건을 transition  $t_4$ 로 명세한다. 시스템의 전체 운영과정은 The Grocery Store System에 소비자가 들어와서 쇼핑을 하고 서비스를 받기 위해 LINE에 대기하다가 자신의 차례가 오면 서비스를 받고 시스템을 빠져나간다. 그러므로 transition  $t_1$ 은 source transition으로 구성하고 transition  $t_4$ 는 sink transition으로 구성한다.

입력함수 I, 출력함수 O, 억제입력함수 f 구성

- 입력함수 I

$$I(t_1) = \phi \quad (\because t_1 \text{이 source transition})$$

$$I(t_2) = \{p_1\}, I(t_3) = \{p_2\}, I(t_4) = \{p_3, p_4\}$$

- 출력함수 O

$$O(t_1) = \{p_1\}, O(t_2) = \{p_2\}, O(t_3) = \{p_3, p_4\}$$

$$O(t_4) = \phi \quad (\because t_4 \text{이 sink transition})$$

- 억제입력함수 f



$$f(t_3) = \{p_3\}, f(t_1) = \emptyset \quad (i=1,2,4)$$

**확률분포함수 구성**

모델링과정에서 상태집합 s에 구성된 상태변수에서 확률변수  $t_i$ 과 E 타입에 표현된  $t_i$ 의 시간은 지수분포함수에 의해 생성된다. 또한, 잔여시간변수  $\sigma$ 의 시간은 지수분포함수에 의해 생성한다고 가정하였다. 그러므로 이들 사건들에 대응하는 각 transition  $t_i(1 \leq i \leq 4)$ 는 지수확률분포함수에 의해 생성된다.

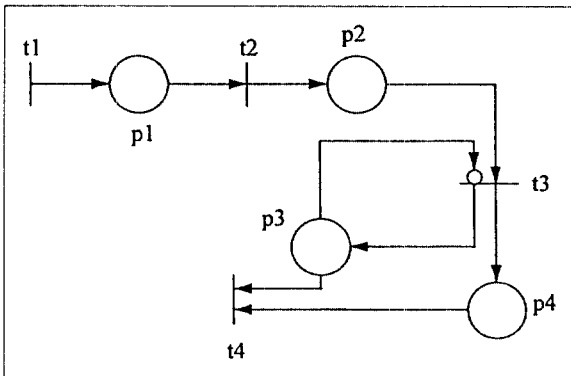
**초기마킹  $\mu_0$  구성**

모델을 변환하고자 하는 시간 t에서 The Grocery Store System의 상태가 쇼핑지역에 소비자가 없고 LINE에 대기 중인 소비자가 없으며 CHECKOUT이 idle 상태에 있다면, SPN의 초기마킹은  $\mu_0=(0,0,0,0)$ 가 된다.

3) The Grocery Store System의 변환된 SPN 모델

1.  $P = \{ p_1, p_2, p_3, p_4 \}$
2.  $T = \{ t_1, t_2, t_3, t_4 \}$
3.  $I(t_1) = \emptyset, I(t_2) = \{p_1\}, I(t_3) = \{p_2\}, I(t_4) = \{p_3, p_4\}$
4.  $O(t_1) = \{p_1\}, O(t_2) = \{p_2\}, O(t_3) = \{p_3, p_4\}, O(t_4) = \emptyset$
5.  $f(t_3) = \{p_3\}, f(t_1) = \emptyset \quad (i=1,2,4)$
6. transition 시간 확률 분포 함수  
 $t_1$             10        지수분포함수  
 $t_2, t_3, t_4$      $\infty$         지수분포함수  
 단,  $\infty$  : 시간이 생성되지 않은 상태
7.  $\mu_0 = (0,0,0,0)$

(a) The Grocery Store System에 대한 SPN 형식론



(b) The Grocery Store System에 대한 SPN 그래프

4) The Grocery Store System의 SPN에 의한 해석적 분석

변환된 SPN 모델에서 transition과 place의 의미는 다음과 같다.

- $t_1$  : The Grocery Store System에 소비자 도착사건
- $t_2$  : 소비자가 쇼핑지역 떠남사건
- $t_3$  : CHECKOUT의 서비스 시작사건
- $t_4$  : The Grocery Store System에서 소비자 떠남사건
- $p_1$  : 소비자 쇼핑상태
- $p_2$  : 소비자가 서비스를 받기 위한 대기상태
- $p_3, p_4$  : CHECKOUT busy상태

해석적 분석을 위하여 transition  $t_1$ 의 점화대기시간은 구간  $[0, 40]$ 를 기준으로 하고 transition  $t_2, t_3, t_4$ 의 점화대기시간은 구간  $[0, 15]$ 를 기준으로 하여 난수(random number)를 생성한다. <표 1>은 초기마킹  $\mu_0=(0,0,0,0)$ 에서 도달성 집합을 구성하는데 있어서 분석시간을 연속적인 시간상에서 180분 동안 수행한 예를 제시하고 있다.

The Grocery Store System의 SPN 모델은 transition 집합에서 언제나 잠재적으로 점화가능한 transition이 존재하여 생존성이 있으며, transition이 점화되어 생성되는 도달성 집합으로부터 시스템의 도달가능한 상태를 예측할 수 있다. 예를 들면  $\mu_1=(1,0,0,0)$ 은 쇼핑지역에 1명의 소비자가 있고 LINE에는 아무도 대기중인 소비자가 없으며 CHECKOUT은 idle 상태이다.

5) The Grocery Store System에 대한 시뮬레이션 결과 예측

① transition  $t_1$ 의 구간을  $[0, 40], [0, 30], [0, 20], [0, 10]$ 으로 설정하여  $t_1$ 이 점화하기 위한 대기시간을 생성하고 네 구간에 대해 모두 연속적인 시간상에서 180분을 기준으로 10회에 걸쳐 관찰한다. 이때, The Grocery Store에 도착하는 소비자 수에 대한 변화를 1800분 동안 관찰하면 <그림 3>의 (a)와 같이 나타난다(총 1800분 관찰).

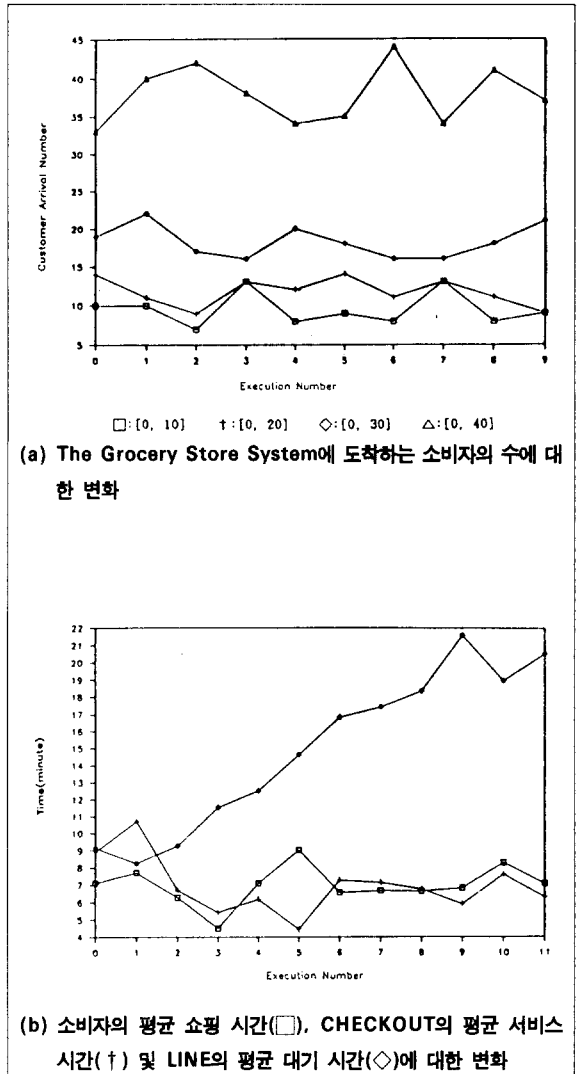
② The Grocery Store System에 대한 소비자의 평균 쇼핑 시간, CHECKOUT의 평균 서비스 시간 및 QUEUE의 평균 대기 시간은 각 구간마다 180분을 기준으로 총 2160

**〈표 1〉 The Grocery Store System에 대한 SPN에 의한 해석적 분석**

점화 transition	R(N)	점화가능 transition	점화대기시간
초기상태	$\mu_0 = (0,0,0,0)$	$t_1$	$t_1=4$
$t_1$	$\mu_1 = (1,0,0,0)$	$t_1, t_2$	$t_1=3, t_2=11$
$t_1$	$\mu_2 = (2,0,0,0)$	$t_1, t_2$	$t_1=24, t_2=8$
$t_2$	$\mu_3 = (1,1,0,0)$	$t_1, t_2, t_3$	$t_1=16, t_2=8, t_3=11$
$t_2$	$\mu_4 = (0,2,0,0)$	$t_1, t_3$	$t_1=8, t_3=3$
$t_3$	$\mu_5 = (0,1,1,1)$	$t_1, t_4$	$t_1=5, t_4=14$
$t_1$	$\mu_6 = (1,1,1,1)$	$t_1, t_2, t_4$	$t_1=1, t_2=4, t_4=9$
$t_1$	$\mu_7 = (2,1,1,1)$	$t_1, t_2, t_4$	$t_1=18, t_2=3, t_4=8$
$t_2$	$\mu_8 = (1,2,1,1)$	$t_1, t_2, t_4$	$t_1=15, t_2=13, t_4=5$
$t_4$	$\mu_9 = (1,2,0,0)$	$t_1, t_2, t_3$	$t_1=10, t_2=8, t_3=3$
$t_3$	$\mu_{10} = (1,1,1,1)$	$t_1, t_2, t_4$	$t_1=7, t_2=5, t_4=2$
$t_4$	$\mu_{11} = (1,1,0,0)$	$t_1, t_2, t_3$	$t_1=5, t_2=3, t_3=6$
$t_2$	$\mu_{12} = (0,2,0,0)$	$t_1, t_3$	$t_1=2, t_3=3$
$t_1$	$\mu_{13} = (1,2,0,0)$	$t_1, t_2, t_3$	$t_1=16, t_2=4, t_3=1$
$t_3$	$\mu_{14} = (1,1,1,1)$	$t_1, t_2, t_4$	$t_1=15, t_2=3, t_4=13$
$t_2$	$\mu_{15} = (0,2,1,1)$	$t_1, t_4$	$t_1=12, t_4=10$
$t_4$	$\mu_{16} = (0,2,0,0)$	$t_1, t_3$	$t_1=2, t_3=1$
$t_3$	$\mu_{17} = (0,1,1,1)$	$t_1, t_4$	$t_1=1, t_4=14$
$t_1$	$\mu_{18} = (1,1,1,1)$	$t_1, t_2, t_4$	$t_1=29, t_2=8, t_4=13$
$t_2$	$\mu_{19} = (0,2,1,1)$	$t_1, t_4$	$t_1=21, t_4=5$
$t_4$	$\mu_{20} = (0,2,0,0)$	$t_1, t_3$	$t_1=16, t_3=1$
$t_3$	$\mu_{21} = (0,1,1,1)$	$t_1, t_4$	$t_1=15, t_4=10$
$t_4$	$\mu_{22} = (0,1,0,0)$	$t_1, t_3$	$t_1=5, t_3=12$
$t_1$	$\mu_{23} = (1,1,0,0)$	$t_1, t_2, t_3$	$t_1=14, t_2=0, t_3=7$
$t_2$	$\mu_{24} = (0,2,0,0)$	$t_1, t_3$	$t_1=14, t_3=7$
$t_3$	$\mu_{25} = (0,1,1,1)$	$t_1, t_4$	$t_1=7, t_4=8$
$t_1$	$\mu_{26} = (1,1,1,1)$	$t_1, t_2, t_4$	$t_1=34, t_2=2, t_4=1$
$t_4$	$\mu_{27} = (1,1,0,0)$	$t_1, t_2, t_3$	$t_1=33, t_2=1, t_3=6$
$t_2$	$\mu_{28} = (0,2,0,0)$	$t_1, t_3$	$t_1=32, t_2=5$
$t_3$	$\mu_{29} = (0,1,1,1)$	$t_1, t_4$	$t_1=27, t_4=12$
$t_4$	$\mu_{30} = (0,1,0,0)$	$t_1, t_3$	$t_1=15, t_3=10$
$t_3$	$\mu_{31} = (0,0,1,1)$	$t_1, t_4$	$t_1=5, t_4=0$
$t_4$	$\mu_{32} = (0,0,0,0)$	$t_1$	$t_1=5$
$t_1$	$\mu_{33} = (1,0,0,0)$	$t_1, t_2$	$t_1=7, t_2=7$
$t_2$	$\mu_{34} = (0,1,0,0)$	$t_1, t_3$	$t_1=0, t_3=1$
$t_1$	$\mu_{35} = (1,1,0,0)$	$t_2, t_3$	$t_2=14, t_3=1$

분을 관찰하고 540분마다 transition  $t_1$ 의 구간을  $[0, 40], [0, 30], [0, 20], [0, 10]$ 으로 변화시켜 분석하였다. 소비자의 평균 쇼핑 시간은 총 관찰 시간동안 transition  $t_2$ 에 대한 점화대기시간의 총합을 점화가 발생한 총 수로 나눈 값이 되고 CHECKOUT의 평균 서비스 시간은 총 관찰 시간동안

transition  $t_4$ 에 대한 점화대기시간의 총합을 점화가 발생한 총 수로 나눈 값이 된다. QUEUE의 평균 대기 시간은 CHECKOUT이 busy 상태에서 대기중인 소비자가 있는 경우의 총 시간 진행값을 transition  $t_3$ 의 점화발생한 수로 나눈 값이다 이에 대한 각 시간변화에 대한 결과는 〈그림 3〉의 (b)와 같다.



**〈그림 3〉 The Grocery Store System에 대한 시물레이션 결과 예측**

## 6. 결론

해석적 분석을 위한 SPN 형식론을 이용하여 DEVS 형식론에 의한 시뮬레이션 결과를 미리 예측하는 방법을 소개하였다. 이는 간접적으로 미리 시뮬레이션의 수행 정도를 파악하여 보다 신뢰할 수 있고 정확한 분석결과를 제공하기 위해 이론적 기반을 마련하였다. 그러므로 대규모의 자료수집을 통한 많은 횟수의 실험을 신뢰성을 가지고 수행할 수 있어 궁극적으로 수행시간의 낭비를 초래하는 문제점을 해결할 수 있음을 보였다. 추후 연구방향은 이산사건 시뮬레이션을 위한 모델링 월드 뷰를 액티비티 조사, 프로세스 상호작용으로 확장하여 적용하고자 한다.

## 참고문헌

- [1] R. Davies & R. O'keefe, *Simulation Modelling with PASCAL*, Englewood Cliffs, NJ: Prentice Hall, Inc., 1989.
- [2] P.J. Denning, J.B. Dennis, & J.E. Qualitz, *Machine, Languages, and Computation*, Englewood Cliffs, NJ: Prentice Hall, Inc., 1978.
- [3] P.J. Haas & G.S. Schedler, "Stochastic Petri Net Representation of Discrete Event Simulation," *IEEE Trans. on Soft. Eng.*, Vol. SE-15, No.4, pp.381-393, Apr. 1989.
- [4] S.V. Hoover & R.F. Perry, *Simulation a Problem-Solving Approach*, Addison-Wesley Pub. Comp. Inc., New York, 1989.
- [5] A.M. Law & W.D. Kelton, *Simulation Modelling and Analysis*, McGraw-Hill Book Comp., 1982.
- [6] K. H. Lee & J. Favrel, "Hierarchical Reduction Method for Analysis and Decomposition of Petri Nets," *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. SMC-15, No. 2, pp.272-280, Mar./Apr. 1985.
- [7] M.K. Molloy, "Performance Analysis using Stochastic Petri Nets," *IEEE Trans. Computer*, Vol. C-31, No. 9, pp.913-917, Sep. 1982.
- [8] M.K. Molloy, "Discrete Time Stochastic Petri Nets," *IEEE Trans. on Soft. Eng.*, Vol. SE-11, No. 4, pp. 417-423, Apr. 1985.
- [9] J.L. Peterson, *Petri Net Theory and The Modeling of Systems*, Englewood Cliffs, NJ: Prentice Hall, Inc., 1981.
- [10] B.P. Zeigler, *Theory of Modelling and Simulation*, A Wiley Interscience Pub., 1984.
- [11] W.M. Zuberek, "D-Timed Petri Nets and Modelling of Timeouts and Protocols," *Trans. of the SCS*, Vol. 4, No. 4, pp.331-357, Oct. 1988.
- [12] 손 진곤, "Petri Net과 DEVS 형식론을 이용한 컴퓨터 통신 프로토콜의 모델링," 고려대학교 박사학위 논문, 1991.
- [13] 정 영식, 황 종선, 백 두권, "SPN을 이용한 DEVS 모델의 타당성 검사," 한국시뮬레이션학회 논문지, Vol. 1, No.1, pp.77-86, Dec. 1992.
- [14] 정 영식, 황 종선, 백 두권, "통합적 시스템 분석을 위한 모델변환 처리기의 설계 및 구현," 한국정보과학회 논문지, Vol. 20, No. 5, pp.717-728, May. 1993.
- [15] Young-Sik Jeong, Chong-Sun Hwang, Doo-Kwon Baik, Jin-Gon Shon, "A Model Transformation Processor Based on DEVS and SPN Formalisms," *The Society for Computer Simulation*, '93 SCSC Conference(at Boston in U.S.A.), Jul. 1993.

## ● 저자소개 ●

**정영식**

1987년 고려대학교 졸업

1989년 고려대학교 석사학위 취득

1993년 고려대학교 박사학위 취득

현 재 원광대학교 공과대학 컴퓨터공학과 전임

관심분야 : 모델링/시물레이션, 분산시스템, ICAI, 멀티미디어 CAI 등