

## 이벤트의 선택 확률을 고려한 시간 넷의 분석 알고리즘 및 응용

임재걸\* · 주재훈\*\*

### A New Analytical Algorithm of Timed Net with Probabilities of Choices and Its Application

Jae-Geol Yim\* · Jae-Hun Joo\*\*

#### ■ Abstract ■

For an analysis of the performance of a computer system, the minimum cycle time method has been widely used. The minimum cycle time method is a mathematical technique with which we can find the minimum duration time needed to fire all the transitions at least once and coming back to the initial marking in a timed net. A timed net is a modified version of a Petri net where a transition is associated with a delay time. In the real world, an event is associated with a probability of occurrence. However, a timed net is not equipped with any facility of specifying probability of event occurrence. Therefore, the minimum cycle time method applied on a timed net can easily overlook probabilities of occurrences of events and yield a wrong result.

We are proposing 'Timed Net with Probabilities of Choices' where a transition can be associated with both delay time and a probability of occurrence. We also introduce an algorithm for minimum cycle time analysis on a 'Timed Net with Probabilities of Choices'. As an example of application, we are performing an analysis of the location based service system using 'Timed Net with Probabilities of Choices'.

Keyword : System Analysis, Performance Analysis, Location Based Service, Minimum Cycle Time, Petri Net, Timed Net

논문접수일 : 2005년 3월 8일      논문게재확정일 : 2005년 11월 8일

\* 동국대학교 컴퓨터멀티미디어학과

\*\* 동국대학교 전자상거래학과

## 1. 서론

페트리 넷은 1962년에 처음 소개된 이후 시스템 성능 테스트와 통신 규약의 일관성 및 타당성 테스트 등의 분야에 성공적으로 사용되었다. 또한 이는 분산소프트웨어시스템, 분산데이터베이스 시스템, 병렬시스템, 유연생산시스템, 사무자동화시스템, 형식언어 등을 비롯한 다양한 분야에서 모델링과 분석 도구로서 각광을 받고 있다[5].

페트리 넷 성능 분석 방법 중 반복적으로 수행되는 과업의 작업 시간을 측정하는 수단으로 '최소 순회시간(minimum cycle time)' 분석 방법이 있다. 페트리 넷의 전이(transition)에, 연관된 사건이 실행되는데 걸리는 시간을 첨부하여 놓은 것을 '시간 넷'이라 하고, '시간 넷'으로 생산 공정의 모델을 구축하여 주기적으로 수행되는 일련의 작업이 실행되는데 걸리는 시간을 '시간 넷(timed net)'을 분석하여 알아내는 방법이 Hillion[3]에 소개되었다.

최소 순회시간을 찾는 기존 알고리즘의 효율성을 개선하는 연구 결과가 Dasdan and Gupta[1]에 발표되었으며, 모든 장소의 입력 간선과 출력 간선이 정확히 하나인 특수한 페트리 넷의 최소 순회시간을 찾는 방법이 Nakamura and Silva[6]에 소개되었다. Rodriguez-Beltran and Ramfrez-Trevino[9]에는 순회시간이 주어졌을 때, 실제 순회시간이 주어진 순회시간이 되도록 초기 마킹을 결정하는 문제에 대한 해가 소개되었다.

실세계의 사건은 실행되는데 걸리는 시간뿐만 아니라 선택될 확률과도 연관되어 있다. 예를 들어, 데이터베이스의 읽기와 쓰기라는 사건을 고려하면, 읽기라는 사건이 실행되는데 걸리는 시간과 쓰기라는 사건이 실행되는데 걸리는 시간이 상이할 뿐 아니라, 읽기라는 사건이 선택될 확률과 쓰기라는 사건이 선택될 확률이 서로 다르다. 그러나 기존의 시간 넷은 사건이 실행되는데 걸리는 시간만 고려할 뿐, 선택될 확률을 고려하지 못했다. 현재까지의 시간 넷에 관한 연구에서는 사건이 발생할 확률을 고려하지 못했지만, 그 확률을 고려하는 경우 시간 넷

의 응용 범위는 더욱 확대될 것이다. 따라서 본 논문은 시간 넷에 사건의 선택 확률을 고려한 '선택 확률 시간 넷'을 제안하고, 이를 위치기반서비스(Location Based Service : LBS) 시스템을 분석하는데 적용한 예를 제시한다.

본 논문은 다음과 같이 구성된다. 2장에서는 장소, 전이, 마킹 등 페트리 넷 관련 용어와 전이에 시간이 연합된 시간 넷의 개념, 시간 넷의 최소 순회 시간 분석 방법을 포함한 기존의 연구를 소개한다. 3장에서는 시간 넷의 전이에 사건의 발생 확률이 연합된 선택 확률 시간 넷을 제안하고, 선택 확률 시간 넷의 최소 순회시간 분석 알고리즘을 고안한다. 4장에서 위치기반서비스 시스템 분석에 제안한 방법을 적용한 예를 보이고, 기존의 방법을 적용한 결과와 비교한다.

## 2. 기존의 연구

여기서는 선택 확률 시간을 고안하기에 앞서 그 기초가 되는 기존의 최소 순회시간 분석 방법을 소개한다. 최소 순회시간 방법에 대한 설명을 페트리 넷에 대한 기초 지식이 없더라도 이해할 수 있도록, 예를 통하여 설명하고자 한다[2, 5, 7, 8].

페트리 넷의 전이에 시간이 첨부되어 있는 것을 시간 넷이라 하며, 간단한 시간 넷의 예는 [그림 1]과 같다. [그림 1]은 데이터베이스 검색과 갱신 작업을 병렬로 실행 중인 프로세스를 나타낸다[5].  $p_1$ 이라고 쓰인 원은 실행 중인 프로세스를 나타내며, 이 원 안에 쓰인  $k$ 는 현재 실행 중인 프로세스가  $k$ 개 있다는 뜻이다. 각 프로세스는 계산과 같은 고유의 작업을 수행하다가 데이터베이스에 접근하여 검색이나 갱신 작업을 한다. 갱신 작업의 시작을 나타내는 것이  $t_2$ 라고 쓰인 사각형이고, 검색 작업의 시작을 나타내는 것은  $t_1$ 이라고 쓰인 사각형이다. 갱신 작업 중인 프로세스는  $p_4$ 라고 쓰인 원 안에, 검색 중인 프로세스는  $p_2$ 라고 쓰인 원 안에 머문다.  $t_4$ 라고 쓰인 사각형과  $t_3$ 이라고 쓰인 사각형은

각각 갱신 작업과 검색 작업의 종료를 의미하며, 작업을 종료한 프로세스는 다시  $p_1$ 이라고 쓰인 원으로 되돌아간다.  $p_3$ 이라고 쓰인 원은 프로세스가 공유하는 데이터가 머무는 장소이며, 본 모델은 갱신 작업시 데이터베이스 전체를 독점한다고 가정하였다. 검색시에는 원하는 모든 프로세스가 동시에 검색을 수행할 수 있다. 다음에서 이 시간 넷의 작동 원리를 설명하고자 한다.

페트리 넷은 이분유향그래프(bipartite directed graph)이다. 이분이란 정점이 두 가지 그룹으로 분할되며, 동일한 그룹에 속하는 정점들 간에는 간선이 존재하지 않는 것을 의미하며, 유향이란 간선에 방향이 있다는 것을 의미한다. [그림 1]의 정점들은 사각형 모양과 원 모양으로 분류된다. 사각형 모양의 정점을 전이(transition)라 하고, 원 모양의 정점을 장소(place)라고 일컫는다. 실세계의 현상을 페트리 넷으로 나타낼 때, 일반적으로 사건은 전이로, 자원은 장소에 나타내는데, 사건이 수행되면 상태가 전이하고, 자원이 존재하는 곳은 장소임으로 페트리 넷의 정점들의 분류의 이름을 그렇게 명명한 것이다.

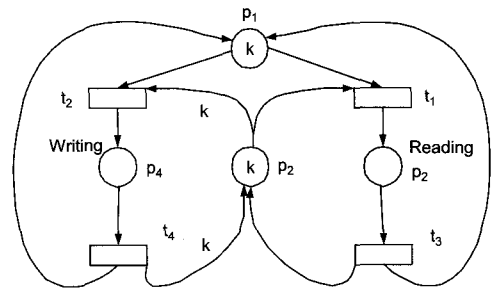
그래프의 정점들이 이름을 갖듯이 장소와 전이의 각 원소도 보통 고유 이름을 갖는다. 실제 시스템을 나타내는 모델의 경우에는 정점과 연합된 시스템의 구성 요소를 나타내는 의미 있는 이름을 사용하지만, [그림 1]의 경우에는 무의미한 일련번호를 사용하여 장소는  $p_1, p_2, p_3$ 이라 하고, 전이는  $t_1, t_2, t_3, t_4$ 라고 명명하였다. [그림 1]의 경우에는 모든 전이 각각에 지연시간이 부착되어 있으며, 그 값은 변수  $d_1, d_2, d_3, d_4$ 로 표기되었다. 그러나 지연시간이 0일 경우에는 표기를 생략할 수 있다.

장소는 일반적으로 자원을 나타내므로, 장소의 이름은 해당 자원 이름을 쓴다. 가용 자원의 분포를 나타내기 위하여 각 장소에 자원의 수를 표기하는데, [그림 1]의 경우에는 장소  $p_1$ 에 해당하는 자원  $k$ 개가 사용 가능함을  $p_1$ 의 내부에  $k$ 라는 레이블로 표기 하였으며, '  $p_1$ 에  $k$ 개의 토큰이 있다'라고 말한다. 장소  $p_3$ 에도  $k$ 개의 토큰이 있음을 알 수 있다.

실세계에서 사건이 발생하여 상태가 변화하는 현상을 페트리 넷에서는 전이의 발화로 시뮬레이션한다. 발화 규칙은 다음과 같다.

- (1) 하나의 전이  $t$ 에 대하여, 입력 장소  $p$ 가 최소한  $W(p, t)$  만큼의 토큰을 갖고 있으면,  $t$ 는 장전되었다고 한다. 단,  $W(p, t)$ 는 간선  $(p, t)$ 의 레이블이다.
- (2) 장전된  $t$ 는 발화될 수도 있고, 아니 될 수도 있다.
- (3)  $t$ 발화는  $W(p, t)$  만큼의 토큰을 각 입력 장소  $p$ 에서 제거하고,  $W(p, t)$  만큼의 토큰을 각 출력 장소에 더하여 준다. 단,  $W(p, t)$ 는 간선  $(p, t)$ 의 레이블이다.

[그림 1]의 전이  $t_2$ 를 예로 들면,  $t_2$ 의 입력 장소는  $p_1$ 과  $p_3$  두 개다. 간선  $(p_1, t_2)$ 의 레이블은 그룹에 명시되어 있지 않은데, 일반적으로 간선의 묵시적인 레이블은 1이므로, 간선  $(p_1, t_1)$ 의 레이블은 1이라고 할 수 있다. 한편, 간선  $(p_3, t_2)$ 의 레이블은  $k$ 라고 쓰여 있다. 발화 규칙에 의하여  $t_2$ 가 발화하려면  $p_3$ 에  $k$ 개의 토큰이 있어야 하고,  $t_2$ 가 발화하면  $p_3$ 에 있는  $k$ 개의 토큰이 모두 소비된다. 한편,  $t_1$ 의 발화는 장소  $p_1$ 과 장소  $p_3$ 에서 각각 하나씩의 토큰을 사용함으로  $t_1$ 은 계속 여러 번 발화가 가능하다. 이에 반하여,  $t_2$ 의 경우에는  $p_3$ 에  $k$ 개의 토큰이 있어야 함으로,  $p_2$ 에 토큰이 하나라도 있으면 발화가 불가능하고,  $t_2$ 가 발화하여  $p_4$ 에 토큰



[그림 1] 데이터베이스 검색과 갱신 작업을 수행하는 프로세스를 나타내는 시간 넷 모형

큰이 있으면  $t_1$ 의 발화가 불가능하다. 즉, 어떤 시점에서 갱신은 단 하나의 프로세스만 실행할 수 있는데 반하여, 검색은 다수의 프로세스가 동시에 실행 가능하다.

초기에 각 장소에 놓인 토큰의 수를 나열한 벡터를 초기마킹이라 한다. [그림 1]의 경우에 초기마킹  $M_0$ 은 다음과 같다.

$$M_0 = (k, 0, k, 0).$$

그래프 이론에서, 그래프를 나타내는 일반적인 방법으로 발생행렬(incidence matrix)을 사용한다. [그림 1]의 발생행렬은 다음과 같다.

$$A = \begin{matrix} & \begin{matrix} p_1 & p_2 & p_3 & p_4 \end{matrix} \\ \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \end{matrix} & \begin{bmatrix} -1 & 1 & -1 & 0 \\ -1 & 0 & -k & 1 \\ 1 & -1 & 1 & 0 \\ 1 & 0 & k & -1 \end{bmatrix} \end{matrix}$$

**[정의 1]**  $Ay=0$ 을 만족하는 벡터  $y$ 는 식 (1)의 성질을 만족하므로 S-불변(S-invariant)이라고 한다. 식 (1)의  $M_0$ 은 초기마킹,  $M_i$ 는  $M_0$ 으로부터 도달 가능한 임의의 마킹이다.

$$y^T M_i = y^T M_0 \quad (1)$$

[그림 1]의 경우에는 다음과 같은 S-불변이 있다.

$$\begin{aligned} (1 \ 1 \ 0 \ 1)^T &= y_1, \\ (0 \ 1 \ 1 \ k)^T &= y_2. \end{aligned}$$

**[정의 2]**  $A^T x=0$ 을 만족하는 벡터  $x$ 를 T-불변(T-invariant)이라 한다.

발생행렬  $A$ 의 각 항  $a_{ij}$ 는 전이  $i$ 의 발화 결과 장소  $j$ 로 입력되는 토큰의 수와, 장소  $j$ 에서 소모되는 토큰의 수를 나타내므로, 각 전이를 T-불변의 해당 항만큼 발화하면 본래의 마킹으로 돌아오는 성질이 있다.

**[정리 1]** 각 전이를 T-불변의 해당 항만큼 발화하면 본래의 마킹으로 돌아오는 성질이 있다.

[그림 1]의 경우에는 다음과 같은 T-불변들이 있다.

$$\begin{aligned} x_1 &= (1 \ 0 \ 1 \ 0)^T, \\ x_2 &= (0 \ 1 \ 0 \ 1)^T, \\ x_3 &= (1 \ 1 \ 1 \ 1)^T. \end{aligned}$$

T-불변의 정의에 의하여 [정리 1]의 역도 참이다. 즉, 주어진 마킹으로부터 발화를 시작하여 이 마킹으로 다시 되돌아오는 발화순서에 나타나는 전이들의 회수로 구성된 벡터  $x_j$ 는 T-불변이다.

**[정리 2]** 주어진 마킹으로부터 발화를 시작하여 이 마킹으로 다시 되돌아오는 발화순서에 나타나는 전이들의 회수로 구성된 벡터  $x_j$ 는 T-불변이다.

예를 들어, [그림 1]에서  $t_1$ 과  $t_3$ 을 차례로 발화하면 다시 [그림 1]로 되돌아온다. 따라서 이 발화순서의 발화 회수 벡터,  $(1 \ 0 \ 1 \ 0)$ 은 T-불변이다.

발생행렬의 음수 항만 절대 값으로 바꾸고, 나머지 항은 0으로 대치하여 얻은 행렬, 즉 각 장소로 들어오는 간선의 가중치만 표기한 행렬을  $A^-$  행렬이라 하며, [그림 1]의 경우에는 다음과 같다.

$$A^- = \begin{matrix} & \begin{matrix} p_1 & p_2 & p_3 & p_4 \end{matrix} \\ \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \end{matrix} & \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & k & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & k & 1 \end{bmatrix} \end{matrix}$$

[그림 1]의 각 전이에 연관된 지연 시간을 나타내는 대각행렬  $D$ 는 다음과 같다.

$$D = \begin{matrix} & \begin{matrix} t_1 & t_2 & t_3 & t_4 \end{matrix} \\ \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \end{matrix} & \begin{bmatrix} d_1 & 0 & 0 & 0 \\ 0 & d_2 & 0 & 0 \\ 0 & 0 & d_3 & 0 \\ 0 & 0 & 0 & d_4 \end{bmatrix} \end{matrix}$$

최소 순회시간을 찾는 식은 Murata[5]에 소개되었고, 식 (2)와 같다. 단,  $x$ 는  $x_3$ 과 같이 양수 항으로 구성된  $T$ -불변이다.

$$\text{최소 순회시간} = \text{Max}\{y_k^T(A^-)^T D x / y_k^T M_0\} \quad (2)$$

[그림 1]의 시간 넷에 식 (2)를 적용하면, 다음과 같다.

$$(A^-)^T D x = [d_1 + d_2 d_3 d_1 + k d_2 d_4]^T$$

$$y_1 = (1 \ 1 \ 0 \ 1)^T \text{일때,}$$

$$y_1^T (A^-)^T D x = d_1 + d_2 + d_3 + d_4,$$

$$y_1^T M_0 = (1 \ 1 \ 0 \ 1) (k \ 0 \ k \ 0)^T = k$$

임으로,

$$y_1^T (A^-)^T D x / y_1^T M_0 = (d_1 + d_2 + d_3 + d_4) / k$$

이다.

$$y_2 = (0 \ 1 \ 1 \ k)^T \text{일때,}$$

$$y_2^T (A^-)^T D x = d_1 + k d_2 + d_3 + k d_4,$$

$$y_2^T M_0 = k$$

임으로,

$$y_2^T (A^-)^T D x / y_2^T M_0 = [k(d_2 + d_4) + (d_1 + d_3)] / k$$

이다.

따라서 최소 순회시간은 다음과 같다.

$$\text{Max}\{(d_1 + d_2 + d_3 + d_4) / k, [k(d_2 + d_4) + (d_1 + d_3)] / k\} = d_2 + d_4 + (d_1 + d_3) / k.$$

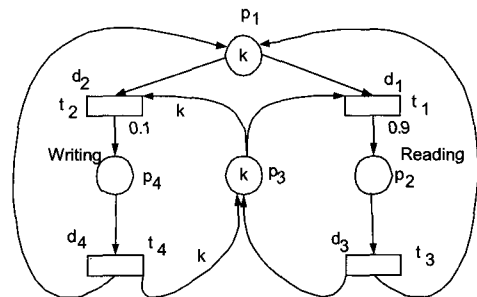
즉, 모든 전이를 정확히 한 번씩 발화하고 초기 마킹으로 돌아오는데 걸리는 시간이  $d_2 + d_4 + (d_1 + d_3) / k$ 이다. 따라서  $k$ 개의 토큰 각각이 모든 전이를 한 번씩 발화하고 초기마킹으로 돌아오는데 걸리는 시간은  $k(d_2 + d_4) + d_1 + d_3$ 이 된다. 데이터베이스에서 갱신 작업과 검색 작업의 발생 빈도를 보면 검색 작업의 빈도가 훨씬 높음을 알 수 있다. 검색 작업의 확률을 0.9, 갱신 작업의 확률을 0.1이라고 한다면,  $k$ 개의 프로세스가 검색이나 갱신 작업을

한 번씩 수행하는데 걸리는 시간은  $0.1k(d_2 + d_4) + 0.9(d_1 + d_3)$ 이라야 한다. 왜냐하면  $k$  프로세스 중 0.1 즉 0.1  $k$  프로세스가 갱신 작업을 수행하고 0.9  $k$  프로세스가 검색 작업을 수행하는데 갱신 작업은 한 번에 하나씩만 수행이 가능하므로  $0.1k(d_2 + d_4)$  시간이 필요하고, 0.9  $k$  프로세스의 검색은 동시에 수행이 가능하므로  $0.9(d_1 + d_3)$  시간이 필요하기 때문이다. 3장에는 선택 확률 시간 넷과 그 분석 알고리즘이 제안되며, 이 방법으로 최소 순회시간이  $0.1k(d_2 + d_4) + 0.9(d_1 + d_3)$ 임을 정확히 분석해 보려고 한다.

### 3. 선택 확률 시간 넷과 알고리즘

실세계의 사건은 선택될 확률과 연관되어 있음에도 불구하고, 시간 넷은 사건이 선택될 확률을 고려하지 않는다. 여기서는 시간 넷의 전이에 선택될 확률을 연합한 페트리 넷을 '선택 확률 시간 넷'이라 정의하고, 선택 확률 시간 넷의 최소 순회시간을 찾는 방법을 제안하고자 한다.

선택 확률 시간 넷의 예로 [그림 2]를 들 수 있다. [그림 2]는 [그림 1]의 시간 넷과 동일하나 전이  $t_1$ 과  $t_2$ 에 선택 확률 0.9와 0.1이 각각 연합되어 있는 점이 다르다. 전이  $t_1$ 과  $t_2$ 는 모두 장소  $p_1$ 의 출력 간선에 연결되어 있으며, 이러한 경우에  $t_1$ 과  $t_2$ 는  $p_1$ 의 출력 전이라고 한다.



[그림 2] 선택 확률 시간 넷의 예

페트리 넷 분야에서 주어진 정점  $v$ 의 출력 정점

〈표 1〉 선택 확률 시간 넷의 정의

선택 확률 시간 넷  $N$ 은 7가지 요소로 구성된다.  
 $N = (P, T, F, W, DT, Prob, M_0)$ , 단,  
 $P = \{p_1, p_2, \dots, p_m\}$ 은 장소(place)라는 유한 집합,  
 $T = \{t_1, t_2, \dots, t_n\}$ 은 전이(transition)라는 유한 집합,  
 $F$  : 유향간선이라는  $(P \times T)$ 와  $(T \times P)$ 의 합집합의 부분집합,  
 $W$  :  $F \rightarrow \{1, 2, 3, \dots\}$ 은 유향간선의 가중치를 결정하는 가중치 함수,  
 $DT$  :  $T \rightarrow \{1, 2, 3, \dots\}$ 은 전이의 지연 시간을 결정하는 함수,  
 $Prob$  :  $T \rightarrow (0, 1)$ 은 전이의 선택 확률을 결정하는 함수. 단,  $\sum_{t_i \in T} Prob(t_i) = 1$ .  
 $M_0$  :  $P \rightarrow \{0, 1, 2, 3, \dots\}$ 은 초기에 각 장소에 놓인 토큰의 수를 결정하며, 초기 마킹(marking)이라고 함.  
 단,  $P$ 와  $T$ 의 교집합은 공집합이고,  $P$ 와  $T$ 의 합집합은 공집합이 아님.

(output vertices)을  $v \cdot$ 으로 표기하고, 입력 정점(input vertices)은  $\cdot v$ 로 표기한다. 즉, [그림 2]의 경우에는  $p_1 \cdot = \{t_1, t_2\}$ 이다. 장소  $p_1$ 에 놓인 토큰은  $t_1$  발화에 사용되든지 아니면  $t_2$  발화에 사용될 수 있다. 따라서 이 경우의  $t_1$ 과  $t_2$ 처럼 입력 장소를 공유하는 전이를 선택 관계라 하고, ' $t_1$  선택  $t_2$ ' 혹은 '선택 ( $t_1, t_2$ )'라고 표기한다.

**[정의 3]** 어떤 두 전이  $t_a$ 와  $t_b$ 가 입력 장소를 공유하면 이들 두 전이 간에 선택 관계가 있다고 하고, ' $t_a$  선택  $t_b$ ' 혹은 '선택 ( $t_a, t_b$ )'로 표기한다. 선택 관계의 전이는 선택 확률과 연합되며, 전이  $t_a$ 의 선택 확률을  $Prob(t_a)$ 로 표기한다.

확률의 정의에 의하여 선택 관계에 있는 전이들의 선택 확률의 합은 1이어야 한다. 즉, 모든 장소에 대하여 다음 식이 성립한다.

$$\sum_{t_i \in P_i} Prob(t_i) = 1.$$

어떤 장소의 출력 전이가 둘 이상이라도, 발화할 전이를 선택하는데 영향을 주지 않는 경우에는 선택 확률을 지정하지 않는다. 선택 확률 시간 넷을 정의하면 <표 1>과 같다.

**[정의 4]** 주어진 벡터  $x$ 의 0이 아닌 항들의 집합을 그 벡터의 받침(support)이라고 하며,  $\|x\|$ 로 표기

한다.  $T$ -불변 중 받침이 다른  $T$ -불변의 받침의 부분집합이 아닌  $T$ -불변을 '최소 받침(minimal support)  $T$ -불변'이라고 한다.

예를 들어, [그림 1]의  $x_1$ 과  $x_2$ 는 최소 받침  $T$ -불변이지만,  $x_1$ 과  $x_2$ 의 합으로 구할 수 있는  $x_3$ 은 그렇지 않다. 일반적으로 벡터  $x$ 의  $i$ 항을 가리키기 위하여  $x[i]$ 와 같이 대괄호의 첨자를 사용한다. 예를 들어,  $x_1[1]=1, x_1[2]=0, x_1[3]=1, x_1[4]=0$ 이다.  $x$ 가  $T$ -불변일 때, 모든 전이  $i$ 를  $x[i]$ 번 발화하는 발화순서는 원래의 마킹으로 되돌려 놓는 성질이 있다. 예를 들어, [그림 2]의 경우 현재 마킹은  $(n001)$ 인데,  $t_1$ 을 한번 ( $x_1[1]=1$ ) 발화하고  $t_3$ 을 한번 ( $x_1[3]=1$ ) 발화하면 마킹이 다시  $(n001)$ 이 된다. 그러나  $x$ 가  $T$ -불변이라는 것이 모든 전이  $i$ 를  $x[i]$ 번 발화할 수 있다는 것을 의미하지는 않는다. 즉,  $A^T x_a = 0$ 을 만족하는 벡터  $x_a$  중에는 모든 전이  $i$ 를  $x_a[i]$ 번 발화하는 것이 불가능한 경우도 있다.  $A^T x_a = 0$ 이고, 모든 전이  $i$ 를  $x_a[i]$ 번 발화하는 것이 가능할 때,  $x_a$ 를 발화 가능  $T$ -불변이라고 한다.

**[정의 5]**  $A^T x_a = 0$ 이고, 모든 전이  $i$ 를  $x_a[i]$ 번 발화하는 것이 가능할 때,  $x_a$ 를 발화 가능  $T$ -불변이라고 한다.

**[정의 6]**  $A^T x = 0$ 을 만족하는  $x > 0$ 이면서  $x$ 가 발화가능한 벡터  $x$ 가 존재하는 페트리 넷을 일관성(consistent) 있다고 한다.

최소 순회시간은 페트리 넷의 모든 전이를 일회 이상 발화하고 초기 마킹으로 되돌아오는 데 걸리는 최소 시간이기 때문에, 최소 순회시간 분석에서 다루는 페트리 넷은 일관성이 있는 넷이다. 나아가서 분석하고자 하는 시스템의 처리 과정은 일련의 과정으로 구성된다. 따라서 선택 확률 시간 넷에는 처리 과정의 첫 공정을 기다리는 자원이나 요소를 나타내는 토큰이 놓이는 장소인 홈 장소가 있다. 예를 들어, [그림 2]의 경우에는 읽거나 쓰기 연산을 하기 위하여 대기하는 프로세스에 해당하는 토큰이 놓이는 장소  $p_1$ 이 홈 장소이다.

선택 관계의 전이에 선택 확률이 정의되듯이, 일련의 전이로 구성된 발화 순서와 발화 순서에 출현하는 각 전이들의 회수로 구성된 벡터에도 선택 확률이 연합된다. 벡터의 선택 확률을 정의하기 위하여 묶음(group)이라는 용어를 사용한다.

**[정의 7]** 선택 관계가 있는 전이들의 부분집합을 묶음(group)이라 한다.

**[정의 8]** 묶음의 선택 확률은 모든 원소 전이들의 선택 확률의 합으로 정의한다.

**[정의 9]** 발화 가능  $T$ -불변  $x$ 가 주어질 때,  $x$ 의 선택 확률은 다음과 같이 구한다.

- 단계 1.  $x$ 받침들을 선택 관계로 묶어 묶음을 형성한다.
- 단계 2. 단계 1에서 구한 묶음 각각에 대한 선택 확률을 구한다.
- 단계 3. 각 묶음의 선택 확률을 모두 곱하여  $x$ 의 선택 확률로 한다.

<예 1> [그림 2]에서  $Prob(t_1) = 0.9$ ,  $Prob(t_2) = 0.1$  일 때, 각  $T$ -불변의 선택 확률을 구하여 보자.

$x_1 = (1 \ 0 \ 1 \ 0)^T$ 의  $T$ -불변 선택 확률은 다음과 같이 구한다.

단계 1에서 받침들을 선택 관계로 묶은 결과 다음과 같은 두 개의 묶음을 얻는다:  $g_1 = \{t_1\}$ ,  $g_2 = \{t_3\}$

단계 2에서 [정의 8]을 적용하여 각 묶음의 선택 확률,  $g_1$ 의 선택 확률 0.9,  $g_2$ 의 선택 확률 1을 구한다.

단계 3에서  $T$ -불변의 선택 확률의 정의에 의하여  $x_1$ 의 선택 확률,  $0.9 \times 1 = 0.9$ 를 구한다.

$x_2 = (0 \ 1 \ 0 \ 1)^T$ 의  $T$ -불변 선택 확률은 다음과 같이 구한다.

단계 1에서  $g_1 = \{t_2\}$ ,  $g_2 = \{t_4\}$ .

단계 2에서  $g_1$ 의 선택 확률 = 0.1,  $g_2$ 의 선택 확률 = 1을 구한다.

단계 3에서  $x_2$ 의 선택 확률 =  $0.1 \times 1 = 0.1$ 을 구한다.

한편,  $T$ -불변  $x_3 = (1 \ 1 \ 1 \ 1)^T$ 의 선택 확률은 다음과 같이 구한다.

단계 1에서  $g_1 = \{t_1, t_2\}$ ,  $g_2 = \{t_3\}$ ,  $g_3 = \{t_4\}$ 를 구한다.

단계 2에서  $g_1$ 의 선택 확률 =  $Prob(t_1) + Prob(t_2) = 1$ ,  $g_2$ 의 선택 확률 = 1,  $g_3$ 의 선택 확률 = 1.

단계 3에서  $T$ -불변의 선택 확률의 정의에 의하여  $x_3$ 의 선택 확률,  $1 \times 1 \times 1 = 1$ 을 구한다.

**[정의 10]** 홈 장소를 가진 일관성 있는 선택 확률 시간 넷을 적당한(well formed) 선택 확률 시간 넷이라 한다.

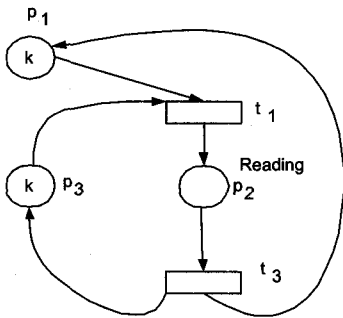
**[정의 11]** 주어진 페트리 넷 PN의  $T$ -불변  $x$ 에 관한 PN의 부페트리 넷은  $x$ 의 받침과 이들과 간선으로(길이 1인 경로) 연결된 정점들로 구성된 페트리 넷이다.

<표 2> 선택 확률 시간 넷의 최소 순회시간 분석 알고리즘

입력 : 적당한 선택 확률 시간 넷. 초기 마킹에서 홈 장소에 놓인 토큰의 수를  $n$ 이라 함.

- 단계 1 : 최소 받침  $T$ -불변(minimal support  $T$ -invariant)을 구한다.
- 단계 2 : 홈 장소를 기준으로 최소 받침  $T$ -불변들을 다음과 같이 합병한다.
  - 2.1 각 최소 받침  $T$ -불변에 대한 부 페트리 넷을 작성한다.
  - 2.2 홈 장소를 포함하지 않은 부 페트리 넷이 존재하는 동안
    - 2.2.1 홈 장소를 포함하지 않은 부 페트리 넷을 선택하여 그와 만나는 이웃 부 페트리 넷에 합병한다.
  - 2.3 합병된 부 페트리 넷 각각에 대한  $T$ -불변을 구하여 홈 장소를 포함하는  $T$ -불변이라 한다.
- 단계 3 : 홈 장소를 포함하는  $T$ -불변  $x_i$ 의 선택 확률  $p_i$ 를 구한다.
- 단계 4 : 홈 장소를 포함하는  $T$ -불변  $x_i$ 에 (선택 확률  $\times n$ )을 곱하여  $p_i \times n \times x_i$ 를 구한다.
- 단계 5 : 모든 홈 장소를 포함하는  $T$ -불변  $x_i$ 에 대한  $p_i \times n \times x_i$ 의 합을 발화가능 양의  $T$ -불변  $x$ 로 한다.
- 단계 6 : 다음 식을 적용한다 :  $Max(y_1^T(A^-)^T Dx / y_k^T M_0)$

예를 들어, [그림 1]의  $T$ -불변  $x_1 = (1 \ 0 \ 1 \ 0)$ 에 관한 부페트리 넷은 [그림 3]과 같다.



[그림 3]  $T$ -불변  $x_1 = (1 \ 0 \ 1 \ 0)$ 에 관한 [그림 1]의 부페트리 넷

[정의 12] 두 부페트리 넷이 공유하는 정점이 존재할 때, 이 둘은 서로 이웃(neighbor)한다고 한다.

여기서는 [정의 3]~[정의 12]를 기반으로 선택 확률 시간 넷에서 홈 장소의 모든 토큰이 시간 넷을 한 번씩 순회하는 데 걸리는 시간을 찾는 알고리즘, 선택 확률 시간 넷의 최소 순회시간 분석 알고리즘을 <표 2>와 같이 제안한다.

예를 들어, [그림 2]에 제안한 알고리즘을 적용하면 다음과 같이 최소 순회시간을 구한다.

1. 단계 1에서 다음과 같은 최소 받침  $T$ -불변

(minimal support  $T$ -invariant)을 구한다.

$$x_1 = (1 \ 0 \ 1 \ 0)^T,$$

$$x_2 = (0 \ 1 \ 0 \ 1)^T.$$

- 2. 단계 2의 결과는 단계 1의 결과와 동일하다.
- 3.  $x_1$ 의 선택 확률 0.1과  $x_2$ 의 선택 확률 0.9를 각각 구한다.
- 4. 단계 4에서 다음과 같은 가중치를 고려한  $T$ -불변을 구한다.

$$p_1 \times k \times x_1 = (0.1k \ 0 \ 0.1k \ 0)^T,$$

$$p_2 \times k \times x_2 = (0 \ 0.9k \ 0 \ 0.9k)^T.$$

- 5. 단계 5에서 다음과 같은 양의 발화 가능  $T$ -불변  $x$ 를 구한다.

$$x = (0.1k \ 0.9k \ 0.1k \ 0.9k)^T.$$

- 6. 다음과 같이 단계 6을 수행한다.

$$(A^-)^T Dx = [0.6kd_1 + 0.1kd_2 \ 0.9kd_3$$

$$0.9kd_1 + 0.1k^2d_2 \ 0.1kd_4]^T$$

$y_1(1 \ 1 \ 0 \ 1)^T$  일 때,

$$y_1^T(A^-)^T Dx = 0.9kd_1 + 0.1kd_2 + 0.9kd_3$$

$$+ 0.1kd_4,$$

$$y_1^T M_0 = (1 \ 1 \ 0 \ 1)(k \ 0 \ k \ 0)^T = k$$

임의로,  $y_1^T(A^-)^T Dx / y_1^T M_0 = (0.9kd_1 + 0.1kd_2 + 0.9kd_3 + 0.1kd_4) / k$  이다.



$y_2(0 \ 1 \ 1 \ k)^T$ 일 때,

$$y_2^T(A^-)^T D x = 0.9kd_1 + 0.1k^2d_2 + 0.9kd_3 + 0.1k^2d_4,$$

$$y_2^T M_0 = (0 \ 1 \ 1 \ k)(k \ 0 \ k \ 0)^T = k$$

$$\text{임으로, } y_2^T(A^-)^T D x / y_2^T M_0 = (0.9kd_1 + 0.1k^2d_2 + 0.9kd_3 + 0.1k^2d_4) / k \text{이다.}$$

따라서 최소 순회시간은 다음과 같다.

$$\begin{aligned} & \text{Max}\{(0.9kd_1 + 0.1kd_2 + 0.9kd_3 + 0.1kd_4) / k, \\ & [0.1k^2(d_2 + d_4) + 0.9k(d_1 + d_3)] / k\} \\ & = 0.1k(d_2 + d_4) + 0.9(d_1 + d_3). \end{aligned}$$

예를 들어,  $n=100$ 이라면, 최소 순회시간은  $0.9d_1 + 10d_2 + 0.9d_3 + 10d_4$ 다. 100 프로세스 중 10개가 갱신 작업, 90개가 검색 작업을 수행하는데 매 갱신마다 하나씩 실행해야 함으로  $10(d_2 + d_4)$  시간이 소요되고, 검색은 동시에 실행이 가능함으로 100개의 프로세스가 검색 작업을 수행하는데 걸리는 시간은  $d_1 + d_3$  단위시간이다. 전체 프로세스 중 90%만 검색을 실행함으로 분석 결과는  $0.9(d_1 + d_3)$  시간이 필요하다고 하였다. 이로부터 제안하는 방법에 의한 분석 결과가 정확함을 알 수 있다.

선택 확률 시간 넷의 최소 순회시간 분석 알고리즘의 타당성을 검증하기 위하여 단계 2에서 구한  $T$ -불변의 선택 확률의 합이 1임([정리 4])을 보여야 한다. 예를 들어, [그림 2]의 경우에는 단계 2의 결과 반드시  $x_1$ 과  $x_2$ 를 발화가능  $T$ -불변으로 생성하여야  $T$ 선택 확률의 합이 1이 되어, 모든 사건의 발생 확률의 합은 1이 되어야 하는 확률의 조건을 만족하게 된다. 이를 증명하기 위하여 ‘ $T$ -불변이 다수이면 출력전이가 다수인 장소가 존재한다’라는 [정리 3]을 이용한다.

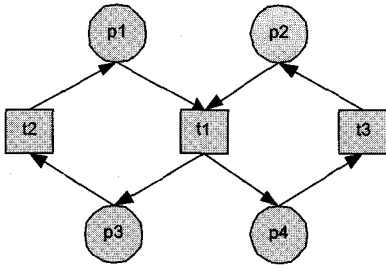
**[정리 3]** 적당한 선택 시간 넷에서 모든 장소 각각에 대한 출력 전이가 유일하면 최소받침  $T$ -불변도 유일하다.

<증명> 모든 전이의 출력 장소도 또한 유일한 경우를 살펴보자. 이 경우에는  $|T|=|P|$ 이다. 또한 적당한 선택 시간 넷의 정의에 의하여 주어진 넷은 연결그래프가 된다. 따라서 홈 장소에서 홈 장소로 되돌아오는 순회가 유일하고 그것이 유일한 최소받침  $T$ -불변이 된다.

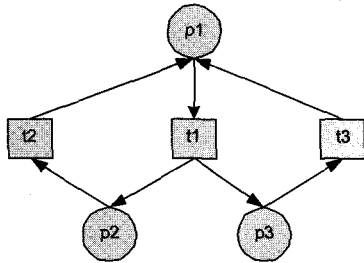
어떤 전이  $t_k$ 의 출력 장소가 여럿,  $p_1, p_2, \dots, p_i$ 인 경우에,  $x$ 라는 최소받침  $T$ -불변이외에  $x_a$ 라는 최소받침  $T$ -불변이 있다면,  $x_a$ 의 어떤 구성 요소의 입력 장소는 반드시  $p_1, p_2, \dots, p_i$  중 일부이고 나머지는  $x$ 의 어떤 구성 요소의 입력 장소라야 한다.  $x_a$ 의 어떤 구성 요소의 입력 장소의 집합을  $Pa$ 라 하고  $x$ 의 구성 요소의 입력 장소들의 집합을  $Pb$ 라 하자.  $x$ 와  $x_a$ 는 각각  $T$ -불변임으로  $x$ 에 해당하는 발화순서도  $t_k$ 의 발화 결과  $Pb$ 의 후손 장소에 파생한 모든 토큰은 물론  $Pa$ 의 후손 장소에 파생한 모든 토큰을 소비하고  $t_k$ 의 모든 입력 장소에 각각 하나씩의 토큰을 놓을 수 있어야 하고,  $x_a$ 에 해당하는 발화순서도  $t_k$ 의 발화 결과  $Pa$ 의 후손 장소에 파생한 모든 토큰은 물론  $Pb$ 의 후손 장소에 파생한 모든 토큰을 소비하고  $t_k$ 의 모든 입력 장소에 각각 하나씩의 토큰을 놓을 수 있어야 한다. 따라서  $Pb$ 의 후손 장소 중에는  $x$ 를 구성하는 전이의 입력이기도 하고  $x_a$ 를 구성하는 전이의 입력이기도 한 장소가 존재해야 한다. 즉, 출력 전이가 둘 이상인 장소가 존재하게 된다. 이것은 모든 장소에 대한 출력 전이가 유일하다는 가정에 모순이다. 따라서 최소받침  $T$ -불변은 유일해야 한다.

참고로 [그림 4]는 모든 장소의 출력전이가 유일하며, 전이  $t_1$ 의 출력 장소가 여럿이더라도, 최소받침  $T$ -불변,  $(t_1 \ t_2 \ t_3)$ 이 유일한 경우의 페트리 넷 예이고, [그림 5]는 전이  $t_1$ 의 출력 장소가 여럿이고 모든 장소의 출력 전이가 유일하여  $T$ -불변이 존재하지 않는 경우의 페트리 넷 예이며, [그림 6]은 전이  $t_1$ 의 출력 장소가 여럿이고, 최소받침  $T$ -불변이 두 개라면,  $\{(1 \ 1 \ 0), (1 \ 0 \ 1)\}$ , 출력 전이가

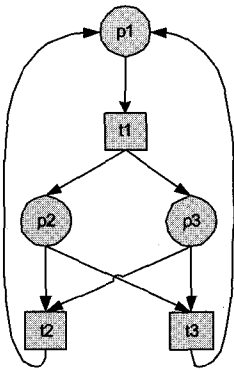
두 개인 장소가,  $p_2$ 와  $p_3$ , 반드시 존재함을 보이는 페트리 넷 예이다.



[그림 4] 최소받침  $T$ -불변이 유일한 페트리 넷



[그림 5]  $T$ -불변이 존재하지 않는 예



[그림 6] 최소받침  $T$ -불변이 두 개이면 선택이 발생함

[정리 4] 단계 2에서 구한 모든 홈 장소를 포함하는  $T$ -불변  $x_i$ 의 선택 확률  $p_i$ 의 합은 1이다.

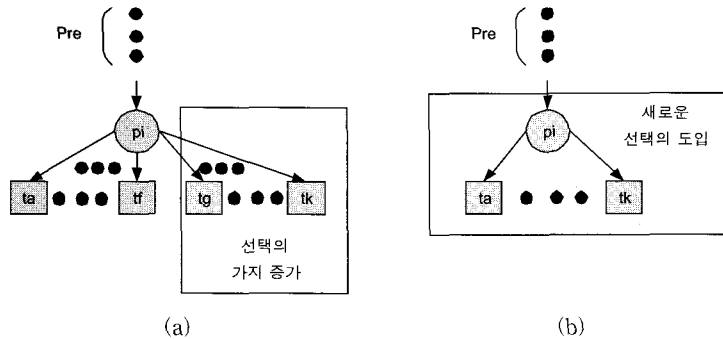
<증명> 주어진 페트리 넷의 모든 장소에 대하여 출력 전이가 유일하면 [정리 3]에 의하여 단계 2에

서 구한 홈 장소를 포함하는  $T$ -불변은 유일하며, 그 자체가 양의 수로 구성되고, 각 전이의 선택 확률이 1임으로 이  $T$ -불변의 선택 확률도 1이다. 예를 들어, [그림 4]의 유일한  $T$ -불변은  $(1\ 1\ 1)^T$ 이고, 이것의 선택 확률은  $1 \times 1 \times 1 = 1$ 이다.

출력전이가 유일하지 않은 장소가  $k$  개일 때 단계 2에서 구한 모든 홈 장소를 포함하는  $T$ -불변  $x_i$ 의 선택 확률  $p_i$ 의 합은 1이라고 가정하고 출력 전이가 유일하지 않은 장소가  $k+1$ 개일 때 단계 2에서 구한 모든 홈 장소를 포함하는  $T$ -불변  $x_i$ 의 선택 확률  $p_i$ 의 합은 1임을 증명하자.

새로운 선택([정의 3] 참조)이 도입되는 경우는 [그림 7](a)처럼 기존의 선택의 가치가 증가하는 경우와 [그림 7](b)처럼 새로운 장소와 출력 전이가 도입되는 경우다. 주어진 넷이 적당한 선택 확률 시간 넷임으로 새로 도입된 전이들을 포함하는  $T$ -불변이 반드시 존재한다. [그림 7]에서 Pre는 기존의 홈 장소를 포함하는  $T$ -불변 중, 홈 장소에서 새로 도입된 선택에까지 이르는 경로를 구성하는 전이들의 발화 회수 벡터들이다. 선택의 수가  $k$ 개일 때 선택 확률의 합은 1이라고 하였으므로 Pre의 선택 확률과 이 경로를 지나지 않는 홈 장소를 포함하는 다른  $T$ -불변들의 선택 확률의 합은 1이다.

[그림 7](a)에서 전이  $t_a, \dots, t_k$  각각의 선택 확률의 합은 1이고, 벡터의 선택 확률의 정의에 의하여 선택 관계의 두 전이,  $t_i, t_j$ 가 모두 같은  $T$ -불변  $x_i$ 의 받침이면  $x_i$ 의 선택 확률을 구할 때  $Prob(t_i) + Prob(t_j)$ 를 다른 전이들의 선택 확률에 곱하기 때문에 전이  $t_g, \dots, t_k$ 를 포함하는  $T$ -불변들을 비롯한 모든 Pre를 공유하는  $T$ -불변들의 선택 확률의 합은 Pre의 선택 확률과 동일하다. 다른 한편,  $t_i, t_j$ 가 서로 다른  $T$ -불변  $x_i, x_j$ 에 각각 속하면 모든  $T$ -불변의 선택 확률을 합산할 때  $Prob(t_i) + Prob(t_j)$ 가 됨으로 출력 전이가 유일하지 않은 장소가  $k$ 개일 때 단계 2에서 구한 모든 홈 장소를 포함하는  $T$ -불변  $x_i$ 의 선택 확률  $p_i$ 의 합이 1이라



[그림 7] 새로운 선택이 도입되는 두 가지 유형

면, 출력전이가 유일하지 않은 장소가  $k+1$ 개일 때 단계 2에서 구한 모든 홈 장소를 포함하는  $T$ -불변  $x_i$ 의 선택 확률  $p_i$ 의 합도 1이다.

[그림 7](b)의 경우도 [그림 7](a)의 경우와 마찬가지로 전이  $t_a, \dots, t_k$  각각의 선택 확률의 합은 1이고, 벡터의 선택 확률의 정의에 의하여 선택 관계의 두 전이,  $t_i, t_j$ 가 모두 같은  $T$ -불변  $T_i$ 의 받침이면  $T_i$ 의 선택 확률을 구할 때  $Prob(t_i) + Prob(t_j)$ 를 다른 전이들의 선택 확률에 곱하기 때문에 모든 Pre를 공유하는  $T$ -불변들의 선택 확률의 합은 Pre의 선택 확률과 동일하고,  $t_i, t_j$ 가 서로 다른  $T$ -불변  $x_i, x_j$ 에 각각 속하면 모든  $T$ -불변의 선택 확률을 합산할 때  $Prob(t_i) + Prob(t_j)$ 가 됨으로, [그림 7](a)의 경우와 마찬가지로 출력전이가 유일하지 않은 장소가  $k$ 개일 때 단계 2에서 구한 모든 홈 장소를 포함하는  $T$ -불변  $x_i$ 의 선택 확률  $p_i$ 의 합이 1이라면, 출력전이가 유일하지 않은 장소가  $k+1$ 개일 때 단계 2에서 구한 모든 홈 장소를 포함하는  $T$ -불변  $x_i$ 의 선택 확률  $p_i$ 의 합도 1이다.

#### 4. LBS 시스템에의 적용 예

위치기반서비스 제공자는 일반적으로 포털 사이트를 운영한다. 즉, 위치기반서비스 외에도 호텔이나 음식점 예약 서비스, 오락 서비스, 정보 검색 서비스, 등 일반적인 포털 시스템이 제공하는 모든 서

비스도 함께 제공한다. 그러므로 전체 가입자 중, 위치기반서비스 관련 메뉴를 선택하는 비율이 시스템 분석에서 고려되어야 시스템이 서비스할 수 있는 적정 가입자 수를 결정할 수 있다.

위치기반서비스에는 사용자의 현재 위치를 실시간으로 파악하여 현재 위치를 고려한 서비스를 제공하는 경우가 대부분이지만, 사용자가 지정하는 위치에 관련된 정보를 제공하는 경우도 적지 않다. 현재 위치를 실시간으로 파악하는 경우에도 가입자가 사용하는 단말기의 종류에 따라 사용자가 전송해온 명령어에 사용자 위치가 수록되어 있는 경우도 있고 위치제공자에게 가입자의 위치를 파악하여 줄 것을 의뢰해야 하는 경우도 있다. 각각의 경우마다 처리 시간이 상이함으로 정확한 시스템 분석을 하려면 각 경우의 비율을 반드시 고려해야 한다. 그러나 기존의 시간 넷에는 사건의 발생 확률을 표시하는 기능이 미비하고, 결과적으로 시간 넷의 최소 순회시간 분석 방법도 사건의 발생 확률을 고려하지 않는다. 따라서 본 논문이 제안하는 선택 확률 시간 넷이 필요하다.

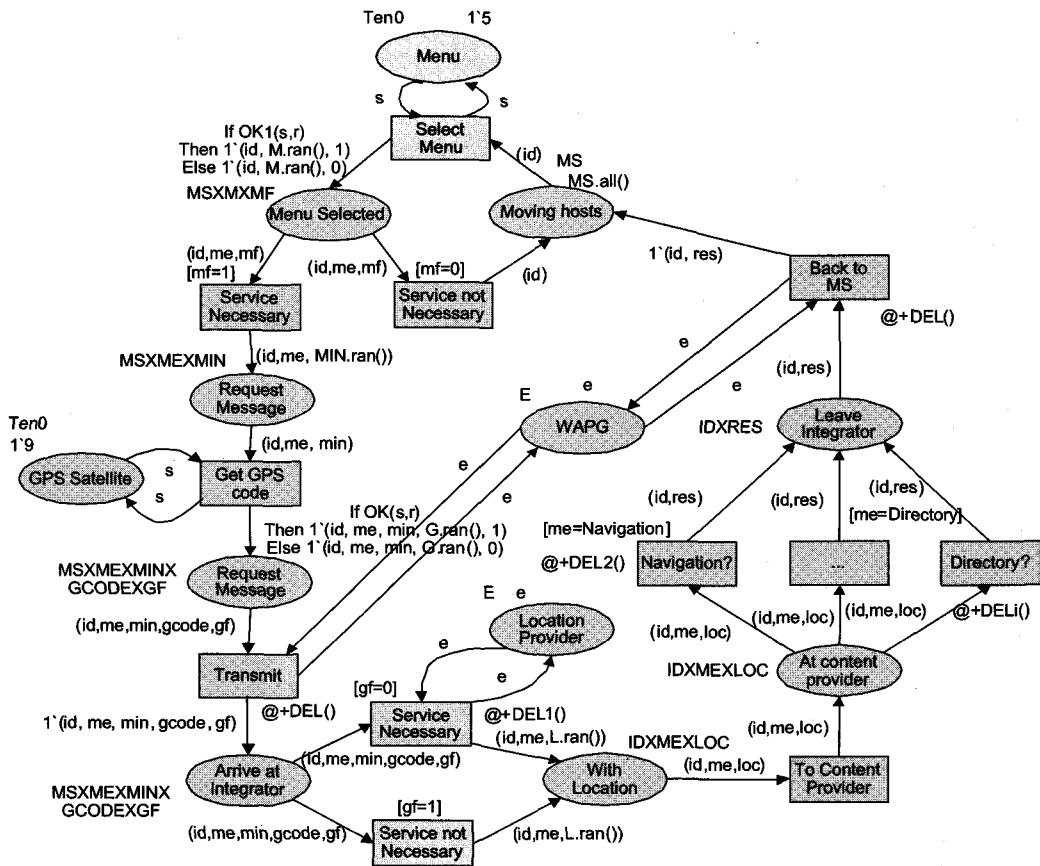
여기서는 본 연구에서 제안한 선택 확률 시간 넷을 이용한 위치기반서비스 시스템을 분석한 예를 제시한다.

Krishnamurthy[4]에 의하면 전통적인 HTTP/XML 방식의 위치기반서비스 시스템의 구성은 다음과 같다. 서비스 제공자(contents provider)는 일반적으로 서비스 포털 사이트를 운영한다. 사용자는 이 포털

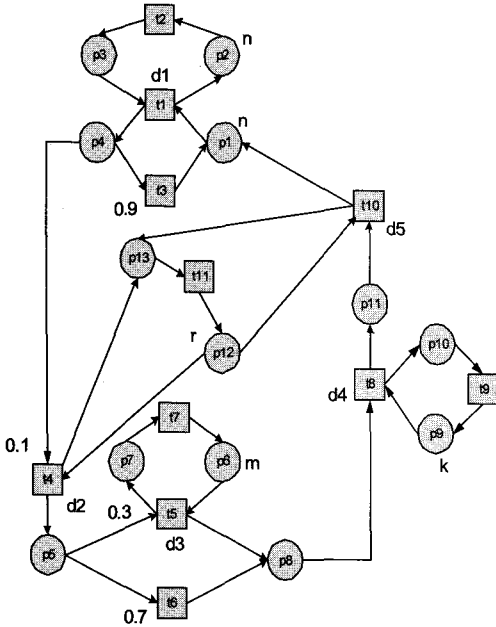
사이트의 URL을 알고, 브라우저를 사용하여 이곳에 접근한다. 포털 사이트의 웹 페이지는 WAPG(Wireless Application Protocol Gateway)의 도움을 받아 사용자 단말기에 출력된다. 사용자는 웹 브라우저의 메뉴를 선택하여 서비스를 요청한다. 사용자에게 서비스를 제공하려면 사용자의 위치 정보를 반드시 알아야 한다. 게이트웨이는 사용자가 요구한 명령을 응용 개발 플랫폼인 개방형 서비스 통합 플랫폼(이하 통합플랫폼이라 함)에 전송한다. 통합플랫폼은 사용자 요구로부터 MIN/MSI(Mobile Identification Number/International Mobile Subscriber Identification)와 같은 이동단말기의 정보를 얻는다. 위치제공자(location provider)로부터 단말기의 위치 정보를 얻는다. 이 위치정보를 가지고 통합플랫폼

은 서비스 제공자로부터 필요한 데이터를 얻는다. 그리고 이 정보를 사용자 단말기에 전달한다.

이것을 페트리 넷으로 표현하면 [그림 8]과 같고, [그림 8]을 선택 확률 시간 넷으로 표현하면 [그림 9]와 같다. [그림 9]의 장소  $p_4$ 에서 선택 ( $t_3, t_4$ )가 발생하며,  $Prob(t_3)=0.9, Prob(t_4)=0.1$ 이다. 이것은 가입자가 선택한 메뉴의 10%만이 디렉터리 서비스, 항법 서비스 등 위치기반서비스일 뿐, 나머지 90%는 오락이나 메시지 서비스와 같은 여타의 서비스임을 나타낸다. 선택 ( $t_5, t_6$ )는 30%만 위치 제공자의 도움으로 가입자의 위치를 파악하고, 나머지 70%는 사용자가 지정된 위치를 사용하거나 자체적으로 실행하는 간단한 조작으로 위치를 파악한다.



[그림 8] LBS 시스템의 페트리 넷 모형



[그림 9] LBS 시스템을 나타내는 선택 확률 시간 넷

[그림 9]의 페트리 넷을 행렬로 표기하면  $A$ 와 같다. 선택 확률 시간 넷의 최소 순회시간 분석 알고리즘의 단계 1에서 다음과 같은 최소 받침  $T$ -불변을 구한다.

$$x_1 = (1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)^T,$$

$$x_2 = (1 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 1 \ 1 \ 1 \ 2)^T,$$

$$x_3 = (1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 1 \ 2)^T.$$

[그림 9]의 홈 장소는  $p_1$ 이며,  $x_1, x_2, x_3$ 은 모두 홈 장소를 포함하는  $T$ -불변이다. 단계 3에서 각각의 선택 확률을 구하면 다음과 같다.

$$x_1 = 0.9,$$

$$x_2 = 0.07,$$

$$x_3 = 0.03.$$

단계 4에서 홈 장소를 포함하는  $T$ -불변 각각에 가중치를 곱하여 다음을 구한다.

$$0.9nx_1 = (0.9n \ 0.9n \ 0.9n \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)^T,$$

$$0.07nx_2 = (0.07n \ 0.07n \ 0 \ 0.07n \ 0 \ 0.07n \ 0 \ 0.07n \ 0 \ 0.07n \ 0.07n \ 0.07n \ 0.07n \ 0.14n)^T,$$

$$0.03nx_3 = (0.03n \ 0.03n \ 0 \ 0.03n \ 0.03n \ 0 \ 0.03n \ 0.03n \ 0.03n \ 0.06n)^T.$$

양의  $T$ -불변  $x$ 를 구하면 다음과 같다.

$$x = (n \ n \ 0.9n \ 0.1n \ 0.03n \ 0.07n \ 0.03n \ 0.1n \ 0.1n \ 0.1n \ 0.2n)^T. \quad (3)$$

식 (3)의 의미는  $t_1$ 과  $t_2$ 를  $n$ 번 발화하고, ...,  $t_{11}$ 을  $0.2n$ 회 발화함으로써, 모든 전이들을 빠짐없이 발화하고 원래의 마킹으로 되돌아오는 발화순서가 있다는 것이다.

$$A = \begin{matrix} & P_1 & P_2 & P_3 & P_4 & P_5 & P_6 & P_7 & P_8 & P_9 & P_{10} & P_{11} & P_{12} & P_{13} \\ \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \\ t_{10} \\ t_{11} \end{matrix} & \begin{bmatrix} -1 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & -1 & -1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 \end{bmatrix} \end{matrix}$$

[그림 9]에 대한  $S$ -불변은 다음과 같다 :

$$y_1^T = (0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0), \quad (4)$$

$$y_2^T = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1),$$

$$y_3^T = (0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0),$$

$$y_4^T = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0),$$

$$y_5^T = (1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \ 0 \ 0).$$

[그림 9]의 경우, 식 (2)에 사용되는 각 구성 요소를 살펴보면,  $x$ 는 식 (3)에 주어진  $T$ -불변,  $y_k$ 는 식 (4)에 주어진 다섯 개의  $S$ -불변,  $A^- = [a_{ij}^-]_{n \times m}$  with  $a_{ij}^- = w(P_i, t_j)$ , 대각행렬  $D = d_n$ ,  $i = 1, 2, \dots$ ,

$n$ , 그리고 초기 마킹  $M_0$ 은 다음과 같다.

$$M_0 = (n \ n \ 0 \ 0 \ 0 \ m \ 0 \ 0 \ k \ 0 \ 0 \ r \ 0)$$

$n$ 은 이동물체의 수,  $r$ 은 무선통신 에이전트의 수,  $m$ 은 위치제공자의 처리장치 수,  $k$ 는 콘텐츠 제공자의 처리장치 수를 의미한다. [그림 9]의 다섯 S-불변 각각에 대한 식 (2)의 분모를 구하면 다음과 같다.

$$\begin{aligned} y_1^T M_0 &= n, \quad y_2^T M_0 = r, \quad y_3^T M_0 = m, \\ y_4^T M_0 &= k, \quad y_5^T M_0 = n. \end{aligned}$$

[그림 9]의 행렬  $A^-$ 는 다음과 같다.

$$A^- = \begin{matrix} & P_1 & P_2 & P_3 & P_4 & P_5 & P_6 & P_7 & P_8 & P_9 & P_{10} & P_{11} & P_{12} & P_{13} \\ \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \\ t_{10} \\ t_{11} \end{matrix} & \begin{bmatrix} +1 & 0 & +1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & +1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & +1 & +1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & +1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \end{matrix}$$

또한 지속시간 행렬  $D$ 는 다음과 같다.

$$D = \begin{matrix} & t1 & t2 & t3 & t4 & t5 & t6 & t7 & t8 & t9 & t10 & t11 \\ \begin{matrix} t1 \\ t2 \\ t3 \\ t4 \\ t5 \\ t6 \\ t7 \\ t8 \\ t9 \\ t10 \\ t11 \end{matrix} & \begin{bmatrix} d1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & d2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & d3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & d4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & d5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{matrix}$$

따라서,  $(A^-)^T D$ 의 결과는 다음과 같다.

$$(A^-)^T \cdot D = \begin{matrix} p1 \\ p2 \\ p3 \\ p4 \\ p5 \\ p6 \\ p7 \\ p8 \\ p9 \\ p10 \\ p11 \\ p12 \\ p13 \end{matrix} \begin{bmatrix} d1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ d1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & d2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & d3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & d3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & d4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & d4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & d5 & 0 \\ 0 & 0 & 0 & d_2 & 0 & 0 & 0 & 0 & 0 & 0 & d5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

결과적으로,

$$\begin{aligned} (A^-)^T D x &= (nd_1 \ 0 \ nd_1 \ 0.1nd_2 \ 0.03nd_3 \ 0.03nd_3 \\ &\quad 0.03nd_3 \ 0 \ 0.1nd_4 \ 0.1nd_4 \ 0 \ 0.1nd_5 \\ &\quad 0.1n(d_2 + d_5) \ 0)^T \text{이다.} \end{aligned}$$

$$y_1^T (A^-)^T D x / y_1^T M_0 = nd_1 / n = d_1,$$

$$y_2^T (A^-)^T D x / y_2^T M_0 = 0.1n(d_2 + d_5) / r,$$

$$y_3^T (A^-)^T D x / y_3^T M_0 = 0.03nd_3 / m,$$

$$y_4^T (A^-)^T D x / y_4^T M_0 = 0.1nd_4 / k,$$

$$\begin{aligned} y_5^T (A^-)^T D x / y_5^T M_0 &= (nd_1 + 0.1nd_2 + 0.03nd_3 \\ &\quad + 0.1d_4 + 0.1nd_5) / n \\ &= d_1 + 0.1d_2 + 0.03d_3 \\ &\quad + 0.1d_4 + 0.1d_5. \end{aligned}$$

따라서 [그림 9]에 대한 최소 순회시간은 다음과 같다.

$$\begin{aligned} &Max\{d_1, 0.1n(d_2 + d_5) / r, 0.03nd_3 / m, 0.1nd_4 / k, \\ &\quad (d_1 + 0.1d_2 + 0.03d_3 + 0.1d_4 + 0.1d_5)\} \quad (5) \end{aligned}$$

지속시간  $d_1$ 은 사용자가 선택한 메뉴를 판단하여 위치기반서비스 서버로 송신할 것인지를 판단하는 시간이고,  $d_2$ 는 단말기에서 통합플랫폼으로 메시지를 전달하는 통신 시간이며,  $d_5$ 는 통합플랫폼에서 단말기로 결과를 전달하는 통신시간이다.  $d_3$ 는 위치제공자의 처리시간인데, 위치제공자의 처리

장치의 수가 증가하면 처리 시간이 짧아질 수 있다.  $d_4$ 는 콘텐츠 제공자의 처리시간이며, 이 시간도 역시 처리장치의 수가 증가하면 짧아질 수 있다.

예를 들어, 지연 시간과 자원의 수가 다음과 같은 경우를 고려하여 보자.

$$d_1 = 50, \quad d_2 = 100, \quad d_3 = 500, \quad d_4 = 1,000, \quad d_5 = 100, \\ n = 1,000, \quad k = 1, \quad m = 1, \quad r = 1.$$

최소 순회시간은 다음과 같다.

$$\text{MAX}\{50, 20,000, 1,500, 100,000, 185\} = 100,000$$

최소 순회시간 100,000의 의미는 이 LBS 시스템의 반응시간이 100,000 단위 시간이며, 이는 콘텐츠 제공자에 의하여 결정된다는 것이다. 따라서 반응시간 100,000 단위 시간이 서비스의 질과 관련한 사용자의 요구 사항을 만족하면, 위치정보 제공자나 통신과 같은 다른 자원을 절약하여 시스템 비용을 줄이는 방안을 모색할 수 있으며, 그 반대로 사용자 요구 사항을 만족시키지 못하면 콘텐츠 제공자를 개선하여 지연 시간을 짧게 하든지, 처리장치를 늘리는 방법을 고려해야 한다.

제한하는 방법을 사용하지 않으면, 양의  $T$ -불변  $x$ 를 다음과 같이 구할 수 있다.

$$x = x_1 + x_2 + x_3 = (3 \ 3 \ 1 \ 2 \ 1 \ 1 \ 1 \ 2 \ 2 \ 2 \ 4)^T.$$

이 경우,  $(A^-)^T D x = (3d_1 \ 0 \ 3d_1 \ 2d_2 \ d_3 \ d_3 \ 0 \ 2d_4 \ 0 \ 2d_4 \ 2d_5 \ 2(d_2 + d_5) \ 0)^T$ 이며, 식 (2)의 값은 다음과 같다.

$$y_1^T (A^-)^T D x / y_1^T M_0 = 3d_1/n,$$

$$y_2^T (A^-)^T D x / y_2^T M_0 = 2(d_2 + d_5)/r,$$

$$y_3^T (A^-)^T D x / y_3^T M_0 = d_3/m,$$

$$y_4^T (A^-)^T D x / y_4^T M_0 = 2d_4/k,$$

$$y_5^T (A^-)^T D x / y_5^T M_0 = (3d_1 + 2d_2 + d_3 + 2d_4 + 2d_5)/n.$$

따라서 [그림 9]에 대한 최소 순회시간은 다음과

같이 된다.

$$\text{Max}\{3d_1/n, 2(d_2 + d_5)/r, d_3/m, 2d_4/k, \\ (3d_1 + 2d_2 + d_3 + 2d_4 + 2d_5)/n\}$$

지연 시간과 자원의 수를 선택 확률 시간 넷과 동일하게 설정하면, 최소 순회시간은 다음과 같다.

$$\text{MAX}\{0.15, 400, 500, 2,000, 3.05\} = 2,000.$$

이 경우에는  $x$ 가  $t_1$ 을 세 번 발화함으로  $x$ 를 약 333번 수행해야 장소  $p_1$ 의  $n$  토큰을 모두 발화에 사용할 수 있다. 따라서  $n$  가입자가 모두 서비스를 받는데 걸리는 시간은  $2,000 \times 333 = 666,000$  단위 시간으로 계산된다. 이 시간은 선택 확률 시간 넷에서 계산한 100,000 단위 시간과 큰 차이가 난다. 즉, 사건의 발생 확률을 고려한 시스템 분석에서는 선택 확률 시간 넷을 사용해야 함을 알 수 있다.

## 5. 결 론

실세계 시스템에서는 어떤 사건이 선택되면 다른 사건은 발생되지 않는 선택적인 경우가 대부분이다. 또한 각 사건이 선택될 확률도 다르다. 그러나 현재까지 연구된 실세계의 사건을 나타내는 시간 넷의 전이에는 선택될 확률이 명시되어 있지 않았다. 이는 결국 모든 사건의 선택 확률이 동일하다는 비현실적인 가정을 하고 있다는 문제점이 내재되어 있음을 의미한다.

본 논문에서는 시간 넷의 전이에 선택 확률을 명시할 수 있는 '선택 확률 시간 넷'을 최초로 제안하고, 그 최소 순회시간을 찾는 알고리즘을 제시하였다. 또한 자연어로 표현된 위치기반서비스 시스템을 '선택 확률 시간 넷'으로 표현하고, 그 알고리즘을 적용하여 정확한 최소 순회시간을 분석하였다. 가입자는 일반적으로 포털 사이트에 접속하여 위치기반서비스를 요구한다. 따라서 가입자는 위치기반서비스를 선택할 수도 있고 다른 종류의 서비스를 선택할 수도 있다. 위치기반서비스 중 일부는 위치

제공자의 서비스를 필수로 하지만 나머지는 그렇지 않다. 또한 위치기반서비스에 속하는 서비스의 종류도 다양하기 때문에 결국 이들을 나타내는 전이에는 필연적으로 선택될 확률이 연합되어야 한다. 따라서 위치기반서비스를 나타내는 페트리 넷 모델도 [그림 9]와 같은 선택 확률 시간 넷이 된다. [그림 9]에 보이는 선택 확률 시간 넷은 최소 순회시간 분석에 적당하나, 자연어로 표현된 위치기반서비스 시스템에 대한 설명과는 거리가 크다. 따라서 [그림 9]를 구하기 전에 자연어 설명에 가까운 [그림 8]을 먼저 구한 다음, 이로부터 [그림 9]를 구하였다.

제안하는 선택 확률 시간 넷의 최소 순회시간 분석 방법이 기존의 최소 순회시간 분석과 다른 점은  $T$ -불변을 구할 때 전이의 선택 확률을 고려한다는 것이다. 이를 위하여, 최소반침  $T$ -불변들을 이웃한  $T$ -불변들과 합병하고, 합병 결과들을 선택 확률을 바탕으로 구한 가중치를 고려하여 합함으로써  $T$ -불변을 구하는 알고리즘을 <표 2>에 제공하였다.

최소 순회시간 분석 결과는 일목요연한 하나의 수학적식으로 표현되며, 시스템의 병목을 정확히 지적하여 준다. 따라서 시스템의 성능을 개선하려면 바로 그 병목을 개선하면 된다. 반대로 시스템의 성능이 충분할 경우에는, 병목을 기준으로 다른 처리 과정에 할당된 자원들은 절감하여도 전체시스템의 성능에 아무런 영향을 주지 않는다는 의미를 분석 결과가 나타내기 때문에 자원을 절감하는 방안도 시사한다. 즉, 최소 순회시간 분석 방법은 사용자 요구 사항을 만족하면서, 경비를 최소화하는 최적의 시스템 설비 구현 방안을 찾아주는 도구로 사용될 수 있다.

최소 순회시간 분석 방법을 사용하려면 시스템 명세를 페트리 넷 모형으로 변형해야 한다. 이 단계는 응용분야와 페트리 넷 분야에 대한 전공 지식을 소유하고, 창의적인 사고 능력을 소유한 숙련가만이 수행할 수 있다. 그러나 이러한 숙련가를 실제 산업 현장에서 찾기가 쉽지 않다. 향후에는 자연어 명세를 페트리 넷 모형으로 변형하는 자동시스템 개발에 관하여 연구하고자 한다.

## 참 고 문 헌

- [1] Dasdan, A. and R.K. Gupta, "Faster Maximum and Minimum Mean Cycle Algorithms for System Performance Analysis," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, Vol.17, No.10(1998), pp. 889-899.
- [2] Desel, J., Wolfgang Reisig, and Grzegorz Rozenberg (Eds.), *Lectures on Concurrency and Petri Nets, Advances in Petri Nets, Lecture Notes in Computer Science*, Vol. 3098, Springer-Verlag, 2004, ISBN : 3-540-22261-8.
- [3] Hillion, Hervé P., "Timed Petri Nets and Application to Multi-Stage Production Systems," *Lecture Notes in Computer Science*, Vol.424 ; *Advances in Petri Nets* 1989, Berlin, Germany, Springer-Verlag(1990), pp. 281-305.
- [4] Krishnamurthy, N., "Using SMS to Deliver Location-based Services," *Proceedings of 2002 IEEE International Conference on Personal Wireless Communications (Proceedings of ICPWC'2002)*, Dec. 15-17, 2002, pp.177-181.
- [5] Murata, T., "Petri nets : Properties, Analysis and Applications," *Proceedings of the IEEE*, Vol.77. No.4(April 1989), pp.541-580.
- [6] Nakamura, M. and M. Silva, "Cycle Time Computation in Deterministically Timed Weighted Marked Graphs," *Proceedings. ETFA '99. 7th IEEE International Conference on Emerging Technologies and Factory Automation 1999*, Vol. 2(Oct. 18-21, 1999), pp.1037-1046.
- [7] Peterson, J.L., *Petri Net Theory and the Modeling of Systems*, Prentice-Hall, N.J.,



- 1981, ISBN : 0-13-661983-5.
- [8] Reisig, W., *Petri Nets, An Introduction*, EATCS, *Monographs on Theoretical Computer Science*, W.Brauer, G. Rozenberg, A. Salomaa (Eds.), Springer Verlag, Berlin, 1985.
- [9] Rodriguez-Beltran, J. and A. Ramfrez-Trevino, "Minimum Initial Marking in Timed Marked Graphs," *2000 IEEE International Conference on Systems, Man, and Cybernetics*, Vol. 4(Oct. 8-11, 2000), pp. 3004-3008.