

추계적 페트리 넷를 이용한 대기시스템의 제어모형

이광식* · 이효성**

Control Models for Queueing Systems Using Stochastic Petri Nets.

Kwang Sik Lee · Hyo-Seong Lee

(Abstract)

In this paper, a threshold policy is considered for the Markovian queueing system with server vacations. The threshold policy considered in this paper has the following form: "when the number of customers present in the system increases to N , the server is turned on and serves customers until the system becomes empty". In this paper, we show how the finite capacity or finite population queueing system under a threshold policy can be modeled by the stochastic Petri net. The performance evaluation of the model is carried out using the software called "SPNP". Some examples are also presented in which it is shown that how the optimal threshold policies can be obtained under a linear cost structure.

1. 서론

대기시스템을 경제적으로 운용하기 위해서는 대기시스템을 효율적으로 제어하기 위한 정책이 필요하다. 대기시스템의 제어에 대한 연구는 주어진 비용구조하에서 시스템을 가장 경제적으로 운용할 수 있는 정책을 도출함을 목적으로 하며 서비스 속도나 서버의 서비스 시점, 고객의 도착률 등을 조절함으로써 대기시스템의 제어가 이루어진다[1, 2]. 특히 서버의 유휴기간이 끝나고 서비스가 시작될 때 준비비용(setup cost)이 발생하는 경우에는 빈번한 준비비용의 발생을 피하기 위하여 대기중인 고객의 수가 특정 한계치를 넘어섰을 때 서비스를 시작하는 제어정책을 사용하게 된다. 이런 형태의 제어정책을 한계치를 이용한 제어

정책(threshold control policy)이라 하며, 이에 대해서는 Yadin[3], Heyman[4, 5], Bell[6] 등의 연구를 기점으로 그동안 많은 연구가 수행되어져 왔다. 최근에는 서버 휴가모형과 관련하여 한계치를 이용한 대기시스템의 제어정책이 활발히 연구되어지고 있다. Kella[7]는 한계치를 이용하여 $M/G/1$ 서버 휴가 시스템의 최적 제어정책을 연구하였으며 Lee와 Srinivasan[8]은 한계치를 이용하여 $M^x/G/1$ 서버 휴가시스템의 최적 제어정책을 연구하였다. Lee외[9]은 부가변수법(supplementary variable technique)을 이용하여 동일한 제어정책을 사용하는 $M^x/G/1$ 서버 휴가시스템의 운용특성을 연구하였다.

앞에서 언급된 연구는 모두 대기시스템의 용량이 무한하고 입력원(input source)의 크기가 무한하다는

* 고등기술연구원

** 경희대학교 공과대학 산업공학과

가정 하에서 수행되었다. 그러나 실사회에 존재하는 대기시스템은 용량이 유한하거나 대기시스템을 이용하는 고객의 수가 한정되어 있는 경우가 많다. 용량이 유한한 경우의 한계치를 이용한 제어정책 연구는 Hersh와 Brosh[10], Teghem[11], Lee[12] 등에 의해 수행되었으나 이들 연구에서는 중요한 비용요소인 고객의 대기비용(waiting cost)이 고려되지 않았다는 결함이 지적된다. 본 연구에서는 대기시스템의 용량이 유한하거나 입력원의 크기가 유한한 경우의 대기시스템의 제어정책을 추계적 페트리 넷(stochastic Petri net)을 이용하여 모형화하고 대기비용을 포함한 비용구조하에서 단위기간당 기대비용을 최소화 해주는 최적제어정책을 구해 보고자 한다.

페트리 넷을 이용한 접근방법은 추계적 이론을 이용한 접근방법에 비하여 다음과 같은 장점이 존재한다. 첫째, 페트리 넷은 추계적 이론에 대한 지식을 갖고 있지 않은 비전문가라도 단기간에 습득할 수 있으며, 이를 통하여 분석하고자 하는 추계적 시스템을 쉽게 모형화할 수 있다. 둘째, 페트리 넷을 구성하는 place와 transition의 수는 추계적 시스템이 취할 수 있는 상태의 수(number of states)에 비하여 일반적으로 극히 작다. 따라서 추계적 시스템의 페트리 넷 표현은 추계적 모형에 비하여 단순하다.

분석하고자 하는 추계적 시스템이 페트리 넷으로 모형화되면, 성능분석은 페트리 넷에 내재되어 있는 marking process를 추계적 이론을 통해 분석함으로써 수행된다. 그러나 페트리 넷 그래프가 주어지면 이러한 성능분석 과정을 자동으로 수행해 주는 소프트웨어가 Chiola[13], Ciardo[14] 등에 의해 개발되어 있다. 따라서 이러한 소프트웨어를 이용하면 사용자는 단지 시스템을 페트리 넷으로 표현해 주기만 함으로써 시스템의 필요한 성능분석치를 제공받을 수 있게 된다. 본 연구에서는 용량이 유한하거나 입력원의 크기가 유한한 대기시스템의 한계치를 이용한 제어정책을 추계적 페트리 넷으로 모형화한 후 Ciardo에 의해 개발된 소프트웨어인 SPNP(Stochastic Petri Net Package)를 이용하여 제어정책의 성능분석을 수행하고자 한다. 시스템의 성능분석이 SPNP에 의해 자동적으로 수행되기 위해서는 추계적 시스템의 상태가 모든 시

점에서 기억부재성질(memoryless property)을 가지고 있어야만 한다. 따라서 본 연구에서 분석될 대기시스템은 도착시간 간격과 서비스 시간이 지수분포를 따르는 대기시스템으로 국한된다. 본 연구와 관련된 기존의 연구로는 Ibe와 Trivedi[15]의 연구를 들 수 있다. 이 연구에서는 제어가 존재하지 않는 대기시스템을 페트리 넷으로 모형화한 후 SPNP를 이용하여 시스템의 성능분석치를 구하는 방법이 제시되었다. 따라서 본 연구는 Ibe와 Trivedi의 연구를 제어가 존재하는 경우로 확장한 연구이며, 시스템의 성능분석치 뿐만 아니라 선형비용구조를 가정한 후 단위기간당 기대비용을 최소화해 주는 최적 제어정책을 본 연구에서는 구하게 된다.

2. 페트리 넷의 개요

본 절에서는 페트리 넷에 익숙하지 않은 독자를 위해 추계적 페트리 넷에 대해 간단히 설명하고자 한다. 페트리 넷은 이산형 사건 동적 시스템(discrete event dynamic system)에서 정보와 제어의 흐름을 모형화하는 데 적합한 그래프 모델이다. 페트리 넷은 가용한 자원을 토큰(token)으로 표시하며, 토큰의 이동조건을 논리적으로 표현하고 그 이동과정을 추적함으로써 시스템의 동적특성을 정확히 모델링할 수 있게 하여준다. 특히 페트리 넷은 concurrency, synchronization, fork-join, conflict 등 특수한 형태의 activity를 정확하게 표현할 수 있는 표현력을 구비하고 있어 이러한 특성을 지닌 동적시스템을 모델링하는 데 이상적인 도구이다. 페트리 넷은 시스템의 모델링, validation, verification, 성능분석등을 통해 시스템의 설계단계에서 이용될 수 있으며, 시스템이 구축된 후에는 시스템의 상태를 감시(monitor)하고 복잡한 활동을 제어하는 제어기(controller)로서의 역할로 전환 가능해 시스템의 전순기(life cycle)를 통해 활용되어질 수 있는 유용한 도구로 평가되어 진다.

페트리 넷에서는 시스템의 상태를 marking이라 표현하며 marking은 각 place에 존재하는 토큰의 수에 의해 표시된다. 시간형 페트리 넷(timed Petri net)에서는 토큰의 이동조건이 충족되면(enabled) 일정한

시간(transition time)이 경과한 후 토큰의 이동이 이루어지며 특히 transition time이 0(immediate transition)이거나 지수분포를 따를 경우(exponential transition)의 페트리 넷을 일반형 추계적 페트리 넷(generalized stochastic Petri net)라 부른다.

본 연구에서는 일반형 추계적 페트리 넷을 편의상 추계적 페트리 넷이라 부르도록 하겠다. 페트리 넷에서는 transition time이 경과해 토큰의 이동이 이루어지는 현상을 transition이 발사(fire)되었다고 하며, transition이 발사되면 토큰이 이동되고 이에 따라 marking에 변화가 생긴다. 시간에 따른 marking의 변화과정을 marking process라 부르며, 추계적 페트리 넷에서는 marking process가 연속시간형 마코브 체인(continuous time Markov chain)과 동일함을 증명할 수 있다. SPNP 등 소프트웨어에서는 추계적 페트리 넷 그래프가 주어지면 시스템의 상태의 수(number of states)가 지나치게 크지 않을 경우 marking process를 자동적으로 분석함으로써 시스템의 성능평가치를 제공하여 주고 있다.

3. 추계적 페트리 넷을 이용한 대기시스템의 제어모형

본 연구에서 분석하고자 하는 대기 시스템 및 그 제어정책은 다음과 같다. 서비스를 받기 위한 고객의 도착시간 간격과 서버가 한명의 고객을 서비스하는데 소요되는 시간은 모두 지수분포를 따른다. 서비스는 한명의 서버에 의해 개인별로 이루어진다. 일단 서비스가 시작되면 서버는 대기 시스템에 고객이 존재하지 않을 때까지 서비스를 계속한다(exhaustive service discipline). 서비스가 끝나 대기 시스템에 고객이 존재하지 않으면 서버의 가동기간(busy period)이 끝나고 유휴기간(idle period)이 시작된다. 유휴기간은 도착한 고객의 수가 일정한계치 N 을 넘어서면 끝나게 되고, 유휴기간이 끝나는 대로 바로 가동기간이 시작된다. 이와 같은 형태의 제어정책을 N -정책이라 부르며 N -정책은 유휴기간중에 취하는 서버의 행태(behavior)에 따라 다음과 같이 세가지 형태로 분류되어질 수 있다. 첫째, 유휴기간동안 서버가 시스

템을 떠나지 않고 고객을 기다리는 경우(서버의 휴가가 존재하지 않는 경우), 둘째, 가동기간이 끝나면 서버는 휴가를 반복해서 떠나며, 서버가 휴가에서 돌아온 순간 고객의 수가 제어 한계치, N ,을 최초로 넘어서면 서비스를 시작하는 모형(서버 복수 휴가 모형), 셋째, 서버는 휴가를 한번 떠나며, 서버가 휴가에서 돌아온 순간, 고객의 수가 N 미만이면 N 에 도달될 때까지 서버가 시스템에서 기다리는 모형(서버 단일 휴가모형)으로 분류된다. 본 연구에서는 위의 세 경우에 대해 페트리 넷을 이용한 제어모형을 수립하고자 한다. 본 연구에서는 또한, 가동기간이 시작되기전 일정한 기간의 준비시간(setup time)이 필요하거나 서버가 서비스 도중 고장이 발생하는 경우도 분석된다.

3.1 서버의 휴가가 존재하지 않는 경우의 페트리 넷 모형

(a) 입력원의 크기가 유한한 경우(기계 수리공 모형)

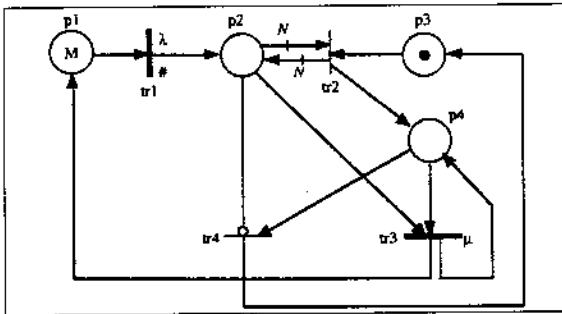
서버의 휴가가 존재하지 않는 모형은 가동기간이 끝나도 서버가 대기 시스템을 떠나지 않고 N 명의 고객이 도착할 때까지 시스템 내에서 고객을 기다리는 경우의 모형이다. 입력원(input source)은 M 명이라 가정하며 대기 시스템의 용량은 M 명의 고객이 모두 수용될 수 있을 정도로 크다고 가정한다. 따라서 본 모형에서는 n 명의 고객이 대기 시스템 내에 존재하면 잠재적 고객의 수는 $(M-n)$ 이 된다. 고객의 도착률은 상태 의존적이며 잠재적 고객의 수가 k 명일 경우 다음고객이 도착할 때까지의 소요시간은 평균이 $\frac{1}{\lambda_k}$ 인 지수분포를 따른다. 서버의 수는 1명이며 고객 한명이 서비스 받는 데 소요되는 시간은 평균이 $\frac{1}{\mu}$ 인 지수분포를 따른다. 이 모형은 문헌에서 기계수리공 모형(machine repairman model 또는 machine interference model)이라 불리며 M 대의 기계가 가동되는 상황에서 한명의 수리공이 고장난 기계의 수리를 담당하는 상황을 나타낸다. 이때 수리공은 빈번한 준비비용의 발생을 피하기 위하여 고장난 기계의 수가 N 에 이를 때까지 수리의 시작을 지연시키는 N -정책을 사용할 수

있다. N -정책을 사용하는 기계 수리공 문제를 추계적 페트리 넷트를 이용하여 모형화 하면 <그림 1>과 같다. <그림 1>에서 immediate transition은 선(bar)으로, exponential transition은 굵은 선(thick bar)으로 표시되었으며 <그림 1>을 보다 상세히 설명하면 다음과 같다.

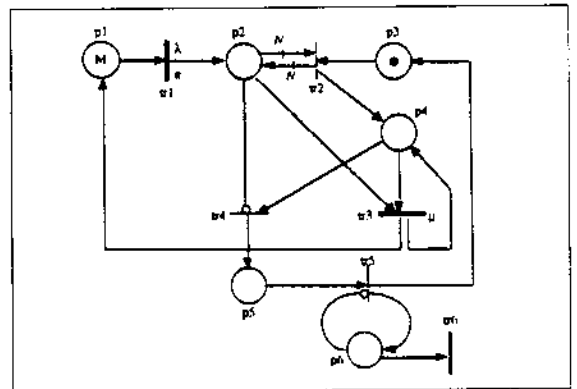
place p_1 은 잠재적 고객을 나타내며 p_2 는 대기 시스템에 존재하는 고객을 나타낸다. place p_3 는 유휴상태에 있는 서버를 나타내고 p_4 는 가동상태에 있는 서버를 나타낸다. 따라서 <그림 1>에 표시된 시스템의 초기상태는 대기 시스템 내에 고객이 한명도 없고 서버는 유휴상태에 있음을 나타낸다. Transition tr_1 은 고객의 도착을 나타내는 transition으로 고객의 도착률이 잠재적 고객의 수에 의존하므로 tr_1 의 발사율(firing rate)도 p_1 에 있는 토큰의 수에 따라 달라진다. 이와 같이 marking dependent한 tr_1 의 발사율은 <그림 1>에서 #에 의해 표시되어 있다. tr_2 는 p_2 에 토큰이 N 개가 있고 p_3 에 토큰이 존재하여야만 enabled되며, tr_2 가 발사되면 p_3 의 토큰은 p_4 로 이동하여 서버가 가동상태로 바뀌었음을 나타내 준다. tr_3 는 서비스가 진행중임을 의미하며 tr_3 가 발사되면 한개의 토큰(고객)이 서비스를 끝내고 p_1 으로 이동되어 잠재적 고객이 된다. 이때 서버는 또다른 고객을 서비스하기 위하여 p_4 로 돌아오며, 이러한 과정은 p_2 에 토큰이 존재하지 않을 때까지 반복된다. tr_3 가 발사된 후 p_2 에 토큰이 더이상 존재하지 않으면 tr_4 가 발사되며 p_4 에 있던 토큰은 p_3 로 이동하여 서버가 유휴상태로 바

뀌었음을 나타내 준다.

N -정책하에서 단위기간당 발생하는 기대비용을 구할 수 있기 위해서는 대기 시스템에 존재하는 평균 고객수와 사이클(유휴기간이 시작된 시점으로부터 다음 유휴기간이 시작된 시점까지의 간격)의 기대길이를 구할 수 있어야만 한다. SPNP에서는 각 place에 존재하는 토큰수의 기대치와 안정상태확률 그리고 exponential transition의 throughput(단위시간당 기대 발사수)을 자동적으로 계산하여 제공하여 준다. 대기 시스템에 존재하는 평균 고객 수는 p_2 에 존재하는 토큰의 기대치와 동일하므로 쉽게 구하여질 수 있다. 사이클의 기대길이는 tr_4 의 throughput의 역수이므로 tr_4 의 throughput을 구할 수 있으면 계산가능하다. 그러나 tr_4 는 immediate transition이므로 SPNP로 부터 throughput 값을 제공받을 수 없으며 페트리 넷트의 구조상 다른 exponential throughput으로부터도 구하여질 수 없다. 따라서 <그림 2>에서와 같이 p_5, p_6, tr_5, tr_6 을 추가하여야만 SPNP로 부터 사이클의 기대길이를 구할 수 있게 된다. <그림 2>에서 tr_4 가 발사된 후 p_6 에 토큰이 존재하지 않으면 tr_5 가 즉시 발사된다. tr_5 가 발사되면 토큰하나가 p_6 로 이동되고 이에 의해 tr_6 가 enabled 된다. tr_6 의 발사율은 tr_3 의 발사율 μ 와 비교할 수 없을 정도로 크게 정해 줌으로써 tr_6 는 enabled되자마자 발사되도록 만들어 준다. 따라서 tr_4 가 발사되는 시점에서는 p_6 에 토큰이 존재하지 않으며 tr_5 가 발사된 직후 tr_6 도 발사되므로 exponential transition tr_6



<그림 1> 입력원의 크기가 유한한 대기 시스템의 페트리 넷트 모형

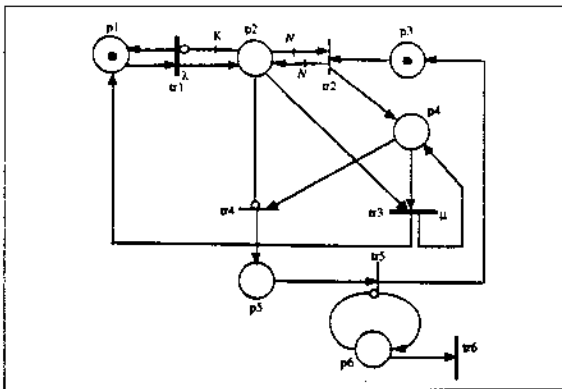


<그림 2> 사이클의 기대길이를 구하기 위한 페트리 넷트 모형

의 throughput은 immediate transition tr_4 의 throughput과 동일한 값을 갖게 된다.

(b) 대기 시스템의 용량이 유한한 경우

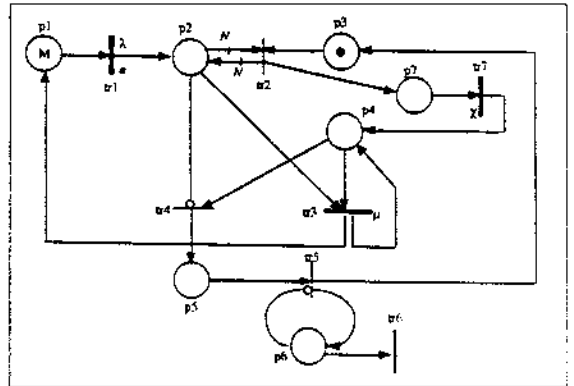
다음은 입력원의 크기는 무한하나 대기 시스템의 용량이 유한하여 유실되는 고객(lost customer)이 발생하는 경우이다. 대기 시스템의 용량은 K 이며, 대기 시스템에 K 명의 고객이 존재할 때 도착한 고객은 대기 시스템으로의 입장이 허용되지 못하고 유실된다. 서비스를 받기 위한 고객은 도착률이 λ 인 포아송 과정에 따라 도착하며 그외의 상황은 기계수리공 모형과 동일하다. 이 모형을 페트리 넷트로 표현하면 <그림 3>과 같다. <그림 3>에서는 포아송 과정에 따라 도착하는 고객의 도착과정을 나타내기 위하여 p_1 에는 토큰하나를 상존 시키고 tr_1 의 발사율은 λ 로 고정시켰다. 또한 대기 시스템에 존재하는 고객의 수가 K 명이하일 경우에만 tr_1 이 enabled 될 수 있도록 inhibitor arc을 사용하였다. 그 외의 페트리 넷트 표현은 <그림 2>와 동일하다.



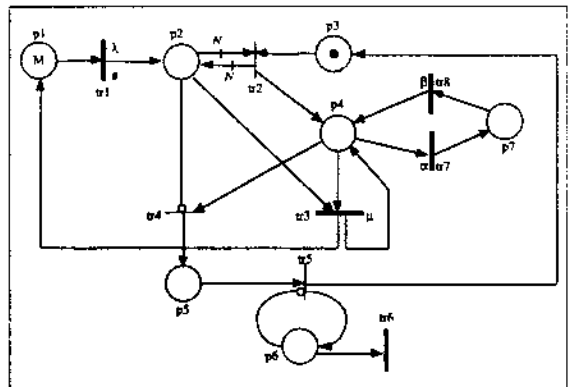
<그림 3> 용량이 유한한 대기 시스템의 페트리 넷트 모형

서버가 서비스를 시작할 때 작업준비시간(setup time)이 소요되는 경우와 서비스중 서버의 고장이 발생하는 경우의 대기 시스템도 추계적 페트리 넷트로 쉽게 모형화할 수 있다. <그림 4>는 입력원의 크기가 M 으로 유한하며 작업준비시간의 분포가 평균 $\frac{1}{\lambda}$ 인 지수분포를 따를 경우의 페트리 넷트 모형이다. <그림 4>는 <그림 2>에 작업준비중임을 나타내는 p_7 과

tr_7 을 추가시켰으며, 이에 의해 작업준비시간이 존재하는 상황이 페트리 넷트로 정확히 표현된다. <그림 5>는 입력원의 크기가 M 으로 유한하며 서비스중 서버의 고장이 발생할 수 있는 경우의 페트리 넷트 모형이다. 고장이 발생할 때까지의 시간은 평균 $\frac{1}{\alpha}$ 인 지수분포를 따르고, 고장을 수리하는 데 소요되는 시간은 평균 $\frac{1}{\beta}$ 인 지수분포를 따른다고 가정한다. 이와 같은 상황도 <그림 2>에 p_7 , tr_7 , tr_8 을 단순히 추가함으로써 페트리 넷트로 정확히 표현됨을 알 수 있다.



<그림 4> 준비시간이 존재하는 대기 시스템의 페트리 넷트 모형



<그림 5> 서버의 고장이 발생할 수 있는 대기 시스템의 페트리 넷트 모형

3.2 서버 단일 휴가 시스템의 페트리 넷트 모형

서버 단일 휴가 시스템에서는 가동기간이 끝난후 서버는 단 한번만의 휴가를 갖는다. 휴가기간의 길

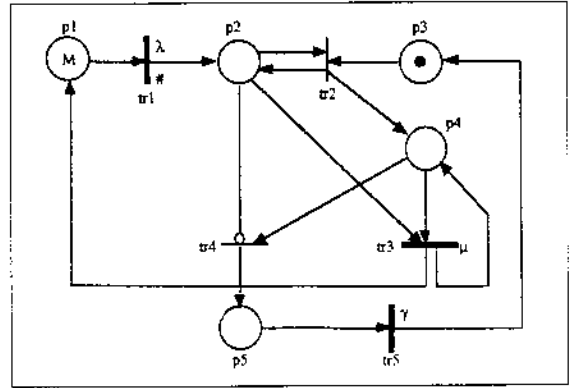
이는 평균 $\frac{1}{\gamma}$ 인 지수분포를 따르며, 휴가에서 돌아왔을 때 시스템에 대기중인 고객의 수가 N 이상이면 서버는 즉시 서비스를 시작하고 N 미만이면 고객의 수가 N 에 이를 때까지 시스템을 떠나지 않고 기다린다. 이와 같이 서버가 단일 휴가만을 취할 때 입력원의 크기가 M 으로 유한한 대기 시스템의 페트리 넷 모형은 <그림 6>과 같다. <그림 6>에서는 가동기간이 끝난 후 서버가 휴가중임을 표현하기 위하여 <그림 1>에 p_5 와 tr_5 를 추가하였으며 이에 대한 설명은 다음과 같다. 가동기간이 끝나 p_2 에 더이상의 토큰(고객)이 존재하지 않으면 p_4 의 토큰은 p_5 로 이동된다. p_5 는 서버가 휴가중임을 나타내며, 토큰이 p_5 에 체류하는 기간은 평균 $\frac{1}{\gamma}$ 인 지수분포를 따른다. tr_5 가 발사되면 p_5 의 토큰은 p_3 로 이동되어 서버가 시스템에 돌아왔음을 나타내 준다. tr_5 가 발사되어 토큰이 p_3 로 이동된 순간 p_2 에 존재하는 토큰의 수가 N 이상이면 tr_2 가 발사되어 p_3 의 토큰은 p_4 로 이동된다. 반면에 p_2 에 존재하는 토큰의 수가 N 미만이면 tr_2 는 p_2 에 존재하는 토큰의 수가 N 에 도달될 때까지 발사되지 않는다. 여기서 유의해야 할 점은 서버 단일 휴가 시스템은 휴가가 존재하지 않는 시스템과 tr_2 의 enabling 조건에 차이가 있다는 사실이다. 즉 서버 단일 휴가 시스템에서는 휴가가 존재하지 않는 시스템과 달리 p_2 에 존재하는 토큰의 수가 N 을 초과할 경우에도 tr_2 가 enabled 된다. 기존의 페트리 넷에서는 이와 같은 enabling조건은 쉽게 표현하기 힘들었으나 SPNP에서는 enabling function을 도입함으로써 marking dependent한 enabling조건을 쉽게 표현할 수 있게 하였다. 따라서 SPNP를 이용할 경우 tr_2 의 enabling 조건은 enabling function을 이용하여 <표 1>과 같이 별도로 정의하여야만 한다.

<표 1> tr_2 의 enabling function

transition	enabling function
tr_2	if (mark("p2") >= N) return(1); return(0);

<표 1>의 enabling function은 p_2 에 있는 토큰의 수 (mark("p2"))가 N 이상이면 tr_2 가 발사되고(return

(1)), 그렇지 않으면 발사되지 않음(return(0))을 의미한다. 또한 <그림 6>에서 볼 수 있듯이, 서버 단일 휴가 시스템에서는 휴가가 존재하지 않는 시스템과 달리 별도의 transition을 추가하지 않아도 tr_5 의 throughput로부터 사이클의 기대길이를 쉽게 구할 수 있다.



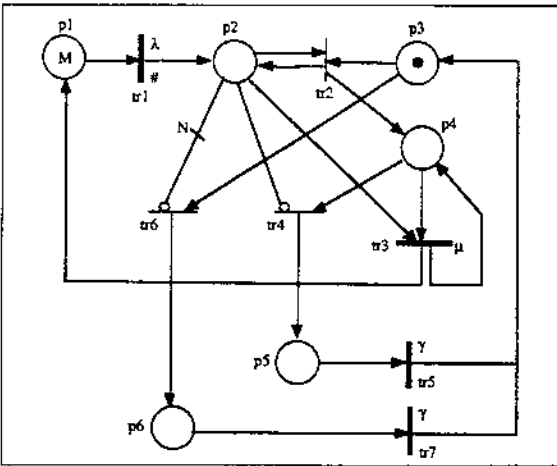
<그림 6> 서버 단일 휴가 시스템의 페트리 넷 모형

3.3 서버 복수 휴가 시스템의 페트리 넷 모형

N -정책을 사용하는 서버 복수 휴가 시스템에서는 서비스가 끝나 대기 시스템 내에 고객이 존재하지 않으면 서버는 휴가를 떠난다. 휴가기간의 길이는 평균 $\frac{1}{\gamma}$ 인 지수분포를 따른다고 가정한다. 휴가에서 돌아온 서버는 대기시스템에서 대기중인 고객의 수가 N 을 넘어섰는지 조사한다. 만일 대기중인 고객의 수가 N 미만이면 서버는 또다시 휴가를 떠나게 되고 N 이상이면 서버는 즉시 서비스를 시작하게 된다. 이러한 과정이 서버가 휴가에서 돌아왔을 때 대기중인 고객의 수가 N 이상일 때 까지 반복 수행된다. N -정책을 사용하는 서버 복수 휴가 시스템에서 입력원의 크기가 M 으로 유한한 경우의 페트리 넷 모형은 <그림 7>과 같다.

<그림 7>에서는 복수 휴가과정을 표현하기 위하여 <그림 1>에 $p_5, p_6, tr_5, tr_6, tr_7$ 를 추가하였다. <그림 7>에서 tr_2 의 enabling조건은 enabling function에 의해 정의되어야 하며 이는 <그림 6>의 tr_2 의 enabling function과 동일하다. <그림 7>에서 p_5 는 서버가 첫

번째 휴가중임을 나타내고 p6는 서버버가 두번째 이후의 휴가중에 있음을 나타낸다. tr5는 첫번째 휴가기간을 나타내고 tr6은 서버버가 휴가에서 돌아왔을 때 대기중인 고객의 수가 N미만이면 enabled되는 immediate transition이다. 서버버가 휴가에서 돌아왔을 때 p2에 존재하는 고객의 수가 N미만일 경우에 tr6가 enabled 될 수 있도록 p2와 tr6는 inhibitor arc에 의해 연결되어 있다. tr7은 서버버가 두번째 이후의 휴가를 떠났을 때의 휴가기간을 나타내며, tr5와 tr7을 별도로 그린 이유는 tr5의 throughput을 이용하여 사이클의 기대길이를 구하기 위함이다.



〈그림 7〉 서버버 복수 휴가 시스템의 페트리 넷 모형

경우에 따라서는 첫번째 휴가기간이 두번째 이후의 휴가기간과 상이한 분포를 따르는 서버버 휴가 시스템도 존재한다. 이 경우에는 tr5의 firing rate와 tr7의 firing rate를 달리 줌으로써 시스템을 페트리 넷으로 모형화 할 수 있다.

4. 최적제어정책

본 절에서는 선형비용구조를 가정한 후 3절에서 분석한 서버버 휴가 시스템의 최적제어정책을 SPNP를 이용하여 구해 보고자 한다. 본 연구에서 가정하는 비용구조는 다음과 같다. 서버버가 서비스를 시작할 때 마다 준비비용(setup cost) A가 발생하고 고객 한명이

단위기간동안 대기 시스템에서 체류함에 따라 대기비용(waiting cost) k가 발생한다. 만일 대기 시스템의 용량이 유한하여 유실되는 고객이 발생하는 경우에는 유실되는 고객 한명당 유실비용 π가 발생한다고 가정한다. 따라서 단위기간당 발생하는 기대비용은 단위기간당 발생하는 기대 준비비용, 단위기간당 발생하는 기대 대기비용 및 단위기간당 발생하는 기대 유실비용의 합으로 표현되며 이들 비용은 각각 다음과 같이 구하여질 수 있다.

단위기간당 발생하는 기대 준비비용은 한 사이클 동안 준비비용이 한번 발생하므로 준비비용 A를 사이클의 기대길이 T로 나누면 구하여 진다. 단위기간당 발생하는 기대 대기비용은 시스템에 존재하는 평균 고객수를 L이라 하면 kL로 표현된다. 또한 단위기간당 발생하는 기대 유실비용은 단위기간당 유실되는 기대 고객수를 ξ라 하면 πξ로 표현된다. 따라서 단위기간당 발생하는 기대비용 C는 다음과 같이 주어진다.

$$C = \frac{A}{T} + kL + \pi\xi$$

최적 제어정책 N*는 비용함수 C의 값을 최소화하는 N의 값이며 이를 구하기 위해서는 N의 값이 주어졌을 때 T, L, ξ의 값을 구할 수 있어야만 한다. L의 값은 p2에 존재하는 토큰의 기대치로서 SPNP로부터 직접 구해지며 T의 값도 3절에서 설명한 바와 같이 SPNP를 통하여 직접 구하여 질 수 있다. 〈그림 3〉에서와 같이 용량이 K로 유한하고 고객의 도착률이 λ인 대기 시스템의 경우 단위기간당 유실되는 기대 고객수는 도착률 λ에 시스템 내에 존재하는 고객의 수가 K일 안정상태확률을 곱하면 구하여 질 수 있다. 이때 시스템 내에 존재하는 고객수가 K일 안정상태확률은 p2에 존재하는 토큰의 수가 K일 확률이므로 SPNP에 의해 자동적으로 구하여지며, 따라서 특정한 제어정책이 주어질 경우의 비용함수 C의 계산을 위해 필요한 모든 데이터는 SPNP에 의해 자동적으로 계산되어 짐을 알 수 있다.

비용함수 C의 최소치를 구하기 위해서는 불록성질이나 unimodality 성질 등 바람직한 수학적 특성을 C

가 보유하고 있는지 규명하는 것이 필요하다. 그러나 비용함수 C 의 이와 같은 특성을 수학적으로 규명하기는 매우 힘들다. 본 연구에서는 많은 예제를 통한 실험을 함으로써 비용함수 C 가 N 에 대해 unimodal한 함수임을 관측할 수 있었다. 이러한 관측결과를 토대로 효율적인 탐사기법(search technique)을 이용하면 함수 C 의 값을 최소화 하여주는 최적제어정책을 쉽게 구할 수 있게 된다. 사용가능한 탐사기법으로는 fibonacci법, bisection법 등 unimodal한 함수의 최적해 탐사에 적용 가능한 임의의 선 탐사기법(line search technique)을 들 수 있다.

<예제1> 예제1은 준비시간과 서버의 복수휴가가 모두 존재하는 모형을 다루며 <그림 4>와 <그림 7>의 혼합모형이라 볼 수 있다. 시스템의 용량은 무한하고 입력원의 크기는 10이며 서버는 유휴기간중 복수휴가를 취한다. 서버가 서비스를 시작하기 위해서는 지수분포를 따르는 준비시간이 필요하며 준비시간의 평균은 $\frac{1}{\lambda}$ 이다. 기타 필요한 시스템 파라미터 및 비용 데이터는 다음과 같다.

$$\lambda = 1.0, \mu = 10.0, \gamma = 1.0, \chi = 2.0, A = 30, h = 1.$$

<예제2> 예제2는 시스템의 용량이 유한하며 고객이 도착률 1인 포아송과정에 따라 도착한다는 점을 제외하면 예제1과 모든 상황이 동일하다. 즉 예제1에서와 같이 서버는 복수 휴가를 취하고 서비스를 시작하기 위해서는 준비시간이 소요된다. 그러나 시스템의 용량이 유한하므로 예제1에서와는 달리 유실되는 고객이 발생한다. 시스템의 용량이 10이고 유실되는 고객 한명당 5의 비용이 발생하며 그 외의 데이터는 1과 동일하다.

예제 1, 2의 결과가 <표 2>에 정리되어 있다. 탐사기법을 사용하면 모든 N 에 대해 C 의 값을 구할 필요가 없으나 표2에서는 비용함수 C 의 특성을 보여주기 위하여 사용가능한 모든 N 에 대하여 C 의 값을 표기하였다. <표 2>에서 볼 수 있듯이 비용함수 C 는 N 에 대해 unimodal한 함수임이 관측된다.

<표 2> 예제1, 예제2의 결과(*는 최적 제어정책을 표시함)

예제 1		예제 2		
한계치(N)	단위기간당 기대비용(C)	한계치(N)	단위기간당 기대비용(C)	유실되는 고객비용
1	14.294	1	13.090	0.094
2*	14.283	2	12.655	0.099
3	14.307	3*	12.622	0.106
4	14.332	4	12.843	0.117
5	14.369	5	13.234	0.130
6	14.417	6	13.745	0.147
7	14.481	7	14.346	0.168
8	14.569	8	15.014	0.194
9	14.701	9	15.737	0.226
10	14.954	10	16.499	0.263

5. 결론

본 연구에서는 대기 시스템의 용량이 유한하거나 입력원의 크기가 유한한 경우의 대기 시스템에 대한 제어정책을 추계적 페트리 넷트를 이용하여 모형화 하였다. 본 연구에서는 한계치를 이용한 제어정책인 N -정책을 고려하였으며, 준비비용, 대기비용 및 고객 유실비용을 포함하는 선형비용구조 하에서 단위기간당 기대비용을 최소화 하여주는 최적 N -정책을 구하였다. 이를 위해 본 연구에서는 분석하고자 하는 제어정책을 우선 페트리 넷트로 표현한 후 SPNP라는 소프트웨어를 이용하여 시스템의 성능분석과정을 자동으로 수행하였다.

페트리 넷트에 대한 기본지식은 쉽게 습득할 수 있으며 일반적으로 대기 시스템의 제어모형을 페트리 넷트로 표현하기도 어렵지 않다. 따라서 대기이론에 대한 비전문가라도 본 연구에서 제안한 방법을 이용하면 대기 시스템의 최적 제어정책을 쉽게 구할 수 있으리라 판단된다. 또한 본 연구에서 제안한 방법은 서비스 시간이 phase type 분포를 따르거나 한명의 서버가 둘이상의 대기 시스템을 번갈아 서비스하는 순환 서버 시스템과 같이 보다 복잡한 대기 시스템에 대하여도 적용가능하다. 그러나 SPNP를 이용한 시스템의 성능분석은 첫째, 시간의 경과에 따른 시스템의 상태변환 과정이 마코브과정을 따라야 하고, 둘째, 시스템의 상태의 수가 너무 크지 않아야 한다는 제약은

만족하여야만 하며, 이러한 제약은 SPNP뿐만 아니라 페트리 넷를 통하여 추계적 시스템의 성능분석을 수행하고자 할 경우 부딪히는 일반적인 한계라 할 수 있다.

【인용문헌】

- [1] Grabill, T. D., D. Gross and N. Magazine, "A Classified Bibliography of Research on Optimal Design and Control of Queues," *Opns. Res.*, Vol. 25, pp.219~232, 1977
- [2] Teghem, J., "Control of the Service Process in a Queueing System," *Euro. J. of Oper. Res.*, Vol. 23, pp.141~158, 1986
- [3] Yadin, M. and P. Naor, "Queueing System with a Removable Service Station," *Oper. Res. Quart.*, Vol. 14, pp.393~405, 1963
- [4] Heyman, D. P., "Optimal Operating Policies for M/G/1 Queueing Systems," *Opns. Res.*, Vol. 16, pp.362-382, 1968
- [5] Heyman, D. P., "The T-Policy for the M/G/1 Queue," *Mgmt. Sci.*, Vol. 23, pp.775~778, 1977
- [6] Bell, C. E., "Characterization and Computation of Optimal Policies for Operating an M/G/1 Queueing System with Removable Server," *Opns. Res.*, Vol. 19, pp.208~218, 1971
- [7] Kella, O., "The Threshold Policy in the M/G/1 Queue with Server Vacations," *Naval Research Logistics*, Vol. 36, pp.111~123, 1989
- [8] Lee, H. S. and M. M. Srinivasan, "Control Policies for the M^x/G/1 Queueing System," *Mgmt. Sci.*, Vol. 35, pp.708~721, 1989
- [9] Lee, H. W., S. S. Lee and K. C. Chae, "Operating Characteristics of M^x/G/1 Queue with N-Policy," *Queueing Systems*, Vol. 15, pp.389~399, 1994
- [10] Hersh, M and I. Brosh, "The Optimal Strategy Structure of an Intermittently Operated Service Channel," *Euro. J. of Oper. Res.*, Vol. 5, pp.133~141, 1980
- [11] Teghem, J., "Optimal Control of a Removable Server in an M/G/1 Queue with Finite Capacity," *Euro. J. of Oper. Res.*, Vol. 23, pp.141~158, 1987
- [12] Lee, H. S., "Optimal Control of the M^x/G/1/K Queue with Multiple Server Vacations," *Comp. and O.R.*, to appear in 1995
- [13] Chiola, G., "A Graphical Petri Net Tool for Performance Analysis," *Proc. 3rd International Workshop on Modeling Techniques and Performance Evaluation*, Paris, France, pp.297~308, 1987
- [14] Ciardo, G. and J. K. Muppala, "Manual for the SPNP Package Version 3.1," *Duke University*, Durham, 1991
- [15] Ibe, O. C. and K. S. Trivedi, "Stochastic Petri Net Analysis of Finite Population Vacation Queueing Systems," *Queueing Systems*, Vol. 8, pp.111~128, 1991



이광식(李光植)

1989년 경희대학교 산업공학과 졸업 (학사)

1992년 동대학교 산업공학과 졸업 (석사)

1994년~ 현재 대우 고등기술 연구원
관심분야: 제조시스템 모델링, CIM



이효성(李曉成)

1978년 서울대학교 산업공학과 졸업 (학사)

1980년 한국과학기술원 산업공학과 졸업 (석사)

1988년 University of Michigan 산업공학과 졸업 (박사)

1982년~ 현재 경희대학교 산업공학과 부교수

관심분야: 대기행렬이론, 제조시스템 및 통신시스템의 성능평가