Journal of the Korean Institute of Industrial Engineers
Vol. 27, No. 1, pp. 89-94, March 2001.

타임아웃이 있는 $M/E_k/c/N$ 대기시스템의 성능분석

류지현[1] · 전치혁[2†]

[1]앤더슨컨설팅/[2]포항공과대학교 산업공학과

Performance Analysis of $M/E_k/c/N$ Time-out Queueing System

Jihyun Ryu[1] · Chi-Hyuck Jun[2]

[1]Product Market Unit, Andersen Consulting, Seoul

[2]Department of Industrial Engineering, Pohang University of Science and Technology, Pohang

There are many queueing systems where customers wait for service up to a certain amount of time and leave the system if they are not served during that time. This paper considers a finite capacity multi-server queueing system with Poisson input and Erlang service time, where a customer becomes a lost customer when his service has not begun within an exponential patient time after his arrival. Performance measures such as average queue length, the average number of customers in service, and the proportion of lost customers can be obtained exactly through the proposed numerical solution procedure.

Keywords : impatient customer, lost customer, multi-server, finite capacity, Erlang service.

1. 서 론

유한개의 서버와 대기공간(큐)을 포함하는 대기시스템에 있어 서비스를 기다리는 고객들은 대기공간에서의 기다리는 시간이 클 경우 서비스를 포기하려는 성질을 가지고 있다. 즉, 어떤 한정된 시간 내에 서비스를 받지 못하면 대기공간을 이탈하여 시스템을 빠져나가게 된다. 이러한 상황은 전화 및 통신망 시스템을 포함한 일상생활 어디서나 빈번히 발생하고 있다. 서비스를 제공하는 서버는 빠른 서비스 속도의 단일서버 혹은 느린 서비스 속도의 복수서버로 모델링이 가능하다. 따라서 복수서버의 타임아웃 대기시스템은 고객의 성급한(impatient) 특성을 고려한 일반적인 대기시스템으로 분석의 가치가 높다.

M/G/1 타임아웃 대기시스템의 경우, 타임아웃 시간이 상수 혹은 확률변수일 때 평균 큐 대기시간, 고객이탈 비율 등의 성능지표가 이미 Baccelli *et al.*(1984), De Kok and Tijms (1985)에 의하여 도출된 바 있다. 복수서버 타임아웃 대기시스템에 관해서는 타임아웃 시간이 상수이고 서비스 시간이 일반분포인 대기시스템의 고객이탈 비율이 Boots and Tijms (1999)에 의해 근사적으로 도출되었고, 서비스 시간이 지수분포를 따르고 타임아웃 시간이 일반분포인 대기시스템의 경우 고객이탈 비율, 평균 큐 대기시간 등의 성능지표가 Brandt and Brandt (1999)에 의하여 도출된 바 있다. 그러나 타임아웃 시간이 확률변수이고 서비스 시간이 지수분포가 아닌 복수서버 시스템의 성능분석은 현재까지 이루어져 있지 않은 상황이다.

본 논문의 목적은 타임아웃 시간이 지수분포를 가지는 $M/E_k/c/N$ 타임아웃 대기시스템($M/E_k/c/N+M$)의 분석방법을 제안하는 데 있다. 서비스 시간으로 얼랑(Erlang)분포를 채택함으로써 지수분포보다 다양한 분포의 특성을 고려할 수 있으며, 일반적인 분포를 얼랑분포의 형태로서 조절함으로써 근사치를 제시할 수도 있기 때문이다. 대기공간을 이탈하는 고객들의 비율, 평균 큐 길이, 평균 서버점유 고객수 등이 주요 성능지표이며 이를 도출하기 위하여 연속시간 마코브체인(Continuous Time Markov Chain; CTMC)을 통해 시스템 방정식을 유도한다. 시스템 방정식의 해는 수치적으로 계산되며 이를 근간으로 성능지표들을 도출한다.

이 논문은 제1회 IE/MS 사이버학술대회에서 우수논문으로 선정된 후 수정과 심사를 거쳐 게재가 확정되었음.
본 연구는 서울대학교 복잡계통계연구센터를 통한 한국과학재단의 지원에 의하여 수행되었음.
† 연락저자 : 전치혁 교수, 790-784 경북 포항시 남구 효자동 산31, 포항공과대학교 산업공학과, Fax : 054-279-2870, e-mail : chjun@postech.ac.kr
2000년 4월 접수, 1회 수정 후, 2000년 12월 게재 확정.

2. 대기모형 및 기호

본 연구에서 고려하는 대기시스템의 모형은 <그림 1>과 같다. 복수서버의 수는 c이며, 시스템 용량은 N으로 제한되어 있다. 즉, 대기공간은 $(N-c)$개이다.

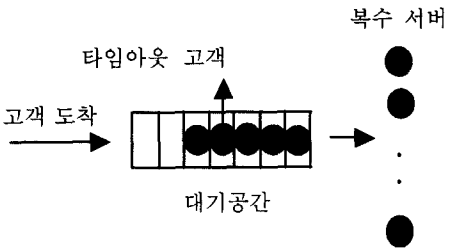

그림 1. 대기모형.

각 고객의 도착은 시간당 도착률 λ를 갖는 포아송 과정을 따른다. 도착시 대기공간이 모두 점유되어 있을 경우 도착한 고객은 시스템에 입장하지 못한다. 동일한 서버들 중 하나 이상이 서비스를 하고 있지 않는 경우 고객은 도착 즉시 이용 가능한 서버 중 하나를 임의 선택하여 점유하고 서비스를 받게 되며 서버가 모두 서비스 중일 때 도착한 고객은 대기공간에서 서비스를 기다리게 된다. 서비스 시간 S는 각 고객에 대하여 독립적이고 동일하게(independent and identically distributed; iid) 평균이 k/μ인 얼랑분포 Erl(k, μ)를 따른다. 서비스를 기다리는 고객은 타임아웃 시간동안 서비스를 받지 못하면 성급하게(impatiently) 대기공간을 이탈하게 된다. 여기서 타임아웃 시간 I는 확률변수이며, 각 고객에 대해 iid Exp(γ)분포를 따른다.

3. 시스템 안정상태 확률 도출

3.1 CTMC 모델링

얼랑분포 Erl(k, μ)를 따르는 서비스 시간은 평균 $1/\mu$인 지수 확률변수 k개의 합으로 간주할 수 있으므로 서비스 중인 고객의 상태를 1부터 k까지로 표시할 수 있다. 이에 따라 $N_i(t)$를 시간 t에 얼랑분포의 i번째 단계의 서비스에 있는 고객수라 하자. 또한, $N_q(t)$를 시간 t에 큐에 있는 고객수라하면, 타임아웃 시간을 지수분포로 가정하였으므로 시스템 상태$\{(N_q(t), N_1(t), N_2(t), .., N_i(t), .., N_{k-1}(t), N_k(t)), t\geq 0\}$는 CTMC를 따른다. 이때 안정상태 확률의 정의는 다음과 같다.

$$\lim_{t\to\infty} P\{N_q(t)=n_q, N_1(t)=n_1, \cdots, N_i(t)=n_i, \cdots, N_k(t)=n_k\} = P_{(n_q, n_1, \cdots, n_i, \cdots, n_k)} \quad (1)$$

임의의 상태 $(n_q, n_1, .., n_i, .., n_k)$를 중심으로 $0<n<c$와 $c<n<N$의 경우에 대한 상태전이 다이어그램은 각각 <그림 2>, <그림 3>과 같다.

시스템에 있는 고객수를 $n=n_q+\sum_{j=1}^{k} n_j$라 할 때 <그림 2>의 $0<n<c$인 경우는 대기공간에 고객이 없는 경우$(n_q=0)$이므로 ❶의 상태전이는 새로운 고객이 도착하는 경우를 나타낸다. ❷의 상태전이는 얼랑 서비스 분포의 i번째 단계$(i=1,\cdots,k-1)$에 있는 고객들 중 한 고객이 $i+1$번째 단계로 접어드는 경우를 나타낸다. ❸은 얼랑 서비스 분포의 k번째 단계에 있는 고객들 중 한 고객이 서비스를 끝내고 시스템을 떠나는 경우를 나타낸다. <그림 3>의 $c<n<N$인 경우는 대기공간에 고객이 기다리고 있는 경우이므로 새로운 고객이 도착하는 ①

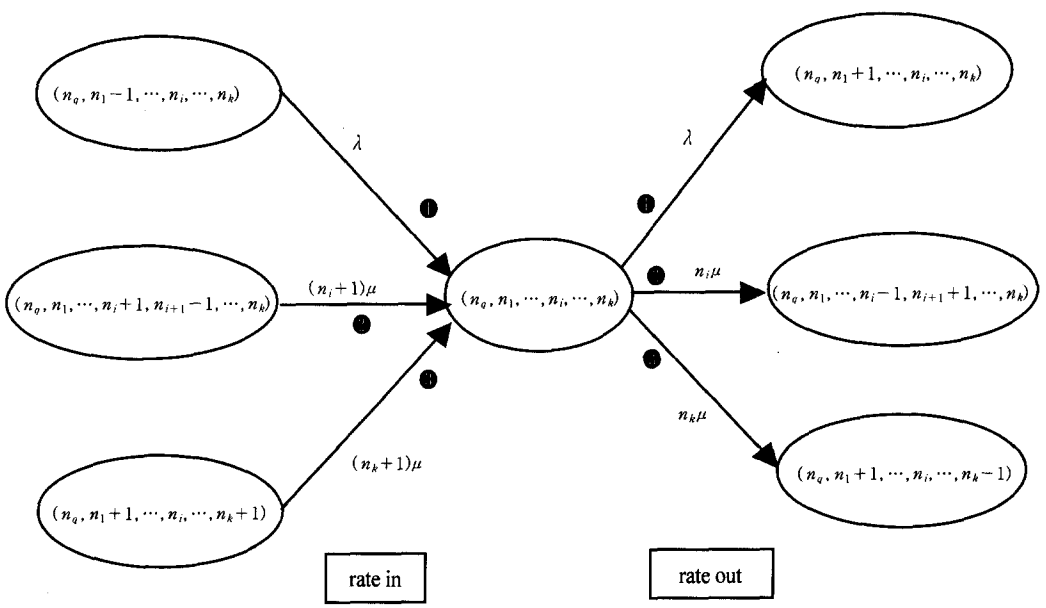

그림 2. 상태전이 다이어그램 $(0<n<c)$.

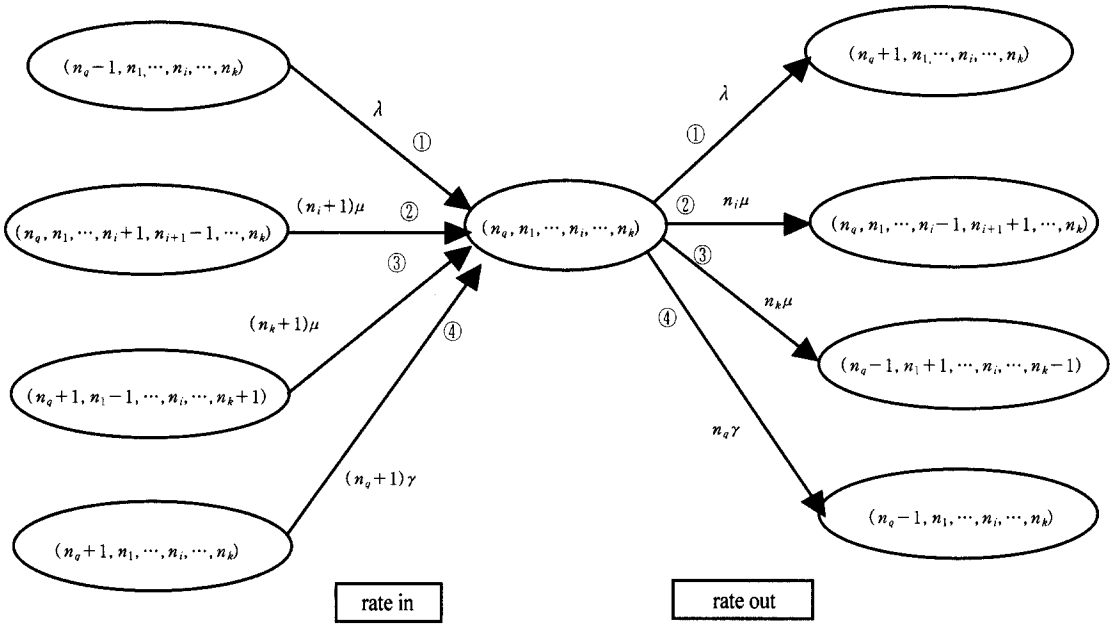

그림 3. 상태전이 다이어그램 ($c < n < N$).

의 상태전이에 있어 $Nq(t)$가 증가하게 된다. ②, ③도 <그림 2>의 상태전이와 같은 상황이며, ③의 경우 하나의 서버가 사용가능해지므로 $Nq(t)$가 감소하게 된다. ④는 대기공간에서 기다리는 고객들 중 한 고객이 대기공간을 이탈하는 경우를 나타낸다. $n=0$, c, N인 경우에 대하여서도, <그림 2> 및 <그림 3>과 유사하게 상태전이를 정의할 수 있다.

따라서 본 대기모형의 평형 방정식(balance equation)은 다음과 같다.

$$n=0:\ \lambda P_{(n_q, n_1, \cdots, n_k)} = \mu P_{(n_q, n_1, \cdots, n_k+1)} \tag{2a}$$

$$0<n<c:\ (\lambda+n\mu)P_{(n_q, n_1, \cdots, n_k)} = \lambda P_{(n_q, n_1, \cdots, n_k)} + \sum_{i=1}^{k-1}(n_i+1)\mu P_{(n_q, n_1, \cdots, n_i+1, n_{i+1}-1, \cdots, n_k)} + (n_k+1)\mu P_{(n_q, n_1, \cdots, n_k+1)} \tag{2b}$$

$$n=c:\ (\lambda+n\mu)P_{(n_q, n_1, \cdots, n_k)} = \lambda P_{(n_q, n_1, \cdots, n_k)} + \sum_{i=1}^{k-1}(n_i+1)\mu P_{(n_q, n_1, \cdots, n_i+1, n_{i+1}-1, \cdots, n_k)} + (n_k+1)\mu P_{(n_q+1, n_1-1, \cdots, n_k+1)} + \gamma P_{(n_q+1, n_1, \cdots, n_k)} \tag{2c}$$

$$c<n<N:\ (\lambda+n_q\gamma+c\mu)P_{(n_q, n_1, \cdots, n_k)} = \lambda P_{(n_q-1, n_1, \cdots, n_k)} + \sum_{i=1}^{k-1}(n_i+1)\mu P_{(n_q, n_1, \cdots, n_i+1, n_{i+1}-1, \cdots, n_k)} + (n_k+1)\mu P_{(n_q+1, n_1-1, \cdots, n_k+1)} + (n_q+1)\gamma P_{(n_q+1, n_1, \cdots, n_k)} \tag{2d}$$

$$n=N:\ (n_q\gamma+c\mu)P_{(n_q, n_1, \cdots, n_k)} = \lambda P_{(n_q-1, n_1, \cdots, n_k)} + \sum_{i=1}^{k-1}(n_i+1)\mu P_{(n_q, n_1, \cdots, n_i+1, n_{i+1}-1, \cdots, n_k)} \tag{2e}$$

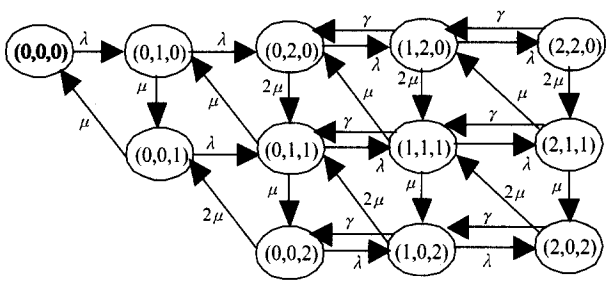

그림 4. 상태전이 다이어그램($k=2$, $c=2$, $N=4$).

참고로 $k=2$, $c=2$, $N=4$인 시스템의 상태전이 다이어그램은 <그림 4>와 같다.

3.2 수치적 방법

CTMC의 복잡한 구조로 인하여 식 (2)와 같은 평형방정식들의 해인 안정상태 확률을 일반적으로 도출해내기가 어려우므로 Seelen(1986)의 수치적 방법을 이용한다. 본 방법은 Ph/Ph/c 대기시스템에 대한 시스템 안정상태 확률을 도출하기 위하여 제안된 것이지만 본 문제 해결에도 적용될 수 있다. 수치적 방법의 적용을 위해 평형 방정식을 다음과 같이 기술한다.

$$D_0 \vec{p}_0 = V_0 \vec{p}_0 + W_1 \vec{p}_1 \tag{3a}$$

$$D_n \vec{p}_n = U_{n-1} \vec{p}_{n-1} + V_n \vec{p}_n + W_{n+1} \vec{p}_{n+1},\ n=1, \cdots, N-1 \tag{3b}$$

$$D_N \vec{p}_N = U_{N-1} \vec{p}_{N-1} + V_N \vec{p}_N \tag{3c}$$

위에서 사용된 기호는 아래와 같다.

m_n : 시스템에 n명의 고객이 존재할 때 가능한 상태의 수

$\vec{p}_n$: m_n개의 원소로 이루어진 열벡터; 열벡터의 각 원소는 $p_{n,i}$

$p_{n,i}$: 시스템에 n명의 고객이 존재할 때 가능한 m_n개의 안정상태 확률에 대응된다($i=1,\cdots, m_n$).

D_n : (n,i) 상태에서 나가는 전이율들로 구성된 전이행렬; 크기는 $m_n \times m_n$

V_n : (n, j) 상태들로부터 (n, i) 상태로 들어오는 전이율들로 구성된 전이행렬; 크기는 $m_n \times m_n$

U_{n-1} : $(n-1, j)$ 상태들로부터 (n, i) 상태로 들어오는 전이율들로 구성된 전이행렬; 크기는 $m_n \times m_{n-1}$

W_{n+1} : $(n+1, j)$ 상태들로부터 (n, i) 상태로 들어오는 전이율들로 구성된 전이행렬; 크기는 $m_n \times m_{n+1}$

예를 들어, $k=2$, $c=2$, $N=4$인 시스템에서 전이행렬 D_n, U_{n-1}, V_n, W_{n+1}들을 구성한 예는 다음과 같다.

$$n=0:\ D_0=(\lambda),\ V_0=(0),\ W_1=(0\ \mu)$$

$$n=1:\ D_1=\begin{pmatrix}\lambda+\mu & 0\\ 0 & \lambda+\mu\end{pmatrix},\ U_0=\begin{pmatrix}\lambda\\ 0\end{pmatrix},$$

$$V_1=\begin{pmatrix}0 & 0\\ \mu & 0\end{pmatrix},\ W_2=\begin{pmatrix}0 & \mu & 0\\ 0 & 0 & 2\mu\end{pmatrix}$$

$$n=2:\ D_2=\begin{pmatrix}\lambda+2\mu & 0 & 0\\ 0 & \lambda+2\mu & 0\\ 0 & 0 & \lambda+2\mu\end{pmatrix},\ U_1=\begin{pmatrix}\lambda & 0\\ 0 & \lambda\\ 0 & 0\end{pmatrix},$$

$$V_2=\begin{pmatrix}0 & 0 & 0\\ 2\mu & 0 & 0\\ 0 & \mu & 0\end{pmatrix},\ W_3=\begin{pmatrix}\gamma & \mu & 0\\ 0 & \gamma & 2\mu\\ 0 & 0 & \gamma\end{pmatrix}$$

$$n=3:\ D_3=\begin{pmatrix}\lambda+2\mu+\gamma & 0 & 0\\ 0 & \lambda+2\mu+\gamma & 0\\ 0 & 0 & \lambda+2\mu+\gamma\end{pmatrix},$$

$$U_2=\begin{pmatrix}\lambda & 0 & 0\\ 0 & \lambda & 0\\ 0 & 0 & \lambda\end{pmatrix},\ V_3=\begin{pmatrix}0 & 0 & 0\\ 2\mu & 0 & 0\\ 0 & \mu & 0\end{pmatrix},$$

$$W_4=\begin{pmatrix}2\gamma & \mu & 0\\ 0 & 2\gamma & 2\mu\\ 0 & 0 & 2\gamma\end{pmatrix}$$

$$n=4:\ D_4=\begin{pmatrix}2\mu+2\gamma & 0 & 0\\ 0 & 2\mu+2\gamma & 0\\ 0 & 0 & 2\mu+2\gamma\end{pmatrix},$$

$$U_3=\begin{pmatrix}\lambda & 0 & 0\\ 0 & \lambda & 0\\ 0 & 0 & \lambda\end{pmatrix},\ V_4=\begin{pmatrix}0 & 0 & 0\\ 2\mu & 0 & 0\\ 0 & \mu & 0\end{pmatrix}$$

위의 예에서 보듯, 규칙적으로 $\vec{p}_n$의 원소를 정렬함으로써 각 전이행렬의 원소가 일정한 규칙에 의해 배열됨을 알 수 있다. 본 연구에서는 이러한 일정한 규칙을 이용하여 k, c, N이 일반적으로 주어질 때 각 전이행렬들을 구성할 수 있는 알고리듬과 컴퓨터 프로그램을 개발하고 이를 해 산출에 이용하였다.

해의 산출을 위하여 반복적 방법을 사용하는 데 여기에 사용된 기호의 정의는 아래와 같다.

$\vec{p}_h^{\,h}$: $\vec{p}_n$의 h번째 iteration결과

$p_{n,i}^h$: $\vec{p}_h^{\,h}$의 i번째 원소, $p_{n,i}$의 h번째 iteration결과

f_n, g_n : 정규화 계수

e_n : m_n개의 원소를 1로 갖는 열 벡터

이때, $p_{n,i}^h$을 구하는 iteration 식은 다음과 같다.

$$n=0:\ p_{n,i}^h = g_n \sum_{j=1}^{m_{n+1}} W_{n+1}(i,j)\,p_{n+1,j}^{h-1}/D_n(i,i) \tag{4a}$$

$$0<n<N:\ p_{n,i}^h = g_n\Big(\sum_{j=1}^{m_{n-1}} U_{n-1}(i,j)\,p_{n-1,j}^h + \sum_{j=1,i>1}^{i-1} V_n(i,j)\,p_{n,j}^h + f_n \sum_{j=i+1,i<m_n}^{m_n} V_n(i,j)\,p_{n,j}^{h-1} + g_n \sum_{j=i}^{m_{n+1}} W_{n+1}(i,j)\,p_{n+1,j}^{h-1}\Big)/D_n(i,i) \tag{4b}$$

$$n=N:\ p_{n,i}^h = g_n\Big(\sum_{j=1}^{m_{n-1}} U_{n-1}(i,j)\,p_{n-1,j}^h + \sum_{j=1,i>1}^{i-1} V_n(i,j)\,p_{n,j}^h + f_n \sum_{j=i+1,i<m_n}^{m_n} V_n(i,j)\,p_{n,j}^{h-1}/D_n(i,i) \tag{4c}$$

$$f_0=\Big[\sum_{n=0}^{N}\sum_{i=1}^{m_n} p_{n,i}^{h-1}\Big]^{-1}$$

$$f_n=\frac{u_{n-1}^T\,\vec{p}_{n-1}^{\,h}}{w_n^T\,\vec{p}_n^{\,h-1}},\ n=1,\cdots,N$$

$$g_n=f_n\frac{u_{n-1}^T\,\vec{p}_{n-1}^{\,h}}{w_{n+1}^T\,\vec{p}_{n+1}^{\,h-1}},\ n=0,\cdots,N-1$$

$$u_n^T=e_{n+1}^T U_n,\quad W_n^T=e_{n-1}^T W_n$$

$\vec{p}_n^{\,h}$는 $n=0$부터 $n=N$까지, 각 n들에 대해서 $p_{n,i}^h$는 $i=1$부터 $i=mn$까지 순차적으로 구한다. $p_{n,i}^0$들은 임의의 non-trivial 해로 설정하고 방정식의 해가 수렴할 때까지 반복한다. f_0는 안정상태 확률들의 합을 1로 유지시키기 위한 정규화 계수이다. $f_n(n=1,...,N)$과 $g_n(n=0,\cdots,N-1)$은 거시적 상태평형을 유지시키기 위한 보정에 사용된다. 거시적 상태에서의 평형방정식은 다음과 같다.

$$u_n^T\,\vec{p}_n = w_{n+1}^T\,\vec{p}_{n+1},\ n=0,1,\cdots,N-1 \tag{5}$$

f_n과 g_n의 정규화 계수들은 수치적 방법의 수렴속도를 증대시키기 위해 iteration 도중에 해를 보정한다.

4. 성능지표 유도

시스템에 고객이 n명 있을 안정상태 확률은 다음과 같이 산출

된다.

$$P_n = \sum_{i=1}^{m_n} p_{n,i} \tag{6}$$

이를 이용하여 평균 큐 길이 L_q 및 평균 서버점유 고객수 L_s는 각각 식 (7), (8)과 같이 도출된다.

$$L_q = \sum_{n=c+1}^{N} (n-c) P_n \tag{7}$$

$$L_s = \sum_{n=1}^{c} nP_n + c \sum_{n=c+1}^{N} P_n \tag{8}$$

주요 관심사인 시스템에 입장한 고객들 중 서비스를 받지 못하고 큐를 이탈하는 고객들의 비율 π를 구하기 위하여 Brumelle (1971)이 제안한 비용구조(cost structure)를 이용한다. 이는 Little's Law의 확장된 개념으로, 고객이 어떤 비용구조에 따라 시스템에 금액을 지불한다면 다음과 같은 등식이 성립한다.

시스템이 버는 단위 시간당 평균 이익
= 고객의 도착률 × 한 고객이 시스템에 지불하는 평균금액

서버에 입장하는 고객이 서버에 있는 동안 단위시간당 1원씩 시스템에게 지불한다면 다음과 같은 등식이 성립한다.

$$L_s = \lambda(1-P_N)(1-\pi)E[S] \tag{9}$$

이 비용구조에 따르면 시스템은 서버에 있는 고객수의 time average인 L_s만큼 단위시간당 수익을 거둔다(좌변). 반면 고객은 단위시간당 $\lambda(1-P_N)(1-\pi)$의 비율로 서버에 도착하게 되고 한 고객은 $E[S]$ 만큼의 금액을 지불한다(우변). 시스템이 버는 금액과 고객들이 지불하는 총 금액은 같으므로 좌변과 우변 사이에는 등식이 성립한다. 이를 이용하면 π를 얻을 수 있다.

$$\pi = 1 - \frac{L_s}{\lambda(1-P_N)E[S]} \tag{10}$$

5. 수치 예

본 논문에서 제시한 수치적 방법의 적합성을 알아보기 위하여 간단한 수치 예를 고려한다. 즉, 고객들의 도착률 λ는 1, 타임아웃 시간의 평균 $1/\gamma$ 은 2로 설정하였고 서버의 용량 c는 5로, 시스템 용량 N은 10으로 설정하였다. 순수한 k만의 영향을 고려하기 위해 서비스 시간의 평균 k/μ는 5로 고정하였다.

얼랑분포의 모수 k, μ 중 k의 변화에 따른 L_q, L_s, π의 변화는 각각 <표 1>, <표 2>, <표 3>과 같다. 수치적 방법으로 계산한 해와 시뮬레이션에 의하여 도출된 값을 실었다. 시뮬레이션 결과로 성능지표의 점추정치(point estimate)와 추정치의 99 % 신뢰구간(confidence interval)을 함께 제시하였다. <표 1>, <표 2>, <표 3>으로부터 수치적 방법을 통해 도출한 L_q, L_s, π는 시뮬레이션 추정치와 동일한 경향을 가지면서 k에 따라 변화함을 알 수 있다. 추정치와의 오차는 ±0.01이내이며 추정치의 신뢰구간을 벗어나지 않는다. 따라서 본 연구가 활용한 수치적 방법이 정확히 수렴된 해를 제공함을 알 수 있다.

표 1. k의 변화에 따른 L_q의 변화

k	L_q	
	수치적 방법	시뮬레이션 추정치(신뢰구간)
2	0.40910	0.41160(0.38604,0.43717)
3	0.40476	0.41221(0.39569,0.42873)
4	0.40239	0.39533(0.37666,0.41400)
5	0.40086	0.39881(0.38095,0.41666)
6	0.39979	0.40175(0.39417,0.40934)
7	0.39898	0.39653(0.38807,0.40499)
8	0.39835	0.40099(0.38770,0.41428)

표 2. k의 변화에 따른 L_s의 변화

k	L_s	
	수치적 방법	시뮬레이션 추정치(신뢰구간)
2	3.96590	3.95312(3.92316,3.98308)
3	3.97751	3.97718(3.94613,4.00823)
4	3.98385	3.97532(3.94889,4.00175)
5	3.98790	3.96738(3.94172,3.99304)
6	3.99075	3.98928(3.96031,4.01825)
7	3.99288	3.99116(3.97949,4.00283)
8	3.99455	4.00600(3.98056,4.03144)

표 3. k의 변화에 따른 π의 변화

k	π	
	수치적 방법	시뮬레이션 추정치(신뢰구간)
2	0.20502	0.20425(0.19688,0.21161)
3	0.20281	0.20268(0.19811,0.20725)
4	0.20160	0.19934(0.19225,0.20642)
5	0.20083	0.19729(0.19327,0.20130)
6	0.20029	0.20008(0.19832,0.20184)
7	0.19988	0.20026(0.19904,0.20148)
8	0.19956	0.19969(0.19448,0.20490)

서비스 분포의 평균을 같도록 설정하였으므로 k의 변화에 따른 성능지표의 변화는 미미하다. k값의 증가에 따라 L_q, π는 미세하게 감소하는 경향을 보이는 반면, L_s는 증가하는 경향을 보인다. k값의 증가에 따른 L_q의 변화를 <그림 5>에 나타내었다.

<그림 5>의 범례에서 n은 수치적 방법을 이용하여 도출한 해이고, s는 시뮬레이션에 의한 점추정치이다. 수직방향의 선

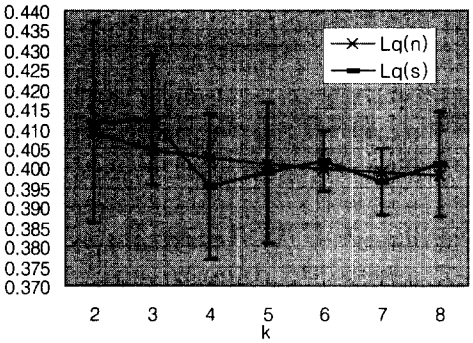

그림 5. k의 증가에 따른 L_q의 변화.

분은 시뮬레이션 추정치의 신뢰구간을 나타낸다. k 값이 증가함에 따라 L_q는 미세하게 감소한다. 곡선의 추이로 보아 k가 커짐에 따라 어떤 값으로 수렴할 것임을 추측할 수 있다.

6. 결 론

본 연구에서는 M/Ek/c/N+M 대기시스템의 성능분석 방법을 제시하였다. 평균 큐 길이 및 평균 서버점유 고객수, 큐 이탈 고객 비율의 성능지표들은 본 연구의 방법을 통해 정확한 해를 구할 수 있다. 한편, M/G/c/N 대기시스템을 M/Ph/c/N 대기시스템으로 근사할 수 있음을 고려하면 본 연구의 방법은 M/G/c/N+M 대기시스템의 성능분석으로 확장할 수 있다. 즉, phase 타입 서비스 시간을 모델링할 수 있는 CTMC를 정의하고 본 연구의 해 도출 절차를 이용하면 M/Ph/c/N+M 대기시스템의 성능지표를 얻을 수 있고, 이 값들을 M/G/c/N+M 대기시스템의 성능지표 근사에 사용할 수 있다.

참고문헌

Baccelli, F., *et al.* (1984), Single-server queues with impatient customers, *Adv. Appl. Prob.*, **16**, 887-905.

Boots, N. K. and Tijms, H. C. (1999), A multiserver queueing system with impatient customers, *Mgmt. Sci.*, **45**, 444-448.

Brandt, A. and Brandt, M. (1999), On the M(n)/M(n)/s queue with impatient calls, *Performance Eval.*, **35**, 1-18.

Brumelle, S. L. (1971), On the relation between customer and time averages in queues, *J. Appl. Prob.*, **8**, 508-520.

De Kok, A. G. and Tijms, H. C. (1985), A queueing system with impatient customers, *J. Appl. Prob.*, **22**, 688-696.

Seelen, L. P. (1986), An algorithm for Ph/Ph/c queues , *European J. Oper. Res.*, **23**, 118-227.