J. Soc. Korea Ind. Syst. Eng Vol. 39, No. 2 : 1-10, June 2016
http://dx.doi.org/10.11627/jkise.2016.39.2.001

ISSN : 2005-0461(print)
ISSN : 2287-7975(online)

On the Exact Cycle Time of Failure Prone Multiserver Queueing Model Operating in Low Loading

Woo-Sung Kim* · Dae-Eun Lim**†

*School of Management and Economics, Handong Global University
**Dept. of System and Management Engineering, Kangwon National University

낮은 교통밀도 하에서 서버 고장을 고려한 복수 서버 대기행렬 모형의 체제시간에 대한 분석

김우성* · 임대은**†

*한동대학교 경영경제학부
**강원대학교 시스템경영공학과

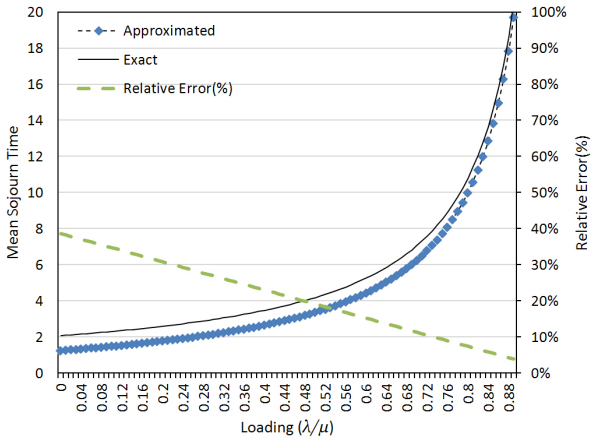
In this paper, we present a new way to derive the mean cycle time of the G/G/m failure prone queue when the loading of the system approaches to zero. The loading is the relative ratio of the arrival rate to the service rate multiplied by the number of servers. The system with low loading means the busy fraction of the system is low. The queueing system with low loading can be found in the semiconductor manufacturing process. Cluster tools in semiconductor manufacturing need a setup whenever the types of two successive lots are different. To setup a cluster tool, all wafers of preceding lot should be removed. Then, the waiting time of the next lot is zero excluding the setup time. This kind of situation can be regarded as the system with low loading. By employing absorbing Markov chain model and renewal theory, we propose a new way to derive the exact mean cycle time. In addition, using the proposed method, we present the cycle times of other types of queueing systems. For a queueing model with phase type service time distribution, we can obtain a two dimensional Markov chain model, which leads us to calculate the exact cycle time. The results also can be applied to a queueing model with batch arrivals. Our results can be employed to test the accuracy of existing or newly developed approximation methods. Furthermore, we provide intuitive interpretations to the results regarding the expected waiting time. The intuitive interpretations can be used to understand logically the characteristics of systems with low loading.

Keywords : G/G/m Queue, Mean Cycle Time Approximation, Failure Prone Queue, Absorbing Markov Chain

1. 서 론

서버 고장(server failure)을 고려한 G/G/m 대기행렬 모형은 오랜 시간동안 생산 공정을 비롯한 다양한 시스템의 성능척도를 추정하는데 사용되어 왔다. 단일 서버로 구성된 모형에 관하여는 내재점 마코프체인(embedded Markov chain)이나 부가변수법(supplementary variable method) 등 다양한 방법을 통하여 평균 체제시간(mean sojourn time)과 같은 성능척도에 관한 정확한 해를 구하는 방법이 제시되어 있으나, 복수 서버 시스템의 경우 마코비안 대기행렬 시스템[10]을 제외하면 정확한 평균 체제시간을 구

Received 26 November 2015; Finally Revised 4 January 2016; Accepted 13 January 2016
† Corresponding Author : del@kangwon.ac.kr



<Figure 1> Comparison of Exact and Approximated Cycle Time of the M/M/1 Queue with Server Breakdown

하는 방법은 현재까지 알려진 바가 없다. 이론적으로는 부가변수법과 같은 방법을 통하여 분석할 수 있지만 서버의 수가 증가함에 따라 상태(state)의 수도 지수적으로 증가하기 때문에 정확한 해(exact solution)를 산출해내기 쉽지 않다. 그렇기 때문에 복수 서버로 구성된 시스템의 경우 근사(approximation) 해법을 통하여 주로 시스템의 성능척도들을 산출하게 된다. 보다 복잡한 시스템의 경우 시뮬레이션을 이용하여 성능을 평가하는 경우도 많다[6]. 많은 근사해법들이 서버 고장을 고려한 대기행렬 모형의 평균 체제시간을 분석하기 위해 제시되었고, 이러한 근사해법들은 평균 체제시간에 관한 근사해를 제공함과 동시에 시스템을 직관적으로 이해하는데 유용한 정보를 제공한다[4, 12]. 하지만 근사해법을 이용하면 필연적으로 오차가 발생할 수밖에 없는데, 이에 대한 예제가 <Figure 1>에 제시되어 있다. <Figure 1>은 서버 고장을 고려한 $M/M/1$ 대기행렬 모형에 대해 체제시간을 두 가지 방법으로 구하여 비교한 것이다. 'Exact'는 정확한 해의 값이며, 'Approximated'는 Hopp and Spearman[4]이 제시한 근사법으로 얻은 결과이다. 상대오차를(정확한 체제시간 값 −근사해법을 통해 계산된 체제시간 값)÷(정확한 체제시간 값)로 정의하고 <Figure 1>에서 점선으로 표시하였다. 상대오차는 교통밀도가 높아지면서 감소하는 것을 볼 수 있다. 생략된 0.9 이상의 교통밀도 수준에서는 상대오차가 5% 미만이었다. 하지만, 0.2 이하의 낮은 교통밀도에서는 30% 이상의 큰 오차를 보인다. 이는 대부분의 생산 공정에서 경제적인 이유 등으로 높은 서버 이용률을 목표로 시스템을 운용하기 때문에 근사해법들이 높은 교통밀도(heavy traffic)에 초점을 두고 개발되었기 때문이다. 그럼에도 불구하고 복잡하게 구성된 생산 시스템에는 적지 않은 공정들이 낮은 교통 밀도 하에서 운영되고 있으며 이러한 공정들의 체제시간 또한 시스템 전체의 성능척도에 영향을 미치게 된다. 특히, 낮은 교통 밀도 하에서 운영된다고 하더라도 서버 고장이나 셋업(setup)과 같은 사건들로 인해 추가적으로 시스템에서 지체가 발생하게 되므로 전체 시스템의 성능척도를 산출하기 위해서는 이러한 공정들의 체제시간 또한 분석되어야 한다.

본 논문에서 분석하는 고장이 발생하는 낮은 교통밀도의 시스템은 반도체 제조공정에서 찾아볼 수 있다. 먼저 병목 공정이 있는 시스템의 예를 들 수 있다. 서버 고장이나 셋업 등 서비스 시간 외에 지체(delay)를 일으키는 사건들이 발생하지 않는, 공정들이 직렬로 연결된 시스템의 지체시간 분석을 가정하자. Avi-itzhak[1]의 결과에 의하면 서비스 시간이 확정적인 시간을 갖는 직렬 대기행렬(tandem queue) 모형의 경우 병목 공정 후에서는 지체가 발생하지 않으며, 따라서 자재(lot 또는 WIP)의 지체시간을 분석할 때는 병목 공정 후의 공정들은 고려하지 않는다. 즉, 병목공정의 후속 공정들에서 자재들이 머무는 시간은 각 공정의 서비스 시간의 합으로 표현된다. 또한, 자재들이 직렬 대기행렬 모형에서의 총 지체시간은 병목현상을 서비스 시간으로 갖는 단일서버 대기행렬 모형의 지체시간과 같음을 증명하였다. 여기서 지체가 없는 후속 공정은 본 연구의 대상이 되는 낮은 교통밀도를 갖는 공정으로 간주할 수 있을 것이다. 이 결과를 바탕으로, Morrison[11]과 Kim and Morrison[8]은 확정적인 서비스 시간을 갖는 직렬 대기행렬 모형을 통하여 반도체 제조라인 내 클러스터 장비의 지체 시간을 분석하였다. 위 논문들은 공통적으로 시스템의 서버 고장과 같은 사건을 고려하지 않으며, 따라서 병목 공정의 후속 공정에서는 지체가 발생하지 않는다고 가정하고 있다. 이러한 가정 때문에 병목 공정 후 공정들은 일반적으로 소수의 서버로 운영되지만 서버 고장과 같은 사건들에 의해 병목현상 후 낮은 교통밀도 하에서 운영되는 공정에서도 지체가 생길 수 있으며, 이는 시스템 전체의 체제시간에 영향을 미친다. 또 다른 낮은 교통밀도를 갖는 시스템의 예는 셋업이 필요한 반도체 클러스터 장비에서도 찾아볼 수 있다. 반도체 공정의 경우 생산 중 제품의 종류가 바뀌는 경우 클러스터 장비의 셋업이 필요한데 장비 셋업을 위해서는 셋업이 필요한 공정을 포함한 선행 공정에 웨이퍼가 존재하지 않아야만 한다[9]. 이 경우 같은 lot 타입 의 연속된 lot들을 하나의 고객으로 간주하면, 클러스터 장비는 셋업 과정으로 인하여 낮은 교통밀도 하에서 운영됨을 알 수 있다.

기존에 제시된 분석 방법들과 한계를 살펴보자. 서버 고장을 고려한 대기행렬 시스템의 초기 연구로 White and Christie[13]의 것을 찾아볼 수 있다. 서버 고장을 고려한 $M/G/1$ 대기행렬 모형의 성능척도에 관하여는 Avi-Itzhak and Naor[2], 그리고 Gaver[3]에 의해 분석되어 있다. 위 논

문들에서는 서버 고장은 포아송 과정Poisson process)으로 일어나며, 수리 기간은 일반 분포를 따른다고 가정한다. 고객의 도착과정이 복합(compound) 포아송 과정을 따르고 서비스 시간은 얼랑 분포를 따르는, 서버 고장이 발생하는 대기행렬 모형에 대해서는 Kim and Kang[5]에 의해 분석되었다. 복수 서버 시스템에 대하여는 고장과정과 수리과정이 포아송 분포를 따르는 $M/M/m$ 대기행렬 모형에 관한 안정상태 고객 수 분포를 Mitrany and Avi-Itzhak [10]이 제시하였다. 그러나 서버 고장을 고려한 $G/G/m$ 대기행렬 모형에 대하여는 평균 체제시간에 관한 해를 정확히 구하는 방법이 제시된 바가 없다. 서버 고장을 고려한 $G/G/m$ 대기행렬 모형의 분석은 대부분 근사해법에 의존하고 있는데 현재 쓰이는 근사해법들은 Hopp and Spearman [4]과 Whitt[14]에 의해 개발되었다. Morrison and Martin [12]에 의해 확장된 Whitt[14]의 결과는 IBM 반도체 생산공정을 분석하는데 사용되었다. 이 근사해법들은 높은 교통밀도에서는 만족할만한 결과를 제공하지만 낮은 교통밀도에서는 상대적으로 큰 오차를 보이고 있다. 마지막으로 본 연구의 초기 결과들은 Kim and Morrison[7]에 제시되어 있다.

위의 사례들과 기존 연구들을 볼 때 낮은 교통밀도 하에서 서버 고장을 고려한 복수 서버 대기행렬 모형의 분석의 필요성을 찾아 볼 수 있다. 본 논문에서는 0에 가까운 낮은 교통밀도 하에서 근사해가 아닌 정확한 평균 체제시간을 분석한다. 본 논문의 결과를 이용하여 시스템의 특성을 정확하게 파악할 수 있으며 부수적인 효과도 기대할 수 있다. 이는 정확한 해를 제시하므로 추후 새롭게 개발된 근사해법을 검증하는데 사용될 수 있다는 점이다. 또한 본 논문에서는 새로운 분석방법을 제시하는데, 서비스 시간이 단계형 확률 변수를 따르는 대기행렬 모형들과 집단 도착 대기행렬 모형들을 2차원의 전이율 다이어그램 (transition rate diagram)을 갖는 마코프 체인 모형으로 분석하는 방법이다. 또한 비병목 공정의 체제시간에 관한 분석에 적용하는 방법들이 추가되었다.

본 논문은 다음과 같이 구성된다. 제 2장에서는 먼저 모형을 수학적으로 정의한다. 고객들의 도착간은 일반분포를, 서비스 시간은 지수분포를 갖는 대기행렬 시스템과 단계형 확률변수인 시스템이 낮은 교통밀도 하에서 흡수 마코프 체인으로 분석될 수 있음을 보이고 평균 체제시간을 분석한다. 동일한 방법으로 고객 집단 도착 시스템도 분석할 수 있음을 보인다. 제 3장에서는 제 2장의 결과를 이용하여 고객들의 도착간격과 서비스 시간 모두 일반분포를 따르는 대기행렬 모형의 평균 체제시간을 재생 이론을 이용하여 분석한다. 제 4장에서는 본 논문의 결과를 사용하여 기존의 근사해법들을 검증해본다. 제 5장에서는 본 연구의 결과와 방법에 대한 의의를 논의한다.

2. 서비스 시간이 단계형 확률변수를 따르는 대기행렬 모형 분석

본 장에서는 논문의 모형을 수학적으로 정의하고 낮은 교통밀도 하에서 서버고장을 고려한 $G/M/m$ 대기행렬 모형의 평균 체제시간을 분석한다. 서버 고장을 고려한 $M/M/m$ 대기행렬 모형의 평균 체제시간에 관한 결과는 Mitrany and Avi-Itzhak[10]에 의해 분석되었지만, 그 방법은 도착간격과 서비스 시간의 분포가 지수분포일 때만 적용 가능하다. 본 논문에서는 흡수 마코프체인을 통하여 평균 체제시간을 분석하는 새로운 방법을 제시한다. 동일한 방법으로 서비스 시간이 단계형 확률변수를 따르는 대기행렬 모형 또한 분석할 수 있다.

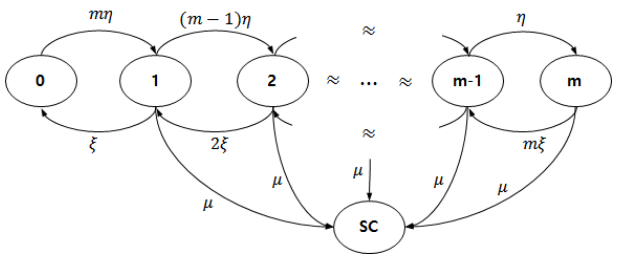
주요 가정들은 다음과 같다. 먼저, 대기행렬 모형의 교통밀도가 0에 가깝다. 교통밀도가 낮기 때문에 임의의 고객이 시스템에 도착했을 때 시스템 안에는 다른 고객은 존재하지 않는다고 가정한다. 따라서 고객의 지체는 서버 고장으로 인해서만 발생한다. 각각의 서버에서는 고장이 발생해 서비스가 불가한 상태(고장 상태, failure period)와 서비스가 가능한 상태(운영 상태, available period)가 반복적으로 번갈아가며 일어나는데, 이를 각각의 기간에 머무는 시간이 지수분포를 따르는 교대 재생과정(alternating renewal process)으로 가정한다. 서버가 고장나서 수리 후에 고객의 서비스가 처음부터 반복되는지 여부에 따라 축출-계속형(preemptive-resume) 시스템과 축출-반복형(preemptive-repeat) 시스템으로 나뉘는데 본 논문에서는 축출-계속형 시스템을 가정한다. 축출-계속형 시스템에서는 서버 고장에 의해 중단되기 전까지 받은 고객의 서비스의 양은 유효하며 수리 후 재개되는 서비스는 축출 시점부터 계속된다. 축출 반복형은 서비스가 서버 고장에 의해 중단되면, 그때까지 받은 고객의 서비스는 무효로 간주한다. 서비스 시간이 지수분포를 따를 때에는 지수분포의 무기억 속성(memoryless property)에 의해 두 시스템은 동일한 행태를 보이기 때문에 $G/M/m$ 모형을 다루는 본 장에서는 축출-계속형과 축출-반복형으로 구분하는 것이 수학적으로는 무의미하다. 그러나 다음 장에서는 서비스 시간이 일반분포를 따르는 모형을 다루는데 이 모형에서는 축출-계속형과 축출-반복형의 수학적 분석에 차이가 발생한다. 또 다른 가정으로, 임의의 고객이 서비스 도중 서비스 받고 있던 서버의 고장이 발생할 경우 다른 서버로 즉시 이동하여 서비스를 계속하여 받으며 이동시간은 무시한다. 만약 모든 서버가 고

장 상태가 된다면 고객은 서버 중 하나가 수리될 때까지 기다리게 되며 이전까지 받았던 서비스는 유효하다. 이전에 언급했다시피 교통밀도가 0에 가깝기 때문에 서버 고장에 의한 대기시간의 발생을 제외하고는 다른 대기시간은 발생하지 않는다. 이제 본 논문에서 사용되는 모형의 매개변수(parameter)들을 정의한다. 각 서버의 서비스율(service rate), 고장율(failure rate)과 수리율(repair rate)을 각각 μ, ξ와 η로 정의한다. 각 서버는 고장 상태와 가동상태가 교대로 반복되는데 가동 기간과 수리기간은 각각 위에 정의된 고장율과 수리율을 갖는 지수분포를 따르며 서로 독립이라고 가정한다.

위와 같은 가정들을 바탕으로 시스템을 흡수 마코프 체인 모형(absorbing markov chain)을 이용하여 모델링한다. 안정상태(steady-state)에서 도착하는 임의의 고객은 교통밀도가 0에 가까울 때, 다른 고객이 아무도 없는 상태의 시스템에 도착하게 되는데, 도착하는 고객이 보게 되는 가용한(고장나지 않은) 서버의 수를 나타내는 확률변수를 N이라고 정의한다($N \in \{0, 1, \cdots, m\}$). 시스템이 안정상태에 있고 서버 고장과 수리 과정이 독립이므로, N은 식 (1)과 같은 이항분포(binomial distribution)을 따르게 된다. 이에 관한 엄밀한 수학적인 증명은 Avi-Itzhak[1]에 제시되어 있다.

$$P(N=n)=\alpha_n=\binom{m}{n}\left(\frac{\eta}{\xi+\eta}\right)^n\left(\frac{\xi}{\xi+\eta}\right)^{m-n}. \qquad (1)$$

마코프 체인 모형의 각 상태 $\{0, 1, \cdots, m\}$을 가용한 서버의 수로 가정하면, <Figure 2>와 같은 전이율 다이어그램을 갖는 연속시간 시간동질(continuous-time homogeneous) 마코프 체인 모형을 얻을 수 있다. 상태 n에 있을 때 상태 $n+1$로의 전이율(transition rate)은 $(m-n)\eta$이며, 상태 $n-1$로의 전이율은 $n\xi$이다(n이 0이나 m일 경우는 제외한다). 상태 SC는 서비스의 완료(service completion)를 나타내는 흡수 상태이다. 가용한 서버가 1개 이상이라면, 각 상태에서 μ의 전이율로 흡수 상태(absorbing state)로 전이가 이루어지게 되며, 이는 서비스가 완료되었음을 의미한다. 따라서, 마코프 체인에서 흡수 상태로 전이가 이루어질 때까지 걸리는 시간은 고객의 평균 체제 시간과 동일한데, 이는 낮은 교통밀도를 갖는다는 특수한 가정 때문이다. 상태들을 $\{0, 1, \cdots, m, SC\}$의 순서로 배열 하면 다음과 같은 전이율 행렬 Q를 얻을 수 있으며 행렬 Q를 다음과 같이 분할한다. 이 때, 행렬 $\Theta_{i,j}$는 모든 원소가 0이며 크기가 $i \times j$인 행렬이다.



<Figure 2> Transition Rate Diagram for the G/M/m queue Operating in Low Loading

$$Q=\begin{pmatrix} -m\eta & m\eta & 0 & \cdots & 0 & 0 & 0 \\ \xi & -(m-1)\eta-\xi-\mu & (m-1)\eta & \cdots & 0 & 0 & \mu \\ 0 & 2\xi & -(m-2)\eta-\xi-\mu & \cdots & 0 & 0 & \mu \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -\eta-(m-1)\xi-\mu & \eta & \mu \\ 0 & 0 & 0 & \cdots & m\xi & -m\xi-\mu & \mu \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \end{pmatrix},$$

$$Q=\begin{pmatrix} T & T^0 \\ \Theta_{1,m+1} & \Theta_{1,1} \end{pmatrix}$$

$$T=\begin{pmatrix} -m\eta & m\eta & 0 & \cdots & 0 & 0 \\ \xi & -(m-1)\eta-\xi-\mu & (m-1)\eta & \cdots & 0 & 0 \\ 0 & 2\xi & -(m-2)\eta-\xi-\mu & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -\eta-(m-1)\xi-\mu & \eta \\ 0 & 0 & 0 & \cdots & m\xi & -m\xi-\mu \end{pmatrix},$$

$$T^0=(0,\ \mu,\ \mu,\ \cdots,\ \mu).$$

행렬 T와 T^0를 구성한 결과가 제시되어 있다. T와 T^0는 크기가 각각 $(m+1)\times(m+1)$과 $(m+1)\times 1$인 행렬이다. 행렬 T는 상태 SC를 제외한 일시 상태(transient state)들 간의 전이율을 나타내는 행렬이다. 이 때, 확률변수 Y를 흡수상태로 전이될 때 까지 걸리는 시간이라고 정의하고 Y의 확률밀도함수와 라플라스 변환을 각각 $f(y)$와 $F^*(\theta)$로 정의하면 흡수 마코프 체인 모형에 관한 기존의 결과를 이용하여 다음과 같은 결과를 얻을 수 있다.

$$\begin{aligned} f(y) &= Ae^{Ty}(-Te), \\ F^*(\theta) &= A(\theta I - T)^{-1}(-Te), \\ E(Y) &= A(-T)^{-1}e, \\ A &= (\alpha_0, \alpha_1, \cdots, \alpha_m). \end{aligned} \qquad (2)$$

A는 $(m+1)$개의 원소로 구성된 행 벡터이며, 마코프체인의 초기확률상태를 나타낸다. e는 모든 원소를 1로 갖는 열벡터이다. 확률 변수 Y가 마코프체인이 흡수 상태로 전이될 때까지 걸리는 시간을 나타내므로, $E(Y)$는 낮은 교통밀도 하에서 고객의 평균 체제시간을 의미한다. 예를 들어, 서버가 두 개인 경우, T와 A는 다음과 같다.

$$T=\begin{pmatrix} -2\eta & 2\eta & 0 \\ 2\xi & -\xi-\eta-\mu & \eta \\ 0 & 2\xi & -2\xi-\mu \end{pmatrix},$$

$$A=\left(\left(\frac{\xi}{\eta+\xi}\right)^2,\ 2\left(\frac{\xi}{\eta+\xi}\right)\left(\frac{\eta}{\eta+\xi}\right),\left(\frac{\eta}{\eta+\xi}\right)^2\right).$$

식 (2)을 통하여 구한 고객의 평균 체제 시간은 식 (3)과 같다.

$$
\begin{aligned}
E(Y) &= \alpha(-T)^{-1}e \\
&= \frac{2\xi^4+2\eta^3(\eta+\mu)+4\xi\eta^2(2\eta+\mu)+\xi^3(8\eta+3\mu)+\xi^2(12\eta^2+5\eta\mu+\mu^2)}{2\eta(\eta+\xi)^2\mu(2\xi+\eta+\mu)} \\
&= \frac{1}{\mu}+\left(\frac{\xi}{\eta+\xi}\right)^2\left(\frac{1}{2\eta}\right)+\left(\frac{1}{2\eta}\right)\frac{\xi^2(2\xi^2+2\eta(\eta+\mu)+\xi(4\eta+\mu))}{(\eta+\xi)^2\mu(2\xi+\eta+\mu)}
\end{aligned} \tag{3}
$$

식 (3)을 해석하면 우리는 시스템의 행태를 이해할 수 있다. 식 (3)번의 마지막 수식의 첫 번째 항은 서비스 시간을 나타낸다. 두 번째 항은 임의의 고객이 도착했을 때 모든 서버가 고장상태임을 나타낸다. $(\xi/(\eta+\xi))^2$의 확률로 고객이 도착했을 때 두 서버 모두 고장 상태이고, 이 때 고객은 두 서버 중 하나의 수리가 완료될 때까지 기다려야 하며 평균적으로 $(1/2\eta)$시간만큼 기다리게 된다. 세 번째 항은 고객이 서비스를 받는 도중 모든 서버에 고장이 발생해 수리될 때까지 지체되는 시간을 나타낸다.

위와 같은 분석법은 서비스 시간의 분포가 지수분포일 때 뿐 아니라, 지수분포들로 구성되는 단계형 확률변수에도 적용 가능하다. 서비스 시간이 단계형 확률변수를 따를 때에는 상태가 2차원 형태로 구성된다. 예를 들어, 서비스 시간이 얼랑 분포(Erlang(n,μ))를 따르는 대기행렬의 모형의 경우 가용한 서버 수를 i로, 잔여 서비스 단계를 j로 나타내자. 상태 (i,j)에 대한 전이율 다이어그램은 <Figure 3>과 같다. 일시 상태들을 $\{(0,n),(1,n),\cdots,(m,n),(0,n-1),(1,n-1),\cdots,(m-1,1),(m,1)\}$과 같이 배열하면 일시 상태 간의 전이율을 나타내는 행렬 T는 다음과 같다.

$$
T=\begin{pmatrix}
C_1 & C_0 & \Theta_{m+1,m+1} & \cdots & \Theta_{m+1,m+1} \\
\Theta_{m+1,m+1} & C_1 & C_0 & \cdots & \Theta_{m+1,m+1} \\
\Theta_{m+1,m+1} & \Theta_{m+1,m+1} & C_1 & \cdots & \Theta_{m+1,m+1} \\
\vdots & \vdots & \vdots & \cdots & \vdots \\
\Theta_{m+1,m+1} & \Theta_{m+1,m+1} & \Theta_{m+1,m+1} & \cdots & C_1
\end{pmatrix},
$$

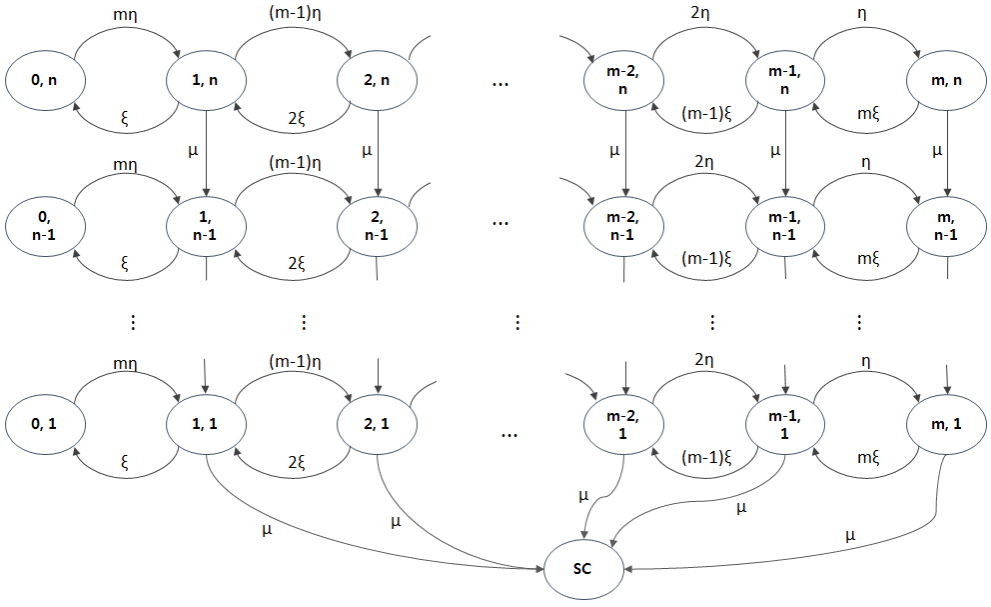
단, C_1과 C_0는 다음과 같이 정의된다.

$$
C_1=\begin{pmatrix}
-m\eta & m\eta & 0 & \cdots & 0 & 0 \\
\xi & -(m-1)\eta-\xi-\mu & (m-1)\eta & \cdots & 0 & 0 \\
0 & 2\xi & -(m-2)\eta-2\xi-\mu & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & -\eta-(m-1)\xi-\mu & \eta \\
0 & 0 & 0 & \cdots & m\xi & -m\xi-\mu
\end{pmatrix},
$$

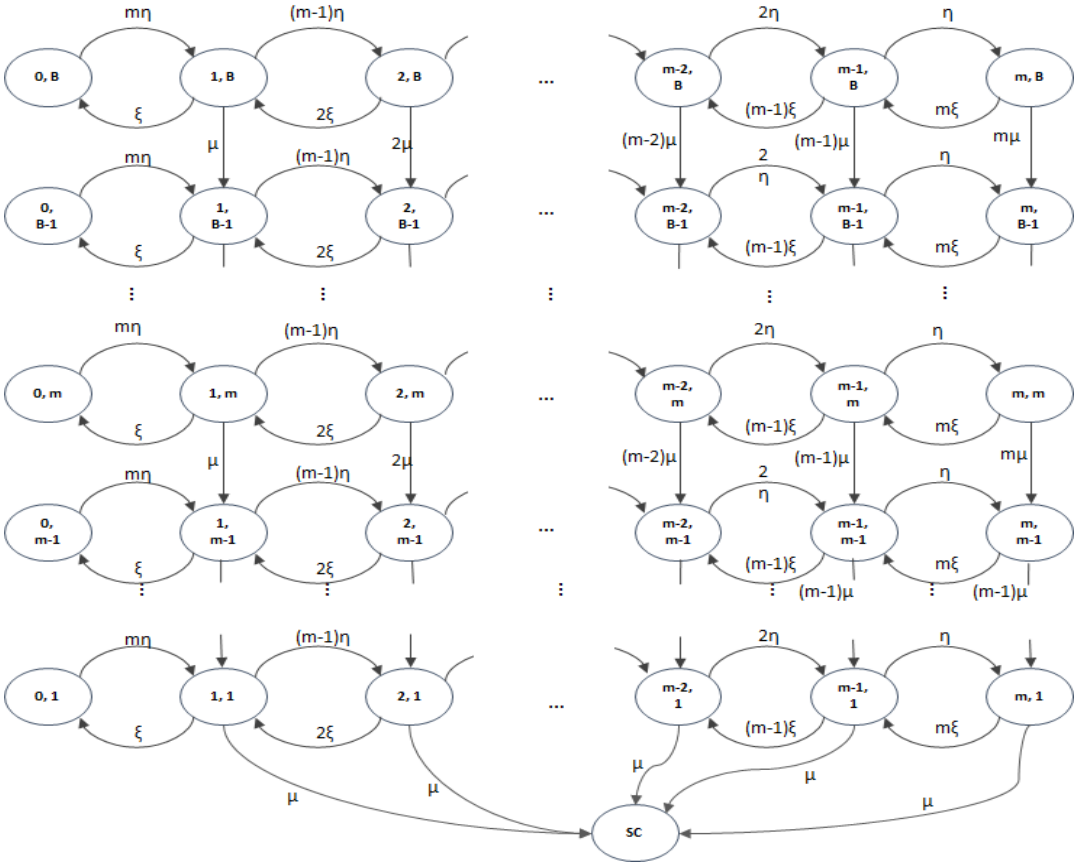
$$
C_0=\begin{pmatrix}
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \mu & 0 & \cdots & 0 & 0 \\
0 & 0 & \mu & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & \mu & 0 \\
0 & 0 & 0 & \cdots & 0 & \mu
\end{pmatrix}.
$$

위 전이율 행렬 T와 식 (2)의 결과를 이용하면 시스템의 성능척도들을 계산할 수 있다. 초기 상태 확률 벡터 A는 상태 (i,n)에 해당하는 원소들만 확률값을 가지며 그 값은 α_i이다.

이러한 분석 방법은 고객이 집단으로 도착하는 대기행렬 모형에도 적용될 수 있다. 고객이 집단으로 도착하며 집단 크기는 범위가 1부터 B까지의 값을 갖는 이산 확률변수(discrete random variable) X를 따른다고 가정하자. 각 고객의 서비스 시간은 서비스율이 μ인 지수분포를 따른다. 확률변수 X의 확률 질량 함수(probability mass func-



<Figure 3> Transition Rate Diagram for a Queueing Model with Erlang Service Time Distribution



<Figure 4> Transition Rate Diagram for a Queueing Model with Batch Arrival

tion)를 $P(X=i)=b_i(i=1,2,\cdots,B)$라고 정의하면, 전이율 다이어그램은 <Figure 4>와 같다(단, $B>m$). 상태 (i,j)의 원소는 각각 가용한 서버의 수와 도착한 집단의 크기를 나타낸다. 예를 들어, 상태 $(m-1,B)$는 크기 B인 집단이 도착했을 때, 가용한 서버의 수는 $m-1$임을 나타낸다. 이 경우 도착한 고객 중에 $m-1$명의 고객이 도착 즉시 가용한 서버에서 서비스를 받게 되며 시스템의 서비스율은 $(m-1)\mu$이다. 비슷한 방식으로 일시 상태간의 전이율 행렬을 구한 후 식 (2)를 이용하면 평균체제시간을 구할 수 있다. 상태 (i,j)에 대한 초기상태 확률은 α_i와 b_j의 곱이다. 이와 같이 단계형 확률 변수 뿐 아니라 집단 도착 대기행렬 모형의 경우에도 흡수 마코프 체인을 이용하여 체제시간을 구할 수 있다.

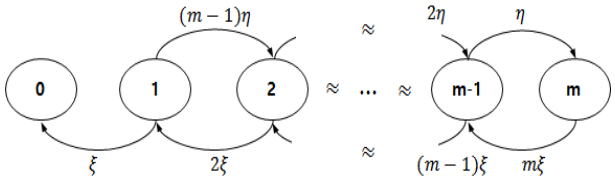
3. 서비스 시간이 일반 분포를 따른 대기행렬 모형 분석

이번 장에서는 낮은 교통 밀도 하에서 서비스 시간이 일반 분포를 따르는 서버 고장을 고려한 G/G/m 모형을 분석한다. 제 2장에서와 동일하게 서버의 가용기간과 고장기간은 지수분포를 따른다고 가정하면 마코프 체인 모형과 재생 이론 (renewal theory)을 이용하여 대기행렬 모형의 체제시간의 정확한 값을 구할 수 있다. 서비스 시간의 분포가 일반분포를 따르기 때문에 서비스 시간을 나타내는 확률 변수를 S로, 이의 확률밀도함수를 $f_s(t)$로 정의하자. 제 2장의 식 (3)을 관찰해 보면, 낮은 교통 밀도 하에서 체제시간은 세 가지 항들의 합으로 구성된다. 첫 번째로, 고객은 서비스 시간만큼 시스템에 머무르게 된다. 두 번째로, 고객이 시스템에 도착했을 때 모든 서버가 고장 상태일 경우, 고객은 서버가 수리될 때까지 기다린다. 세 번째로, 고객이 서비스 받는 도중 모든 서버의 고장이 발생하면 고객은 서버가 수리될 때까지 기다리게 된다. 모든 서버 중 하나의 서버가 수리 되는 즉시 고객은 수리가 완료된 서버로 이동하여 즉시 서비스를 이어서 받는 것을 참고하자. 확률 변수 B를 고객의 서비스 시간 동안 모든 서버가 동시에 고장 상태에 있게 되는 사건 수로 정의하면 평균 체제시간은 다음과 같은 식 (4)로 표현될 수 있다.

$$\lim_{\lambda\to 0}E[CT]=\frac{1}{\mu}+\left(\frac{\xi}{\eta+\xi}\right)^m\left(\frac{1}{m\eta}\right)+\left(\frac{1}{m\eta}\right)E[B] \qquad (4)$$

식 (4)에서 $E[B]$를 제외한 모든 항들은 시스템 파라미터로 주어져 있기 때문에 $E[B]$를 구하면 평균 체제시간을 구할 수 있다. 우리는 두 단계를 이용하여 $E[B]$를 구할 수 있다. 먼저, 흡수 마코프 체인 모형을 이용하여 고객의 서비스 도중 모든 서버가 고장 날 때까지 걸리는 평균 시간을 계산한다. 그 후에, 모든 서버가 고장 나는 사건의 수를 재생 이론을 이용하여 계산하면 $E[B]$를 구할 수 있다.

먼저, 흡수 마코프 체인 모형을 이용하여 모든 서버가 고장 날 때까지 걸리는 시간을 계산한다. 마코프 체인의 상태를 안정상태에서 가용한 서버 숫자로 정의하면 마코프 체인은 <Figure 5>와 같다.



<Figure 5> Transition Rate Diagram for an Absorbing Markov Chain Model

상태 0과 상태 m을 제외하면 상태 n에서 상태 $n+1$로의 전이율은 $(m-n)\eta$이고 상태 $n-1$로의 전이율은 $n\xi$이다. 흡수 상태 0은 모든 서버가 고장인 상태를 의미한다. 이 때, 마코프 체인이 흡수 상태로 전이될 때까지 걸리는 시간은 모든 서버가 고장상태로 전이될 때까지 걸리는 시간이다. 상태들을 $\{1, 2, \cdots, m, 0\}$과 같이 재배열하면 일시 상태간의 전이율을 나타내는 행렬 T는 식 (5)와 같다.

$$T=\begin{pmatrix} -(m-1)\eta-\xi & (m-1)\eta & \cdots & 0 & 0 \\ 2\xi & -(m-2)\eta-2\xi & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & -\eta-(m-1)\xi & \eta \\ 0 & 0 & \cdots & m\xi & -m\xi \end{pmatrix}. \quad (5)$$

식 (2)를 통하여 흡수할 때까지 걸리는 시간의 확률 밀도함수와 라플라스 변환, 그리고 평균을 구할 수 있는데, 흡수 할 때 까지 걸리는 시간이 체제 시간이었던 제 2장과는 달리, 위 모형의 흡수할 때까지 걸리는 시간은 서비스 도중 모든 서버가 고장 상태가 될 때까지 걸리는 시간을 나타낸다. 초기확률을 나타내는 벡터 A는 $(\alpha_0+\alpha_1, \alpha_2, \alpha_3, \cdots, \alpha_m)$이다. 벡터 A의 첫 번째 항이 $\alpha_0+\alpha_1$인 이유는 고객이 도착시점에 모든 서버가 고장 상태일 경우에는 서버가 수리 될 때까지 걸리는 시간이 식 (4)의 두 번째 항에 반영되어 있고, 수리 후에는 가용한 서버 수가 하나인 상태에서 서비스를 시작하기 때문이다. 위 마코프 체인이 흡수 상태로 전이될 때까지 걸리는 시간을 재생이론의 재생간격으로 보고, 시간 t까지 발생한 재생 사건의 수를 $N(t)$로 정의하자. $E[B]$는 서비스 시간동안 모든 서버에 일어나는 평균 횟수를 나타내므로 서비스 시간에 조건을 취하면 다음과 같이 $E[B]$의 값을 얻을 수 있다.

$$E[B]=\int_{-\infty}^{\infty} E[N(t)]f_S(t)dt. \quad (6)$$

식 (6)에서 볼 수 있듯이, $E[N(t)]$의 값을 계산하면 E[B]의 값을 구할 수 있다. $E[N(t)]$의 라플라스 변환을 $M^*(\theta)$로 정의하면 재생 이론을 통하여 다음의 식 (7) 같은 결과를 얻는다.

$$\begin{aligned} M^*(\theta) &= \frac{G^*(\theta)}{\theta(1-F^*(\theta))}, \\ G^*(\theta) &= A_G(\theta I - T)^{-1}(-Te), \\ F^*(\theta) &= A_F(\theta I - T)^{-1}(-Te). \end{aligned} \quad (7)$$

초기확률 벡터는 각각 다음과 같다.

$$\begin{aligned} A_G &= (\alpha_0+\alpha_1, \alpha_2, \cdots, \alpha_m), \\ A_F &= (1, 0, \cdots, 0). \end{aligned} \quad (8)$$

이는 처음에 고객이 서버에 도착하게 될 때 보게 되는 가용한 서버 수는 이항분포를 따르지만, 그 후 서비스 도중 모든 서버에 고장이 일어나게 될 경우, 수리 완료가 될 때는 항상 하나의 서버만 가용한 상태가 되기 때문이다. 식 (7)의 라플라스 변환에 역변환을 취하여 식 (6)에 대입한 뒤 적분을 계산하면 평균 체제 시간을 구할 수 있다.

위 방법을 이용하여 대기행렬 모형의 체제시간을 구해보자. $M/M/2$ 대기행렬 모형의 체제시간에 관한 결과는 식 (3)에 제시되어 있지만, 위 방식으로도 동일한 결과를 얻음을 확인할 수 있다. 전이율 행렬 T와 초기확률 벡터들은 식 (9)와 같다.

$$\begin{aligned} T &= \begin{pmatrix} -2\eta & 2\eta & 0 \\ 2\xi & -\xi-\eta-\mu & \eta \\ 0 & 2\xi & -2\xi-\mu \end{pmatrix}, \\ A_G &= \left(\left(\frac{\xi}{\eta+\xi}\right)^2+2\left(\frac{\xi}{\eta+\xi}\right)\left(\frac{\eta}{\eta+\xi}\right), \left(\frac{\eta}{\eta+\xi}\right)^2\right), \\ A_F &= (1, 0). \end{aligned} \quad (9)$$

식 (7)로부터 $G^*(\theta)$와 $F^*(\theta)$를 구하면 다음과 같다.

$$G^*(\theta)=\frac{\xi^2(2\xi^2+2\eta(\theta-\eta)+\xi(4\eta+\theta))}{(\eta+\xi)^2(2\xi^2+\theta(\eta+\theta)+\xi(4\eta+\theta))}, \quad (10)$$

$$F^*(\theta)=\frac{\xi^2(2\xi^2+2\eta(\theta-\eta)+\xi(4\eta+\theta))}{2\xi^2+4\xi\eta+\xi\theta+\eta\theta+\theta^2}.$$

위 식을 (7)을 대입하여, $M^*(\theta)$를 구하면 다음과 같다.

$$M^*(\theta)=\frac{\xi^2(2\xi^2+2\eta(\theta-\eta)+\xi(4\eta+\theta))}{\theta(\xi+\eta)^2(4\eta\xi+\theta(\eta+\theta))}, \quad (11)$$

역변환을 취하고 식 (6)에 대입하면 $E[B]$를 구할 수 있다.

$$E(B)=\frac{\xi^2(2\xi^2+2\eta(\eta+\mu)+\xi(4\eta+\mu))}{(\eta+\xi)^2\mu(2\xi+\eta+\mu)}. \quad (12)$$

위에서 구한 값을 식 (4)에 대입하면 식 (3)과 일치하는 것을 확인할 수 있다.

4. 응용 사례

본 장에서는 앞의 장에서 구한 방법론들이 실제 생산 공정에서 어떻게 적용될 수 있는지 방법들을 제시하고자 한다.

4.1 비병목 공정의 체제시간에 관한 분석

본 논문의 방법론이 교통밀도가 0에 가까이 낮은 상황에서의 체제시간에 관한 결과라는 것에서 착안하면, 복잡한 생산 시스템의 비병목 공정들의 체제시간에 관한 유용한 정보를 제공하는 데 사용될 수 있다. 대부분의 생산 공정의 경우 병목 공정 후에 있는 공정들은 적은 수의 서버들로 운영되고 있다. 하지만 고장이나 셋업과 같은 사건들이 자주 일어나는 상황에서 적은 수의 서버로 공정을 운영할 경우, 체제시간이 길어지게 되므로 이러한 상황을 분석하여 적절한 수의 서버를 배치하는 것이 필요하다. 수치 예제를 통하여 우리는 서버 수가 증가함에 따라 체제시간이 어떻게 감소하는지 살펴볼 수 있다. <Table 1>에서 서비스 시간이 지수분포일 때 다양한 고장율에 따른 체제시간이 계산되어 있다. 고장율이 증가함에 따라 체제시간 또한 증가하는 것은 자명하나, 우리는 어느 정도 체제시간이 증가하는지를 분석함으로서 시스템에 관한 유용한 정보를 얻을 수 있다. 먼저, 서비스율(μ)이 5이기 때문에, 서버 고장을 무시하면 체제 시간이 0.2의 값을 갖게 됨을 참고하자. 서버 고장이 발생하고 발생율이 1이 될 경우, <Table 1>에 따르면 고장의 영향을 받지 않기 위해서는(고장이 없는 시스템과 같은 성능을 내기 위해서는) 시스템의 서버 수가 3대 이상이어야 한다. 마찬가지로, 만일 서버 3대로 운영되고 있는 시스템에서 고장율이 1에서 5로 다섯 배가 증가한다면, 비슷한 성능을 얻기 위해서 서버 4대를 더 설치해야 함을 의미한다. 이와 같은 분석을 통하여, 비병목 공정에서의 서버의 대수를 산정하는데 유용한 정보를 얻을 수 있다.

<Table 1> Mean Sojourn Time for the G/M/m Queueing Model

	Sojourn Time(μ = 5)					
Number of Servers(m)	$\xi=1,$ $\eta=10$	$\xi=3,$ $\eta=10$	$\xi=5,$ $\eta=10$	$\xi=10,$ $\eta=10$	$\xi=15,$ $\eta=10$	$\xi=20,$ $\eta=10$
1	0.2291	0.2831	0.3333	0.45	0.56	0.6667
2	0.2021	0.2142	0.2311	0.2804	0.332	0.3838
3	0.2002	0.2029	0.2091	0.2334	0.2634	0.2954
4	0.2	0.2006	0.2029	0.2152	0.2336	0.2551
5	0.2	0.2001	0.2009	0.2072	0.2188	0.2335
6	0.2	0.2	0.2003	0.2035	0.2107	0.2211
7	0.2	0.2	0.2001	0.2017	0.2063	0.2135
8	0.2	0.2	0.2	0.2008	0.2037	0.2087
9	0.2	0.2	0.2	0.2004	0.2022	0.2057
10	0.2	0.2	0.2	0.2002	0.2013	0.2037

4.2 근사해법의 이론적 검증

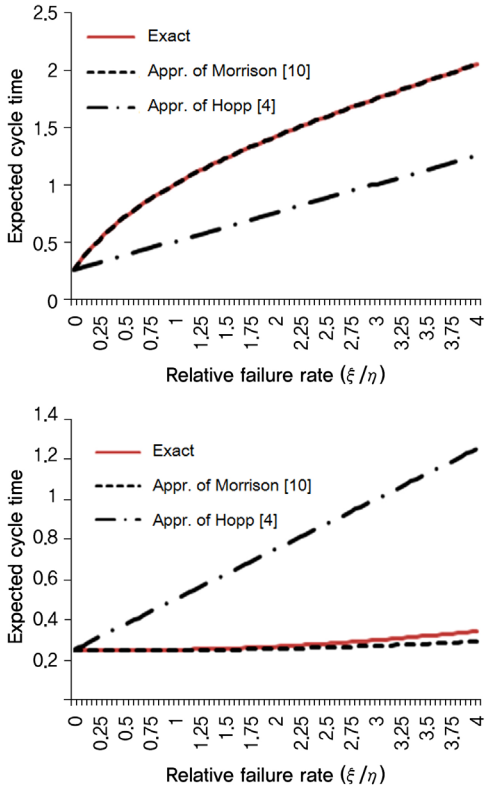
앞의 장에서 구한 방법이 정확한 해임을 참고하자. 정확한 해를 이용하여 체제시간에 관한 근사해법들을 이론적으로 테스트할 수 있다. 이 장에서는 본 논문의 결과를 이용하여 Hopp and Spearman[4]와 Kim and Morrison[9]의 서버고장을 고려한 G/G/m 대기행렬 모형의 체제시간에 관한 근사해법들을 테스트한다. 위 두 논문에 제시된 근사해법 식은 각각 식 (13)과 식 (14)와 같다.

$$E[CT]\approx\frac{1}{\mu^*}+\frac{1}{\mu^*}\left(\frac{C_{SE}^2+C_A^2}{2}\right)\frac{(\rho^*)^{-1+\sqrt{2m+2}}}{m(1-\rho^*)}, \quad (13)$$

$$E[CT]\approx(T+H+P)+(1-A)^mE[R_m]+\frac{(\phi/\mu)+m_I}{1-(1-A)^m}+\frac{1}{\mu_e}\left(\frac{C_{SE}^2+C_A^2}{2}\right)\frac{(\rho_e)^{-1+\sqrt{2m+2}}}{m(1-\rho_e)}. \quad (14)$$

각각의 근사해법 식의 변수들은 Hopp and Spearman[4]와 Morrison and Martin[12]의 정의를 따른다.

위 식에서, 교통밀도가 0에 가까워지면 낮은 교통밀도 하에서 체제시간에 관한 근사해를 구할 수 있다. 결과가 <Figure 6>에 제시되어 있다. X축은 고장율을 수리율로 나눈 상대적 고장율로서 수리하여 서비스가 가능한 상태에 비해 상대적으로 고장이 얼마나 자주 발생하는지를 나타낸다. 그림에서 확인할 수 있듯이 식 (14)를 통하여 얻은 근사해가 식 (13)의 결과보다 더 정확한 해를 제공함을 확인할 수 있다. 이와 같이, 제시된 방법은 근사해법의 정확도를 이론적으로 시험하는 데에 사용될 수 있다.



<Figure 6> Exact and Approximated Cycle Time for the G/M/1 and G/M/5 Failure Prone Queues

5. 결 론

G/G/m 대기행렬 모형에 관한 많은 연구결과들이 있지만 아직까지 체제시간에 관한 정확한 해를 구하는 방법은 제시되지 않았다. 그렇기 때문에, G/G/m 대기행렬을 분석할 때에 대부분의 경우 근사해법을 사용하게 된다. 근사해법들은 시스템에 관한 유용하고 직관적인 정보를 제공하지만 근사해법을 검증하는 것은 쉽지가 않기 때문에 대부분의 경우 시뮬레이션이나 하나의 서버로 이루어진 마코비안 대기행렬 모형과 같은 모형들을 이용해 왔다.

본 논문에서는 낮은 교통 밀도 하에서 서버고장을 고려한 복수 서버 대기행렬 시스템의 체제시간에 관한 분석을 수행하였다. 낮은 교통 밀도 하에서 고장 과정과 수리과정이 포아송 과정일 때, 흡수 마코프 체인과 재생이론을 이용하여 체제시간에 관한 정확한 해를 얻을 수 있다. 위 방식은 고객의 집단 도착을 가정한 마코비안 복수서버 시스템의 경우에도 적용될 수 있다. 이러한 방법론은 비병목 공정과 같이 낮은 교통밀도 하에서 운영되고 있는 시스템의 체제시간에 관하여 유용한 정보를 제공하고 있으며, 근사해법을 이론적으로 테스트 하는데 사용될 수도 있다. 또한 본 연구에서는 평균 체제시간에 대한 식을 제시하고 그에 대한 직관적인 해석을 제공하는데, 이는 시스템의 특성을 논리적으로 이해하는데 큰 도움이 될 것으로 생각한다.

Acknowledgement

This paper has been supported by 2015 Handong Global University Research Fund (Project number : 20150096). This study was supported by 2015 Research Grant from Kangwon National University.

References

[1] Avi-Itzhak, B., A Sequence of Service Stations with Arbitrary Input and Regular Service Times, *Management Science*, 1965, Vol. 11, No. 5, pp. 565-571.

[2] Avi-Itzhak, B. and Naor, P., Some Queueing Problems with the Service Station Subject to Breakdown, *Operations Research*, 1963, Vol. 11, No. 3, pp. 303-320.

[3] Gaver, D.P., A Waiting Line with Interrupted Service, including Priorities, *Journal of the Royal Statistical Society : Series B*, 1962, Vol. 24, No. 1, pp. 73-90.

[4] Hopp, W.J. and Spearman, M.L., Factory Physics : Foundations of Manufacturing Management, 2nd ed., McGraw-Hill, New York, 2001.

[5] Kim, C.-O. and Kang, K.-S., A Single Server Queue Operating under N-Policy with a Renewal Break down Process, *Journal of the Society of Korea Industrial and Systems Engineering*, 1996, Vol. 19, No. 39, pp. 205-218.

[6] Kim, W.K., Design and Evaluation of Double Ended Queueing Model Extension by Simulation, *Journal of The Korean Institute of Plant Engineering*, 2012, Vol. 17, No. 3, pp. 13-23.

[7] Kim, W.-S. and Morrison, J.R., On Cycle Time Ap-

proximations for the Failure Prone G/G/m Queue : Theoretical Justification of a Practical Approximation, *Proceedings of the 2011 International Conference on Control, Automation and System(ICCAS)*, 2011, pp. 1558-1563.

[8] Kim, W.-S. and Morrison, J.R., On Equilibrium Probabilities for the Delays in Deterministic Flow Lines with Random Arrivals, *IEEE Transactions on Automation Science and Engineering (IEEE), a special issue on selected papers from IEEE CASE 2013*, 2015, Vol. 12, No. 1, pp. 62-74.

[9] Kim, W.-S. and Morrison, J.R., The Throughput Rate of Serial Production Lines with Deterministic Process Times and Random Setups : Markovian Models and Applications to Semiconductor Manufacturing, *Computers and Operations Research*, 2015, Vol. 53, No. 1, pp. 288-300.

[10] Mitrany, I.L. and Avi-Itzhak, B., A Many-Server Queue with Service Interruptions, *Operations Research*, 1968, Vol. 16, No. 3, pp. 628-638.

[11] Morrison, J.R., Deterministic Flow Lines with Applications, *IEEE Transactions on Automation Science and Engineering*, 2010, Vol. 7, No. 2, pp. 228-239.

[12] Morrison, J.R. and Martin, D.P., Practical Extensions to Cycle Time Approximations for the G/G/m-queue with Applications, *IEEE Transactions on Automation Science and Engineering*, 2007, Vol. 4, No. 4, pp. 523-532.

[13] White, H. and Christie, L., Queueing with Preemptive Priorities or with Breakdown. *Operations Research*, 1958, Vol. 6, pp. 79-95.

[14] Whitt, W., Approximations for the GI/G/m Queue, *Production and Operations Management*, 1993, Vol. 2, No. 2, pp. 114-161.

ORCID

Woo-Sung Kim | http://orcid.org/0000-0001-9444-2712
Dae-Eun Lim | http://orcid.org/0000-0002-6591-4968