

## Poisson 수요과정을 갖는 두 품목 콘웍시스템의 성능평가

박찬우<sup>1†</sup> · 김수민<sup>2</sup> · 방우형<sup>3</sup> · 이효성<sup>2</sup>

<sup>1</sup>한국철도기술연구원 시험인증안전센터 / <sup>2</sup>경희대학교 산업경영공학과 / <sup>3</sup>LIG 넥스원(주) 연구개발본부

### Performance Evaluation of a Two-Product CONWIP System with Poisson Demand Processes

Chan-Woo Park<sup>1</sup> · Su-min Kim<sup>2</sup> · Woo-hyung Bang<sup>3</sup> · Hyo-Seong Lee<sup>2</sup>

<sup>1</sup>Testing and Certification Center, Korea Railroad Research Institute

<sup>2</sup>Department of Industrial and Management Engineering, Kyung Hee University

<sup>3</sup>R&D Center, LIG Nex1 Co., Ltd.

In this study we consider a flow line CONWIP system in which two types of product are produced. The processing times of each product type at each station follow an independent exponential distribution and the demands for the finished products of each type arrive according to a Poisson process. The demands that are not satisfied instantaneously are either backordered or lost according to the number of unsatisfied demands that exist at their arrival instants. For this system we develop an approximation method to obtain the performance measures such as steady state probabilities of the number of parts of each product type at each station, mean waiting times of backordered demands and the proportion of backordered demands. For the analysis of the proposed CONWIP system, we model the CONWIP system as a two class closed queueing network with a synchronization station and analyze the closed queueing network using a product-form approximation method for multiple classes developed by Baynat and Dallery. In the approximation method, each subsystem is analyzed using a matrix geometric method. Comparisons with simulation show that the approximation method provides fairly good results for all performance measures.

**Keywords:** CONWIP System, Multi-Class Closed Queueing Network, Product-Form Approximation Method, Matrix Geometric Method, Performance Evaluation

#### 1. 서론

콘웍(CONstant Work In Process, CONWIP)정책은 밀어내기식 제어정책(push control policy)과 끌어당기기식 제어정책(pull control policy)을 결합한 혼합제어정책(hybrid control policy)으로서 지난 20여 년 동안 많은 관심을 받아 왔다. 콘웍정책하에서는 수요가 발생하여 완제품 재고가 인출되면 즉시 원자재 투입지시가 이루어지므로 생산시스템 내의 재공품(완제품 포함)의 수가 일정하게 유지된다. 수요발생에 의하여 생산지시가 이루어지므로 이러한 측면에서 보면 콘웍정책은 끌어당기

기식 제어정책의 특성을 지니고 있다. 그러나 원자재가 첫 번째 작업장에 투입되면 마지막 작업장에서의 공정을 마친 후 완제품으로 창고에 보관될 때까지 특별한 제어 없이 밀어내기 방식에 의해 가공이 이루어진다. 따라서 이러한 측면에서 보면 콘웍정책은 밀어내기식 제어정책의 특성을 지니고 있다.

콘웍정책에 대해서는 그동안 많은 연구가 수행되어 왔다. Spearman(1992)은 stochastic coupling 방식을 사용하여 콘웍정책과 간판정책의 성능을 비교하였다. 그 결과 고객 만족도(재고고갈 확률, 재고부족으로 대기 중인 고객의 대기시간 등) 측면에서 콘웍정책이 간판정책에 비해 우수함을 증명하였다. 고

† 연락저자 : 박찬우, 437-757 경기도 의왕시 철도박물관로 176 한국철도기술연구원, Tel : 031-460-5545, Fax : 031-460-5579, E-mail : cwpark@krii.re.kr

2012년 12월 6일 접수; 2013년 1월 25일 수정본 접수; 2013년 1월 28일 게재 확정.

객만족도 측면 외에도 콘워프정책은 제어방식이 용이하고, 각 공정에서 생산준비 시간이 비교적 긴 경우에도 적용될 수 있다는 점 등 부수적인 장점이 있는 것으로 연구되었다(Spearman and Zazanis, 1992). 콘워프정책을 다양한 전통적 생산제어정책과 비교하기 위한 연구도 활발하게 수행되었으며 대부분의 연구 결과에서 콘워프정책은 전통적인 생산제어정책에 비하여 우수한 것으로 보고되고 있다(Bonvik *et al.*, 1993; Buzacott and Shantikumar, 1993).

콘워프정책은 시스템 내에 존재하는 재공품의 수가 일정하게 유지되므로 폐쇄형 대기네트워크(closed queueing network)로 모델링하여 분석할 수 있다. 그러나 콘워프정책을 폐쇄형 대기네트워크로 모델링하면 승법형 대기네트워크(product-form queueing network)의 조건이 충족되지 못하므로 콘워프정책의 분석을 위해서는 근사적기법(approximation method)을 활용할 수밖에 없다. 폐쇄형 대기네트워크의 근사적기법으로 다양한 기법이 연구되었으나, 일반적으로 Marie(1979)에 의해 개발된 승법형 근사법이 가장 우수한 기법으로 평가된다(Baynat and Dallery, 1993). Marie의 근사법이 제안된 이후 이를 이용하여 콘워프시스템을 분석하기 위한 다양한 연구가 수행되었다. Mascolo *et al.*(1996)는 완제품에 대한 수요가 Poisson 과정에 따라 발생하는 콘워프시스템의 성능분석 방법을 연구하였다. 이 연구에서는 각 작업장에서의 가공시간이 2단계 Coxian 분포를 따른다는 가정하에서 시스템의 안정상태 확률을 구하기 위한 근사적기법이 제시되었다. Lee and Lee(1998)는 Marie의 기법을 이용하여 완제품에 대한 수요가 복합 포아송과정에 따라 발생하는 콘워프시스템을 분석하였고, Park and Lee(2006)는 이 연구를 확장하여 완제품에 대한 수요가 복합 포아송과정을 따르고 각 작업장의 가공시간이 Coxian-2 분포를 따를 경우의 콘워프시스템을 근사적으로 분석하기 위한 기법을 개발하였다.

위에서 언급한 연구는 모두 한 종류의 제품이 생산되는 콘워프시스템을 분석한 연구이다. 그러나 최근에는 혼류생산이 가능하도록 생산라인을 설계하는 추세이므로 다품목 생산이 가능한 콘워프시스템에 대한 연구가 매우 중요하다. 다품목 생산이 가능한 콘워프시스템은 다계층 폐쇄형 대기네트워크(multi-class closed queueing network)로 모델링할 수 있으므로 다품목 콘워프시스템을 분석하기 위해서는 다계층 폐쇄형 대기네트워크의 근사적기법을 활용해야 한다. 다계층 폐쇄형 대기네트워크의 근사적기법으로는 Marie의 기법을 다계층 폐쇄형 대기네트워크로 확장한 Baynat and Dallery(1996)의 승법형근사법(product-form approximation method), Whitt(1994)의 기법을 폐쇄형 대기네트워크로 확장한 Satyam and Krishnamurthy(2008)의 모수분해법(parametric decomposition method), 그리고 평균치분석법(mean value analysis)에 기초한 근사적기법을 들 수 있다. 이들 연구 중에서는 Baynat과 Dallery의 승법형근사법이 다양한 환경에서 가장 좋은 결과를 제공해 주는 우수한 기법으로 평가된다. 그러나 Baynat과 Dallery의 기법은 하부네트워크(sub-network)의 분석이 불가능한 경우가 많고 특히 계층의 수가 세 개 이상일 경우에는 분석과정이 크게 복잡해지고 해의 정확도도

떨어지는 것이 문제로 대두된다.

본 연구에서는 Baynat과 Dallery의 기법을 이용하여 두 종류의 제품이 생산되는 콘워프시스템의 성능척도를 구하기 위한 근사적기법을 개발하고자 한다. 하부네트워크의 분석을 위해서 행렬기하법(matrix geometric method)을 활용하였으며 재고가 존재하지 않을 때 도착하는 수요에 대해서는 대기수요로 남는 경우와 유실되는 경우가 모두 고려될 수 있도록 가정을 일반화하였다.

## 2. 모형 설명

본 연구에서는 두 종류의 제품(제품 1, 제품 2)이 생산되는 콘워프시스템을 분석한다. 본 연구에서 분석하고자 하는 콘워프시스템은 흐름식 생산라인 시스템으로  $M$ 개의 작업장과 완제품 재고창고로 구성되며 각 작업장에는 1대의 기계가 존재한다. 작업장  $i$ 에서의 제품 1의 가공시간은 평균  $1/\mu_{1,i}$ 인 지수분포를 따르고 제품 2의 가공시간은 평균  $1/\mu_{2,i}$ 인 지수분포를 따른다. 작업장  $i$ 에서 가공을 기다리는 반제품이 두 개 이상일 경우에는 제품의 종류와 관계없이 선입선출(FIFO) 규칙에 따라 다음 가공물이 결정된다. 시스템 내에는  $N_1$ 개의 간판 1과  $N_2$ 개의 간판 2가 존재하며, 시스템 내에 존재하는 모든 제품(반제품 및 완제품)은 간판이 부착된 채로 이동, 대기, 가공, 그리고 저장 과정을 거치기 때문에 시스템 내에 존재하는 제품 1의 수는 항상  $N_1$ 개, 제품 2의 수는 항상  $N_2$ 개가 된다. 제품 1에 대한 수요는 수요율  $\lambda_1$ 의 포아송과정에 따라 발생하고 제품 2에 대한 수요는 수요율  $\lambda_2$ 의 포아송과정에 따라 발생하며, 두 제품의 수요과정은 서로 독립이다.

외부로부터 완제품  $r$ 에 대한 수요가 발생할 때 완제품창고에 제품  $r$ 의 재고가 존재하면 수요는 즉시 충족되나 제품  $r$ 의 재고가 존재하지 않으면 수요는 대기하거나 유실된다( $r = 1, 2$ ). 즉, 제품  $r$ 의 수요가 도착한 시점에 대기 중인 제품  $r$ 의 수요가  $B_r$  미만이면 새로 도착한 수요는 미충족수요로 남아 대기하게 되며,  $B_r$  이상이면 새로 도착한 수요는 유실되는 것으로 가정한다. 따라서  $B_r$ 이 무한대이면 추후납품의 경우(backordered)로 귀착되며,  $B_r$ 이 0이면 수요가 유실되는 경우(lost sales)로 귀착된다. 완제품과 수요가 결합되어 수요가 충족되면 완제품과 수요는 즉시 시스템을 빠져 나가고 완제품에 부착되어 있던 간판은 떨어져 작업장 1 앞에 대기 중인 제품  $r$ 의 원자재에 부착된다. 간판이 부착된 원자재는 즉시 생산라인에 투입되고 새로이 생산라인에 투입된 원자재는 마지막 가공이 완료되어 완제품창고에 저장될 때까지 밀어내기 방식으로 가공이 이루어진다. 이 외에도 본 연구에서는 다음의 사항을 가정한다.

- 작업장 1 앞에는 항상 충분한 양의 원자재가 쌓여 있다.
- 각 작업장의 대기공간은 충분히 크므로 봉쇄(blocking)가 발생하지 않는다.

- 부품의 작업장 간 이동시간은 무시할 수 있을 정도로 짧다.
- 각 작업장의 작업은 다른 작업장과 독립적으로 수행된다 (asynchronous system).

본 연구에서 분석하고자 하는 콘웍시스템을 그림으로 표현하면 <Figure 1>과 같다. 간판을 고객이라 간주할 때 시스템 내에 두 종류의 고객이 존재하고 두 종류의 고객 수는 항상  $N_1$ ,  $N_2$ 로 일정하므로 본 시스템은 2-계층 폐쇄형 대기네트워크 (two-class closed queueing network)로 모델링될 수 있다. 완제품창고에서는 완제품과 수요가 결합되어야만 완제품이 시스템을 이탈할 수 있으므로 완제품창고는 동기화 스테이션이라 할 수 있다. 서비스시간이 지수분포를 따르지만 고객의 계층에 따라 서비스율이 상이하고 시스템 내에 동기화 스테이션이 존재하므로 본 시스템은 Basket *et al.*(1975)에 의해 제안된 BCMP-네트워크의 요건(Buzacott and Shantikumar, 1993)을 충족하지 않는다(여기서 “BCMP”라는 명칭은 Basket *et al.*(1975)의 저자들의 이름의 두문자에서 비롯된 것임). 따라서 작업장의 수와 간판의 수가 극히 적은 경우를 제외하고는 정확한 해를 구하기 힘들며 성능척도를 구하기 위해서는 근사적기법 또는 시뮬레이션에 의존할 수밖에 없다. 그러나 시뮬레이션을 이용하여 요구되는 정확도의 해를 구하기 위해서는 많은 시간과 노력을 투자해야 하므로 적은 노력으로 실제 해에 근접한 해를 제공해 줄 수 있는 분석적기법의 개발이 필요하다. 본 연구에서는 두 품목 콘웍시스템의 성능척도를 구하기 위해 Baynat and Dallery(1996)가 다계층 폐쇄형 대기네트워크의 근사해를 구하기 위하여 개발한 승법형근사법(product-form approximation method)을 이용하고자 한다.

### 3. 승법형근사법을 이용한 분석

#### 3.1 승법형근사법의 개요

Baynat과 Dallery의 기법은 Marie(1979)의 기법을 다계층 폐쇄형 대기네트워크로 확장한 기법으로 승법형요건이 충족되지 않는 다계층 폐쇄형 대기네트워크를 근사적으로 분석하기 위해 개발된 기법이다. Baynat과 Dallery의 기법에서는 각 계층의 행태를 지수분포 서비스 시간과 상태중속 서비스율을 갖는 단일계층 승법형 대기네트워크로 근사화시켜 분석하며, 계층- $r$  고객의 행태를 대변하는 단일계층 승법형 대기네트워크를 계층- $r$  동등승법형 네트워크(class- $r$  equivalent product-form network)라 부른다. 따라서 계층이  $R$ 개면  $R$ 개의 동등승법형 네트워크가 만들어진다. 동등승법형 네트워크의 형태는 모든 계층에 대해 동일하며 서버의 상태중속 서비스율에서만 차이가 난다. Baynat과 Dallery의 기법은 2단계로 구성된다. 1단계는 원 네트워크를  $K$ 개의 하부네트워크로 분해한 후  $R$ 개의 동등승법형 네트워크를 만드는 단계이고, 2단계는 원 네트워크와 유사한 행태를 갖도록 동등승법형 네트워크를 구성하는 서버의 상태중속 서비스율을 결정하는 단계이다. 각각의 단계에 대하여 아래에서 간략히 설명한다.

#### (1) 동등승법형 네트워크 구축

동등승법형 네트워크를 만들기 위해서는 먼저 주어진 대기네트워크를 몇 개의 하부네트워크로 분해한다. 계층- $r$  동등승법형 네트워크는 각각의 하부네트워크를 서비스 시간이 지수분포를 따르고 서비스율이 상태중속인 단일 서버로 치환함으로써 만들어진다. 따라서 원 네트워크를  $K$ 개의 하부네트워크

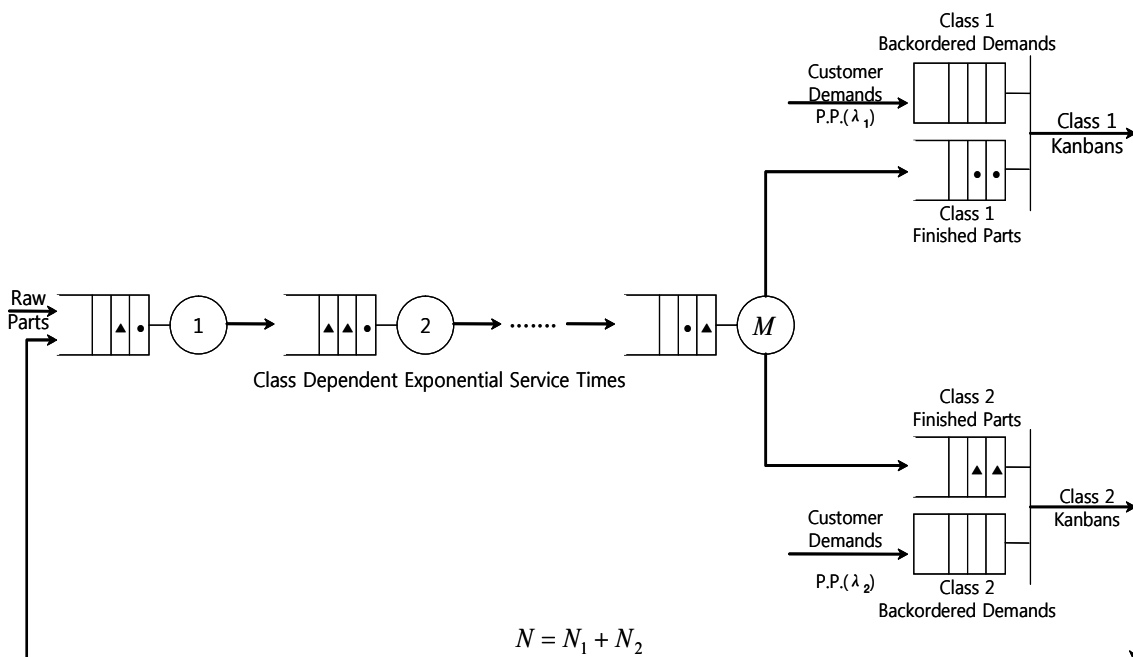


Figure 1. Two-product CONWIP system with poisson demand processes

로 분해하였다면 각 계층별 동등승법형 네트워크는  $K$ 개의 단일 서버로 구성되며 각 서버의 서비스 시간은 서비스율이 상태종속인 지수분포를 따른다.

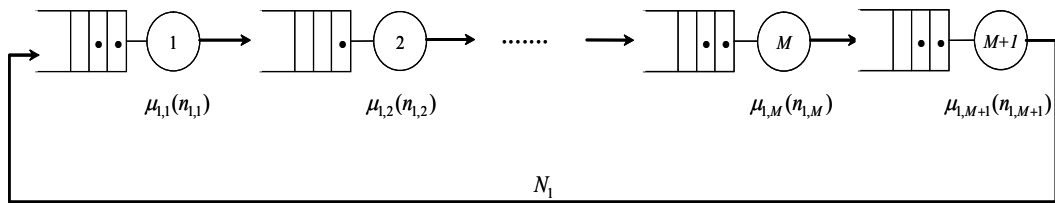
근사적분석이 만족할 만한 결과를 얻기 위해서는 원 네트워크를 하부네트워크로 분해할 때 네 가지 조건이 충족되어야 한다(Baynat and Dallery, 1996). 승법형근사법을 적용하는 과정에서 가장 어려운 부분은 하부네트워크의 분석이다. 일반적으로 하부네트워크는 형태가 복잡할수록 분석이 어려워지므로 분석을 용이하게 하기 위해서는 하부네트워크의 형태가 가능한 단순해지도록 분해하여야 한다. 하부네트워크의 형태를 가장 단순하게 만들기 위해서는 원 네트워크를 구성하는 각각의 스테이션이 별도의 하부네트워크를 이루도록 분해하면 된다. 이와 같이 분해하여도 Baynat and Dallery(1996)에서 규정한 네 가지 조건이 충족된다면 이러한 방법이 가장 이상적인 하부네트워크 분해방법이라 할 수 있다. 본 연구에서는 각각의 스테이션이 별도의 하부네트워크를 이루도록 대기네트워크를 분해하여도 네 가지 조건이 충족되므로 모든 동등승법형 네트워크는  $M+1$ 개의 단일 서버로 구성된다. 본 연구에서는 계층이 2개 이므로 동등승법형 네트워크는 <Figure 2>에서와 같이 2개가 존재한다. 본 연구에서는  $i$ 번째 하부네트워크를  $R^i$ 라 표현하고, 계층- $r$  동등승법형 네트워크에서  $R^i$ 에 대응하는 서버를  $S^{r,i}$ 라 표현하기로 한다.

원 네트워크에서는 계층 간 상호 영향을 미치는 두 개의 계층이 동일 네트워크 내에 존재하나 승법형근사법에서는 각 계층을 분리하여 각 계층의 행태를 별도의 단일계층 승법형 대기네트워크로 단순화시킨다. 단순화 과정으로 인해 동등승법형 네트워크가 원 네트워크 내에 존재하는 각 계층의 행태를 제대로 표현할 수 있을지에 대해 의문이 제기될 수 있다. 그러나 다른 계층과의 상호 영향이 모두 고려되도록 동등승법형 네트워크

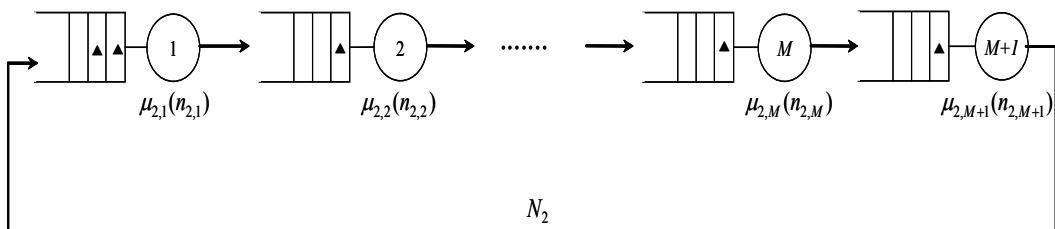
의 서비스율을 결정하기 때문에 단일계층 네트워크로도 각 계층의 행태를 비교적 정확하게 표현할 수 있다. 동등승법형 네트워크의 서비스율이 모두 결정되면 동등승법형 네트워크를 분석함으로써 원 네트워크의 성능 평가치를 구할 수 있다.

(2) 동등승법형 네트워크의 서비스율 결정

하부네트워크로의 분해가 끝나고 동등승법형 네트워크의 형태가 결정되면 동등승법형 네트워크를 구성하는 서버의 서비스율을 결정해야 한다. 동등승법형 네트워크의 서비스율을 구하기 위해서는 각각의 하부네트워크를 고객의 도착과정이 계층별로 독립이고 계층별 도착률이 하부네트워크 내 해당 계층의 고객 수에 종속하며, 도착간격이 지수분포를 따르는 개방형 대기네트워크로 간주하여 분석한다. 우선  $i$ 번째 하부네트워크  $R^i$ 에 계층- $r$  고객이  $n_{r,i}$ 명 있을 때 계층- $r$  고객의 상태종속 도착률,  $\lambda_{r,i}(n_{r,i})$ 을 알고 있다고 가정하자. 그러면 임의의 대기행렬 분석기법을 이용하여 하부네트워크  $R^i$ 에 존재하는 계층- $r$  고객의 안정상태확률을 구할 수 있고, 안정상태확률을 통하여  $R^i$ 에 계층- $r$  고객이  $n_{r,i}$ 명 있을 때 단위 시간당  $R^i$ 를 이탈하는 계층- $r$  고객의 평균수를 구할 수 있다. 이를 상태종속 산출률(state dependent throughput)이라 부르며  $v_{r,i}(n_{r,i})$ 로 표현하기로 한다. 상태종속 산출률  $v_{r,i}(n_{r,i})$ 는 계층- $r$  동등승법형 네트워크의 서버  $S^i$ 에 존재하는 고객의 수가  $n_{r,i}$ 일 경우의  $S^i$ 의 상태종속 서비스율  $\mu_{r,i}(n_{r,i})$ 로 치환하여 사용될 수 있다. 계층- $r$  동등승법형 네트워크를 구성하는 모든 서버의 상태종속 서비스율을 구하게 되면 승법형 대기네트워크를 분석할 수 있는 임의의 알고리즘을 사용하여 계층- $r$  동등승법형 네트워크를 분석할 수 있고, 분석 결과 서버  $S^i$ 에 존재하는 고객의 수가  $n_{r,i}$ 일 경우의 산출률(state dependent throughput)  $X_{r,i}(n_{r,i})$ 을 구할 수 있다. 서버  $S^i$ 의 상태종속 산출률  $X_{r,i}(n_{r,i})$



(a) class-1 equivalent product-form network



(b) class-2 equivalent product-form network

Figure 2. Equivalent product-form networks

은 다시 하부네트워크  $R^i$ 의 상태종속 도착률  $\lambda_{r,i}(n_{r,i})$ 로 치환하여 사용될 수 있다.

요약해 보자면 동등승법형 네트워크의 상태종속 서비스율  $\mu_{r,i}(n_{r,i})$ 이 구해지면 이로부터 하부네트워크의 상태종속 도착률  $\lambda_{r,i}(n_{r,i})$ 을 구할 수 있고,  $\lambda_{r,i}(n_{r,i})$ 이 구해지면 이로부터 승법형대기네트워크의 상태종속 서비스율,  $\mu_{r,i}(n_{r,i})$ 을 구할 수 있게 된다. 그러나 동등승법형 네트워크의 상태종속 서비스율과 하부네트워크의 상태종속 도착률은 모두 미지의 값이므로 이를 구하는 문제는 고정점 문제(fixed point problem)가 되며, 이를 구하기 위해서는 반복적 기법을 사용해야만 한다. 하부네트워크 및 동등승법형 네트워크의 상세한 분석방법은 제 3.2절에서 설명한다.

### 3.2 하부네트워크 분석

#### (1) $R^i$ 분석( $i = 1, \dots, M$ )

앞서 설명한 바와 같이 하부네트워크  $R^i$ ( $i = 1, \dots, M$ )는 고객의 도착과정이 계층별로 독립이고 계층별 도착률은 하부네트워크 내에 존재하는 해당 계층의 고객 수에 의해 결정되며, 도착간격이 지수분포를 따르는 개방형 대기네트워크로 간주하여 분석한다(Figure 3 참조). 하부네트워크  $R^i$ 에서 서버는 1명이고 계층-1의 서비스시간은 평균  $1/\mu_{1,i}$ 인 지수분포를 따르며 계층-2의 서비스 시간은 평균  $1/\mu_{2,i}$ 인 지수분포를 따른다. 만일 하부네트워크 내에 존재하는 계층- $r$ 의 고객 수가  $n_{r,i}$ 일 때의 상태종속 도착률  $\lambda_{r,i}(n_{r,i})$ 을 알고 있다면 하부네트워크는 다음과 같이 분석할 수 있다( $r = 1, 2$ ).

각 계층의 도착간격과 서비스 시간이 모두 지수분포를 따르므로 하부네트워크의 고객 수 확률은 연속시간형 마코브체인

으로 분석이 가능하다. 그러나 선입선출 서비스원칙하에서 하부네트워크  $R^i$ 를 정확히 분석하기 위해서는 대기상태에 있는 각각의 고객이 어떤 계층에 속하는지에 대한 정보가 필요하다. 따라서  $N_1, N_2$ 의 값이 커지면 분석이 현실적으로 불가능해진다. 이와 같은 문제를 해결하기 위해서 본 연구에서는 선입선출원칙을 무작위 서비스원칙(random service rule)으로 대체하여 분석하고자 한다. 무작위 서비스원칙은 대기 중인 고객을 무작위로 선정하여 다음 서비스를 제공받을 고객을 결정하는 서비스원칙이다. 무작위서비스원칙은 선입선출원칙에 비해 분석이 간략히 이루어질 수 있으며, 선입선출원칙을 적용할 경우와 유사한 결과가 산출되는 것으로 보고되어 있어 선입선출 원칙 대신 유용하게 사용될 수 있다(Baynat and Dalry, 1996).

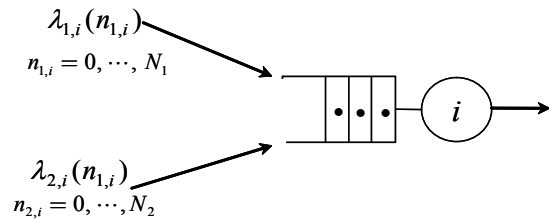


Figure 3. Sub-network  $R^i, i = 1, \dots, M$

무작위 서비스원칙 하에서는 대기 중인 고객의 순서를 고려하지 않아도 되므로 시스템의 상태를 다음과 같이 간략히 정의할 수 있다.  $c$ 를 하부네트워크  $R^i$ 에서 서비스 받고 있는 고객의 종류,  $m_1$ 을 하부네트워크  $R^i$ 에 존재하는 계층-1의 대기 고객 수,  $m_2$ 를 하부네트워크  $R^i$ 에 존재하는 계층-2의 대기 고객 수라 하면 하부네트워크  $R^i$ 의 상태 공간  $ST_i$ 은 다음과 같

Table 1. Transition Rate Matrix for  $N_1 = 2, N_2 = 2$

	(0, 0, 0)	(2, 0, 0)	(2, 0, 1)	(1, 0, 0)	(1, 0, 1)	(1, 0, 2)	(2, 1, 0)	(2, 1, 1)	(1, 1, 0)	(1, 1, 1)	(1, 1, 2)	(2, 2, 0)	(2, 2, 1)
(0, 0, 0)	$\eta_i(0, 0, 0)$	$\lambda_{2,i}(0)$		$\lambda_{1,i}(0)$									
(2, 0, 0)	$\mu_{2,i}$	$\eta_i(2, 0, 0)$	$\lambda_{2,i}(1)$				$\lambda_{1,i}(0)$						
(2, 0, 1)		$\mu_{2,i}$	$\eta_i(2, 0, 1)$					$\lambda_{1,i}(0)$					
(1, 0, 0)	$\mu_{1,i}$			$\eta_i(1, 0, 0)$	$\lambda_{2,i}(0)$				$\lambda_{1,i}(1)$				
(1, 0, 1)		$\mu_{1,i}$			$\eta_i(1, 0, 1)$	$\lambda_{2,i}(1)$				$\lambda_{1,i}(1)$			
(1, 0, 2)			$\mu_{1,i}$			$\eta_i(1, 0, 2)$					$\lambda_{1,i}(1)$		
(2, 1, 0)				$\mu_{2,i}$			$\eta_i(2, 1, 0)$	$\lambda_{2,i}(1)$				$\lambda_{1,i}(1)$	
(2, 1, 1)					$\frac{1}{2}\mu_{2,i}$		$\frac{1}{2}\mu_{2,i}$	$\eta_i(2, 1, 1)$					$\lambda_{1,i}(1)$
(1, 1, 0)				$\mu_{1,i}$					$\eta_i(1, 1, 0)$	$\lambda_{2,i}(0)$			
(1, 1, 1)					$\frac{1}{2}\mu_{1,i}$		$\frac{1}{2}\mu_{1,i}$			$\eta_i(1, 1, 1)$	$\lambda_{2,i}(1)$		
(1, 1, 2)						$\frac{1}{3}\mu_{1,i}$		$\frac{2}{3}\mu_{1,i}$			$\eta_i(1, 1, 2)$		
(2, 2, 0)									$\mu_{2,i}$			$\eta_i(2, 2, 0)$	$\lambda_{2,i}(1)$
(2, 2, 1)										$\frac{2}{3}\mu_{2,i}$		$\frac{1}{3}\mu_{2,i}$	$\eta_i(2, 2, 1)$

이 표현된다.

$$ST_i = \{(c, m_1, m_2) | c=0, 1, 2, 0 \leq m_r \leq N_r \cdot \delta(c \neq 0) - \delta(c=r), r=1, 2\} \quad (1)$$

여기서,  $\delta(x)$ 는 조건  $x$ 가 참이면 1, 아니면 0을 가지는 조건부 함수를 나타내며, 식 (1)에서 만약  $c=0$ 이면 하부네트워크  $R^i$ 에 고객이 없음을 의미한다.

만일 상태 공간  $ST_i$ 의 모든 상태를 하부네트워크  $R^i$ 에 존재하는 계층-1 고객 수를 기준으로 사전식 배치(lexicographic order)를 하면 제품별 간판 수  $N_1$ 과  $N_2$ 가 각각 2일 경우의 상태 전이율 행렬(transition rate matrix)은 <Table 1>과 같다. <Table 1>에서  $\eta_i(c, m_1, m_2)$ 는 다음과 같이 표현된다.

$$\eta_i(c, m_1, m_2) = -[\lambda_{1,i}(m_1 + \delta(c=1)) + \lambda_{2,i}(m_2 + \delta(c=2)) + \mu_{1,i} \cdot \delta(c=1) + \mu_{2,i} \cdot \delta(c=2)]$$

$N_1$ 과  $N_2$ 가 일반적인 값을 취할 경우 모든 상태를 계층-1 고객 수를 기준으로 사전식 배치를 하면 상태전이율 행렬은 <Figure 4>와 같이 6종류의 소행렬로 블록(block)화될 수 있다.

$$\mathbf{Q}_i = \begin{bmatrix} \hat{\mathbf{T}}_i & \hat{\mathbf{A}}_i & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \hat{\mathbf{S}}_i & \mathbf{T}_i(1) & \mathbf{A}_i(1) & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_i(2) & \mathbf{T}_i(2) & \mathbf{A}_i(2) & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \cdots & \mathbf{S}_i(N_1-1) & \mathbf{T}_i(N_1-1) & \mathbf{A}_i(N_1-1) \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{S}_i(N_1) & \mathbf{T}_i(N_1) \end{bmatrix}$$

Figure 4. Transition rate matrix  $\mathbf{Q}_i$

전이율 행렬  $\mathbf{Q}_i$ 를 구성하는 소행렬의 성분은 다음과 같이 표시할 수 있다.

$$\hat{\mathbf{A}}_i = \{\hat{a}_{j,k}^i\}, j=1, \dots, N_2+1, k=1, \dots, 2N_2+1$$

$$\hat{a}_{j,k}^i = \begin{cases} \lambda_{1,i}(0), & \text{if } (j=k=1) \text{ or } (j \geq 2, k=N_2+j), \\ 0, & \text{o/w.} \end{cases} \quad (2)$$

$$\mathbf{A}_i(n) = \{a_{j,k}^i(n)\}, n=1, \dots, N_1-1,$$

$$j=1, \dots, 2N_2+1, k=1, \dots, 2N_2+1,$$

$$a_{j,k}^i(n) = \begin{cases} \lambda_{1,i}(n), & \text{if } k=j, \\ 0, & \text{o/w.} \end{cases} \quad (3)$$

$$\hat{\mathbf{T}}_i = \{\hat{t}_{j,k}^i\}, j=1, \dots, N_2+1, k=1, \dots, N_2+1,$$

$$\hat{t}_{j,k}^i(n) = \begin{cases} \lambda_{2,i}(j-1), & \text{if } 1 \leq j \leq N_2, k=j+1, \\ \mu_{2,i}, & \text{if } 2 \leq j \leq N_2+1, k=j-1, \\ -\lambda_{1,i}(0) - \lambda_{2,i}(0), & \text{if } k=j=1, \\ -\lambda_{1,i}(0) - \lambda_{2,i}(j-1) - \mu_{2,i}, & \text{if } 2 \leq j \leq N_2+1, k=j. \\ 0, & \text{o/w.} \end{cases} \quad (4)$$

$$\mathbf{T}_i(n) = \{t_{j,k}^i(n)\}, n=1, \dots, N_1,$$

$$j=1, \dots, 2N_2+1, k=1, \dots, 2N_2+1,$$

$$t_{j,k}^i(n) = \begin{cases} \lambda_{2,i}(j-1), & \text{if } 1 \leq j \leq N_2, k=j+1, \\ \lambda_{2,i}(j-N_2-1), & \text{if } N_2+2 \leq j \leq 2N_2, k=j+1, \\ \frac{n}{n+j-N_2-2} \mu_{2,i}, & \text{if } N_2+2 \leq j \leq 2N_2+1, k=j-N_2-1, \\ \frac{j-N_2-2}{n+j-N_2-2} \mu_{2,i}, & \text{if } N_2+3 \leq j \leq 2N_2+1, k=j-1, \\ -\lambda_{1,i}(n) - \lambda_{2,i}(j-1) - \mu_{1,i}, & \text{if } 1 \leq j \leq N_2+1, k=j, \\ -\lambda_{1,i}(n) - \lambda_{2,i}(j-N_2-1) - \mu_{2,i}, & \text{if } N_2+2 \leq j \leq 2N_2+1, k=j, \\ 0, & \text{o/w.} \end{cases} \quad (5)$$

여기서,  $\lambda_{r,i}(N_r) = 0, r=1, 2$ .

$$\hat{\mathbf{S}}_i = \{\hat{s}_{j,k}^i\}, j=1, \dots, 2N_2+1, k=1, \dots, N_2+1$$

$$\hat{s}_{j,k}^i = \begin{cases} \mu_{1,i}, & \text{if } 1 \leq j \leq N_2+1, k=j, \\ 0, & \text{o/w.} \end{cases} \quad (6)$$

$$\mathbf{S}_i(n) = \{s_{j,k}^i(n)\}, n=2, \dots, N_1,$$

$$j=1, \dots, 2N_2+1, k=1, \dots, 2N_2+1,$$

$$s_{j,k}^i(n) = \begin{cases} \frac{n-1}{n+j-2} \mu_{1,i}, & \text{if } 1 \leq j \leq N_2+1, k=j, \\ \frac{j-1}{n+j-2} \mu_{1,i}, & \text{if } 2 \leq j \leq N_2+1, k=j+N_2, \\ 0, & \text{o/w.} \end{cases} \quad (7)$$

전이율 행렬  $\mathbf{Q}_i$ 는 삼각대각(tridiagonal) 형태를 취하고 있으므로 행렬기하법(matrix geometric method)을 적용하여 분석할 수 있다(Neuts, 1981). 하부네트워크  $R^i$ 의 상태  $(c, m_1, m_2)$ 의 안정상태확률을  $P_i(c, m_1, m_2)$ 라 정의했을 때, 소행렬로 표현된 균형방정식은 다음과 같다.

$$\mathbf{P}_i(0) \cdot \hat{\mathbf{T}}_i + \mathbf{P}_i(1) \cdot \hat{\mathbf{S}}_i = \mathbf{0} \quad (8)$$

$$\mathbf{P}_i(0) \cdot \hat{\mathbf{A}}_i + \mathbf{P}_i(1) \cdot \mathbf{T}_i(1) + \mathbf{P}_i(2) \cdot \mathbf{S}_i(2) = \mathbf{0} \quad (9)$$

$$\mathbf{P}_i(n-1) \cdot \mathbf{A}_i(n-1) + \mathbf{P}_i(n) \cdot \mathbf{T}_i(n) + \mathbf{P}_i(n+1) \cdot \mathbf{S}_i(n+1) = \mathbf{0}, n=2, 3, \dots, N_1-1. \quad (10)$$

$$\mathbf{P}_i(N_1-1) \cdot \mathbf{A}_i(N_1-1) + \mathbf{P}_i(N_1) \cdot \mathbf{T}_i(N_1) = \mathbf{0}. \quad (11)$$

여기서,  $\mathbf{P}_i(0) = \{P_i(0, 0, 0), P_i(2, 0, 0), P_i(2, 0, 1), \dots, P_i(2, 0, N_2-1)\}$ ,  $\mathbf{P}_i(n) = \{P_i(1, n-1, 0), P_i(1, n-1, 1), \dots, P_i(1, n-1, N_2)\}$ ,  $P_i(2, n, 0), P_i(2, n, 1), \dots, P_i(2, n, N_2-1)\}$ 을 나타낸다( $n=1, \dots, N_1$ ).

위의 균형방정식은 반복적인 구조를 가지므로  $\mathbf{P}_i(n)$ 은 함수  $\mathbf{R}_i(n)$ 을 이용해 다음과 같이 순환식으로 표현할 수 있다.

$$\mathbf{P}_i(n) = \mathbf{P}_i(n-1) \cdot \mathbf{R}_i(n), \quad n=1, \dots, N_1. \quad (12)$$

여기서  $\mathbf{R}_i(n)$ 은 다음과 같다.

$$\mathbf{R}_i(n) = \begin{cases} -\hat{\mathbf{A}}_i \cdot [\mathbf{T}_i(1) + \mathbf{R}_i(2) \cdot \mathbf{S}_i(2)]^{-1}, & \text{if } n=1, \\ -\mathbf{A}_i(n-1) \cdot [\mathbf{T}_i(n) + \mathbf{R}_i(n+1) \cdot \mathbf{S}_i(n+1)]^{-1}, & \text{if } n=2, \dots, N_1-1, \\ -\mathbf{A}_i(N_1-1) \cdot \mathbf{T}_i(N_1)^{-1}, & \text{if } n=N_1. \end{cases} \quad (13)$$

식 (13)에서  $\mathbf{R}_i(1)$ 은  $(N_2+1) \times (2N_2+1)$  행렬이고,  $2 \leq n \leq N_1$ 에 대하여  $\mathbf{R}_i(n)$ 은  $(2N_2+1) \times (2N_2+1)$  행렬이다.

따라서  $\mathbf{P}_i(0)$  값이 구해지면 식 (12)에 의해 순차적으로 다른 안정상태확률을 구할 수 있으며,  $\mathbf{P}_i(0)$  값은 식 (14)와 정규화 조건인 식 (15)를 연립하여 구할 수 있다.

$$\mathbf{P}_i(0) \cdot [\hat{\mathbf{T}}_i + \mathbf{R}_i(1) \cdot \hat{\mathbf{S}}_i] = \mathbf{0} \quad (14)$$

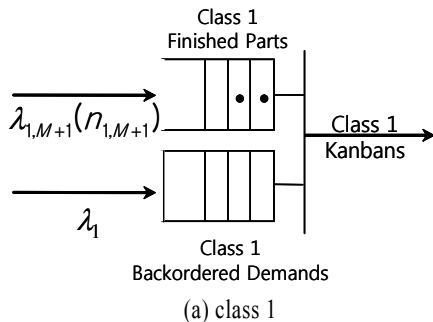
$$\mathbf{P}_i(0) \cdot [\mathbf{e}_i(0) + \sum_{n=1}^{N_1} \mathbf{R}\mathbf{P}_i(n) \cdot \mathbf{e}_i(n)] = \mathbf{1} \quad (15)$$

여기서,  $\mathbf{e}_i(0)$ 은 모든 원소가 1이며 크기가  $(N_2+1)$ 인 열벡터이고,  $\mathbf{e}_i(n)$ 은 모든 원소가 1이며 크기가  $(2N_2+1)$ 인 열벡터이며,  $\mathbf{R}\mathbf{P}_i(n)$ 은 크기  $(N_2+1) \times (2N_2+1)$ 의 행렬로 다음과 같다( $n=1, \dots, N_1$ ).

$$\mathbf{R}\mathbf{P}_i(n) = \prod_{i=1}^n \mathbf{R}(i), \quad n=1, \dots, N_1. \quad (16)$$

식 (12)에 의해  $P_i(c, m_1, m_2)$ 이 구해지면 이를 이용하여 하부네트워크  $R^i$ 에 존재하는 계층별 고객 수에 대한 안정상태 확률  $P_{r,i}(n_{r,i})$ 을 다음과 같이 구할 수 있다.

$$P_{1,i}(n_{1,i}) = \begin{cases} P_i(0, 0, 0) + \sum_{m_2=0}^{N_2-1} P_i(2, 0, m_2), & \text{for } n_{1,i} = 0, \\ \sum_{m_2=0}^{N_2-1} [P_i(1, n_{1,i}-1, m_2) + P_i(2, n_{1,i}, m_2)] + P_i(1, n_{1,i}-1, N_2), & \text{for } n_{1,i} = 1, \dots, N_1. \end{cases} \quad (17)$$



$$P_{2,i}(n_{2,i}) = \begin{cases} P_i(0, 0, 0) + \sum_{m_1=0}^{N_1-1} P_i(1, m_1, 0), & \text{for } n_{2,i} = 0, \\ \sum_{m_1=0}^{N_1-1} [P_i(1, m_1, n_{2,i}) + P_i(2, m_1, n_{2,i}-1)] + P_i(2, N_1, n_{2,i}-1), & \text{for } n_{2,i} = 1, \dots, N_2. \end{cases} \quad (18)$$

(2) 하부네트워크  $R^{M+1}$  분석

하부네트워크  $R^{M+1}$ 은 고객의 계층별 상태종속 도착률이  $\lambda_{r,M+1}(n_{r,M+1})$ 이고 도착간격이 지수분포를 따르는 개방형 대기네트워크로 간주하여 분석한다. 하부네트워크  $R^{M+1}$ 을 분석하기 위해서 상태종속 도착률  $\lambda_{r,M+1}(n_{r,M+1})$ 을 알고 있다고 가정하자. 하부네트워크  $R^{M+1}$ 에서는 각 계층의 고객 수가 상대 계층의 고객 수 및 도착과정, 서비스 과정에 의해 영향을 받지 않으므로 <Figure 5>에서와 같이 두 고객을 분리하여 독립적으로 분석할 수 있다. 제품  $r$ 의 수요는 평균 수요율  $\lambda_r$ 의 포아송 과정에 따라 발생하고, 최대 대기 수요는  $B_r$ 이므로 완제품창고의 완제품  $r$ 의 수를  $k_r$ (만약  $k_r < 0$ 이면,  $|k_r|$ 개의 미충족수요가 완제품을 기다리고 있음을 의미한다)로 나타냈을 때, <Figure 6>과 같이 출생사멸 과정으로 모형화할 수 있다 ( $r=1, 2$ ).

하부네트워크  $R^{M+1}$ 에 완제품이  $k_r$ 개 있을 확률을  $P_r(k_r)$ 이라 했을 때 <Figure 6>의 출생사멸 과정으로부터  $P_r(k_r)$ 은 다음과 같이 표현된다.

$$P_r(k_r) = \begin{cases} \frac{\lambda_{r,M+1}(k_r-1)}{\lambda_r} P_r(k_r-1), & \text{if } 1 \leq k_r \leq N_r, \\ \frac{\lambda_{r,M+1}(0)}{\lambda_r} P_r(k_r-1), & \text{if } -B_r+1 \leq k_r \leq 0, \\ 0, & \text{o/w} \end{cases} \quad (19)$$

식 (19)와 확률의 합이 1이 되어야 한다는 정규화식을 이용하면 계층별 안정상태확률을 구할 수 있으며, 하부네트워크  $R^{M+1}$ 의 계층별 고객 수에 대한 안정상태확률  $P_{r,M+1}(n_{r,M+1})$ 은 다음과 같이 구할 수 있다.

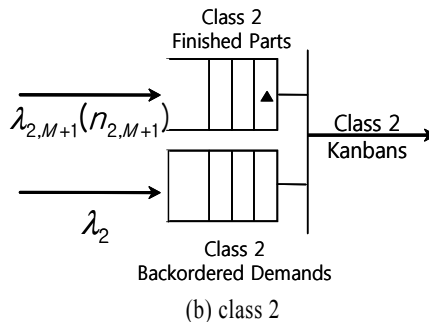


Figure 5. Sub-network  $R^{M+1}$

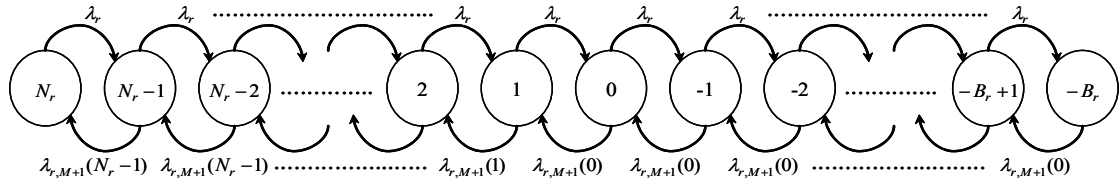


Figure 6. Birth and death process of sub-network  $R^{M+1}$  for class  $r$  ( $r=1, 2$ )

$$P_{r,M+1}(n_{r,M+1}) = \begin{cases} P_{r,M+1}(n_{r,M+1}), & \text{for } n_{r,M+1} > 0, \\ \sum_{k_r=-B_r+1}^0 P_r(k_r), & \text{for } n_{r,M+1} = 0. \end{cases} \quad (20)$$

### 3.3 승법형근사법을 이용한 알고리즘

하부네트워크  $R^i$ 의 안정상태 확률  $P_{r,i}(n_{r,i})$ 가 구해지면 하부네트워크  $R^i$ 에  $n_{r,i}$ 명의 계층  $r$  고객 이 존재할 경우의 상태종속 산출률  $v_{r,i}(n_{r,i})$ 의 값은 다음과 같이 얻을 수 있다 (Buzen, 1973).

$$v_{r,i}(n_{r,i}) = \begin{cases} 0, & \text{for } n_{r,i} = 0, \\ \lambda_{r,i}(n_{r,i}-1) \frac{P_{r,i}(n_{r,i}-1)}{P_{r,i}(n_{r,i})}, & \text{for } n_{r,i} = 1, 2, \dots, N_r. \end{cases} \quad (21)$$

하부네트워크  $R^i$ 의 상태종속 산출률  $v_{r,i}(n_{r,i})$ 의 값이 구해지면 계층- $r$  동등승법형 대기네트워크의 서버  $S^{r,i}$ 의 상태종속 서비스율  $\mu_{r,i}(n_{r,i})$ 은 다음과 같이 구해진다.

$$\mu_{r,i}(n_{r,i}) = v_{r,i}(n_{r,i}), \quad n_{r,i} = 1, \dots, N_r \quad (22)$$

따라서 하부네트워크  $R^i$ 의 상태종속 도착률  $\lambda_{r,i}(n_{r,i})$  값이 주어지면 하부네트워크  $R^i$ 의 안정상태확률을 구할 수 있고 식 (21)와 식 (22)를 이용하여 계층- $r$  동등승법형 네트워크의 상태종속 서비스율  $\mu_{r,i}(n_{r,i})$ 을 구할 수 있다. 앞서 설명한 하부네트워크  $R^i$ 의 분석과정에서 하부네트워크의 상태종속 도착률  $\lambda_{r,i}(n_{r,i})$ 을 알고 있다고 가정하였다. 그러나  $\lambda_{r,i}(n_{r,i})$ 의 값은 구해야 하는 미지의 값이다. 만일 동등승법형 네트워크의 상태종속 서비스율  $\mu_{r,i}(n_{r,i})$ 의 값이 주어지면  $\lambda_{r,i}(n_{r,i})$ 의 값은 다음과 같이 구하여 질 수 있다. 동등승법형 네트워크의 서비스율  $\mu_{r,i}(n_{r,i})$ 의 값을 알고 있다면 승법형 네트워크의 분석에 사용 가능한 임의의 알고리즘을 이용하여 서버  $S^{r,i}$ 의 상태종속 산출률  $X_{r,i}(n_{r,i})$ 을 구할 수 있다. 중첩(convolution) 알고리즘을 이용하면 서버  $S^{r,i}$ 의 상태종속 산출률은 다음 식에 의해 구할 수 있다.

$$X_{r,i}(n_{r,i}) = \begin{cases} e^{r_i} \frac{G_{M+1/i}(N_r - n_{r,i} - 1)}{G_{M+1/i}(N_r - n_{r,i})}, & \text{for } n_{r,i} = 0, \dots, N_r - 1, \\ 0, & \text{for } n_{r,i} = N_r. \end{cases} \quad (23)$$

여기서  $e_{r,i}$ 는 서버  $S^{r,i}$ 의 상대적 방문율이고,  $G_{M+1/i}(n)$ 는  $M+1$ 개의 스테이션 중 스테이션  $i$ 를 제외하고 계산한 정규화 상수다.

동등승법형 대기네트워크를 구성하는 서버  $S^{r,i}$ 의 상태종속 산출률이 구해지면 하부네트워크  $R^i$ 의 상태종속 도착률  $\lambda_{r,i}(n_{r,i})$ 은 다음과 같이 구해진다.

$$\lambda_{r,i}(n_{r,i}) = X_{r,i}(n_{r,i}), \quad n_{r,i} = 0, \dots, N_r \quad (24)$$

앞에서 설명한 바와 같이 하부네트워크의 안정상태확률이 구해지면, 하부네트워크의 상태종속 산출률을 계산할 수 있고, 이 값은 곧 동등승법형 네트워크의 상태종속 서비스율로 사용된다. 또한 동등승법형 네트워크의 상태종속 서비스율이 구해지면 이로부터 동등승법형 대기네트워크의 상태종속 산출률이 계산될 수 있고, 이 값은 곧 하부네트워크의 상태종속 도착률로 사용될 수 있다. 승법형근사법은 동등승법형 네트워크의 상태종속 서비스율  $\mu_{r,i}(n_{r,i})$ 에 대한 초기 값을 부여하는 것으로 시작한다. 초기 값이 부여되면 이를 이용하여 하부네트워크의 상태종속 산출률  $\lambda_{r,i}(n_{r,i})$ 를 계산하고,  $\lambda_{r,i}(n_{r,i})$ 가 구해지면 이를 이용하여 다시  $\mu_{r,i}(n_{r,i})$  값을 구한다. 승법형근사법에서는 이와 같은 과정을  $\mu_{r,i}(n_{r,i})$  값이 수렴될 때까지 반복 수행하며,  $\mu_{r,i}(n_{r,i})$  값이 수렴되면 필요한 성능척도 값들을 계산한다.

#### • 알고리즘

**Step 0 :**  $\mu_{r,i}(n_{r,i})$ 의 초기화

$$(1) \mu_{r,i}(n_{r,i}) = \mu_{r,i}, \quad n_{r,i} = 1, \dots, N_r, \quad i = 1, \dots, M+1, \quad r = 1, 2.$$

**Step 1 :** 계층- $r$  동등승법형 대기네트워크의 분석,  $r = 1, 2$ .

- (1) 계층- $r$  동등승법형 대기네트워크의 정규화 상수  $G(N_r)$  계산.
- (2) 식 (23)을 이용하여 계층별 상태종속 산출률  $X_{r,i}(n_{r,i})$  계산.
- (3) 식 (24)을 이용하여 상태종속 도착률  $\lambda_{r,i}(n_{r,i})$  계산.

**Step 2 :** 하부네트워크  $R^i$  분석,  $i = 1, \dots, M$ .

- (1) 행렬기하법을 이용하여 하부네트워크  $R^i$ 를 분석하여  $P_i(c, m_1, m_2)$ 을 계산.
- (2)  $P_i(c, m_1, m_2)$ 이 구해지면 식 (17)과 식 (18)을 통해  $P_{r,i}(n_{r,i})$ 을 계산.



- (3) 식 (21)을 이용하여 상태종속 산출률  $v_{r,i}(n_{r,i})$  계산.
- (4) 식 (22)를 이용하여 상태종속 서비스율  $\mu_{r,i}(n_{r,i})$  계산.

**Step 3 :** 하부네트워크  $R^{M+1}$  분석

- (1) 식 (19)과 식 (20)을 이용하여 안정상태확률  $P_{r,M+1}(n_{r,M+1})$ 을 계산.
- (2) 식 (21)과 식 (22)를 이용하여 상태종속 서비스율  $\mu_{r,M+1}(n_{r,M+1})$  계산.

**Step 4 :**  $\mu_{r,i}(n_{r,i})$  수렴성 검사

- (1)  $\mu_{r,i}(n_{r,i})$ 가 수렴하면 Step 5로 넘어가고 그렇지 않으면 Step 1로 가서 반복.

**Step 5:** 성능척도 계산

**3.4 시스템의 성능척도 계산**

상태종속 서비스율  $\mu_{r,i}(n_{r,i})$  값이 수렴하면 각 하부네트워크의 안정상태확률로부터 다양한 성능척도를 계산할 수 있다. 우선 동기화 스테이션에서의 평균 재고 수  $\bar{I}$ , 평균 대기 수요 수  $\bar{BI}$ 와 평균 산출률  $\bar{X}$ 은 다음과 같이 구해진다.

$$\bar{I} = \sum_{r=1}^2 \sum_{k_r=1}^{N_r} k_r \cdot P_r(k_r) \tag{25}$$

$$\bar{BI} = \sum_{r=1}^2 \sum_{k_r=-B_r}^{-1} |k_r| \cdot P_r(k_r) \tag{26}$$

$$X = X_{1,M+1} + X_{2,M+1} \tag{27}$$

평균 대기 수요수가 구해지면 Little의 공식에 의해 수요의 평균 대기시간  $\bar{W}$ 을 구할 수 있다.

$$\bar{W} = \bar{BI} / X \tag{28}$$

재고 수에 따른 안정상태확률을 이용하여 도착한 고객이 수용될 확률인  $SL$ 과 즉시 만족될 확률인  $F$ 를 구할 수 있다.

$$SL = \sum_{r=1}^2 (X_{r,M+1} / X) \cdot (1 - P_r(-B_r)) \tag{29}$$

$$F = \sum_{r=1}^2 (X_{r,M+1} / X) \cdot \left( \sum_{n=1}^{N_r} P_r(n) \right) \tag{30}$$

하부네트워크  $R^i$ 에 대해서는 작업장별 기계의 평균 이용률

$\bar{U}_i$ 와 작업장별 평균 가공품수  $\bar{L}_i$ 을 계산할 수 있다.

$$\bar{U}_i = 1 - P_{r,i}(0, 0, 0), \quad i = 1, \dots, M \tag{31}$$

$$\bar{L}_i = \sum_{r=1}^2 \left( \sum_{n=1}^{N_r} n \cdot P_{r,i}(n) \right), \quad i = 1, \dots, M \tag{32}$$

**4. 근사적 분석의 정확도 평가**

본 연구에서 제안한 알고리즘의 성능을 평가하기 위해 많은 실험이 수행되었다. 또한 알고리즘의 정확도를 평가하기 위하여 알고리즘을 통해 얻은 값과 시뮬레이션에서 얻어진 값을 비교했다. 시뮬레이션 언어는 SIMLIB을 사용했으며, 수치분석에는 Visual Basic 6.0이 사용되었다. 시뮬레이션 종료조건으로는 시뮬레이션 시간을  $10^6$ 으로 설정하여 10번 반복했으며, 알고리즘의 종료조건은  $10^{-6}$  이하로 하였다.

본 연구에서는 변수 값이 변화함에 따라 근사적기법의 정확도가 어떻게 변화하는지 알아보기 위해 4개의 예제 세트에 대한 실험을 수행하였다. 예제 A에서는 고객의 도착률을 변화시켜 보았으며, 예제 B는 간판 수의 변화, 예제 C는 서비스율 변화, 예제 D는 작업장 수를 변화시켜 실험해 보았다. 각 예제에 대한 실험데이터는 <Table 2>에 주어지고, 실험결과는 <Table 3> ~ <Table 6>에 제시되어 있다. <Table 3> ~ <Table 6>에서 상대오차(relative error)는 “|알고리즘 결과-시뮬레이션 결과|/시뮬레이션 결과”로 정의된다.

제시된 결과에서 확인되듯이 본 논문에서 제안한 근사법은 대부분의 성능척도에서 상대오차 5% 미만의 만족할만한 수준의 해를 산출해 준다. 그러나 평균 대기수요  $\bar{BI}$ 와 이와 연관된 지표인 수요의 평균 대기시간  $\bar{W}$ 는 다른 성능척도에 비해 상대적으로 높은 오차율을 보여준다(최대 상대오차 28.8%). 그러나 최대 상대오차가 큰 경우에도 절대오차 값은 크지 않으며 이러한 경우에는 상대오차 값이 큰 의미를 갖지 못한다. 즉,  $\bar{BI}$ 와  $\bar{W}$ 의 값이 매우 작을 때에 한해 상대오차 값은 큰 값을 보이며  $\bar{BI}$ 와  $\bar{W}$ 의 값이 증가함에 따라 상대오차 값이 작아짐을 확인할 수 있다.

또한 실험결과로부터 수요의 도착율, 간판 수, 서비스율, 작업장의 수 등이 성능척도에 어떠한 영향을 미치는 지 확인할 수 있다. 예제 A에서는 제품별 수요의 도착률이 증가함에 따라 평균 대기 수요 수와 수요의 평균 대기시간이 증가함을 볼

**Table 2.** Experimental data sets

Experiment set	$\lambda_1$	$\lambda_2$	$K_1$	$K_2$	$M$	$B_1$	$B_2$	$\mu_{1,i}$ ( $i=1, \dots, M$ )	$\mu_{2,i}$ ( $i=1, \dots, M$ )
A	[0.65, 0.85]	[0.39, 0.51]	7	7	5	10	10	2	2
B	0.75	0.45	[5, 9]	[5, 9]	5	10	10	2	2
C	0.75	0.45	7	7	5	10	10	[1.9, 2.7]	[1.8, 2.6]
D	0.6	0.36	7	7	[8, 12]	10	10	2	2

Table 3. Results of experiment set A

$\lambda_1$	$\lambda_2$	Algorithm (Relative Error %)						Simulation (±95% Confidence Interval)					
		$\bar{I}$	$\overline{BI}$	$FR$	$SL$	$\overline{W}$	$X$	$\bar{I}$	$\overline{BI}$	$FR$	$SL$	$\overline{W}$	$X$
0.65	0.39	8.6950	0.2063	0.9210	0.9997	0.1985	1.0395	8.6884	0.1758	0.9227	0.9999	0.1689	1.0407
		(0.0760%)	(17.3493%)	(0.1842%)	(0.0200%)	(17.5252%)	(0.1153%)	(±0.0111)	(±0.0030)	(±0.0008)	(±0.0000)	(±0.0028)	(±0.0009)
0.7	0.42	7.8610	0.4144	0.8727	0.9991	0.3705	1.1185	7.8636	0.3507	0.8760	0.9995	0.3133	1.1196
		(0.0331%)	(18.1637%)	(0.3767%)	(0.0400%)	(18.2573%)	(0.0982%)	(±0.0099)	(±0.0039)	(±0.0007)	(±0.0000)	(±0.0033)	(±0.0006)
0.75	0.45	6.9422	0.7870	0.8055	0.9972	0.6583	1.1955	6.9234	0.6827	0.8084	0.9984	0.5696	1.1985
		(0.2715%)	(15.2776%)	(0.3587%)	(0.1202%)	(15.5723%)	(0.2503%)	(±0.0084)	(±0.0062)	(±0.0009)	(±0.0000)	(±0.0050)	(±0.0006)
0.8	0.48	5.9653	1.3993	0.7192	0.9928	1.1030	1.2686	5.9111	1.2603	0.7194	0.9952	0.9892	1.2740
		(0.9169%)	(11.0291%)	(0.0278%)	(0.2412%)	(11.5042%)	(0.4239%)	(±0.0113)	(±0.0124)	(±0.0014)	(±0.0001)	(±0.0094)	(±0.0006)
0.85	0.51	4.9858	2.3000	0.6196	0.9840	1.7234	1.3346	4.8736	2.1668	0.6141	0.9877	1.6136	1.3428
		(2.3022%)	(6.1473%)	(0.8956%)	(0.3746%)	(6.8047%)	(0.6107%)	(±0.0105)	(±0.0119)	(±0.0013)	(±0.0002)	(±0.0085)	(±0.0004)

Table 4. Results of experiment set B

$K_1$	$K_2$	Algorithm (Relative Error %)						Simulation (± 95% Confidence Interval)					
		$\bar{I}$	$\overline{BI}$	$FR$	$SL$	$\overline{W}$	$X$	$\bar{I}$	$\overline{BI}$	$FR$	$SL$	$\overline{W}$	$X$
5	5	3.5735	2.3329	0.5784	0.9858	1.9779	1.1795	3.4785	2.1258	0.5703	0.9908	1.7871	1.1895
		(2.7311%)	(9.7422%)	(1.4203%)	(0.5046%)	(10.6765%)	(0.8407%)	(±0.0035)	(±0.0153)	(±0.0007)	(±0.0002)	(±0.0128)	(±0.0003)
6	6	5.1907	1.3510	0.7090	0.9938	1.1347	1.1906	5.1500	1.1813	0.7102	0.9964	0.9879	1.1958
		(0.7903%)	(14.3655%)	(0.1690%)	(0.2609%)	(14.8598%)	(0.4349%)	(±0.0085)	(±0.0058)	(±0.0008)	(±0.0001)	(±0.0045)	(±0.0007)
7	7	6.9422	0.7870	0.8055	0.9972	0.6583	1.1955	6.9268	0.6843	0.8085	0.9984	0.5709	1.1985
		(0.2223%)	(15.0080%)	(0.3711%)	(0.1202%)	(15.3092%)	(0.2503%)	(±0.0102)	(±0.0066)	(±0.0008)	(±0.0001)	(±0.0053)	(±0.0007)
8	8	8.7808	0.4608	0.8729	0.9987	0.3847	1.1978	8.7853	0.3978	0.8764	0.9992	0.3318	1.1987
		(0.0512%)	(15.8371%)	(0.3994%)	(0.0500%)	(15.9433%)	(0.0751%)	(±0.0098)	(±0.0046)	(±0.0004)	(±0.0000)	(±0.0039)	(±0.0007)
9	9	10.6767	0.2705	0.9185	0.9994	0.2257	1.1989	10.6891	0.2358	0.9208	0.9996	0.1966	1.1991
		(0.1160%)	(14.7159%)	(0.2498%)	(0.0200%)	(14.8016%)	(0.0167%)	(±0.0106)	(±0.2358)	(±0.0007)	(±0.0000)	(±0.0030)	(±0.0006)

Table 5. Results of experiment set C

$\mu_{1,i}$	$\mu_{2,i}$	Algorithm (Relative Error %)						Simulation (± 95% Confidence Interval)					
		$\bar{I}$	$\overline{BI}$	$FR$	$SL$	$\overline{W}$	$X$	$\bar{I}$	$\overline{BI}$	$FR$	$SL$	$\overline{W}$	$X$
1.9	1.8	5.8259	1.4779	0.7091	0.9922	1.2435	1.1885	5.7778	1.3535	0.7080	0.9945	1.1344	1.1932
		(0.8325%)	(9.1910%)	(0.1554%)	(0.2313%)	(9.6174%)	(0.3939%)	(±0.0142)	(±0.0111)	(±0.0014)	(±0.0001)	(±0.0089)	(±0.0007)
2.1	2.0	7.3521	0.5862	0.8395	0.9983	0.4897	1.1972	7.3525	0.5057	0.8421	0.9991	0.4219	1.1988
		(0.0054%)	(15.9185%)	(0.3088%)	(0.0801%)	(16.0702%)	(0.1335%)	(±0.0091)	(±0.0059)	(±0.0009)	(±0.0001)	(±0.0047)	(±0.0007)
2.3	2.2	8.4775	0.2447	0.9110	0.9996	0.2041	1.1993	8.4869	0.2053	0.9134	0.9998	0.1711	1.1999
		(0.1108%)	(19.1914%)	(0.2628%)	(0.0200%)	(19.2870%)	(0.0500%)	(±0.0124)	(±0.0036)	(±0.0008)	(±0.0000)	(±0.0029)	(±0.0011)
2.5	2.4	9.3032	0.1102	0.9492	0.9999	0.0919	1.1998	9.3117	0.0928	0.9506	1.0000	0.0773	1.1998
		(0.0913%)	(18.7500%)	(0.1473%)	(0.0100%)	(18.8875%)	(0.0000%)	(±0.0064)	(±0.0014)	(±0.0004)	(±0.0000)	(±0.0012)	(±0.0007)
2.7	2.6	9.9242	0.0533	0.9700	1.0000	0.0445	1.1999	9.9306	0.0447	0.9709	1.0000	0.0372	1.1998
		(0.0644%)	(19.2394%)	(0.0927%)	(0.0000%)	(19.6237%)	(0.0083%)	(±0.0061)	(±0.0007)	(±0.0003)	(±0.0000)	(±0.0006)	(±0.0008)

수 있으며, 예제 B에서는 간판 수가 증가할수록 평균 대기 수  
요 수와 수요의 평균 대기시간이 감소함을 알 수 있다. 또한,

예제 C를 통해 제품별 서비스율이 증가함에 따라 평균 재고 수  
가 증가하고, 평균 대기 수요 수와 수요의 평균 대기시간이 감

**Table 6.** Results of experiment set D

M	Algorithm (Relative Error %)						Simulation (± 95% Confidence Interval)					
	$\bar{I}$	$\overline{BI}$	FR	SL	$\overline{W}$	X	$\bar{I}$	$\overline{BI}$	FR	SL	$\overline{W}$	X
8	6.8566	0.6718	0.8172	0.9980	0.7017	0.9573	6.8600	0.5229	0.8235	0.9993	0.5450	0.9595
	(0.0496%)	(28.4758%)	(0.7650%)	(0.1301%)	(28.7523%)	(0.2293%)	(±0.0077)	(±0.0055)	(±0.0009)	(±0.0000)	(±0.0056)	(±0.0006)
9	6.0564	1.0888	0.7505	0.9955	1.1408	0.9544	6.0526	0.8574	0.7575	0.9982	0.8951	0.9578
	(0.0628%)	(26.9886%)	(0.9241%)	(0.2705%)	(27.4494%)	(0.3550%)	(±0.0096)	(±0.0050)	(±0.0010)	(±0.0001)	(±0.0048)	(±0.0006)
10	5.3067	1.6509	0.6776	0.9909	1.7390	0.9493	5.2518	1.3673	0.6782	0.9959	1.4301	0.9561
	(1.0454%)	(20.7416%)	(0.0885%)	(0.5021%)	(21.5999%)	(0.7112%)	(±0.0098)	(±0.0107)	(±0.0011)	(±0.0001)	(±0.0107)	(±0.0006)
11	4.6227	2.3507	0.6034	0.9837	2.4966	0.9416	4.5229	2.0634	0.5954	0.9914	2.1692	0.9512
	(2.2065%)	(13.9236%)	(1.3436%)	(0.7767%)	(15.0931%)	(1.0093%)	(±0.0107)	(±0.0179)	(±0.0014)	(±0.0003)	(±0.0179)	(±0.0005)
12	4.0161	3.1569	0.5326	0.9734	3.3909	0.9310	4.0161	3.1569	0.5326	0.9734	3.3909	0.9310
	(4.0063%)	(7.5385%)	(3.6994%)	(1.0471%)	(9.0041%)	(1.3458%)	(4.0063%)	(7.5385%)	(3.6994%)	(1.0471%)	(9.0041%)	(1.3458%)

소함을 알 수 있다. 예제 D에서는 작업장 수가 증가함에 따라 평균 대기 수요 수와 수요의 평균 대기시간이 증가함을 확인할 수 있다.

### 5. 결론

본 연구에서는 두 종류의 제품이 생산되는 콘웍시스템에 대한 근사적 분석을 수행하였다. 연구 대상인 콘웍시스템을 분석하기 위해서 동기화 스테이션이 존재하는 2-계층 폐쇄형 대기네트워크로 모델링하고 이를 Baynat and Dallery(1996)가 개발한 승법형근사법을 이용해 분석하였다. 승법형근사법을 적용하는 과정에서 만나게 되는 가장 어려운 문제는 하부네트워크의 분석이다. 본 연구에서는 하부네트워크의 형태를 간략히 하기 위해서 원 네트워크를 구성하는 각각의 스테이션이 별도의 하부네트워크를 이루도록 네트워크를 분해했으며, 하부네트워크를 효율적으로 분석하기 위해서 행렬기하법을 사용했다. 본 연구에서 제안한 근사적기법을 다양한 예제에 적용하여 실험해 본 결과 상대오차율이 대부분 5% 미만으로 만족할 만한 정확도를 보이고 있다. 본 연구와 관련된 추후 연구로는 품목의 수가 3개 이상인 경우와 복수 서버의 경우, 그리고 서비스시간이 지수분포를 따르지 않는 경우에 대한 연구를 들 수 있겠다.

### 참고문헌

Baskett, F., Chandy, K.-M., Muntz, R.-R., and Palacios, F.-G. (1975), Open, closed and mixed networks of queues with different classes of customers, *Journal of the ACM*, **22**(2), 248-260.  
 Baynat, B. and Dallery, Y. (1993), A Unified View of Product-form Approximation Techniques for General Closed Queueing Networks, *Performance Evaluation*, **18**(3), 205-224.  
 Baynat, B. and Dallery, Y. (1996), A Product-form Approximation Me-

thod for General Closed Queueing Networks with Several Classes of Customers, *Performance Evaluation*, **24**(3), 165-188.  
 Bonvik, A.-M., Couch, C.-E., and Gershwin, S.-B. (1997), A Comparison of Production-line Control Mechanism, *International Journal of Production Research*, **25**(3), 789-804.  
 Buzacott, J.-A. and Shantikumar, J.-G. (1993), *Stochastic Models of Manufacturing Systems*, Prentice Hall, Englewood Cliffs, N. J.  
 Buzen, J.-P. (1973), Computational Algorithm for Closed Queueing Networks with queueing networks, *Communications of the ACM*, **16**(9), 527-531.  
 Lee, J.-E. and Lee, H.-Y. (1998), Approximate Analysis of a CONWIP system with Compound Poisson Demands, *Korean Journal of Operations Research*, **23**(3), 153-168.  
 Marie, R.-A. (1979), An Approximate Analytical Method for General Queueing Networks, *IEEE Transactions on Software Engineering*, **5**(5), 530-538.  
 Mascolo, M.-D., Frein, Y., and Dallery, Y. (1996), An Analytical Method for Performance Evaluation of kanban Controlled Production Systems, *Operations Research*, **44**(1), 50-64.  
 Neuts, M.-F. (1981), *Matrix Geometric Solutions in Stochastic Models-an Algorithmic Approach*, John Hopkins University Press, Baltimore.  
 Park, C.-H. and Lee, H.-S. (2006), Performance Evaluation of CONWIP system with Compound Poisson Demands and Coxian processing time, *Korean Journal of Operations Research*, **31**(3), 66-79.  
 Satyam, K. and Krishnamurthy, A. (2008), Performance Evaluation of a Multi-product System under CONWIP Control, *IIE Transactions*, **40**(3), 252-264.  
 Spearman, M.-L. (1992), Customer Service in Pull Production System, *Operations Research*, **40**(5), 948-958.  
 Spearman, M.-L., Woodruff, D.-M., and Hopp, W.-J. (1990), CONWIP : A Pull Alternative to Kanban, *International Journal of Production Research*, **28**(5), 879-894.  
 Spearman, M.-L. and Zazanis, M.-A. (1992), Push and Pull Production Systems: Issues and Comparisons, *Operations Research*, **40**(3), 521-532.  
 Whitt, W. (1994), Towards Better Multi-class Parametric Decomposition Approximations for Open Queueing Networks, *Annals of Operations Research*, **48**(3), 221-248.