

예약도착 대기행렬을 활용한 함정정비 최적 예약시간산정에 관한 연구

A Study on the Optimal Appointment Scheduling for the Ship Maintenance with Queueing System with Scheduled Arrivals

저자 (Authors)	고재우, 김각규, 윤봉규 Jae-Woo Ko, Gak-Gyu Kim, Bong-Kyu Yun
출처 (Source)	한국경영과학회지 38(3) , 2013.9, 13-22(10 pages) Journal of the Korean Operations Research and Management Science Society 38(3) , 2013.9, 13-22(10 pages)
발행처 (Publisher)	한국경영과학회 The Korean Operations Research and Management Science Society
URL	http://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE02246430
APA Style	고재우, 김각규, 윤봉규 (2013). 예약도착 대기행렬을 활용한 함정정비 최적 예약시간산정에 관한 연구. 한국경영과학회지, 38(3), 13-22
이용정보 (Accessed)	서울대학교 147.46.182.*** 2020/11/04 18:19 (KST)

저작권 안내

DBpia에서 제공되는 모든 저작물의 저작권은 원저작자에게 있으며, 누리미디어는 각 저작물의 내용을 보증하거나 책임을 지지 않습니다. 그리고 DBpia에서 제공되는 저작물은 DBpia와 구독계약을 체결한 기관소속 이용자 혹은 해당 저작물의 개별 구매자가 비영리적으로만 이용할 수 있습니다. 그러므로 이에 위반하여 DBpia에서 제공되는 저작물을 복제, 전송 등의 방법으로 무단 이용하는 경우 관련 법령에 따라 민, 형사상의 책임을 질 수 있습니다.

Copyright Information

Copyright of all literary works provided by DBpia belongs to the copyright holder(s) and Nurimedia does not guarantee contents of the literary work or assume responsibility for the same. In addition, the literary works provided by DBpia may only be used by the users affiliated to the institutions which executed a subscription agreement with DBpia or the individual purchasers of the literary work(s) for non-commercial purposes. Therefore, any person who illegally uses the literary works provided by DBpia by means of reproduction or transmission shall assume civil and criminal responsibility according to applicable laws and regulations.

예약도착 대기행렬을 활용한 함정정비 최적 예약시간 산정에 관한 연구

고재우¹ · 김각규² · 윤봉규^{2*}

¹해군 작전사령부, ²국방대학교 군사운영분석학과

A Study on the Optimal Appointment Scheduling for the Ship Maintenance with Queueing System with Scheduled Arrivals

Jae-Woo Ko¹ · Gak-Gyu Kim² · Bong-Kyu Yun²

¹Naval Operations Commander

²Department of Operations Research, Korea National Defense University

■ Abstract ■

Queueing system with scheduled arrivals is useful in many fields where both customers' waiting time and servers' operation time (utilization) are important, and arrival time of customers is possible to be controlled. In this paper, we analyzed the operation of ship maintenance with the queueing system with scheduled arrival. Based on the model presented by Pegden and Rosenshine [8], who dealt with exponential service time, we extended the service time distributions to phase-type distribution which is able to include a wide range of real stochastic phenomena. Since most activities in the military are carried out under tight control and schedule, scheduled arrival queue has quite good applicability in this area. In this context, we applied queue with scheduled arrival to the optimal booking time decision for the ship maintenance in the navy

Keyword : Scheduled Arrivals to Queues, Phase-Type Distribution, Optimal Scheduling Interval

논문접수일 : 2013년 01월 31일 논문게재확정일 : 2013년 05월 09일

논문수정일(1차 : 2013년 03월 20일, 2차 : 2013년 05월 08일)

* 교신저자 bkyoon1@gmail.com

1. 서 론

현실에서 일어나는 사건은 발생시점을 예측하기 어려운 경우가 많아 확률분포 형태로 정보를 표현하는 것이 일반적이다. 그러나 최근 기업뿐만 아니라 공공서비스에서 경쟁의 개념이 도입되고 고객 만족도에 의한 성과측정을 실시하는 등 고객의 선택권을 강화하기 위한 움직임이 있으며 그 결과 서비스 품질을 향상시키기 위한 많은 노력을 기울이고 있다. 특히 제공되는 서비스의 가치가 매우 높거나 고객이 대기시간에 민감한 경우 서비스 질을 향상시키거나 최적화된 고객의 대기시간 찾기 위한 방안을 찾고 있다. 예약도착 대기행렬모형(scheduled arrival to queues)은 일반적인 대기행렬시스템과 달리 고객의 도착이 확률적으로 결정되는 것이 아니라 특정 시점에 고객이 도착하도록 설계한 시스템으로써 고객의 도착시간을 적절하게 조절하여 서버를 효율적으로 운용하거나 고객의 대기시간을 줄이기 위해 새롭게 도입된 분석모형이다. 이때 서비스 시간이 고정되어 있지 않은 경우 예약 시스템에서 예약간격은 중요한 의사결정 요소이다. 예약간격을 짧게 잡으면 서버의 활용도는 높아지는 반면 고객의 대기시간이 늘어날 것이고, 예약간격을 길게 잡으면 고객의 대기시간은 줄일 수 있지만 서버의 활용도가 낮아지게 된다.

Bailey[3]는 고객의 대기시간을 최소화하기 위해 외래환자들의 예약 진료시간(appointment scheduling)을 조절함으로써 고객의 만족도를 높이는 최초의 연구를 하였다. 이후 연구는 주로 병원에서 서비스를 받는 환자와 진료를 시행하는 의료진 간의 대기시간과 유힬시간을 최소화하는 연구가 진행되었다[4]. Pegden and Rosenshine[8]은 앞선 특정분야에 대한 예약간격 최소화 연구와 달리 일반화된 예약도착 대기행렬시스템을 제시하였다(제 2.1 절 식 (1) 참고). 이는 시스템의 주된 비용을 고객의 대기비용과 서버의 운용비용으로 구분하고 두 비용의 합이 최소가 되는 고객 도착시점을 도출하는 모형이다. Stein and Cote[9]는 앞선 모형을 마코프체인으로 표현하여 다수 고객의 대기시

간을 효과적으로 산출하는 기법을 제시하였으며, 특히 현실적으로 계산의 용이성과 적용상 문제를 고려하여 도착간격을 균등하게 적용하는 경우에 대한 분석을 실시하였다. 또한 고객별로 예약간격을 달리하는 'Variable Interval Rule'에 대한 연구가 Yang et al.[12]와 Vanden Bosch and Dietz[11]에 의해 진행되었으며, 이후 Hassin and Mendel[5]은 Pegden and Rosenshine[8]의 모형을 기반으로 No-show 확률을 고려한 모형을 제시하였고, Sun et al.[10]는 기존 연구에 대한 다양한 수치해석을 실시하였다.

그러나 앞선 연구들은 분석의 용이성을 위해 시스템의 서비스 시간을 지수분포로 가정하여 분석하였으며, 이는 현실세계의 광범위한 현상들을 설명하는데 한계가 있다. 또한 군(軍)은 조직의 특수성을 고려하였을 때 엄격한 계획과 통제하에서 여러 분야의 시스템을 관리하고 운영할 필요성이 높으므로 예약 도착 모형의 적용은 무엇보다 유용할 것으로 기대된다. 즉, D/PH/1의 모형을 제시함으로써 전통적 대기행렬로는 묘사하기 힘든 현실 상황을 차별화시킨 것에 의의가 있다. 또한 기존의 D/G/1 모형의 특수한 경우를 설명하는데 기여하였다. 따라서 본 연구는 'Variable Interval Rule'에 대한 기존 예약도착 대기행렬시스템 연구를 확장한 고객의 서비스 시간이 단계형 분포(phase-type distribution)를 따를 때 **최적 도착간격을 결정할 수 있는 방안을 제시하고, 해군 함정의 정비 계획에 적용한 사례**를 바탕으로 다양한 분석을 실시한다.

이를 위해 제 2장에서는 기존 예약도착 대기행렬 시스템(scheduled arrival to queue)에 서비스 시간을 단계형 분포로 확장한 윤봉규[2] 연구를 바탕으로 증명부분을 좀 더 보강하고 다양한 경우를 고려하여 살펴본다. 제 3장에서는 확장된 예약도착 대기행렬시스템 모형을 해군 함정 정비계획에 적용하여 다양한 분석을 실시하고, 제시한 모형을 보다 쉽게 적용할 수 있도록 Matlab을 활용한 방법을 소개한다. 제 4장에서는 본 논문에서 연구한 결과 및 향후 연구방향을 제안한다.

2. 예약도착 대기행렬시스템

2.1 예약도착 시스템의 최적 예약간격 분석모형

예약도착 대기행렬시스템에서 최적 도착간격을 도출하기 위한 방안을 살펴보기 위해 먼저 매개변수와 결정변수를 정의하고 다음과 같이 가정한다.

<결정변수>

x_i : i 번째 고객과 $i+1$ 번째 고객의 도착간격,
($i = 0, 1, 2, \dots, K$)

<매개변수>

K : 최대 예약가능 고객수
 N_i : i 번째 예약 고객이 도착할 때 시스템 내에 존재하는 고객수(i 번째 고객은 미포함)
 w_i : i 번째 고객의 시스템 내 평균 대기시간($w_1 = 0$)
 c_w : 단위시간당 고객 대기비용
 c_s : 단위시간당 서버 운용비용
 S_i : x_i 동안 서비스를 받고 떠난 고객수
 J_i : i 번째 고객 도착시 진행중인 서비스 단계
 α : 흡수 마코프체인의 초기 확률벡터
 T : 흡수 마코프체인의 일시상태 전이율 행렬
 S : 서비스 시간
 t : 일시상태에서 흡수상태로의 전이율 행렬, 크기가 m 인 열벡터

<가정>

- 예약 가능한 고객은 총 K 명이다.
- 서비스는 먼저 온 고객이 먼저 서비스를 받고 시스템을 떠나는 FCFS(선입선출) 방식을 따른다.
- 서버는 첫 번째 고객이 도착함과 동시에 운용되기 시작하고 마지막 고객이 서비스를 받고 떠나는 시점에 종료된다. 즉, 서버는 첫 번째 고객이 도착함과 동시에 ON 상태를 시작하여 OFF없이 마지막 고객의 서비스까지 지속된다.

앞선 기호들과 가정들을 바탕으로 시스템에서 발생하는 총 비용의 기대값은 식 (1)과 같다.

$$z = c_w \sum_{i=2}^K w_i + c_s \left\{ \sum_{i=1}^{K-1} x_i + w_K + E[S] \right\} \quad (1)$$

위 식에서 우변의 첫 항은 첫 번째 고객의 대기시간을 제외한 모든 고객의 대기시간을 합한 값에 단위시간당 고객의 대기비용을 곱하여 고객의 총 대기비용을 표현한 것이다. $i=2$ 부터 시작한 것은 첫 번째 고객은 도착과 동시에 서버가 시작되어 대기시간이 발생하지 않기 때문이다. 우변의 둘째 항은 서버의 총 운용시간에 단위시간당 서버의 운용비용을 곱하여 서버의 총 운용비용을 나타낸 것이다. 즉, $\sum_{i=1}^{K-1} x_i$ 는 첫 번째 고객과 마지막 고객과의 도착간격을 의미하며, $w_K + E[S]$ 는 마지막 고객 대기시간과 서비스 시간의 기대치이다.

예약도착 대기행렬시스템의 비용을 결정하는 식 (1)에서 나머지 값들은 모두 상수이나 w_i 값은 고객의 도착간격과 서버의 서비스 시간에 의해 변하게 된다. 따라서 예약도착 대기행렬시스템의 최소 비용을 구하기 위해서는 우선 w_i 에 관한 식을 모두 구하고, z 를 최소화하는 x_i 값을 구하면 된다. 이는 dz/dx_i 를 0으로 만드는 방정식의 해를 구함으로써 도출할 수 있다.

2.2 서비스가 지수분포인 경우의 w_i

예약도착 대기행렬시스템의 목적식 (1)에서 $E[S]$ 에 서비스 평균인 $1/\mu$ 를 대입하면 미지수(Unknown)는 i 번째 고객의 대기시간의 평균인 w_i 만 남게 된다. w_i 는 다음 식 (2)와 같이 i 번째 고객 도착시 보는 고객수에 조건을 걸어 유도할 수 있다.

$$w_i = E[W_i] = \sum_{n=1}^{i-1} E[W_i | N_i = n] P[N_i = n] \quad (2)$$

식 (2)에서 w_i 는 i 번째 고객이 n 명을 볼 확률과 n 명을 볼 경우 대기시간의 기대치의 곱으로 나타

낼 수 있으며, 이때 n 은 최소 0명이지만 0명을 볼 경우 대기시간은 발생하지 않으므로 최소 1이라는 것을 알 수 있다. 또한 i 번째 고객이 도착시 보는 고객수는 최대 $i-1$ 명이다. 식 (2)에서 i 번째 고객이 n 명을 볼 경우 대기시간의 기대치는 서비스 시간이 평균 $1/\mu$ 인 지수분포이므로 n/μ 가 되어 식 (3)과 같이 표현할 수 있다.

$$w_i = \sum_{n=1}^{i-1} \frac{n}{\mu} P[N_i = n] \quad (3)$$

식 (3)으로부터 w_i 를 구하기 위해서는 i 번째 예약 고객 도착시 n 명을 볼 확률인 $P[N_i = n]$ 을 알아야 한다. i 번째 고객이 보는 시스템 내 고객 수는 $i-1$ 번째 고객이 본 고객 수에 $i-1$ 번째 도착한 고객을 더하고 x_{i-1} 동안 서비스를 받고 떠난 고객수를 빼면 구할 수 있으므로 다음과 같은 관계를 유도할 수 있다.

$$N_i = N_{i-1} + 1 - S_{i-1} \quad (4)$$

식 (4)를 통해 N_i 는 무기억속성(memoryless property)을 만족하는 마코프체인임을 알 수 있다.

$P[N_i = n]$ 는 N_{i-1} 에 조건을 걸어 식 (5)와 같이 유도할 수 있다.

$$\begin{aligned} P[N_i = n] &= \sum_{k=(n-1)^+}^{i-2} P[N_{i-1} = k] \cdot P[N_i = n | N_{i-1} = k] \\ &\quad (\text{단, } (n-1)^+ = \max(0, n-1)) \end{aligned} \quad (5)$$

식 (5)에서 i 번째 고객이 n 명을 보기 위해서는 $i-1$ 번째 고객이 최소 $n-1$ 명의 고객을 봐야하며, $i-1$ 번째 고객은 최대로 볼 수 있는 고객수가 $i-2$ 명이므로 k 의 범위가 $n-1$ 에서 $i-2$ 로 정해진다는 것을 알 수 있다. 식 (5)의 우변에 표현되어 있는 $P[N_i = n | N_{i-1} = k]$ 는 $i-1$ 번째 고객이 k 명의 고객을 보고 도착했을 때 i 번째 고객이 n 명을 볼 확률이다. 이는 식 (4)를 통해 S_{i-1} 이 $k+1-n$ 명이 되어야 하므로 특정 시간동안 도착 고객의 수가 포아송 분포를 따르는 포아송 과정(Poisson Process)의 성질

을 이용하여 식 (6)을 유도할 수 있다.

$$P[N_i = n | N_{i-1} = k] = \frac{e^{-\mu x_{i-1}} (\mu x_{i-1})^{k+1-n}}{(k+1-n)!} \quad (6)$$

식 (6)을 식 (5)에 대입하면 다음과 같은 식 (7)을 유도할 수 있으며, 식 (7)을 통해 축차적으로 적용하게 되면 결국 $P[N_i = n]$ 을 구할 수 있다.

$$P[N_i = n] = \sum_{k=(n-1)^+}^{i-2} P[N_{i-1} = k] \cdot \frac{e^{-\mu x_{i-1}} (\mu x_{i-1})^{k+1-n}}{(k+1-n)!} \quad (7)$$

한편, 식 (7)을 통해 구한 $P[N_i = n]$ 을 식 (3)에 대입하여 w_i 를 구하고 그 결과를 식 (1)에 적용하게 되면 x_i 에 관한 시스템의 총 비용을 나타내는 방정식을 도출할 수 있다. 여기서 식 (1)을 최소화할 수 있는 고객 도착간격을 정하기 위해서는 x_i ($i = 1, \dots, k-1$)에 관한 편미분을 하여 그 값이 0이 되도록 하는 x_i 값을 구하면 된다.

2.3 단계형 분포를 적용한 확장된 예약도착 시스템 모형

Neuts[7]에 의해 1980년대 초반에 단계형 분포(phase-type distribution)가 제시된 이후 지수분포의 확장에 관한 많은 연구가 이루어졌다. 단계형 분포는 Erlang 분포와 같이 몇 개의 지수분포의 단계로 이루어진 확률분포로써 지수분포의 무기억속성(memoryless property)을 그대로 유지하여 분석이 용이하며, 시스템을 여러 단계로 구분하여 표현함으로써 보다 광범위한 현실의 복잡한 현상을 모델링 가능하고 정규분포까지 근사 가능한 특징을 가진다. 따라서 제 2.2절에서는 기존 연구에서 제시된 서비스 시간이 지수분포를 적용한 예약도착 시스템의 최적 예약 간격을 알아보았으나, 본 절에서는 현실을 보다 정확히 묘사하기 위하여 서비스 시간을 단계형 분포로 확장했을 때 시스템의 최적 예약 간격을 도출하는 방법을 살펴본다.

상태(State) {0}을 흡수상태, 상태 $\{1, \dots, m\}$ 을 일시상태라고하면, 단계형 확률변수는 식 (8)과 같은 전이율 행렬을 가지는 하나의 흡수상태와 적절한 초기상태확률 벡터 α 를 갖는 연속시간 마코프체인(continuous time Markov chain)이 흡수될 때까지 걸린 시간이다.

$$Q = \begin{bmatrix} 0 & 0 \\ t & T \end{bmatrix} \quad (8)$$

단계형 확률변수의 특징은 다음과 같다.

Theorem 1 : $[(-T)^{-1}]_{r,j}$ 는 단계 r 에서 시작한 단계형 확률 변수가 흡수될 때까지 일시 상태 j 에 머문 시간의 기대치이다.

Theorem 2 : 단계형 확률변수 Y_a, Y_b 가 각각 $Y_a \sim PH(\alpha_a, T_a)$, $Y_b \sim PH(\alpha_b, T_b)$ 이라고 하면, 두 단계형 확률변수의 합 $Y = Y_a + Y_b$ 는 $PH(\alpha, T)$ 가 된다. 단, α, T 는 다음과 같다.

$$\alpha = [\alpha_a, 0, \dots, 0], \quad T = \begin{bmatrix} T_a & t_a \alpha_b \\ 0 & T_b \end{bmatrix} \quad (9)$$

($t_a = -T_a \mathbf{1}$, 단, $\mathbf{1}$ 은 모든 원소가 1인 벡터)

Theorem 1, 2에 대한 증명은 Latouche and Ramaswami[6]와 윤봉규[1, 2]를 참조하기 바란다.

Corollary 1 : $\alpha(-T)^{-1}\mathbf{1}$ 은 일시상태에서 시작하여 흡수될 때까지 걸린 시간의 기대치이다.

Proof] Theorem 1에 의해 초기 시작단계에 조건을 걸면 서비스 시간의 기대치는 다음 식과 같다.

$$E[S] = \sum_{r=1}^m E[S | \text{시작시 } phase = r] \cdot P[\text{시작시 } phase = r] \text{이며, } E[S | \text{시작시 } phase = r] \text{는 } \sum_{j=1}^m (-T)^{-1}_{r,j} \text{이므로 } E[S] = \alpha(-T)^{-1}\mathbf{1} \text{이 된다.}$$

따라서 제 2.1절의 목적식을 단계형 분포를 적용한 모형으로 표현하면 Corollary 1에 의해 다음 식 (10)과 같다.

$$z = c_w \sum_{i=2}^K w_i + c_s \left\{ \sum_{i=1}^{K-1} x_i + w_K + \alpha(-T)^{-1}\mathbf{1} \right\} \quad (10)$$

(단, $\mathbf{1}$ 은 모든 원소가 1인 벡터)

식 (10)은 지수분포 서비스 시간을 가정한 경우와 달리 마지막 고객의 평균 서비스 시간이 $E[S]$ 에서 $\alpha(-T)^{-1}\mathbf{1}$ 로 바뀐 것이다. 서비스 시간의 평균뿐 아니라 도착시 진행중인 서비스는 진행 단계에 따라서 서비스 종료까지의 시간이 달라지므로 식 (2)에서 표현된 w_i 가 다음 식 (11)와 같이 변형된다.

$$w_i = \sum_{n=1}^{i-1} \sum_{j=1}^m E[W_i | N_i = n, J_i = j] \cdot P[N_i = n, J_i = j] \quad (11)$$

즉, w_i 값은 기존의 식에 고객의 서비스 단계를 세 부적으로 표현한 것으로 i 번째 고객이 n 명을 보고, 서비스 단계가 j 인 경우 대기시간의 기대치 $E[W_i | N_i = n, J_i = j]$ 와 i 번째 고객이 n 명의 고객을 보고 이때 서비스 단계가 j 일 확률인 $P[N_i = n, J_i = j]$ 와의 곱으로 표현된다. 또한 $E[W_i | N_i = n, J_i = j]$ 은 Theorem 1을 이용하여, $P[N_i = n, J_i = j]$ 은 단계를 고려하여 식 (12)~식 (14)으로 표현됨을 알 수 있다.

$$E[W_i | N_i = n, J_i = j] \quad (12)$$

$$= [(-T)^{-1}\mathbf{1}]_j + (n-1) \cdot \alpha(-T)^{-1}\mathbf{1}$$

(단, $\mathbf{1}$ 은 모든 원소가 1인 벡터)

식 (12)의 우변 첫 항은 단계 j 에 있는 진행 중인 서비스의 잔여시간의 기대치이며, 두 번째 항은 단계형 확률변수를 따르는 $n-1$ 명의 서비스 시간의 기대치이다.

i 번째 고객이 시스템에 도착하였을 경우 확률은 다음의 두 가지 경우를 고려할 수 있다.

(i) $n \geq 1$ 일 경우(j 의 서비스 단계를 볼 경우)

$$\begin{aligned}
 & P[N_i = n, J_i = j] \\
 &= \sum_{k=(n-1)^+}^{i-2} \sum_{r=1}^m P[N_{i-1} = k, J_{i-1} = r] \\
 &\quad \cdot P[N_i = n, J_i = j | N_{i-1} = k, J_{i-1} = r]
 \end{aligned} \quad (13)$$

(ii) $n=0$ 일 경우(서비스 단계가 없는 경우)

$$\begin{aligned}
 & P[N_i = 0] \\
 &= \sum_{k=\max(1, n-1)}^{i-2} \sum_{r=1}^m P[N_{i-1} = k, J_{i-1} = r] \\
 &\quad \cdot P[N_i = 0 | N_{i-1} = k, J_{i-1} = r] \\
 &\quad + P[N_{i-1} = 0] \cdot P[N_i = 0 | N_{i-1} = 0]
 \end{aligned} \quad (14)$$

식 (13)의 우변 $P[N_i = n, J_i = j | N_{i-1} = k, J_{i-1} = r]$ 은 $i-1$ 번째 고객이 k 명을 보고 도착하고 진행 중인 서비스 단계가 r 이라는 조건하에 i 번째 고객이 n 명의 고객을 보고 도착하고 서비스 단계가 j 일 확률이다. 단계형 분포를 따르는 경우에도 식 (4)가 성립하며 x_{i-1} 동안 서비스를 받고 나간 고객 수 S_{i-1} 은 $k+1-n$ 명이 되어야 한다.

한편, $i-1$ 번째 고객이 k 명을 보고 도착하므로 x_{i-1} 동안 일어나는 서비스 과정은 $k+1$ 개의 단계형 확률변수의 합으로 표현되는 새로운 단계형 확률변수에 따라 결정된다. $k+1$ 명의 서비스 과정은 Theorem 2에 의해 $PH(\alpha_{k+1}, T_{k+1})$ 에 따라 결정된다.

$$\alpha_{k+1} = [\alpha_k, 0, \dots, 0], \quad T_{k+1} = \begin{bmatrix} T_k & t_k \alpha_1 \\ 0 & T_1 \end{bmatrix} \quad (15)$$

(α_1 : r 번째 숫자만 1이고 나머지는 0인 벡터)

이상의 논의를 정리하면 아래와 같은 식 (16)을 얻을 수 있으며, 식 (16)을 유도한 방법의 연장선에서 단계형 확률변수의 특성에 의해 식 (17), 식 (18)을 유도할 수 있다.

$$\begin{aligned}
 & P[N_i = n, J_i = j | N_{i-1} = k, J_{i-1} = r] \\
 &= [e^{T_{k+1}x_{i-1}}]_r \cdot (k+1-n)m+j
 \end{aligned} \quad (16)$$

$$\begin{aligned}
 & P[N_i = 0 | N_{i-1} = k, J_{i-1} = r] \\
 &= 1 - \alpha_{k+1} e^{T_{k+1}x_{i-1}} \mathbf{1}
 \end{aligned} \quad (17)$$

$$P[N_i = 0 | N_{i-1} = 0] = 1 - \alpha e^{T x_{i-1}} \mathbf{1} \quad (18)$$

위의 식 (16)~식 (18)을 식 (13), 식 (14)에 대입하여 i 번째 고객이 도착 시 n 명의 고객을 볼 확률을 구할 수 있으며 이는 식 (19), 식 (20)과 같이 정리할 수 있다.

(i) $n \geq 1$ 일 경우

$$\begin{aligned}
 & P[N_i = n, J_i = j] \\
 &= \sum_{k=(n-1)^+}^{i-2} \sum_{r=1}^m P[N_{i-1} = k, J_{i-1} = r] \\
 &\quad \cdot [e^{T_{k+1}x_{i-1}}]_r \cdot (k+1-n)m+j
 \end{aligned} \quad (19)$$

(ii) $n=0$ 일 경우

$$\begin{aligned}
 & P[N_i = 0] = \sum_{k=\max(1, n-1)}^{i-2} \sum_{r=1}^m P[N_{i-1} = k, J_{i-1} = r] \\
 &\quad \cdot (1 - \alpha_{k+1} e^{T_{k+1}x_{i-1}} \mathbf{1}) \\
 &\quad + P[N_{i-1} = 0] \cdot (1 - \alpha e^{T x_{i-1}} \mathbf{1})
 \end{aligned} \quad (20)$$

식 (19), 식 (20)을 통해 축차적으로 $P[N_i = n, J_i = j]$ 를 구한 값을 식 (11)에 대입하면 w_i 를 구할 수 있으며 이를 통해 식 (10)을 도출할 수 있다. 결국 식 (10)은 x_i 에 관한 시스템의 총 비용을 나타내는 방정식으로 표현되며 x_i 를 편미분하는 방법을 통해 최소값을 구할 수 있다.

3. 사례적용 및 결과분석

시스템을 운영하는 경영자의 입장에서 시스템에서 발생하는 비용을 최소화하고자 하는 노력은 필수적이라 할 수 있다. 이는 군(軍)이라고 예외될 수 없으며 다양한 분야에서 시스템을 효과적으로 운영하고자 하는 노력이 수반되고 있다. 본 장에서는 해군 함정의 정비계획을 사례로 단계형 분포를 적용한 확장된 모형으로 정비를 위한 최적 예약시간을 산출해 본다.

3.1 단계형 분포 적용의 적합성 검증

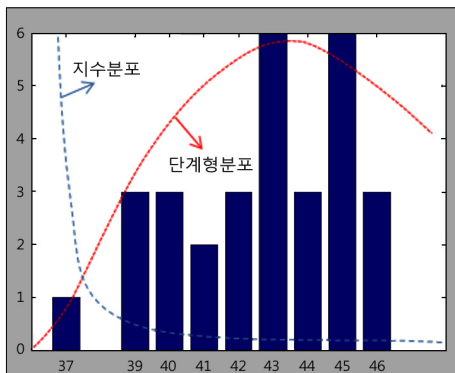
단계형 분포로 확장된 모형을 적용하기 위해 해군 함정정비 시간자료 30여 개를 수집하였으며, 정비단계는 크게 3단계로 구분하였다. 단계별 정비시간과 총 정비시간을 분석한 내용은 <표 1>과

값으며, 데이터의 검증을 위해 Arena 11.0(student version)의 Input Analyzer를 활용하고 이때 귀무가설 H_0 는 서비스가 지수분포를 따르는 것으로 설정하였다. 단, 해군의 함정정비 시간은 군의 정비능력에 대한 세부 수치이므로 '0'으로 처리하였다.

<표 1> 단계별 함정 평균 정비시간 분석

구 분	1단계 (상가/ 선체세척)	2단계 (탑재장비 정비)	3단계 (하가/ 시운전)	전체 (상가/선체세척+ 탑재장비정비+ 하가/시운전)
서비스시간 (일)	0.0000	00.0000	0.0000	00.0000
square error	0.008639	0.004747	0.000831	0.083953
유의확률 (p-value)	0.38	0.395	p > 0.75	p < 0.05

<표 1>로부터 함정 정비시간을 단계별로 구분했을 때가 구분하지 않을 때보다 오차의 크기가 작음을 알 수 있다. 특히, 함정 정비시간을 하나의 지수분포로 적용했을 경우 유의확률이 0.05보다 작을 값은 없으며, 이는 지수분포의 적용이 적합하지 않음을 보여준다. 반면 단계형 분포일 경우 모든 p-value가 유의수준 0.05보다 큰 값을 가져 단계별 지수분포의 가정이 유의함을 알 수 있다. 또한 전체 시간을 단계형 분포로 fitting 시킨 결과를 비교할 경우에 쉽게 단계형 분포가 실제 데이터 값을 잘 반영하고 있다는 것을 알 수 있다. [그림 1]은 단계형 분포와 지수분포의 pdf를 비교한 것이다.



[그림 1] 단계형 분포와 지수분포의 pdf 비교

결국 함정 정비 시 서비스 시간에 대한 분석은 단계형 분포를 적용하는 것이 바람직하다는 것을 알 수 있다.

3.2 지수분포와 단계형 분포 적용시 비용변화

앞선 함정 수리시간의 사례분석을 통해 함정 정비시간은 단계형 분포를 적용하는 것이 바람직하다는 것을 확인하였다. 따라서 지수분포를 적용했을 때의 결과와 단계형 분포를 적용했을 때의 결과의 비교를 통해 각각이 시스템 비용발생에 어떤 영향을 주는지에 대한 분석이 필요하다. <표 2>에서 나타내고 있는 x_1 은 첫 번째 함정의 수리시작 이후 두 번째 함정이 도착할 때까지의 시간간격이고, x_2 는 두 번째 함정의 수리시작 이후 세 번째 함정이 도착할 때까지의 시간간격이다. 실험에서 고객 대기비용(c_w)은 함정이 정비를 해야 함에도 불구하고 대기함으로써 고장 등 위험요소가 발생할 수 있는 단위시간당 발생비용이며, 서버의 운용비용(c_s)은 함정 정비공장에서 함정의 정비를 기다리면서 발생하는 단위시간당 비용을 말한다.

<표 2> 단계형 분포를 지수 분포로 가정시 손실비용

c_w	c_s	지수 분포			단계형 분포			손실 비용
		x_1	x_2	비용	x_1	x_2	비용	
1	5	11.3	24.0	943.7	26.1	49.6	841.3	102.4
1	4	13.9	26.4	756.1	28.5	49.5	684.3	71.8
1	3	17.9	29.9	569.9	31.9	49.9	526.3	43.6
1	2	25.0	35.6	385.7	37.3	51.3	366.4	19.3
1	1	40.7	48.2	204.4	48.3	56.6	201.5	2.9
2	1	60.6	65.0	226.4	61.7	66.0	226.3	0.1
3	1	73.8	76.8	244.1	70.4	73.2	243.6	0.5
4	1	83.7	85.8	258.6	77.0	79.0	257.0	1.6
5	1	91.6	93.3	270.9	82.4	83.9	267.9	3.0

*손실비용 = 지수분 포 비용-단계형 분포 비용.

<표 2>를 통해 $c_w=1$, $c_s=1$ 일 때, x_1 은 지수분포에서 40.7611일, 단계형 분포에서 48.3638일이고, x_2 는 지수분포 48.2574일, 단계형 분포 56.6540일이 되는 것을 알 수 있다. 이와같이 단계형 분포

적용이 적합한 시스템의 경우 서비스 시간을 지수 분포로 가정하면 보다 큰 x_1 값과 x_2 값을 적용하게 됨으로써 결과적으로 단계형 분포에서의 최적값과 차이가 발생한다. 다시 말하면 단계형 분포를 따르는 현실시스템을 지수 분포로 가정할 시 불필요한 비용이 발생하게 된다. 또한 c_s 가 c_w 보다 큰 값을 가질수록 손실비용이 늘어나는 것도 알 수 있다.

3.3 서비스단계 세분화에 따른 x_1 , x_2 값의 변화

<표 2>에서 지수분포는 c_w 와 c_s 의 값에 따라 고객의 도착간격 변화가 민감히 반응하는 반면, 단계형 분포를 적용하였을 경우 상대적으로 변화의 폭이 크지 않음을 확인할 수 있다. 이를 좀 더 알아보기 위해 <표 3>과 같이 단계를 세분화 하였으며, 서비스 과정을 1단계로 고려하였을 경우 지수 분포 적용시와 동일하다. <표 3>은 제 3.2절의 실험에 2단계로 구분된 서비스 과정을 추가하여 분석함으로써 단계를 구분할수록 결과값의 분산이 작아진다는 것을 확인할 수 있다. 이는 단계가 세분화될수록 서비스 종료시점에 대한 예측력이 높아져서 고객의 도착시점을 보다 정확히 예측할 수 있기 때문이다. 즉, 서비스 단계를 100단계로 구분하고 각 단계의 서비스 시간이 1시간인 시스템과 동일한 100시간이 소요되지만 단계가 2단계로 구분되어 있는 시스템을 비교해보자. 만약 현재 서비스 중인 고객이 95단계에 있다면 앞선 시스템에서는 5단계,

즉 5시간 정도 후에 다음고객이 서비스를 받을 수 있을 것이라고 정확한 예측이 가능하지만 2단계로만 구분되어 있는 시스템에서는 95단계에 있는 동일한 고객을 단지 2단계로 인식하고 결과적으로 다음고객이 0~50시간 정도 후에 서비스 받을 것이라고 큰 편차를 갖는 결과값을 제시하게 된다.

3.4 도착간격 특성 및 비용변화에 따른 결과 비교

지금까지 제 2장에서 제시한 예약도착 대기행렬 모형에 의해 각 고객의 도착간격을 독립적으로 고려한 결과를 보았으나 이번 절은 Stein and Cote[9]의 연구에서와 같이 고객의 도착간격(\bar{x})을 동일하게 적용하였다. 이는 서버가 일정한 간격의 예약을 받음으로 해서 서비스의 편리성을 높이고, 변수의 수를 줄임으로써 계산의 용이성을 높여준다. <표 4>는 고객의 도착간격이 다른 경우와 같을 경우에 대한 결과를 나타낸 것이다. 기존의 Stein and Cote[9]의 연구에서 언급되었듯이 고객 대기비용이 상대적으로 서버 운용비용보다 큰 값일수록 고객 도착간격이 같을 경우와 다를 경우의 차이가 작아지게 된다. 그러나 이번 연구의 사례와 같이 함정 정비의 경우 함정 도착간격의 주기가 일단위로 비교적 큰 시간단위로 설정해야할 경우는 고객의 편의성 측면보다는 시스템의 비용측면이 더욱 크게 인식이 되기 때문에 정확한 도착간격을 제시하는 것이 바람직하다고 할 수 있다.

<표 3> 서비스단계 세분화에 따른 고객도착 간격결과

C_w	C_s	서비스단계를 1단계로 구분한 경우		서비스단계를 2단계로 구분한 경우		서비스단계를 3단계로 구분한 경우	
		x_1	x_2	x_1	x_2	x_1	x_2
1	5	11.3	24.0	19.3	35.6	26.1	49.6
1	4	13.9	26.4	21.7	36.6	28.5	49.5
1	3	17.9	29.9	25.3	38.5	31.9	49.9
1	2	25.0	35.6	31.3	42.0	37.3	51.3
1	1	40.7	48.2	44.0	50.8	48.3	56.6
2	1	60.6	65.0	59.8	63.5	61.7	66.0
3	1	73.8	76.8	70.1	72.6	70.4	73.2
4	1	83.7	85.8	77.9	79.7	77.0	79.0
5	1	91.6	93.3	84.1	85.5	82.4	83.9

〈표 4〉 비용변화에 따른 도착간격 특성별 결과

구 분	x_i 가 다를때		x_i 가 같을때
	x_1	x_2	\bar{x}
$C_w = 1, C_s = 5$	11.3	24.0	16.0
$C_w = 1, C_s = 4$	13.9	26.4	18.7
$C_w = 1, C_s = 3$	17.9	29.9	22.7
$C_w = 1, C_s = 2$	25.0	35.6	29.4
$C_w = 1, C_s = 1$	40.7	48.2	44.1
$C_w = 2, C_s = 1$	60.6	65.0	62.7
$C_w = 3, C_s = 2$	73.8	76.8	75.2
$C_w = 4, C_s = 1$	83.7	85.8	84.7
$C_w = 5, C_s = 1$	91.6	93.3	92.4

비용변화에 따른 결과를 살펴보면 c_w 의 비중이 크다는 것은 고객의 대기비용이 크게 고려된다는 것으로써 VIP 고객을 다루는 시스템에 적합하다. 이러한 시스템에서는 고객의 대기비용이 상당히 중요하게 작용할 것이며, 따라서 고객의 불만이 나오지 않도록 서버가 비효율적으로 작동하더라도 고객의 대기시간을 줄이기 위해 도착간격을 충분히 넓히게 될 것이다. 〈표 4〉를 통해 이러한 결과가 현실과 다르지 않음을 알 수 있다. 또한 고객을 중시할수록 도착간격을 넓게 잡을 것이므로 도착간격의 차가 서버를 중시할 때에 비해 둔감해지는 것을 〈표 4〉에서 확인할 수 있다.

3.5 Matlab을 활용한 결과산출 방법

최적 예약시간을 산출하기 위한 기존 방법은 w_i 값을 구함으로써 도출된 목적식을 x_i 에 대하여 편미분한 값을, 0으로 만들어 결과를 산출하였다. 그러나 단계형 분포로 확장할 경우 모든 결과들이 복잡한 행렬과 벡터로 표현된다는 점에서 이 계산과정은 상당히 어렵고 번거로워진다. 실례로 식 (21)은 Pegden and Rosenshine[8]에서 고객이 3명인 경우 최종 목적식에서 각각 x_1 와 x_2 에 대해 편미분한 값을 보여준다.

$$\begin{aligned} \frac{dz}{dx_1} &= -C_w e^{-\mu x_1} - (C_w + C_s) \\ &\quad \times [e^{-\mu(x_1+x_2)} + \mu x_2 e^{-\mu(x_1+x_2)}] + C_s = 0 \\ \frac{dz}{dx_2} &= -(C_w + C_s) \\ &\quad \times [e^{-\mu x_2} + \mu x_1 e^{-\mu(x_1+x_2)}] + C_s = 0 \end{aligned} \quad (21)$$

본 연구에서는 계산과정의 용이성을 위하여 Matlab의 함수를 이용하여 결과를 산출하는 방법을 제안한다. 연구에서 활용한 방법은 *fminsearch* 함수를 이용하는 것으로써 이 함수는 Simplex/Direct Search Method를 이용하여 예약도착 대기행렬시스템의 목적식과 같이 x_i 로 이루어진 다변수의 목적식에 대한 최소값을 구하는 문제에 적합하고, 벡터나 행렬로 표현되어도 계산이 가능하여 단계형 분포를 사용하기도 적합하다. 즉, Matlab의 함수를 이용한 방법은 최종 목적식을 구한 후 추가적인 편미분 작업 없이 최소값을 산출할 수 있다.

〈표 5〉는 비선형방정식의 최적해를 찾는 *fminsearch* 함수를 이용한 결과로써, 이를 통해 *fminsearch* 함수를 이용 시에도 고객 대기비용과 서버 운용비용의 비율에 상관없이 안정적인 해를 산출할 수 있다는 것을 알 수 있다. 또한 *fminsearch* 함수의 결과는 〈표 4〉의 결과와 상대적 오차를 비교하였을 때 충분히 유의미 하다.

〈표 5〉 *fminsearch* 함수를 활용한 C_w 와 C_s 결과

C_w	C_s	비율	결과값	
			x_1	x_2
1	5	1:5	11.3860	24.0811
1	4	1:4	13.9744	26.4683
1	3	1:3	17.9808	29.9148
1	2	1:2	25.0017	35.6227
1	1	1:1	40.7611	48.2574
2	1	2:1	60.6795	65.0692
3	1	3:1	73.8432	76.8077
4	1	4:1	83.7244	85.8976
5	1	5:1	91.6578	93.3375

따라서 편미분을 이용한 기존의 방법보다는 비선형방정식의 최적해를 찾는 *fminsearch* 함수를 이용한 새로운 방법이 수치해석 측면에서 보다 쉽고 안정적이며 기존 방법과 같은 결과를 얻을 수 있기 때문에 적합하다고 할 수 있다. 따라서 수리적으로 전문적인 지식이 부족한 실무자라도 예약도착 대기행렬시스템의 기초적인 지식만으로 제시한 모형을 쉽게 활용할 수 있을 것으로 판단되며, 예

약도착 시스템이 필요한 현실의 다양한 분야에 적용하여 시스템을 보다 효율적으로 운용하는데 크게 기여할 수 있을 것이다.

4. 결론 및 향후 연구방향

본 연구에서는 예약도착 대기행렬 모형에서 총 비용을 최소화할 수 있는 최적 예약 간격을 결정할 수 있는 방법론을 살펴보았다. 예약도착 대기행렬 모형의 최적 예약간격 결정 모형은 도착간격이 사전에 알려져 있다는 특징으로 인해 확률적 도착과정을 갖는 모형과 분석방법이 상이하다. 이번 연구에서는 서비스 시간이 지수분포인 경우에 대해서 분석한 기존 연구를 확장하여 현실 설명력이 크고, 응용 가능성이 높은 단계형 분포에 적용할 수 있는 방안을 제시하였다.

실험을 통하여 우리는 제시한 방법의 효용성 4가지를 살펴보았다. 1) 단계형 분포를 적용하여 분석할 경우 불필요한 비용 낭비를 막고, 2) 단계가 세분화될수록 서비스 종료 시점에 대한 예측력이 높아져서 다음 고객의 도착시점을 정확히 제시할 수 있으며, 3) 고객의 대기비용이 상대적으로 높을 경우 고객의 도착간격의 특성에 상관없이 유사한 값 즉, 충분히 여유있는 간격을 주는 것이 유리함을 알 수 있었다. 또한 4) 최적 예약시간 산출시 기존 방법보다는 쉽게 사용할 수 있는 새로운 방법을 제시하여 전문적인 지식이 없는 실무자가 활용하기에 용이한 방법을 제시하였다. 향후 연구는 현재 Variable Interval을 감안하여 고객수가 확장됨에 따라 계산의 복잡성이 증가하여 발생하는 한계를 극복하는 연구가 필요하다.

참 고 문 헌

- [1] 윤봉규, “단계형 확률과정과 국방분야 응용 사례”, 『국방과학기술』, 제1권, 제1호(2008), pp.13-26.
- [2] 윤봉규, “예약 도착 시스템의 최적 예약시간 산출 모형”, 『국방과학기술』, 제5권, 제3호(2012), pp. 37-45.
- [3] Bailey, N., “A Study of Queues and Appointment Systems in Hospitals Outpatients with Special Reference to Waiting Times,” *Journal of the Royal Statistical Society*, Vol.14(1952), pp.185-199.
- [4] Cayirli, T. and E. Veral, “Outpatient Scheduling in Health Care : A Review of Literature,” *Production and Operations Management Society*, Vol.12.(2003), pp.519-549.
- [5] Hassin, R. and S. Mendel, “Scheduling Arrivals to Queues : A Single-Server Model with No-Shows,” *Management Science*, Vol.54, No.3 (2008), pp.565-572.
- [6] Latouche, G. and V. Ramaswami, *Introduction to Matrix Analytic Methods in Stochastic Modeling*, SIAM, Philadelphia, 1999.
- [7] Neuts, M.F., *Matrix-Geometric Solutions in Stochastic Models*, Dover Publications INC., 1981.
- [8] Pegden, C.D. and M. Rosenshine, “Scheduling Arrivals to Queues,” *Computers and Operations Research*, Vol.17, No.4(1990), pp.343-348.
- [9] Stein W.E. and M.J. Cote, “Scheduling Arrivals to a Queue,” *Computers and Operations Research*, Vol.21, No.6(1994), pp.607-614.
- [10] Sun, W., Z. Tian, and N. Tian, “Performance Analysis of the Appointment Systems with No-Shows,” *International Journal of Information and Management Sciences*, Vol.21, No.1 (2010), pp.57-71.
- [11] Vanden Bosch, P.M. and C.D. Dietz, “Minimizing Expected Waiting in Medical Appointment system,” *IIE Transactions*, Vol.32, No.9(2000), pp.841-848.
- [12] Yang, K.K., M.L. Lu, and S.A. Quek, “A New Appointment Rule for a Single-Server, Multiple-Customer Service System,” *Naval Research Logistics*, Vol.45, No.2(1998), pp.313-326.