SEJONG UNIVERSITY
VISION 2030 WORLD TOP100 UNIVERSITY

P.E.L.
ROUGH CONSENSUS & RUNNING CODE

대기행렬의 기초
- 4장. 복수서버 대기행렬 시스템(1) -

2025.07.14.

Jihye Kim
jihye@pel.sejong.ac.kr
Protocol Engineering Lab., Sejong University

CONTENTS

개요

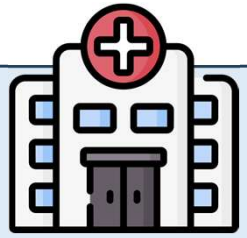
I

A. 복수서버 대기행렬 시스템(Multi-Server Queuing System)(1/2)

- 여러 개의 서버가 고객에게 서비스를 제공하는 대기행렬 시스템
  - 고객들은 서비스를 받기 위해 큐에 줄을 서고, 비어 있는 서버를 통해 서비스를 받은 후에 시스템을 떠남
- 복수서버 대기행렬 시스템은 서비스를 요청하는 고객, 고객이 기다리는 대기열(큐), 서비스를 제공하는 여러 개의 서버로 구성됨

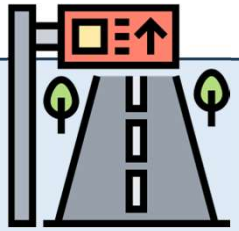
공항 체크인 카운터

- 고객: 비행기 탑승객들
- 큐: 카운터 줄
- 서버: 체크인을 돕는 직원들



병원 접수 창구

- 고객: 진료 받으려는 환자들
- 큐: 접수 창구 앞 줄
- 서버: 진료 접수를 돕는 직원들



고속도로 톨게이트

- 고객: 차량들
- 큐: 톨게이트 진입 줄
- 서버: 톨게이트 수납원들

A. 복수서버 대기행렬 시스템(Multi-Server Queuing System)(2/2)

- 일반적인 복수서버 대기행렬 시스템은 고객의 도착이 포아송분포를 따르고, 고객 서비스시간이 지수분포를 따르며, 서버가 c개인 M/M/c 큐를 활용함

II M/M/c 큐

고객이 시스템에 도착하는 경우,
시스템 상태를 관측하지 않고
무조건 시스템으로 들어오는 M/M/c 큐

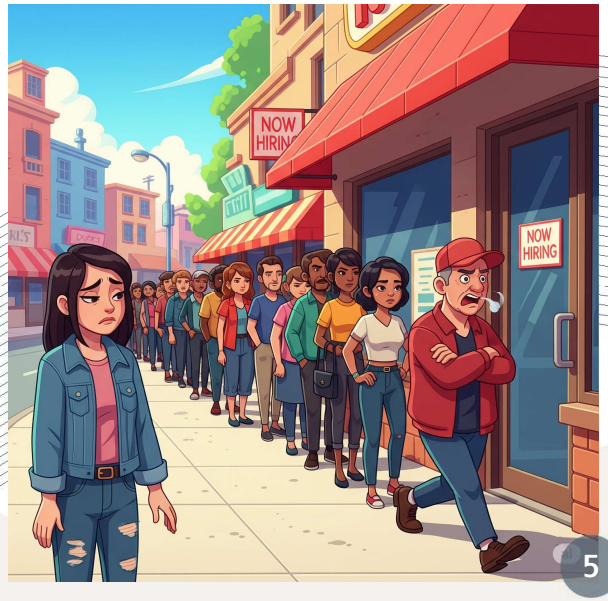
III 고객의 행동을 동반하는 M/M/c 큐

고객이 시스템에 도착하는 경우,
시스템 상태를 관측한 후 자신의 행동
(주저(Balking), 취소(Reneging))을
결정하는 M/M/c 큐

Q. 고객의 행동을 동반하는 M/M/c 큐란?

상황) 유명한 음식점에 갔을 때 손님이 문밖까지 줄을 서서 기다리는 경우,
음식점에 도착한 모든 사람이 그 줄에 무조건 합류하지 않음

1. 주저(Balking): 긴 줄을 보고 바로 발길을 돌림
2. 취소(Reneging): 일단 줄에 합류하여 기다리다가 시간이 지나도 줄이 줄어들지 않아서 더 이상 기다리지 못하고 줄을 이탈하여 다른 곳으로 발길을 돌림



M/M/c 큐

II

II 2 M/M/c 큐

M/M/c
고객의 도착과정 / 고객의 서비스과정 / 서버의 수

A. 개요(1/2)

- 정의
  - 고객 도착과정이 포아송분포를 따르고, 고객 서비스시간이 지수분포를 따르며 서버가 c개이고 버퍼의 크기는 무한한 복수서버 대기행렬 시스템
- 특징
  - 고객이 도착하는 순서대로(FIFO, First-In First-Out) 서비스함
  - 서버는 시스템 내 고객이 존재하는 한 지속적으로 고객을 서비스함
- 켄달(Kendall) 표기방식 기반의 M/M/c 큐
  - 고객 도착과정(Arrival Process): 포아송 분포(Possion Distribution, 표기 M)를 따름
    - α: 고객의 단위시간(Unit Time)당 평균도착률(Average Arrival Rate)
  - 고객 서비스과정(Service Process): 지수분포(M: Markov)를 따름
    - β: 하나의 서버에 의한 고객당 평균서비스율(Average Service Rate)
  - 서버의 수: 동시에 서비스가 가능한 서버가 c개
  - 버퍼의 크기: 고객이 기다리는 버퍼의 크기는 무한대(∞)이므로 표기상 생략됨

*서버군: Group of Servers

(그림 1) M/M/4 큐

M/M/1 큐 시스템 내에는 하나의 서버에 대한 서비스율이 β, 한 서버에 걸리는 부하는 $\rho = \alpha/\beta$ 임

M/M/c 큐 시스템 내에는 서버가 c개 존재하므로, 서비스율은 $c\beta$, 평균 부하는 $\rho = \alpha/c\beta$ 임

II 2 M/M/c 큐

A. 개요(2/2)

병원에는 단 한 명의 의사가 진료를 하는데, 시간당 평균 3명의 손님이 병원에 오고, 의사는 환자당 평균 15분의 진료시간을 소요한다.

- 이 병원은 시간당 고객의 도착률이 3명이고, 서비스율은 $(60/15) = 4$명임
- 병원의 평균 부하는 0.75이며, 의사는 진료시간의 25%를 진료 없이 여유롭게 보낼 수 있음

다른 병원에는 의사가 4명이고, 이 병원에는 시간당 평균 10명의 손님이 오고, 의사는 각자 환자당 평균 15분의 진료시간을 소요한다.

- 이 병원은 시간당 고객의 도착률이 10명이고, 서비스율은 $(60/15) \times 4 = 16$명임
- 병원의 평균 부하는 0.625이며, 각 의사는 진료시간의 37.5%를 진료 없이 여유롭게 보낼 수 있음

*시스템에서 큐에 고객이 쌓이지 않고 안정적으로 동작하기 위해서는 시스템 전체에 걸리는 부하가 1보다 작아야 함: $\rho < 1$, 즉 $\alpha/(c\beta) < 1$ 의 조건을 만족해야 함

B. M/M/c 큐의 상태확률(1/8)

- 시스템 상태확률
  - 시스템 내 고객이 k명인 확률
  - $p_k(t) = P\{N(t) = k\}$
  - $N(t)$: 시간 t에서 시스템 내 고객 수(=버퍼에서 기다리거나 서비스를 받는 중인 고객 수)

- 평행상태(Steady State)에서 시스템 상태확률
  - $\lim_{t\to\infty} p_k(t) = p_k,\ k > 0$
  - 시스템이 시간에 무관하게 일정한 상태를 유지함
  - p_k를 구하기 위해 생성소멸과정(BDP, Birth and Death Process)을 주로 활용함

***평행상태(Steady State)**: 시간이 충분히 지난 후($t \to \infty$)에는 시스템에 들어오고 나가는 고객의 평균이 같아짐에 따라 시간 변화가 시스템에 영향을 미치지 않게 되는 상태

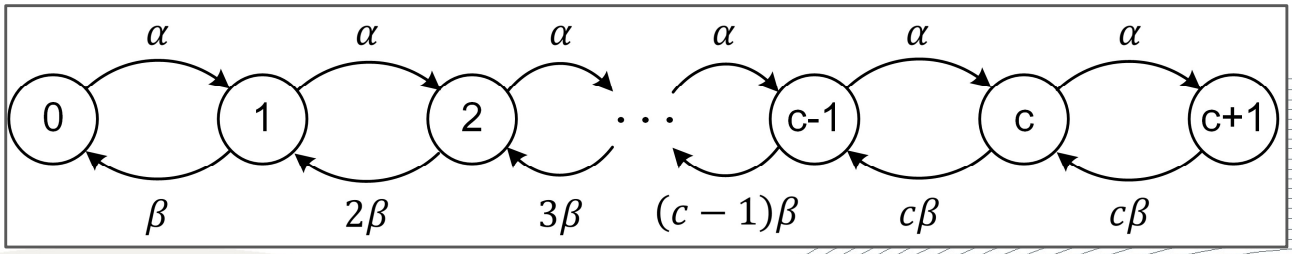
B. M/M/c 큐의 상태확률(2/8)

- 고객 평균도착률 및 평균서비스율
  - 고객 평균도착률(Average Arrival Rate): 시스템 내 고객 수에 무관하게 일정함
    - $\alpha_n = \alpha \ (n = 1, 2, \dots)$
  - 고객 평균서비스율(Average Service Rate): 시스템 내 고객 수에 따라 변동됨
    - $\beta_n = \begin{cases} n\beta, & 0 \le n < c \\ c\beta, & n \ge c \end{cases}$

시간에 따라 다른 상태로
어떻게 변하는지 설명하는 식

- 상태전이방정식(State Transition Equation)
  - 가정사항: M/M/1 큐와 동일하게 고객의 도착 혹은 출발이 동시에 일어나지 않음

$$\alpha p_0 = \beta p_1,$$
$$\alpha p_1 = 2\beta p_2,$$
$$\dots$$
$$\alpha p_{k-1} = k\beta p_k$$



(그림 2) M/M/c 상태전이 구조

- αp_{k-1}: 시스템에 고객 $k-1$명이 들어와서 상태가 k가 될 확률
- $k\beta p_k$: 시스템에 고객 k명이 있다가 서비스를 받고 나가면서 상태가 $k-1$이 될 확률

B. M/M/c 큐의 상태확률(3/8)

- 시스템 상태확률(State Probability)(1/2)
  - $p_k = \begin{cases} \frac{\gamma^k}{k!} p_0, & 0 \le k < c \\ \frac{\gamma^k}{c! c^{k-c}} p_0, & k \ge c \end{cases}$
  - $\gamma = \frac{\alpha}{\beta}$: 시스템에 가해지는 부하(Offered Load)

증명#1

상태:$0 \le k < c$

상태전이방정식에 대해 p_n으로 표기

$$p_1 = \frac{\alpha}{\beta} p_0, \qquad p_2 = \frac{\alpha}{2\beta} p_1 = \frac{\alpha^2}{2\beta^2} p_0,$$
$$p_k = \frac{\alpha}{k\beta} p_{k-1} = \frac{\alpha^2}{k(k-1)\beta^2} p_{k-2}$$

일반식 p_k에 대하여 k를 1씩 감소

$$p_k = \frac{\alpha}{k\beta} p_{k-1} = \frac{\alpha^2}{k(k-1)\beta^2} p_{k-2} = \cdots = \frac{\alpha^k}{k \times (k-1) \times \cdots \times 2 \times 1 \times \beta^k} p_0$$

$\gamma = \frac{\alpha}{\beta}$ 를 대입

$$p_k = \frac{\alpha^k}{k!\,\beta^k} p_0 = \frac{1}{k!} \cdot \left(\frac{\alpha^k}{\beta^k}\right) p_0 = \frac{\gamma^k}{k!} p_0, \qquad 0 \le k < c$$

B. M/M/c 큐의 상태확률(4/8)

- 시스템 상태확률(State Probability)(2/2)
  - $p_k = \begin{cases} \frac{\gamma^k}{k!} p_0, & 0 \le k < c \\ \frac{\gamma^k}{c! c^{k-c}} p_0, & k \ge c \end{cases}$
  - $\gamma = \frac{\alpha}{\beta}$: 시스템에 가해지는 부하(Offered Load)

증명#2

상태: $k \ge c$

서비스율이 $c\beta$로 일정함

$$\alpha p_c = c\beta p_{c+1},$$
$$\alpha p_{c+1} = c\beta p_{c+2},$$
$$\dots$$

$$p_{c+1} = \frac{\alpha}{c\beta} p_c, \qquad p_{c+2} = \frac{\alpha^2}{c^2\beta^2} p_c,$$
$$p_k = \frac{\alpha^{k-c}}{c^{k-c}\beta^{k-c}} p_c = \frac{\gamma^{k-c}}{c^{k-c}} p_c$$

증명#1의 $p_c = \frac{1}{c!}\gamma^c p_0$ 활용

$$p_k = \frac{\gamma^{k-c}}{c^{k-c}} p_c = \frac{\gamma^k}{c!\, c^{k-c}} p_0, \qquad k \ge c$$

B. M/M/c 큐의 상태확률(5/8)

- 시스템 트래픽 부하(Offered Load)
  - $\boldsymbol{\rho} = \frac{\text{고객 도착률}}{\text{고객 서비스율}} = \frac{\alpha}{c\beta} = \frac{\gamma}{c}$

- 시스템 초기 상태확률
  - $\boldsymbol{p_0} = \left(\sum_{k=0}^{c} \frac{\gamma^k}{k!} + \frac{c^c}{c!} \times \frac{\rho^{c+1}}{1-\rho}\right)^{-1}$

증명#3

$$p_0 = \left(\sum_{k=0}^{c} \frac{\gamma^k}{k!} + \sum_{k=c+1}^{\infty} \frac{\gamma^k}{c!\, c^{k-c}}\right)^{-1} = \left(\sum_{k=0}^{c} \frac{\gamma^k}{k!} + \frac{c^c}{c!} \sum_{k=c+1}^{\infty} \rho^k\right)^{-1} = \left(\sum_{k=0}^{c} \frac{\gamma^k}{k!} + \frac{c^c}{c!} \times \frac{\rho^{c+1}}{1-\rho}\right)^{-1}$$

상태확률($\boldsymbol{p_k}$) 활용

트래픽 부하($\boldsymbol{\rho}$) 활용

등비급수 ($\sum_{n=1}^{\infty} ar^{n-1}$)
첫 항: a, 공비: r에서, $|r| < 1$이면 수렴이므로 $\frac{a}{1-r}$ 으로 계산

- 시스템 상태확률
  - $\boldsymbol{p_k} = \begin{cases} \frac{\gamma^k}{k!} p_0, & 0 \le k < c, \\ \frac{c^c \rho^k}{c!} p_0, & k \ge c, \end{cases}$

증명#4

상태: $k \ge c$

초기 상태확률($\boldsymbol{p_0}$)과 트래픽 부하($\boldsymbol{\rho}$) 활용

$$p_k = \frac{\gamma^k}{c!\, c^{k-c}} p_0 = \frac{\gamma^k}{c^k \times c^{-c} \times c!} p_0 = \frac{c^c \rho^k}{c!} p_0$$

B. M/M/c 큐의 상태확률(6/8)

- 새로 도착한 고객이 대기해야 할 확률
  - $P_w = \frac{p_c}{1-\rho}$

n: 고객 수, c: 서버 수, k: 시스템 상태
$k \geq c$인 경우, 새로 도착한 고객이 대기하게 됨

증명#5

상태확률($p_k = \frac{c^c \rho^k}{c!} p_0$, $k \geq c$) 활용

$$P_w = \sum_{k=c}^{\infty} p_k = \sum_{k=c}^{\infty} \frac{c^c}{c!} \rho^k p_0 = \frac{c^c}{c!} \times \frac{\rho^c}{1-\rho} \times p_0$$

등비급수 활용

$\frac{c^c \rho^c}{c!} p_0$는 p_c임

$$P_w = \frac{p_c}{1-\rho}$$

- 시스템 상태가 c인 경우, 새로 도착한 고객이 버퍼에서 대기하는 동안 서비스가 지연되는 현상에 따라 “얼랑 지연공식(Erlang Delay Formula) 또는 얼랑-C 공식(Erlang-C Formula)”이라 함

*얼랑 지연공식(Erlang Delay Formula) 또는 얼랑-C 공식(Erlang-C Formula):
고객이 시스템에 도착했을 때, 서비스를 즉시 받지 못하고 대기할 확률에 대한 공식

$$P_w = \frac{\frac{(c\rho)^c}{c!}}{\sum_{k=0}^{c-1} \frac{(c\rho)^k}{k!} + \frac{(c\rho)^c}{c!} \times \frac{1}{1-\rho}}$$

B. M/M/c 큐의 상태확률(7/8)

예제 4-1

전자회사 서비스센터에는 근무시간에 동시에 서비스하는 서비스요원이 20명이고, 시간당 고객의 도착률은 16명이라고 한다. 그러나 고객이 많이 몰려와서 서비스를 바로 받지 못하고 기다려야 하는 일이 생길 수 있다. 이 회사는 서비스 품질 유지 차원에서 새로 도착하는 고객이 바로 서비스를 받지 못하고 기다릴 확률의 목표치가 1% 미만이도록 서비스를 유지하려 할 때, 서비스센터의 각 서비스요원은 한 명의 고객을 서비스하는 데 걸리는 시간이 적어도 몇 분 이하로 유지되어야 하는지 계산하라.

- 서비스요원: $c = 20$
- 새로 도착한 고객이 대기해야 할 확률: $P_w = 0.01$
- 고객도착률: $\alpha = 16$
- 얼랑-C 공식을 통해 $c = 20, P_w = 0.01$일 때, 이를 단일서버로 환산하여 평균 서비스시간을 구하는 문제
- 얼랑-C표(**부록#2** 참고)에 따라 $\gamma = 10.973$이고, 각 서비스요원이 고객 1명을 서비스하는 데 걸리는 시간은 $\frac{1}{\beta}$을 구하면 됨 ($\gamma = \frac{\alpha}{\beta} < 10.973$에서 $\alpha = 16$임)
  - $\frac{1}{\beta} < \frac{10.973}{16} = 0.686$시간$= 41.15$분

$\therefore$ 평균 41분 이내에 서비스를 마쳐야 함

B. M/M/c 큐의 상태확률(8/8)

문제 4-1

콜센터에는 가장 바쁜 시간대에 시간당 고객 문의전화 도착률이 29건이고, 한 콜당 평균지속시간이 2분이라고 한다. 그러나 고객의 전화문의가 몰려서 서비스를 바로 받지 못하고 기다려야 하는 일도 생길 수 있다. 그래서 이 회사에서는 고객 서비스 품질 유지 차원에서 새로 도착하는 고객이 바로 서비스를 받지 못하고 기다려야 할 확률의 목표치가 1% 미만이 되도록 서비스를 유지하려고 할 때 이 **콜센터는 몇 명의 서비스요원을 준비해야 하는지 계산하라.**

- 새로 도착한 고객이 대기해야 할 확률: $P_w = 0.01$
- 평균 콜 도착률: $\alpha = 29$
- 평균서비스율: $\beta = 30$ (한 콜당 평균서비스시간: $\frac{1}{\beta} = 2$분(=$\frac{2}{60}$시간))
- 시스템 부하: $\gamma = \frac{\alpha}{\beta} = \frac{29}{30} = 0.9667$
- 얼랑-C 공식을 통해 $\gamma = 0.9667, P_w < 0.01$이 되기 위해 필요한 최소 서비스요원의 수를 구하는 문제
- 얼랑-C표(**부록#2** 참고)에 따라 $\gamma > 0.9667$인 값을 찾으면 서비스요원(c)은 최소 5명이 필요함
  - $c = 4$인 경우, 감당 가능한 부하량=0.810, 실제 부하는 $\gamma = 0.9667 > 0.810$이므로 1%가 넘음
  - $c = 5$인 경우, 감당 가능한 부하량=1.259, 실제 부하는 $\gamma = 0.9667 < 1.259$이므로 1% 미만임

  ∴ 최소 5명이 필요함

C. M/M/c 큐의 상태에 대한 기대치(1/4)

- 시스템 내 평균 고객 수(Average Number of Customers in System)
  - $\boldsymbol{E[N] = c\rho + \frac{\rho p_c}{(1-\rho)^2}}$

N는 고객 수, c는 서버 수, ρ는 트래픽 부하, p_c는 고객이 대기하게 될 상태확률

증명#6

$$E[N] = \sum_{n=0}^{\infty} np_n = \sum_{n=0}^{c} np_0 \frac{\gamma^n}{n!} + \sum_{n=c+1}^{\infty} np_c \rho^{n-c} = p_0 \gamma \sum_{n=1}^{c} \frac{\gamma^{n-1}}{(n-1)!} + p_c \sum_{n=1}^{\infty} (n+c)\rho^n$$

$$E[N] = \gamma \sum_{n=0}^{c-1} p_n + p_c \left[\frac{\rho}{(1-\rho)^2} + c\frac{\rho}{1-\rho} \right] = c\rho \left(1 - \frac{p_c}{1-\rho} \right) + \frac{c\rho p_c}{(1-\rho)^2} + \frac{c\rho p_c}{1-\rho} = \boldsymbol{c\rho + \frac{\rho p_c}{(1-\rho)^2}}$$

- 큐 내 고객의 평균 대기시간(Mean Waiting Time of Customers in Queue)
  - $\boldsymbol{E[W_Q] = \frac{p_c}{c\mu(1-\rho)^2}}$

$c\rho$: 평균 서비스 중인 고객 수(서버 하나의 평균 부하가 ρ이고, 서버 c개)

증명#7

$$E[W_Q] = \frac{E[N] - c\rho}{\lambda} = \frac{\boldsymbol{p_c}}{\boldsymbol{c\mu(1-\rho)^2}}$$

리틀의 공식(Little's Law) 활용

$\boldsymbol{E[N] = \lambda \times E[W]}$

(시스템 내 평균 고객 수 = 평균도착률 × 평균체재시간)

**평균 고객 수=서비스 중인 고객 수+대기 중인 고객 수*

파라미터 설명	생성소멸과정	대기행렬
도착률	α	λ
서비스율	β	μ
시스템 부하	$\gamma = \alpha/\beta$	$\rho = \gamma/c$

C. M/M/c 큐의 상태에 대한 기대치(2/4)

버퍼의 수(c)가 2개인 M/M/2 큐에서 생성소멸과정을 통해 해석하면 다음과 같다.

- 고객 평균도착률: $\alpha_n = \alpha, n = 1,2, \ldots$
- 고객 평균서비스율: $\beta_n = \begin{cases} \beta, & n = 1 \\ 2\beta, & n \geq 2 \end{cases}$
- $\gamma = \alpha/\beta$를 활용한 시스템 상태확률: $p_1 = \gamma p_0,\ p_2 = \frac{1}{2}\gamma^2 p_0,\ p_n = 2\left(\frac{\gamma}{2}\right)^n p_0$
- 시스템에 가해지는 부하(Offered Load)가 $\rho = \frac{\alpha}{2\beta} = \frac{\gamma}{2}$일 때 초기 상태확률: $p_0\{1 + 2(\rho + \rho^2 + \cdots)\} = 1$

  즉, $p_0 = \left(1 + 2\frac{\rho}{1-\rho}\right)^{-1} = \frac{1-\rho}{1+\rho}\ (\rho < 1)$
- 시스템 내 고객 수가 n일 상태확률: $p_n = \begin{cases} \frac{1-\rho}{1+\rho}, & n = 0 \\ \frac{2(1-\rho)}{1+\rho}\rho^n, & n > 0 \end{cases}$
- 시스템 내 평균 고객 수: $E[N] = \sum_{n=0}^{\infty} np_n = \sum_{n=1}^{\infty} n\frac{2(1-\rho)}{1+\rho}\rho^n = \frac{2\rho}{1-\rho^2}$
- 고객의 시스템 내 평균 대기시간(Mean Waiting Time): $E[W] = \frac{E[N]}{\alpha} = \frac{2\rho}{\alpha(1-\rho^2)} = \frac{1}{\beta(1-\rho^2)}$

C. M/M/c 큐의 상태에 대한 기대치(3/4)

예제 4-2

고객의 평균도착률은 α이고, 서버의 평균서비스율이 2β인 M/M/1 큐와 같은 평균도착률을 가지되 하나의 서버의 평균서비스율이 β인 M/M/2 큐 중에서 어느 시스템의 고객 체재시간이 더 짧은지 계산하여 비교하라.

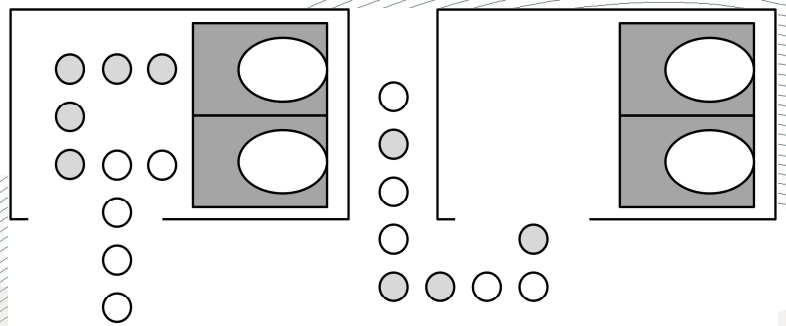
- 평균서비스율 2β인 M/M/1 큐의 시스템 부하: $\rho_1 = \alpha/(2\beta)$
- 하나의 서버의 평균서비스율이 β인 M/M/2 큐의 시스템 부하: $\rho_2 = \alpha/(2\beta)$
- 평균서비스율 2β인 M/M/1 큐의 시스템 내 평균체재시간: $W_1 = \frac{1}{2\beta(1-\rho_1)}$
- 하나의 서버의 평균서비스율이 β인 M/M/2 큐의 시스템 내 평균체재시간: $W_2 = \frac{1}{\beta(1-\rho_2^2)}$
- 두 시스템의 평균체재시간 비교: $W_2 - W_1 = \frac{1}{\beta(1-\rho_2^2)} - \frac{1}{2\beta(1-\rho_1)} = \frac{1}{2\beta(1+\rho)} > 0$
- ∴ M/M/2 큐의 고객 체재시간이 더 짧음
  (e.g., 보통 사람보다 2배나 일을 잘하는 사람 1명이 어떤 일을 마치는 데 걸리는 시간은 보통 사람 2명이 일을 마치는 데 걸리는 시간보다 더 짧음)

*평균체재시간(Mean Sojourn Time): 버퍼에서의 서비스시간과 고객 평균대기시간의 합
*평균대기시간(Mean Waiting Time): 고객이 서비스를 받기 위해 큐에서 기다리는 시간

C. M/M/c 큐의 상태에 대한 기대치(4/4)

(예제 4-2 응용)
변기가 여러 개 있는 화장실에서 줄을 설 때, 2가지 방식을 고려할 수 있다. 이 중 두 번째 방법에서의 대기 시간이 더 짧은 것에 대해 증명하면 다음과 같다.
1) 변기마다 줄을 지어서 서는 방법
2) 모든 사람이 한 줄로 서 있다가 변기 하나가 빌 때 한 명이 들어가는 방법

- 두 방법에서 줄을 서서 기다리는 사람의 수는 동일하다고 가정함
- 방법 1:
  - 하나의 변기에 대한 고객 평균도착률이 $\alpha/2$인 포아송분포를 따름
  - 화장실에서 하나의 변기에 대한 1인당 평균 소비시간은 $1/\beta$이고 지수분포를 따름
  - 시스템 부하는 $\rho_1 = \alpha/2\beta$임. 이는 서버 능력이 2배가 아닌 고객 도착률이 반으로 줄어든 경우이므로 M/M/1 큐 시스템으로 모델링
  - 고객체재시간은 $W_1 = \frac{\rho_1/(1-\rho_1)}{\alpha/2} = \frac{2\rho_1}{\alpha(1-\rho_1)}$
- 방법 2:
  - 고객 평균도착률이 α인 포아송분포를 따름
  - 화장실에서 1인당 평균 소비시간은 $1/\beta$이고 지수분포를 따름
    - 이때 서버가 2개이므로 시스템 전체의 서비스율은 2β임
  - 시스템 부하는 $\rho_2 = \alpha/2\beta$인 M/M/2 큐 시스템으로 모델링
  - 고객체재시간은 $W_2 = \frac{2\rho_2/(1-\rho_2^2)}{\alpha} = \frac{2\rho_2}{\alpha(1-\rho_2^2)}$
- 이때, $W_2 - W_1 = \frac{2\rho}{\alpha}\left(\frac{1}{1-\rho} - \frac{1}{(1-\rho)^2}\right) = -\frac{2\rho^2}{\alpha(1-\rho^2)} < 0$ 이므로 방법 2가 더 짧음



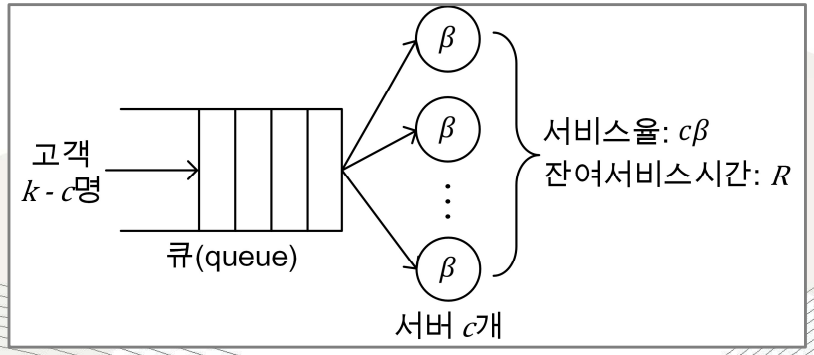
(그림 3) 예제 4-2 응용 그림

D. M/M/c 큐의 대기시간의 분포(1/6)

- 시스템 내 고객의 서비스 대기시간이 임계치 t를 초과할 확률
  - $\boldsymbol{P\{W_Q > t\} = \sum_{k=c}^{\infty} P\{W_Q > t | N = k\} p_k}$
  - $\boldsymbol{W_Q}$: 고객이 시스템 도착 후, 큐에서의 서비스 대기시간(Waiting Time)
  - $\boldsymbol{P\{W_Q > t | N = k\}}$: 고객이 시스템에 도착 당시, 대기 중인 고객 수($\boldsymbol{N}$)가 $\boldsymbol{k}$명임을 알고 있을 때, 해당 고객의 서비스 대기시간($\boldsymbol{W_Q}$)이 $\boldsymbol{t}$보다 클 확률

*전확률이론(Theory of Total Probability)에 의해,
p_k는 고객이 시스템에 도착하기 전에 이미 도착한 고객이 k명 있을 확률임

- 시스템 내 고객의 서비스 대기시간(Waiting Time)(1/2)
  - $\boldsymbol{W_Q = R + \sum_{j=1}^{k-c} W_Q^j}$, $\boldsymbol{k \geq c}$
  - $\boldsymbol{R}$: 서비스 중인 고객 c명의 잔여서비스시간(Residual Service Time)
  - $\boldsymbol{W_Q^j}$: 큐에 있는 $\boldsymbol{j}$번째 고객의 서비스 대기시간(Waiting Time)



서버 $\boldsymbol{c}$개에는 모든 고객이 서비스 중이고,
버퍼에 고객 $\boldsymbol{k - c}$ 명이 서비스 대기 중

(그림 4) M/M/c 큐에서의 고객 서비스 대기

D. M/M/c 큐의 대기시간의 분포(2/6)

- 시스템 내 고객의 서비스 대기시간(Waiting Time)(2/2)
  - 새로운 고객이 시스템의 임의의 시점에 도착했을 때, 서버에서 서비스 중인 고객의 잔여서비스시간(Residual Service Time)은 PASTA(Poisson Arrival See Time Average)의 정리에 의해 평균서비스시간에 근접함

*PASTA(Poisson Arrival See Time Average):
도착과정이 포아송분포를 따르는 시스템은 고객이 지수적인 간격으로 도착함을 의미하며,
도착 고객이 관측한 시스템 상태 분포가 임의의 시점에서 본 시스템 상태 분포와 일치한다는 성질

*지수분포의 기억상실성질(Memoryless Property):
과거 조건이 현재 확률에 영향을 미치지 않는 특성

시간 s동안 서비스하는 중$(X > s)$일 때, 추가로 시간 t가 지나도 서비스가 지속될$(X > t + s)$ 확률은 처음부터 시간 t이상 지속될$(X > t)$ 확률과 동일함

$$P(X > t + s | X > s) = \frac{P(X > t + s)}{P(X > s)} = \frac{e^{-\lambda(t+s)}}{e^{-\lambda}} = \frac{e^{-\lambda t} \cdot e^{-\lambda s}}{e^{-\lambda}} = e^{-\lambda t} P(X > t) = e^{-\lambda t}$$

$$\therefore \boldsymbol{P(X > t + s \mid X > s) = P(X > t)}$$

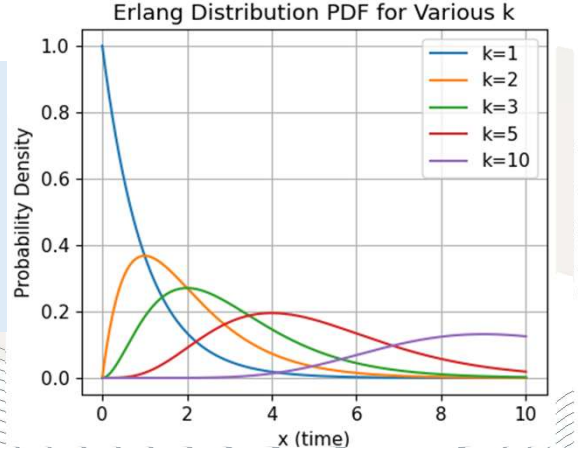
D. M/M/c 큐의 대기시간의 분포(3/6)

- 시스템 내 서버의 고객 서비스시간
  - 각 고객 서비스시간이 평균 $1/\beta$인 지수분포를 따를 때, k명의 독립적인 고객 서비스 시간은 매개변수가 β이며 차수가 k인 얼랑분포를 가짐
  - $E_k(x) = 1 - \sum_{j=0}^{k-1} \frac{(\beta x)^j}{j!} e^{-\beta x}, \quad x \geq 0$
  - x: 독립적인 k명의 고객 서비스시간

*얼랑분포(Erlang Distribution):
고객이 시스템 진입 후 k개의 연속된 서비스를 모두 완료하는 데 걸리는 전체 시간의 분포

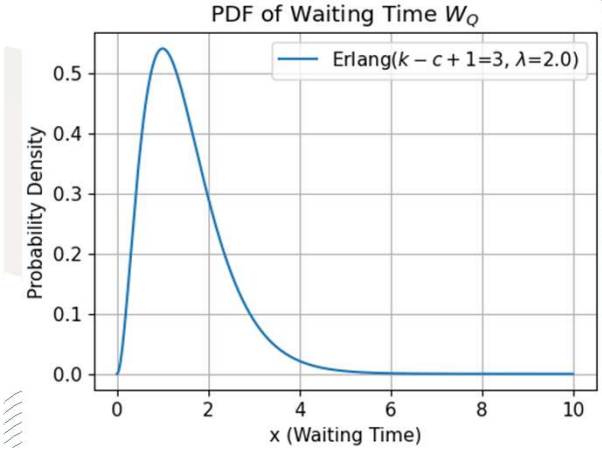
- 서버의 고객 서비스시간의 확률밀도함수(PDF, Probability Density Function)
  - $e_k(x) = \begin{cases} \frac{\beta^k x^{k-1}}{(k-1)!} e^{-\beta x}, & x \geq 0, k \geq 1 \\ \beta e^{-\beta x}, & k = 1 \text{ (지수분포)} \end{cases}$
- 고객의 서비스 대기시간(W_Q) 확률밀도함수(PDF)
  - $e_{k-c+1}(x) = \frac{(c\beta)^{k-c+1} x^{k-c}}{(k-c)!} e^{-c\beta x}, \quad x \geq 0, k \geq c$

큐에서 대기하는 것은 $k \geq c$를 의미함



평균서비스율 $\beta = 1$인 시스템에서, 고객 $k = 2$명의 서비스를 완료하는 데 걸리는 총 시간이 0.5시간 이내일 확률은 약 39.3%이다.

(그림 5) 고객 서비스시간 확률밀도함수



평균서비스율 $\beta = 1$인 시스템에서, $k = 4$명이 있고 서버 $c = 2$개가 있을 때 (2명은 서비스 중, 2명은 대기 중), 새로 도착한 고객의 서비스 대기시간이 2시간 이내일 확률은 약 76.1%이다.

(그림 6) 서비스 대기시간 확률밀도함수

D. M/M/c 큐의 대기시간의 분포(4/6)

- 시스템 내 고객의 서비스 대기시간이 임계치 t를 초과할 확률
  - $\boldsymbol{P\{W_Q > t\} = \sum_{k=c}^{\infty} P\{W_Q > t | N = k\} p_k = \frac{cp_c}{c-\gamma} e^{-(c\beta-\alpha)t},\ t \geq 0}$

증명#8

$$P\{W_Q > t | N = k\} = \int_{x=t}^{\infty} e_{k-c+1}(x)\, dx = \int_{x=t}^{\infty} \frac{(c\beta)^{k-c+1} x^{k-c}}{(k-c)!} e^{-c\beta x}\, dx$$

서비스 대기시간(W_Q) 확률밀도함수 활용

$$P\{W_Q > t\} = \frac{c^c}{c!} p_0 \sum_{k=c}^{\infty} \left[\int_{x=t}^{\infty} \frac{(c\beta)^{k-c+1} x^{k-c}}{(k-c)!} e^{-c\beta x}\, dx \rho^k \right]$$

전확률의 법칙 활용

$$P(A) = \sum_i P(A|B_i) \cdot P(B_i)$$

$\frac{(c\beta)^{k-c+1} x^{k-c}}{(k-c)!}$에서 $c\beta$만 정리하여 $k-c$ 포함 식으로 정리

$$P\{W_Q > t\} = \frac{c^c}{c!} c\beta p_0 \sum_{k=c}^{\infty} \left[\int_{x=t}^{\infty} \frac{(c\beta)^{k-c} x^{k-c}}{(k-c)!} e^{-c\beta x}\, dx \rho^k \right]$$

$c\beta = \frac{\alpha}{\rho}$ 활용

$$P\{W_Q > t\} = \frac{c^c}{c!} c\beta p_0 \sum_{k=c}^{\infty} \left[\int_{x=t}^{\infty} \frac{\left(\frac{\alpha}{\rho}\right)^{k-c} x^{k-c}}{(k-c)!} e^{-c\beta x}\, dx \rho^k \right] = \frac{c^c}{c!} c\beta p_0 \rho^c \sum_{k=c}^{\infty} \left[\int_{x=t}^{\infty} \frac{(\alpha x)^{k-c}}{(k-c)!} e^{-c\beta}\ dx \right]$$

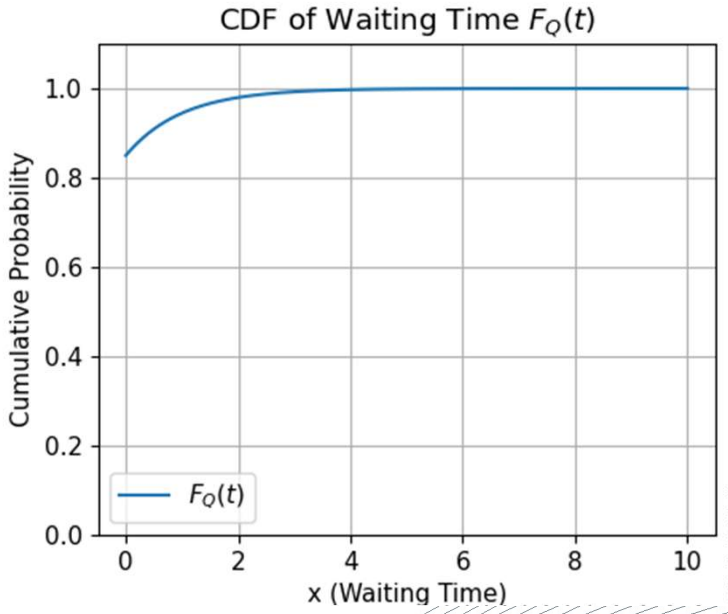
수열 합 기호는 적분기호 안에 들어갈 수 있음

$$P\{W_Q > t\} = \frac{c^c}{c!} c\beta p_0 \rho^c \int_{x=t}^{\infty} \sum_{k=c}^{\infty} \left[\frac{(\alpha x)^{k-c}}{(k-c)!} e^{-c\beta x} \right] dx = c\beta p_c \int_{x=t}^{\infty} e^{-(c\beta-\alpha)x}\, dx = \frac{\boldsymbol{cp_c}}{\boldsymbol{c-\gamma}} \boldsymbol{e^{-(c\beta-\alpha)t}}, \qquad t \geq 0$$

D. M/M/c 큐의 대기시간의 분포(5/6)

- 고객의 서비스 대기시간 누적분포함수(CDF, Cumulative Probability Function)
  - $F_Q(t) = P\{W_Q \le t\} = 1 - \frac{cp_c}{c-\gamma} e^{-(c\beta-\alpha)t},\ t \ge 0$

평균서비스율 $\beta = 1$, 도착률 $\alpha = 2$,
서버 $c = 3$, $p_c = 0.05$인 시스템에서,
새로 도착한 고객이 1분 이내에 서비스를
받게 될 확률은 약 73.6%이다.



(그림 7) 고객 서비스시간 누적분포함수

D. M/M/c 큐의 대기시간의 분포(6/6)

문제 4-2

한 미용실에는 미용 능력이 유사한 미용사가 2명 있다. 이 미용실에는 시간당 평균 2명의 고객이 미용하러 오고, 한 사람당 미용 시간이 평균 20분이다. 그런데 이 미용실에는 미용 시간이 얼마 걸리지 않는 남자 어린이도 오고, 파마를 하는 아주머니도 와서 미용 시간이 각자 다를 수밖에 없다. 이때 한 고객이 미용실에 들어왔는데 그 고객은 1시간 반 뒤에 약속이 있어서 적어도 약속 10분 전에는 미용을 마무리해야 한다. 그 고객이 미용실에 도착한 순간부터 자신의 차례가 올 때까지 기다려야 할 시간이 1시간이 넘을 확률은 얼마일까?

- 미용사 수: $c = 2$
- 고객 도착률: $\alpha = 2$ (명/시간)
- 1인당 평균 서비스시간: 20분, 서비스율: $\beta = 3$ (명/시간)
- 대기시간(W_Q)이 1시간 이상일 확률을 구하는 문제임: $P(W_Q > 1)$
- 시스템 부하: $\gamma = \frac{\alpha}{\beta} = \frac{2}{3},\ \rho = \frac{\gamma}{c} = \frac{1}{3}$
- 시스템 초기 상태확률: $p_0 = \left[\sum_{k=0}^{c} \frac{\gamma^k}{k!} + \frac{c^c}{c!} \times \frac{\rho^{c+1}}{1-\rho}\right]^{-1} \approx 0.6667$
- 시스템 상태확률: $p_c = \frac{c^c \rho^c}{c!} p_0 = 0.1481$
- $P(W_Q > 1) = \frac{c p_c}{c-\gamma} e^{-(c\beta-\alpha)t} = \frac{2\times 0.1481}{2-2/3} e^{-(2\times 3-2)\times 1} \approx 0.00407$
- ∴ 고객이 1시간 이상 기다릴 확률은 약 0.407%이므로, 서비스를 거의 즉시 받을 가능성이 매우 높음

E. M/M/c 큐의 출력과정(1/5)

- 시스템에서의 고객 출력행태(Behavior)
  - 고객 출력행태 관찰을 위해서는 연속하는 두 고객의 출력 사이 간격을 분석해야 함
  - $S_n(t) = P\{N(t) = n$이고 $\tau > t\}$
  - τ: 서로 다른 두 고객의 출력 간격(Interdeparture Time)
  - $N(t)$: 임의의 고객 1명이 나가고 난 직후인 시간 t에서 시스템 상태

- 고객 출력행태(Behavior) 관측시점 (1/2)
  - 관측시점은 보통 시스템 내 한 고객이 서비스를 마친 후 출발하려는 시점이 기준이 됨
  - 2명의 고객이 동시에 시스템에 도착하거나 출발하는 상황은 없다는 가정이 필요함

  - 아주 작은 시간구간 δt를 가정하면 시스템 상태 간 관계식은 다음을 성립함

$$
\begin{aligned}
&S_o(t+\delta t) = S_0(t)(1-\alpha\delta t),\\
&S_n(t+\delta t) = S_{n-1}(t)\alpha\delta t + S_n(t)(1-\alpha\delta t)(1-n\beta\delta t), \qquad n < c,\\
&S_n(t+\delta t) = S_{n-1}(t)\alpha\delta t + S_n(t)(1-\alpha\delta t)(1-c\beta\delta t), \qquad n \geq c
\end{aligned}
$$

  - $S_o(t+\delta t) = S_0(t)(1-\alpha\delta t)$: 시간 t에서 시스템 내 고객이 한 명도 없었고, 시간구간 δt 동안에 한 명의 고객도 도착하지 않아서 시스템 상태가 여전히 0일 확률
  - $S_{n-1}(t)\alpha\delta t$: 시간 t에서 시스템 내 고객이 $n-1$명 있는데, δt 동안 1명이 도착해서 시스템 상태가 n이 된 경우
  - $S_n(t)(1-\alpha\delta t)(1-n\beta\delta t)$: 시간 t에서 시스템 내 고객이 n명 있는데, δt 동안에 1명도 도착하지 않고, 서비스 중인 고객도 서비스를 마치지 않아서 고객 수 변동이 없을 확률

E. M/M/c 큐의 출력과정(2/5)

- 고객 출력행태(Behavior) 관측시점 (2/2)

$$S_o(t+\delta t) = S_0(t)(1-\alpha\delta t),$$
$$S_n(t+\delta t) = S_{n-1}(t)\alpha\delta t + S_n(t)(1-\alpha\delta t)(1-n\beta\delta t), \quad n < c,$$
$$S_n(t+\delta t) = S_{n-1}(t)\alpha\delta t + S_n(t)(1-\alpha\delta t)(1-c\beta\delta t), \quad n \geq c$$

$t \to \infty$일 때 미분 정리를 이용하여 정리

$$\frac{dS_0(t)}{dt} = -\alpha S_0(t),$$
$$\frac{dS_n(t)}{dt} = \alpha S_{n-1}(t) - (\alpha + n\beta)S_n(t), \quad n < c,$$
$$\frac{dS_n(t)}{dt} = \alpha S_{n-1}(t) - (\alpha + c\beta)S_n(t), \quad n \geq c$$

시스템 초기조건($S_n(0) = p_n$)에 따른 일반해 계산 (지수적 감소)

- $S_0(t) = p_0 e^{-\alpha t}$
- $S_n(t) = p_n e^{-\alpha t}$

- 두 고객에 대한 시스템 출발간격이 임계치 t를 초과할 확률
  - $P\{\tau > t\} = \sum_{n=0}^{\infty} S_n(t) = e^{-\alpha t}\sum_{n=0}^{\infty} p_n = e^{-\alpha t}$
  - 고객의 출력간격은 지수분포를 따른다는 것을 의미함
  - 고객의 입력과정이 포아송분포를 따르면, 출력과정도 포아송분포를 따름을 의미함
  - 따라서, 여러 M/M/c 큐가 직렬(Serially)로 연결된 네트워크 큐에 대한 입력이 없는 경우, 모든 큐는 입력과정과 출력과정이 포아송과정으로 모형화될 수 있음

τ: 서로 다른 두 고객의 시스템 출발간격

E. M/M/c 큐의 출력과정(3/5)

*호: 통신시스템에서의 두 장치 간 통신 연결

예제 4-3 (1/3)

고객의 수가 아주 많고 또 채널단위로 고객을 수용하는 무선통신망은 M/M/c 큐를 이용하여 모형화할 수 있다. 이러한 통신시스템에 대한 모델링의 예로 다음과 같은 파라미터를 가정한다.

- 호의 평균도착률: $\alpha = \frac{720\text{호}/\text{시간}}{3600\text{초}/\text{시간}} = 0.2\text{호}/\text{초}$
- 호당 평균채널점유시간: $1/\beta = 180\text{초}$
- 시스템 내 채널 수: $c = 40$
- 시스템에 부과되는 트래픽강도: 평균호도착률과 호당 평균채널점유시간의 곱(단위:얼랑), $\gamma = 0.2180 = 36\ [\text{얼랑}]$
- 시스템 전체에 대한 채널점유도: $\rho = \frac{\gamma}{c} = \frac{36}{40} = 0.9$

위와 같은 가정하에서 다음을 구하라.

(1) 고객이 시스템에 도착하자마자 바로 서비스를 받을 수 있는 확률은 얼마인가?

- 전체확률에서 고객이 기다려야 할 확률을 뺀 확률, 즉 고객이 도착하자마자 바로 서비스받을 수 있는 확률을 구하는 문제임: $1 - E_c(c,\gamma)$
- $E_c(c,\gamma)$는 얼랑-C 공식이며, $E_{40}(40,36) = 1 - \frac{p_c}{1-\rho} \approx 0.5$임
- $\therefore 50\%$

*새로 도착한 고객이 대기해야 할 확률
(=얼랑-C 공식으로도 불림)

$$P_w = \frac{p_c}{1-\rho}$$

E. M/M/c 큐의 출력과정(4/5)

*호: 통신시스템에서의 두 장치 간 통신 연결

예제 4-3 (2/3)

고객의 수가 아주 많고 또 채널단위로 고객을 수용하는 무선통신망은 M/M/c 큐를 이용하여 모형화할 수 있다. 이러한 통신시스템에 대한 모델링의 예로 다음과 같은 파라미터를 가정한다.

- 호의 평균도착률: $\alpha = \frac{720\text{호}/\text{시간}}{3600\text{초}/\text{시간}} = 0.2\text{호}/\text{초}$
- 호당 평균채널점유시간: $1/\beta = 180\text{초}$
- 시스템 내 채널 수: $c = 40$
- 시스템에 부과되는 트래픽강도: 평균호도착률과 호당 평균채널점유시간의 곱(단위:얼랑), $\gamma = 0.2180 = 36$ [얼랑]
- 시스템 전체에 대한 채널점유도: $\rho = \frac{\gamma}{c} = \frac{36}{40} = 0.9$

위와 같은 가정하에서 다음을 구하라.

(2) 고객이 시스템에 도착한 후 서비스를 받기까지 2분 이상 기다려야 할 확률은 얼마인가?

- 한 고객이 자신보다 먼저 도착한 고객의 서비스를 위해 t시간만큼 기다려야 할 확률은 고객의 시스템 내 대기시간이 t보다 클 확률을 의미함: $P\{W_Q > t\} = \frac{cp_c}{c-\gamma} e^{-(c\beta-\alpha)t},\ t \geq 0$
- $c = 40, \gamma = 36, \beta = 1/180, \alpha = 0.2, p_c = P_w(1-\rho)$이고, 여기서 $P_w = E_c(c,\gamma) = 0.5$임 (앞 문제 참고)
- $P\{W_Q > t\} = \frac{cp_c}{c-\gamma} e^{-(c\beta-\alpha)t} = \frac{40\times 0.5}{40-36} e^{-\left(\frac{40}{180}-\alpha\right)120} = 5e^{-\frac{8}{3}} = 0.347$
- $\therefore 34\%$

E. M/M/c 큐의 출력과정(5/5)

*호: 통신시스템에서의 두 장치 간 통신 연결

예제 4-3 (3/3)

고객의 수가 아주 많고 또 채널단위로 고객을 수용하는 무선통신망은 M/M/c 큐를 이용하여 모형화할 수 있다. 이러한 통신시스템에 대한 모델링의 예로 다음과 같은 파라미터를 가정한다.

- 호의 평균도착률: $\alpha = \frac{720\text{호}/\text{시간}}{3600\text{초}/\text{시간}} = 0.2\text{호}/\text{초}$
- 호당 평균채널점유시간: $1/\beta = 180\text{초}$
- 시스템 내 채널 수: $c = 40$
- 시스템에 부과되는 트래픽강도: 평균호도착률과 호당 평균채널점유시간의 곱(단위:얼랑), $\gamma = 0.2180 = 36\ [\text{얼랑}]$
- 시스템 전체에 대한 채널점유도: $\rho = \frac{\gamma}{c} = \frac{36}{40} = 0.9$

위와 같은 가정하에서 다음을 구하라.

(3) 고객의 시스템에 도착한 후 서비스를 받기까지 기다려야 하는 평균대기시간은 얼마인가?

- 고객의 평균대기시간: $E[W]$
- $E[W] = \frac{p_c}{c\beta(1-\rho)^2} = \frac{P_w(1-\rho)}{c\beta(1-\rho)^2} = \frac{P_w}{c\beta(1-\rho)} = \frac{0.5}{\frac{40}{180}(1-0.9)} = 22.5$
- $\therefore 22.5\text{초}$

F. M/M/c/K 큐(1/4)

- 정의
  - 시스템이 수용가능한 고객 수가 K명으로 유한한 M/M/c 큐 시스템

- 특징
  - $K \geq c$여도 K는 유한하므로, 큐에 빈 공간이 없는 경우, 시스템에 도착한 고객이 시스템에 들어오지 못하고 떠나야 하며, 이를 손실(Loss 또는 Blocking)이라고 함
  - 고객 도착 및 서비스과정은 M/M/c 큐와 동일함
  - 큐의 크기가 유한하여 시스템 상태가 K 이하여야 함

- 고객 평균도착률 및 평균서비스율
  - 고객 평균도착률(Average Arrival Rate): $\boldsymbol{\alpha_n = \alpha,\ n = 1, 2, \ldots, K}$
  - 고객 평균서비스율(Average Service Rate): $\boldsymbol{\beta_n = \begin{cases} n\beta, & n < c \\ c\beta, & c \leq n \leq K \end{cases}}$

F. M/M/c/K 큐(2/4)

- 시스템 상태확률(State Probability)
  - $p_k = \begin{cases} \left(\frac{\gamma^k}{k!}\right) p_0, & 0 \le k < c, \\ \left(\frac{c^c \rho^k}{c!}\right) p_0, & c \le k \le K \end{cases}$

고객 도착 및 서비스과정은 M/M/c 큐와 동일하므로,
생성소멸과정에 기반한 상태확률도 동일함

- 시스템 초기 상태확률
  - $p_0 = \left(\sum_{n=0}^{c} \frac{\gamma^n}{n!} + \sum_{n=c+1}^{K} \frac{c^c \rho^K}{c!}\right)^{-1}$

$\gamma = \frac{\alpha}{\beta}, \quad \rho = \frac{\gamma}{c}$

M/M/c 큐 (버퍼 크기가 무한(∞))에서
버퍼 크기만 유한해진 값에 해당함

M/M/c 큐: $p_0 = \left(\sum_{k=0}^{c} \frac{\gamma^k}{k!} + \frac{c^c}{c!} \sum_{k=c+1}^{\infty} \rho^k\right)^{-1}$

F. M/M/c/K 큐(3/4)

- 시스템 내 고객 수가 k명($K \geq k \geq c$)인 경우
  - c명: 서비스를 받는 중인 고객 수
  - $k-c$명: 버퍼에서 대기 중인 고객 수 (버퍼에 들어갈 수 있는 고객 수: 최대 $K-c$명)
  - $K=c$인 경우, 시스템 상태확률은 서버 수와 시스템 내 고객 수가 같은 경우이므로 M/M/c/c 큐라고 부르기도 함

- 시스템이 포화상태인 경우
  - $\boldsymbol{p_c} = \frac{\frac{\gamma^c}{c!}}{\sum_{n=0}^{c}\frac{\gamma^n}{n!}}$
  - 얼랑 손실공식(Erlang Loss Formula) 또는 얼랑-B 공식(Erlang-B Formula)라고 불림
  - 유한한 크기의 버퍼를 가지는 대기행렬시스템의 손실확률을 구할 때 활용됨

***얼랑 손실공식(Erlang Loss Formula) 또는 얼랑-B 공식(Erlang-B Formula):**
고객이 시스템에 도착했을 때, 시스템의 대기 공간이 없어
서비스를 받지 못하고 손실될 확률에 대한 공식

$$p_c = \frac{\frac{\gamma^c}{c!}}{\sum_{n=0}^{c}\frac{\gamma^n}{n!}}$$

F. M/M/c/K 큐(4/4)

예제 4-4

얼랑-B 공식을 이용한 네트워크 설계의 예를 들어보자. 비디오전화의 호 발생 패턴이 포아송분포를 따르고 연결지속시간이 지수함수적으로 분포되어 있다고 가정한다. 표 4-1은 비디오전화 가입자의 특성을 가정한 것이다. 이때 필요한 대역폭은 최소한 얼마 이상이어야 하는가?

<표 4-1> 비디오전화 가입자의 트래픽 파라미터

파라미터	가정치
전체 가입자 수(N)	1,000
최번시 가입자활동률(A)	0.1
호당 평균 연결지속시간$(1/\beta)$	100초
가입자당 평균 호도착률(α)	시간당 1호
연결거부확률 목표치(ε)	1%
하나의 비디오전화가 필요로 하는 대역폭	2 Mbps

- 교환기가 수용가능한 비디오전화의 최대 연결 수: c
- c 이상의 연결은 연결제어기법(Connection Admission Control)을 이용해서 차단된다고 가정함
- 호레벨 서비스품질 요구인 연결거부확률을 목표치 이하로 유지

  하도록 링크를 설계하는 문제는 M/M/c/c을 이용: $E(c,\gamma)=\frac{\frac{\gamma^c}{c!}}{\sum_{i=0}^{c}\frac{\gamma^i}{i!}} \le \varepsilon$
- 호의 평균도착률 $= 1{,}000 \times 0.1 \times 1 = 100$
- 호당 평균지속시간 $= 100$초
- 시스템 부하: $\gamma = (\text{호의 평균도착률} \times \text{호당 평균지속시간})/3600 = (100 \times 100)/3600 = 2.78$
- 비디오전화의 연결거부확률을 1% 이하로 정의하였으므로, 서비스품질 조건을 만족하는 링크 용량 설계 문제는 $E(c,\gamma) \le 0.01$에서 c의 최소값을 찾는 문제임
- 얼랑-B표로부터 $c = 8$임을 알 수 있으나, 한 채널의 비디오전화가 필요로 하는 대역폭이 2 Mbps이므로, 이 비디오전화 시스템이 갖추어야 할 대역폭은 총 $2 \times 8 = 16$ Mbps임

고객의 행동을 동반하는 M/M/c 큐

III

A. M/M/c-B 큐(1/5)

- 정의
  - 고객이 시스템 진입을 주저(Balking)하는 M/M/c 큐 시스템

- 특징
  - 고객의 행동 특성에 따라 다양한 모양의 함수를 가질 수 있음
    - 시스템 내 고객 수가 아주 적어도 고객이 진입을 주저하는 경우도 있고, 고객 수가 상당히 많아도 주저하지 않고 시스템에 진입하는 경우도 있음
    - 대부분은 시스템 내 어느 정도 고객이 있을 때, 새로 도착한 고객이 진입을 주저함

- 고객의 주저확률(Balking Probability)
  - $1 - b_n$
  - $b_n = P\{\text{시간 } t\text{에 도착한 고객이 시스템에 들어감} \mid Q(t) = n\},\ n \geq 0$
    - 시간 t에서 시스템 상태가 $Q(t) = n$인 경우, 고객이 시스템 진입을 결정할 확률

A. M/M/c-B 큐(2/5)

- 고객의 시스템 진입 확률(1/2)
  - 대부분 버퍼에 대기 중인 고객이 없을 경우($n \leq c$) 주저하지 않고 시스템으로 들어가고, 버퍼에 대기 중인 고객이 있을 경우($n > c$) 특정 비율로 시스템에 들어가길 주저함
    - $\boldsymbol{n}$: 시스템 내 고객 수, $\boldsymbol{c}$: 서버 수

  - 경우#1) $n > c$인 경우, 고객의 시스템 진입 확률이 시스템 내 고객 수에 반비례하여 감소함
    - $\boldsymbol{b_n = \begin{cases} 1, & n \leq c, \\ \frac{1}{n-c+2}, & n > c \end{cases}}$
    - e.g., 시스템 내 서버가 2개($n > c$)인 경우, $\boldsymbol{b_n = \frac{1}{n}}$이 되어 시스템 내 고객 수의 역함수가 됨. 즉, $\boldsymbol{b_3 = 0.33, b_4 = 0.25, b_5 = 0.2, \ldots}$

  - 경우#2) $n > c$인 경우, 고객이 시스템 진입 확률이 시스템 내 고객 수에 대해 지수함수적으로 감소함
    - $\boldsymbol{b_n = \begin{cases} 1, & n \leq c, \\ \alpha^{n-c+1}, & n > c, 0 < \alpha < 1 \end{cases}}$
    - e.g., 시스템 내 서버가 2개($n > c$)인 경우, $\boldsymbol{b_n = \alpha^{n-1}}$이 되어 시스템 내 고객 수의 지수함수가 되고, 이 경우에 $\boldsymbol{\alpha < 1}$이므로 $\boldsymbol{b_n}$은 $\boldsymbol{n}$에 대해 지수함수적으로 감소함. 즉, $\boldsymbol{b_3 = 0.25, b_4 = 0.125, b_5 = 0.0625, \ldots}$

A. M/M/c-B 큐(3/5)

- 고객의 시스템 진입 확률(2/2)
  - 대부분 버퍼에 대기 중인 고객이 없을 경우($n \leq c$) 주저하지 않고 시스템으로 들어가고, 버퍼에 대기 중인 고객이 있을 경우($n > c$) 특정 비율로 시스템에 들어가길 주저함
    - n: 시스템 내 고객 수, c: 서버 수

  - 경우#3) $n > c$인 경우, 고객이 시스템에 진입하기를 포기함
    - $b_n = \begin{cases} 1, & n \leq c, \\ 0, & n > c \end{cases}$

구분	그래프 양상	특징	적용 예시
경우#1	역함수적 감소	• 대기열이 길어도 대기하는 상황 • 혼잡이 심해질수록 고객 진입률이 점진적으로 감소함	• e.g., 병원 응급실에서는 대기시간이 길어도 진료받기 위해 대기함. 시스템 내 환자 수가 증가할수록 진료를 즉시 받을 가능성이 감소함에도 불구하고 환자들은 병원 진입을 시도함
경우#2	지수적 감소	• 대기열이 길어지면 대기하는 것을 포기하는 상황 • 혼잡이 조금만 증가해도 진입을 급격히 포기	• e.g., 실시간 스포츠 중계를 시청하고자 할 때, 네트워크 지연이 발생하여 스트리밍 품질이 저하될 경우, 시청자가 스트리밍을 중단하거나 다른 플랫폼으로 이동할 수 있음
경우#3	- (완전 차단)	• 대기열이 존재하지 않는 상황 • 대기할 자리가 없을 시 바로 차단됨	• e.g., 한 지역의 기지국에서 동시 처리 가능한 접속 수는 제한되어 있으나, 특정 시간대에 사용자들이 몰려서 기지국 채널이 모두 점유될 경우, 이후에 접속을 시도하는 사용자는 대기 없이 바로 접속이 거절됨

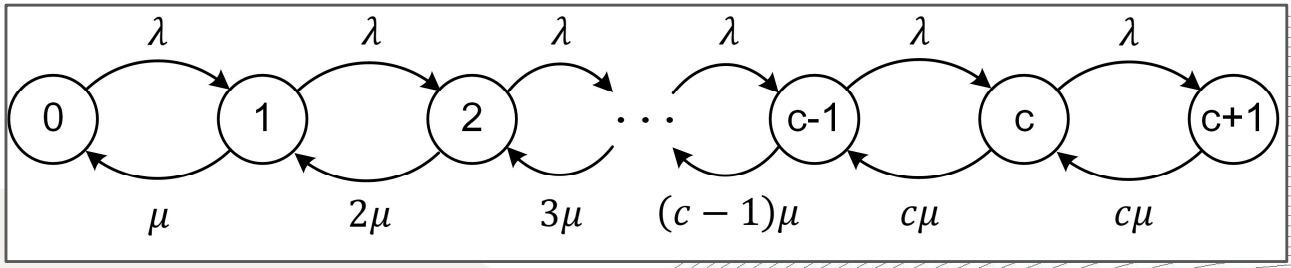
네트워크 설계, 대기 시스템 모델링 시 고객 민감도를 반영해야 하는 경우,
대기열이 존재하는 시스템에서는 경우#1,#2와 같이 진입 확률을 조정하여 시뮬레이션 정확도를 향상할 수 있으나,
5G 무선 접속 시도와 같은 상황에서는 경우#3과 같이 시스템 포화 시 차단하는 구조를 활용할 수 있다.

A. M/M/c-B 큐(4/5)

- 고객 평균도착률 및 평균서비스율
  - 고객 평균도착률(Average Arrival Rate): 시스템 상태에 따라 달라짐
    - $\lambda_n = \begin{cases} \lambda, & n \le c, \\ \lambda b_n, & n > c \end{cases}$
  - 고객 평균서비스율(Average Service Rate): 기존 M/M/c 큐와 동일함
    - $\mu = \begin{cases} n\mu, & 0 \le n \le c, \\ c\mu, & n > c \end{cases}$

- 상태전이방정식(State Transition Equation)
  - M/M/c 큐와 동일하게 고객 도착 및 출발이 동시에 일어나지 않음을 가정함
  - 시스템 상태를 k라고 할 때, $k \le c$에 대한 시스템 상태확률

$$\lambda p_0 = \mu p_1,$$
$$\lambda p_1 = 2\mu p_2, \ldots$$
$$\ldots$$
$$\lambda p_{k-1} = k\beta p_k$$



(그림 8) M/M/c-B 상태전이 구조

- λp_{k-1}: 시스템에 고객 $k-1$명이 들어와서 상태가 k가 될 확률
- $k\beta p_k$: 시스템에 고객 k명이 있다가 서비스를 받고 나가면서 상태가 $k-1$이 될 확률

A. M/M/c-B 큐(5/5)

- 시스템 상태확률(State Probability)

$$\bullet\ \boldsymbol{p_k} = \begin{cases} \left(\frac{\gamma^k}{k!}\right)\boldsymbol{p_0},\ \boldsymbol{k \le c}, \\ \left(\frac{c^c p^k}{c!}\right)\boldsymbol{\beta_k p_0},\ \boldsymbol{k > c} \end{cases}$$

시스템 상태(k)가 서버 수(c)일 때까지는 일반적인 M/M/c 큐와 동일함

증명#9

$0 \le \boldsymbol{k} \le \boldsymbol{c}$

상태전이방정식을 상태 $n > 0$에 대하여 p_n으로 표기

$$p_1 = \frac{\lambda}{\mu}p_0, \qquad p_2 = \frac{\lambda}{2\mu}p_1 = \frac{\lambda^2}{2\mu^2}p_0, \qquad \cdots$$

$\gamma = \frac{\lambda}{\mu}$ 활용

$$p_1 = \gamma p_0, \qquad p_2 = \frac{\lambda}{2\mu}p_1 = \frac{1}{2}\gamma^2 p_0, \qquad \boldsymbol{p_k = \frac{1}{k!}\gamma^k p_0}$$

증명#10

$\boldsymbol{k > c}$

$k = c + 1$인 상태에 대한 상태전이방정식

$$\lambda b_c p_c = c\mu p_{c+1}, \qquad \lambda b_{c+1} p_{c+1} = c\mu p_{c+2}$$

$$p_{c+1} = \frac{b_c}{c}\gamma p_c = \frac{b_c \gamma^{c+1}}{c!\,c}p_0, \qquad p_{c+2} = \frac{b_{c+1}}{c}\gamma p_{c+1} = \frac{b_c b_{c+1}\gamma^{c+2}}{c!\,c^2}p_0$$

$$\lambda b_{k-1} p_{k-1} = c\mu p_k, \qquad p_k = \frac{b_c b_{c+1} \cdots b_{k-1}\gamma^k}{c!\,c^{k-1}}p_0$$

$\rho = \gamma/c$, $\beta_k = b_c b_{c+1} \cdots b_{k-1}$ 활용

$$p_k = \frac{c^c \beta_k \rho^k}{c!}p_0 = \left(\frac{\boldsymbol{c^c p^k}}{\boldsymbol{c!}}\right)\boldsymbol{\beta_k p_0}, \qquad k > c$$

B. M/M/c/K-B 큐(1/4)

- 정의
  - 시스템이 수용가능한 고객 수가 K명으로 유한한 M/M/c-B 큐 시스템

- 상태전이방정식(State Transition Equation)
  - 임의의 $k(k < c)$에 대해 M/M/c-B 큐의 도착 및 서비스 관계가 동일하며, $c < k < K$인 경우에는 다음과 같음
  - $\boldsymbol{\lambda b_{k-1} p_{k-1} = c\mu p_k}$

- 시스템 상태확률(State Probability)
  - $\boldsymbol{p_k = \frac{b_c b_{c+1} \cdots b_{k-1} \gamma^k}{c! c^{k-c}} p_0, \quad c < k \leq K}$

기존 M/M/c-B 큐에서도 활용한 상태확률 $p_n = \frac{\gamma^k}{c! c^{k-c}} p_0$에 진입확률($b_n$)을 곱한 값

- 시스템 초기 상태확률
  - $\boldsymbol{p_0 = \left[\sum_{n=0}^{c} \frac{1}{n!} \gamma^n + \frac{c^c}{c!} \sum_{n=c+1}^{K} \beta_n \rho^n\right]^{-1}}$

β_n: 진입확률(b_n)과 동일

B. M/M/c/K-B 큐(2/4)

- 시스템 상태확률(State Probability)

$\chi_k = b_{k-1} \dots b_1 b_0$을 의미

- $$p_k = \begin{cases} \left(\frac{\chi_k \gamma^k}{k!}\right) p_0, & k \le c, \\ \left(\frac{c^c \chi_k \rho^k}{c!}\right) p_0, & k > c \end{cases}$$

증명#11

$k \le c$

$$\lambda b_0 p_0 = \mu p_1, \qquad \lambda b_1 p_1 = 2\mu p_2, \qquad \lambda b_{k-1} p_{k-1} = k\mu p_k$$

$$p_1 = b_0 \gamma p_0, \qquad p_2 = \frac{1}{2} b_1 \gamma p_1 = \frac{1}{2} b_1 b_0 \gamma^2 p_0, \qquad p_k = \frac{1}{k!} b_{k-1} \dots b_1 b_0 \gamma^k p_0$$

$\chi_k = b_{k-1} \dots b_1 b_0$ 대입

$$\boldsymbol{p_k = \frac{1}{k!} \chi_k \gamma^k p_0}, \qquad k \le c$$

증명#12

$k > c$

상태 $k = c+1$인 상태전이방정식

$$\lambda b_c p_c = c\mu p_{c+1}, \qquad \lambda b_{c+1} p_{c+1} = c\mu p_{c+2}, \qquad \lambda b_{k-1} p_{k-1} = c\mu p_k$$

$$p_{c+1} = \frac{b_c}{c} \gamma p_c, \qquad p_{c+2} = \frac{b_{c+1}}{c} \gamma p_{c+1}, \qquad p_k = \frac{b_{k-1}}{c} \gamma p_{k-1}$$

$$p_{c+1} = \frac{b_c}{c} \gamma p_c = \frac{\chi_{c+1} \gamma^{c+1}}{c!\, c} p_0, \qquad p_{c+2} = \frac{b_{c+1}}{c} \gamma p_{c+1} = \frac{\chi_{c+2} \gamma^{c+2}}{c!\, c^2} p_0,$$

$$p_k = \frac{b_{k-1}}{c} \gamma p_{k-1} = \boldsymbol{\frac{c^c \chi_k \rho^k}{c!} p_0}$$

B. M/M/c/K-B 큐(3/4)

- 시스템 포화상태인 상태확률
  - 고객이 시스템 진입을 거부당할 확률을 의미함 (시스템 상태 $k = K$)
  - $\boldsymbol{p_K = \left(\frac{c^c \chi_K \rho^K}{c!}\right) p_0}$

- 시스템 내 고객 평균 수(Average Number of Customers in System)
  - $\boldsymbol{E[N] = \sum_{k=0}^{K} k p_k}$

- 시스템 내 고객 평균체재시간(Mean Sojourn Time)
  - $\boldsymbol{E[W] = \frac{E[N]}{\lambda(1-p_K)}}$

리틀의 공식(Little's Law) 활용
$\boldsymbol{E[N] = \lambda \times E[W]}$
(시스템 내 평균 고객 수 = 평균도착률 × 평균체재시간)
**평균 고객 수=서비스 중인 고객 수+대기 중인 고객 수*

실제로 시스템에 들어온 고객의 유효 도착률(Effective Arrival Rate),
즉 시스템이 포화상태인 경우를 제외한 실제 유입률로
평균체재시간(Mean Sojourn Time)을 계산해야 함: $\lambda_{eff} = \lambda(1 - p_K)$

B. M/M/c/K-B 큐(4/4)

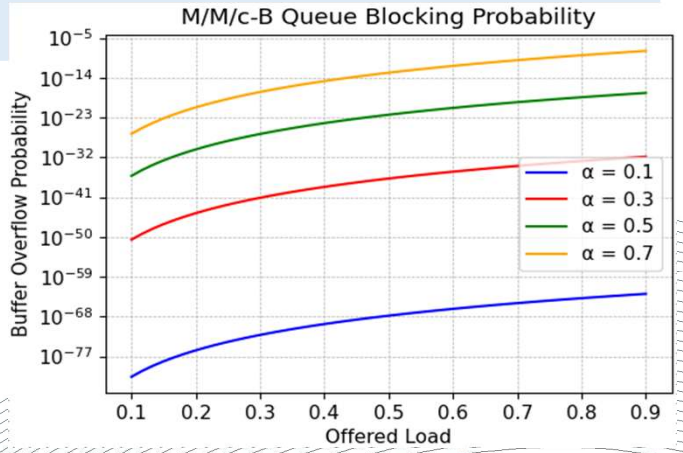
브러킹(Blocking): 고객 수가 K일 경우, 새로 도착한 고객은 버퍼에 빈 공간이 없으므로 시스템 내로 들어올 수 없음

예제 4-5

M/M/c-B 큐에서 각 서버의 서비스율은 1로 정규화된다고 가정하고, $K = 20, c = 10$ 그리고 시스템에 도착한 고객이 시스템 안으로 들어갈 확률인 b_n은 다음과 같이 주어진다고 하자.

$$b_n = \begin{cases} 1, & n \le c, \\ \alpha^{n-c+1}, & n > c, 0 < \alpha < 1 \end{cases}$$

이때 고객의 동작에 대한 파라미터인 $\alpha = 0.1, 0.3, 0.5, 0.7$의 각 경우를 가정하고 시스템의 평균부하가 0.1부터 0.9까지 변화할 경우에 대하여 고객의 브러킹확률(Buffer Overflow Probability)을 구하고, 그 결과가 가지고 있는 물리적 의미를 논하라.



(그림 9) M/M/c-B 큐 브러킹확률

- 브러킹확률(P_K) 즉, 고객 수가 K명일 상태확률: $p_K = \left(\frac{c^c \chi_K \rho^K}{c!}\right) p_0$
- $\chi_k = b_{k-1} \dots b_1 b_0, \quad p_0 = \left[\sum_{n=0}^{c} \frac{\chi_n}{n!} \gamma^n + \frac{c^c}{c!} \sum_{n=c+1}^{\infty} \chi_n \rho^n\right]^{-1}$
- 브러킹확률은 부하의 크기에 민감하게 반응하므로 브러킹확률을 기수가 10인 로그값으로 표시함
  - 그렇게 하지 않는 경우, 작은 부하 값에 대하여 브러킹확률이 거의 모두 0에 근접하게 되어 성능 차이를 구별할 수 없게 됨
- 시스템 부하(Offered Load) 또는 고객의 시스템 안으로 들어갈 확률(b_n)이 증가할수록 브러킹확률(Buffer Overflow Probability)은 높아짐
- 고객이 시스템에 도착했을 때 서버가 모두 서비스 중이어서 고객이 시스템 진입을 주저하는 큐에서는 고객의 동작 관련 파라미터(α)에 대해 대체적으로 브러킹확률(Buffer Overflow Probability)이 낮음
  - 따라서, 고객의 주저행동이 큐에 미치는 효과는 상당히 크다고 할 수 있음

C. M/M/c-R 큐(1/5)

- 정의
  - 시스템 상태와 무관하게 고객이 시스템 진입하여 대기하다가, 시간이 지나도 서비스 차례가 오지 않는 경우에 서비스 받는 것을 중도 포기(Reneging)하는 M/M/c 큐 시스템

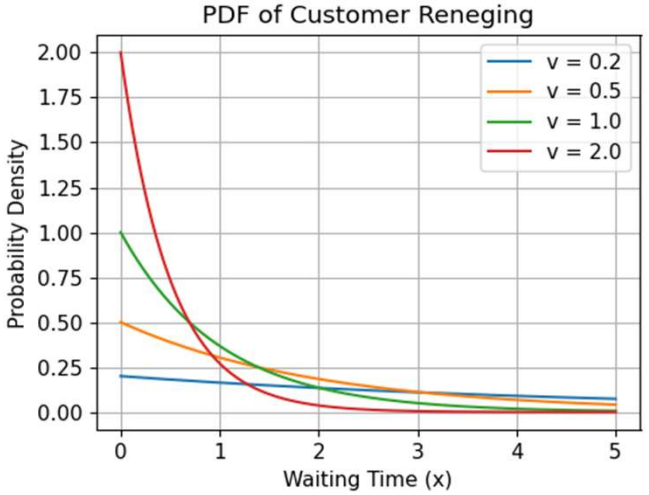
- 고객 중도포기에 관한 확률모형
  - 가정#1) 시스템 내 큐에서 자신의 차례를 기다리는 특정 고객이 시간(d)이 지나도 차례가 오지 않는 경우에 시스템을 떠난다고 가정함
    - 고객의 중도포기 확률은 큐에서 자신보다 앞에 대기중인 고객 수에 따라 지수함수의 형태로 결정되며, 평균 $\boldsymbol{v}$의 속도로 시스템을 떠난다고 가정함
    - $\boldsymbol{v}$: 평균 중도포기율(Reneging Rate)
  - 가정#2) 고객 대기시간이 길어질수록 해당 고객은 큐의 상태가 낮은 위치에 있게 됨
    - 고객의 시스템 도착 당시 큐에 이미 대기중인 고객이 $\boldsymbol{L}$명인 경우, 서버는 먼저 도착한 고객을 순서대로 처리하므로 시간이 지나면 해당 고객이 위치한 큐의 상태가 $\boldsymbol{L}$에서 서버 쪽으로 가까워짐
    - 그러나 해당 고객은 시간이 지날수록 큐에 있을 확률이 낮아짐

위 가정에서 해당 고객의 대기시간(d)과 해당 고객이 위치하고 있는 큐의 상태를 나타내는 변수(x) 간의 관계를 구할 수 있으며, 이로부터 해당 고객의 중도포기에 대한 확률밀도함수 $f(x)$를 구할 수 있음

C. M/M/c-R 큐(2/5)

- 고객 중도포기에 대한 확률밀도함수(PDF)
  - $f(x) = ve^{-vx}$: 고객 지연에 대한 민감도를 의미
    - x가 커질수록 고객이 큐에서 오래 대기한 것으로 간주하며, 고객이 중도포기할 확률은 낮아짐

고객의 평균 중도포기율 $v = 0.5$(시간)인 시스템에서, 고객이 1시간 이내에 중도포기할 확률은 약 39.3%이다.



(그림 10) 고객 중도포기 확률밀도함수

- 고객 중도포기율(Reneging Rate)
  - 시각 t에서 고객 수가 $Q(t) = n$일 때, 고객이 시각 t까지 대기하다가 $(t, t + dt)$ 사이에 시스템을 떠날 확률
  - $P\{t$까지 기다리던 고객이 $(t, t + dt]$사이에 떠남$\big| Q(t) = n\} = \begin{cases} vdt + o(dt), & n > c, \\ 0, & 0 \le n \le c \end{cases}$
  - $Q(t) = n$: 시각 t에서 시스템 내 고객 수
  - c: 시스템 내 서버 수
  - $0 \le n \le c$인 경우, 실제로 큐에서 대기중인 고객이 없으므로 중도포기율은 0임
  - $n > c$인 경우, 큐에 대기중인 고객이 있으므로 중도포기율은 매우 작은 시간구간과 중도포기율의 곱(vdt)으로 표시함

C. M/M/c-R 큐(3/5)

- 고객 평균도착률 및 평균서비스율
  - 고객 평균도착률(Average Arrival Rate): 시스템 상태와 무관함
    - $\boldsymbol{\lambda_n = \lambda,\ n \geq 0}$
  - 고객 평균서비스율(Average Service Rate): 시스템 상태에 따라 달라짐
    - $\mu_n = \begin{cases} n\mu, & 0 \leq n < c, \\ c\mu + (n-c)\nu, & n \geq c \end{cases}$
    - $0 \leq n < c$인 경우, 큐에 대기중인 고객이 없으므로 일반적인 M/M/c 큐와 동일함
    - $n \geq c$인 경우, 큐에 대기중인 고객이 있으므로 중도포기하는 고객이 생길 수 있음
      - 모든 서버가 고객을 서비스 중이므로 서비스율은 최소한 $c\mu$임
      - 중도포기율은 서비스율에 포함되므로, 큐에 대기중인 고객 수인 $n-c$에 평균중도포기율인 ν를 곱한 만큼의 값이 더해져야 함

C. M/M/c-R 큐(4/5)

- 시스템 상태확률(State Probability)
  - $p_k = \begin{cases} \frac{\gamma^k}{k!} p_0, & k \le c, \\ \frac{c^c}{c!} \xi_{k-c} \rho^k p_0, & k > c \end{cases}$
- 시스템 초기 상태확률
  - $p_0 = \left[\sum_{i=0}^{c} \frac{\gamma^i}{i!} + \frac{c^c}{c!} \sum_{j=c+1}^{\infty} \rho^j \xi_{j-e} \right]^{-1}$

증명#13

$0 \le k \le c$

$\gamma = \lambda/\mu$ 적용

$\lambda p_0 = \mu p_1, \quad \lambda p_1 = 2\mu p_2, \quad \lambda p_{k-1} = n\mu p_k$

$p_1 = \gamma p_0, \quad p_2 = \frac{1}{2}\gamma p_1 = \frac{1}{2}\gamma^2 p_0, \quad \boldsymbol{p_k = \frac{1}{k!}\gamma^k p_0}, \quad 0 \le k \le c$

증명#14

$k > c$

상태 $k = c+1, c+2$에 대한 상태전이방정식

$\lambda p_c = (c\mu + v) p_{c+1}, \quad \lambda p_{c+1} = (c\mu + 2v) p_c,$

$p_{c+1} = \frac{\lambda}{c\mu + v} p_c, \quad p_{c+2} = \frac{\lambda^2}{(c\mu + 2v)(c\mu + v)} p_c$

이탈률의 합을 ζ_l로 표기

$\zeta_l = \left((c\mu + v)(c\mu + 2v) \dots (c\mu + lv)\right)^{-1} = \left(\prod_{i=1}^{l} (c\mu + iv) \right)^{-1} = \left((c\mu)^l \prod_{i=1}^{l} \left(1 + \frac{iv}{c\mu}\right) \right)^{-1} = \frac{\xi_l}{(c\mu)^l}$

치환

$\xi_l = \left(\prod_{i=1}^{l} \left(1 + \frac{iv}{c\mu}\right) \right)^{-1}$

$c + l$을 k로 표현

$p_{c+l} = \lambda^l \zeta_l p_c$

$p_k = \lambda^{k-c} \zeta_{k-c} p_c, \quad k > c$

p_c를 대입

$\boldsymbol{p_k = \frac{c^c \rho^k}{c!} \xi_{k-c} p_0}$

상태 $k = c + l$일 때 중도포기율 누적값(ξ_l)

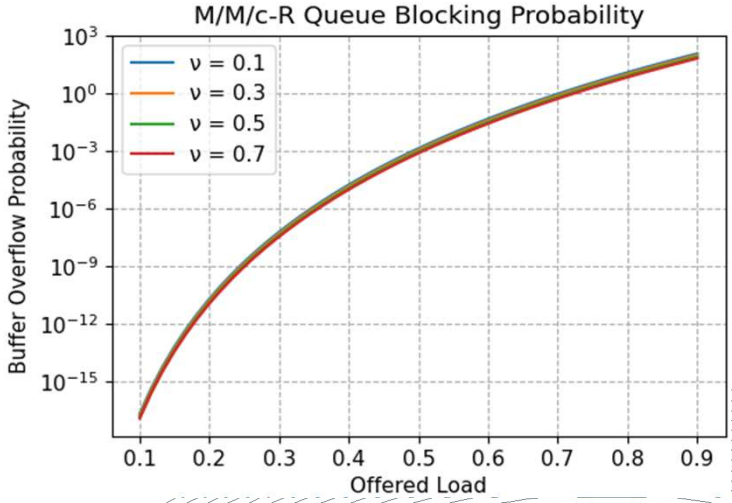
C. M/M/c-R 큐(5/5)

브러킹(Blocking): 고객 수가 K일 경우, 새로 도착한 고객은 버퍼에 빈 공간이 없으므로 시스템 내로 들어올 수 없음

예제 4-6

M/M/c-R 큐에서 각 서버의 서비스율은 1로 정규화된다고 가정하고 $K = 20, c = 10, f(x) = ve^{-v}$ 라고 하자. 이때 $v = 0.1, 0.3, 0.5, 0.7$의 각 경우를 가정하여 평균부하가 0.1부터 0.9까지 변화할 경우에 대하여 고객의 브러킹확률(Buffer Overflow Probability)을 구하고, 그 결과가 가지고 있는 물리적인 의미를 논하라.

- 브러킹확률(P_K) 즉, 고객 수가 K명일 상태확률: $p_k = \frac{c^c}{c!}\xi_{K-c}\rho^K p_0$
- 브러킹확률은 부하의 크기에 민감하게 반응하므로 브러킹확률을 기수가 10인 로그값으로 표시함
  - 그렇게 하지 않는 경우, 작은 부하 값에 대하여 브러킹확률이 거의 모두 0에 근접하게 되어 성능 차이를 구별할 수 없게 됨
- 시스템 부하(Offered Load)가 증가할수록 브러킹확률은 지수적으로 증가하나, 고객의 평균중도포기율(v)이 증가해도 브러킹확률은 크게 변화하지 않음
  - 일단 고객이 큐에 들어온 이상 일정 시간이 지나지 않으면 나가지 않는 성질 때문



(그림 11) M/M/c-R 큐 브러킹확률

- M/M/c-B 큐에 비해 M/M/c-R 큐가 같은 조건에서 높은 브러킹확률을 가짐
  - M/M/c-B 큐는 시스템 서버가 모두 점유된 경우 고객이 아예 시스템 안으로 들어오지 않음
  - M/M/c-R 큐는 시스템 서버가 모두 점유된 상태에서도 고객이 시스템 내로 들어옴

추가예제

IV

추가예제 1

어느 클라우드 서버 인프라에서는 백엔드 처리 서버가 4대로 구성되어 있고, 외부에서 도착하는 요청들을 동시에 처리할 수 있다. 이때 도착하는 요청은 초당 5,000개이며, 각 서버는 평균적으로 하나의 요청을 처리하는 데 0.5ms가 소요된다. 시스템이 안정적인지 분석하고, 평균대기시간과 평균체재시간을 구하여라.

- 평균도착률: $\lambda = 5{,}000\,\text{req/s}$
- 서버 수: $c = 4$
- 평균서비스시간: $0.5\,ms = 0.0005\,s$
- 평균서비스율: $\mu = 2{,}000\,\text{req/s}$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = \frac{5{,}000}{2{,}000} = \frac{5}{2},\ \rho = \frac{\gamma}{c} = \frac{5}{4\times 2} = 0.625$ ($\rho < 1$이므로 안정적임)
- 시스템 초기 상태확률: $p_0 = \left[\sum_{n=0}^{c-1}\frac{\gamma^n}{n!} + \frac{\gamma^c}{c!}\times\frac{1}{1-\rho}\right]^{-1} = \left[\sum_{n=0}^{3}\frac{2.5^n}{n!} + \frac{2.5^c}{4!}\times\frac{1}{1-0.625}\right]^{-1} \approx 0.0737$
- 평균 대기 고객 수: $E[N_Q] = p_0 \times \frac{\gamma^c\times\rho}{c!(1-\rho)^2} = 0.0737 \times \frac{2.5^4\times 0.625}{24\times(1-0.625)^2} \approx 0.533$
- 평균대기시간: $E[W_Q] = \frac{E[N_Q]}{\lambda} = \frac{0.533}{5{,}000} \approx 0.0001066s = 106.6\mu s$
- 평균체재시간: $E[W] = E[W_Q] + \frac{1}{\mu} = 0.0001066 + 0.0005 = 0.0006066s = 606.6\mu s$

추가예제 2

통신사는 1,000명의 기업 고객이 사용하는 VPN 연결 전용 라우터를 구축하고자 한다. 고객은 평균적으로 하루에 3번 VPN 회선을 통해 본사 또는 클라우드로 접속하고, VPN 연결은 평균적으로 30분 동안 유지되며 일정 대역폭을 점유한다. 이때 VPN 연결 요청이 몰릴 때도 연결 거절률이 1% 이하가 되도록 하는 것이 목표이며, 이를 위해 라우터에 필요한 포트 수를 구하여라.

- 평균도착률: $\lambda = \frac{3}{24} \times 1{,}000 = 125$회/시간
- 평균서비스시간: 30분 $= \frac{1}{2}$시간
- 평균서비스율: $\mu = 2$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = \frac{125}{2} = 62.5$
- 연결 거절률이 1% 이하로 정의하였으므로, 얼랑-B 공식을 통해 포트 수(c) 계산: $E(c,\gamma) = \frac{\frac{\gamma^c}{c!}}{\sum_{k=0}^{c}\frac{\gamma^k}{k!}} \leq 0.01$
- 얼랑-B표로부터 $c = 71$일 때, 0.89%로 1% 이하가 됨
- ∴ 최소한 71개의 포트를 갖춘 라우터가 필요함

추가예제 3

이동통신 시스템에서 사용자는 이동 중이며, 특정 지역에서 기지국 A와 기지국 B의 커버리지 경계에 도달하였다. 단말은 먼저 기지국 A로 핸드오버를 시도하며 연결을 요청하고, 기지국 A는 최대 4개의 세션을 동시에 수용할 수 있다. 현재 기지국 A에는 단말이 6명 접속 중이며, 상태에 따라 연결 성공 확률이 낮아진다. 시스템 내 단말 수가 증가할수록 단말이 연결 시도를 즉시 포기할 확률이 증가하며, 이때 단말의 동작에 대한 파라미터를 $\alpha = 0.5$로 가정할 때, 새로 연결을 시도하는 단말이 기지국 A에 연결 시도할 확률과 기지국 B로 우회할 확률은 얼마인가?

- 평균도착률: $\lambda = 10$
- 평균서비스율: $\mu = \frac{1}{20} = 0.05$
- 서버 수: $c = 4$
- 단말 진입 확률: $b_n = \begin{cases} 1, & n \leq 4 \\ \alpha^{n-3}, & n > 4 \end{cases}$
- 기지국 A에 연결 시도할 확률: $b_6 = \alpha^{6-3} = 0.5^3 = 0.125$
- 기지국 B로 우회할 확률: $1 - b_n = 1 - 0.125 = 0.875$
- ∴ 기지국 A에 연결 시도할 확률은 12.5%, 기지국 B로 우회할 확률은 87.5%임

추가예제 4 (1/2)

어느 대규모 IDS 센터에서는 4개의 IDS 분석 서버가 수천 개의 네트워크 센서로부터 실시간으로 침해 탐지 로그를 수집하여 분석한다. 이 센터의 탐지 이벤트는 포아송분포를 따라 초당 평균 300건이 발생하며, 각 IDS 분석 서버는 지수적인 시간 동안 초당 100건의 이벤트를 처리할 수 있다. 이때 이 센터에서의 침해 탐지 로그 분석 서비스를 위한 이벤트의 평균 대기시간과 시스템 내 평균 체재시간을 구하라. 또한, 이 센터의 시스템이 과부하 상태인지 분석하고, 서버의 증설이 필요한지 판단하라.

- 평균도착률: $\lambda = 300$건/초
- 평균서비스율: $\mu = 100$건/초
- 서버 수: $c = 4$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = 3,\ \rho = \frac{\gamma}{c} = \frac{3}{4} = 0.75$ ($\rho < 1$ 이므로 시스템은 안정적인 상태임)
- 시스템 초기 상태확률: $p_0 = \left[\sum_{n=0}^{c-1} \frac{\gamma^n}{n!} + \frac{\gamma^c}{c!} \times \frac{1}{1-\rho}\right]^{-1} = \left[\sum_{n=0}^{3} \frac{3^n}{n!} + \frac{3^c}{4!} \times \frac{1}{1-0.75}\right]^{-1} \approx 0.0377$
- 시스템 상태확률: $p_c = \frac{c^c \rho^c}{c!} p_0 = \frac{4^4 \times 0.75^4}{4!} \times 0.0377 \approx 0.1274$
- 시스템 내 평균 대기 고객 수: $E[L_q] = \frac{\rho p_c}{(1-\rho)^2} = \frac{0.75 \times 0.1274}{(1-0.75)^2} = 1.528$
- 고객 평균 대기시간: $E[W_q] = \frac{E[L_q]}{\lambda} = \frac{1.528}{300} \approx 0.00509$초 $= 5.09$ms
- 시스템 평균체재시간: $E[W] = E[W_q] + \frac{1}{\mu} = 0.00509 + 0.01 = 0.01509$초 $= 15.09ms$

추가예제 4 (2/2)

어느 대규모 IDS 센터에서는 4개의 IDS 분석 서버가 수천 개의 네트워크 센서로부터 실시간으로 침해 탐지 로그를 수집하여 분석한다. 이 센터의 탐지 이벤트는 포아송분포를 따라 초당 평균 300건이 발생하며, 각 IDS 분석 서버는 지수적인 시간 동안 초당 100건의 이벤트를 처리할 수 있다. 이때 이 센터에서의 침해 탐지 로그 분석 서비스를 위한 이벤트의 평균 대기시간과 시스템 내 평균 체재시간을 구하라. 또한, 이 센터의 시스템이 과부하 상태인지 분석하고, 서버의 증설이 필요한지 판단하라.

- 평균대기시간은 5.09ms, 시스템 내 평균 체재시간은 15.09ms이고, 시스템은 안정 상태임
- 서버의 증설 여부를 판단하기 위해서는 시스템 안정성뿐만 아니라 대기 시간이 길거나 대기 중인 고객이 많은 경우, 즉 서비스 품질이 저하되는 경우를 고려 해야 함
- 현재 시스템 평균체재시간은 5.09ms이나 이를 1ms 이하로 서비스를 개선하고자 하는 경우의 시스템 내 평균대기 고객 수: $E[W_q] = \frac{E[L_q]}{\lambda} = 5.09ms, E[L_q] = E[W_q] \times \lambda = 0.001s \times 300 = 0.3$
- 따라서, 서버를 1개 이상 증설하여 총 5개 이상 활용하는 경우, 평균체재시간을 1ms 이하로 유지할 수 있음 (그러나 현재 안정적인 시스템임에 따라 증설이 필수적으로 요구되지 않음)

서버 수(c)	시스템 부하(ρ)	시스템 평균대기시간($E[L_q]$)	평균체재시간($E[W_q]$)
4	0.75	1.53	5.09 ms
5	**0.6**	**0.24**	**0.8 ms**
6	0.5	0.08	0.27 ms

추가예제 5

Oauth 또는 SSO 기반 인증 시스템에서는 사용자의 인증 요청이 집중될 경우, 인증 토큰을 생성하는 서버 클러스터 3대가 병렬로 요청을 처리한다. 사용자 인증 요청은 초당 평균 120건 발생하며, 각 인증 서버는 지수 분포를 따르는 서비스 시간을 갖고 평균적으로 초당 50건의 인증 요청을 처리할 수 있다. 또한, 이 시스템은 서비스의 안전성과 인증 지연 최소화를 동시에 고려하며, 특히 실시간 서비스 품질을 유지하기 위해 전체 인증 처리 지연 시간이 100ms를 초과하지 않을 것이 요구된다. 이때 현재 시스템에서의 요청 처리 평균 대기시간과 안정성을 분석하라. 또한, 인증 지연 100ms 초과를 방지하기 위해 인증 서버를 얼마나 증설해야 하는가?

- 평균도착률: $\lambda = 120$요청/초
- 평균서비스율: $\mu = 50$요청/초
- 서버 수: $c = 3$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = \frac{120}{50} = 2.4,\ \rho = \frac{\gamma}{c} = \frac{2.4}{3} = 0.8$ ($\rho < 1$ 이므로 시스템은 안정적임)
- 시스템 초기 상태확률: $p_0 = \left[\sum_{n=0}^{c-1} \frac{\gamma^n}{n!} + \frac{\gamma^c}{c!} \times \frac{1}{1-\rho}\right]^{-1} = \left[\sum_{n=0}^{2} \frac{2.4^n}{n!} + \frac{2.4^3}{3!} \times \frac{1}{1-0.8}\right]^{-1} \approx 0.0562$
- 시스템 상태확률: $p_c = \frac{c^c \rho^c}{c!} p_0 = \frac{3^3 \times 0.8^3}{3!} \times 0.0562 \approx 0.1296$
- 시스템 내 평균 대기 고객 수: $E[L_q] = \frac{\rho p_c}{(1-\rho)^2} = \frac{0.8 \times 0.1296}{(1-0.8)^2} \approx 2.592$
- 고객 평균 대기시간: $E[W_q] = \frac{E[L_q]}{\lambda} = \frac{2.592}{120} \approx 0.02162$초 $= 21.6$ms
- 시스템 평균체재시간: $E[W] = E[W_q] + \frac{1}{\mu} = 0.02162 + 0.02 = 0.0416$초 $= 41.6ms$
- 전체 인증처리 지연 시간이 100ms를 초과하지 않는다는 것은 $E[W] < 100ms$를 의미하며, 현재 $E[W] = 41.6$ms이므로 서버 증설이 필요하지 않음

추가예제 6 (1/2)

기업의 네트워크 인프라에는 다수의 원격지 클라이언트들이 동시에 VPN 연결을 요청하는 상황이 발생한다. 이 시스템은 사용자의 접속 요청을 처리하기 위해 병렬 VPN 채널을 운영하며, 평균적으로 분당 8건의 VPN 세션 연결 요청이 도착한다. 각 세션은 평균 1분 동안 유지되며, 시스템은 최대 10개의 VPN 채널을 동시에 할당할 수 있다. 이 시스템에서 평균 세션 수와 평균 대기시간을 구하고, 사용자가 도착했을 때 즉시 채널에 할당되는 서비스 가능률을 계산하라.

- 평균도착률: $\lambda = 8$건/분
- 평균서비스율: $\mu = 1$건/분
- 서버 수: $c = 10$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = 8,\ \rho = \frac{\gamma}{c} = \frac{8}{10} = 0.8$ ($\rho < 1$ 이므로 시스템은 안정적임)
- 시스템 초기 상태확률: $p_0 = \left[\sum_{n=0}^{c-1} \frac{\gamma^n}{n!} + \frac{\gamma^c}{c!} \times \frac{1}{1-\rho}\right]^{-1} = \left[\sum_{n=0}^{9} \frac{8^n}{n!} + \frac{8^{10}}{10!} \times \frac{1}{1-0.8}\right]^{-1} \approx 0.000372$
- 시스템 상태확률: $p_c = \frac{c^c \rho^c}{c!} p_0 = \frac{10^{10} \times 0.8^{10}}{10!} \times 0.000372 = 0.0415$
- 고객 평균 대기시간: $E[W_q] = \frac{p_c}{c\mu(1-\rho)^2} = \frac{0.0415}{10 \times 1 (1-0.8)^2} = 0.5187$분
- 고객 평균체재시간: $E[W] = E[W_q] + \frac{1}{\mu} = 0.5187 + 1 = 1.5187$분
- 시스템 내 평균 세션 수: $E[N] = \lambda \times E[W] = 5 \times 1.5187 = 7.5935$
- 시스템 내 평균 대기 세션 수: $E[L_q] = \lambda \times E[W_q] = 5 \times 0.5187 = 2.5935$

추가예제 6 (2/2)

기업의 네트워크 인프라에는 다수의 원격지 클라이언트들이 동시에 VPN 연결을 요청하는 상황이 발생한다. 이 시스템은 사용자의 접속 요청을 처리하기 위해 병렬 VPN 채널을 운영하며, 평균적으로 분당 8건의 VPN 세션 연결 요청이 도착한다. 각 세션은 평균 1분 동안 유지되며, 시스템은 최대 10개의 VPN 채널을 동시에 할당할 수 있다. 이 시스템에서 평균 세션 수와 평균 대기시간을 구하고, 사용자가 도착했을 때 즉시 채널에 할당되는 서비스 가능률을 계산하라.

- 새로 도착한 고객이 대기해야 할 확률: $P_w = \frac{p_c}{1-\rho} = \frac{0.0415}{1-0.8} = 0.2075$
- 서비스 가능률: $1 - P_w = 0.7925$
- ∴ 평균 세션 수는 약 7개이고, 평균 대기시간은 0.5분이며, 서비스 가능률은 약 79%임
  추가적으로, 새로운 고객이 원격 접속을 원하는 경우, 연결을 위해 대기해야 할 확률은 약 20%로 해석할 수 있음

추가예제 7

기업의 VPN 전용 게이트웨이는 동시에 처리할 수 있는 세션 수에 제한이 있는 시스템으로, 사용자의 VPN 접속 요청이 폭주할 경우에 연결 실패가 발생할 수 있다. 이 시스템은 병렬 TLS 세션 서버를 운영하며, 동시에 최대 6개의 VPN 세션만 유지할 수 있다. 이때 VPN 연결 요청은 포아송 분포를 따르며, 평균 도착률은 분당 20건이다. 각 세션은 평균적으로 5분간 유지되며, 병렬로 동작 가능한 서버의 수는 5개이다. 이 시스템에서 사용자가 세션 연결에 실패할 확률과 트래픽 급증을 대비한 세션 연결 사전 차단율을 구하라.

- 평균도착률: $\lambda = 20$건/분
- 평균서비스율: $\mu = \frac{1}{5} = 0.2$건/분
- 서버 수: $c = 5$, 서버 최대 수용량: $K = 6$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = \frac{20}{0.2} = 100$
- 이 시스템은 최대 6개의 세션까지 수용 가능하므로, 7번째 요청은 연결에 실패하게 됨. 따라서, 시스템 포화상태일 때의 얼랑-B 공식을 활용하여 세션 연결 실패 확률을 구할 수 있음
- 얼랑-B 공식(시스템 포화상태): $p_c = \frac{\frac{\gamma^c}{c!}}{\sum_{n=0}^{c}\frac{\gamma^n}{n!}} = \frac{\frac{100^6}{6!}}{\sum_{n=0}^{6}\frac{100^n}{n!}} \approx 0.941$
- 트래픽 급증을 대비한 세션 연결 사전 차단을 위해, 6번째 요청부터 연결에 실패하게 함으로써 시스템 폭주를 방지할 수 있음. 따라서, 최대 5개의 세션 수용을 시스템 포화상태로 간주하고 얼랑-B 공식을 활용하여 사전 차단율을 구할 수 있음
- 얼랑-B 공식: $p_c = \frac{\frac{\gamma^c}{c!}}{\sum_{n=0}^{c}\frac{\gamma^n}{n!}} = \frac{\frac{100^5}{5!}}{\sum_{n=0}^{5}\frac{100^n}{n!}} \approx 0.945$
- ∴ 사용자의 세션 연결 실패 확률은 약 94.1%이고, 세션 연결 사전 차단율은 약 94.5%가 됨

추가예제 8

어느 보안 관제센터에서는 이벤트 로그, 경고 메시지, 탐지 결과 등 다양한 형태의 보안 로그가 초당 50건의 비율로 수집되고 있다. 수집된 로그는 2대의 서버가 병렬로 처리하여 데이터베이스에 저장하거나 실시간 분석하며, 각 서버는 초당 25건의 로그를 처리한다. 그러나 시스템에 유입되는 로그를 일시적으로 저장하는 버퍼의 크기는 10으로 제한되어 있으므로, 버퍼에 빈 공간이 없으면 수집된 로그는 유실될 수 있다. 이때 시스템에서의 로그 유실 확률을 분석하고, 버퍼의 수용량을 얼마나 늘려야 유실률을 1% 이하로 만들 수 있는지 구하여라.

- 평균도착률: $\lambda = 50$건/초
- 평균서비스율: $\mu = 25$건/초
- 서버 수: $c = 2$, 서버 최대 수용량: $K = 10$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = 2$
- 로그 유실 확률은 시스템에 $K = 10$명 이상 있을 때 새로 도착한 요청이 버려질 확률이므로, 얼랑-B 공식을 변형하여 활용할 수 있음: $p_K = \frac{\frac{\gamma^K}{c!c^{K-c}}}{\sum_{n=0}^{c-1}\frac{\gamma^n}{n!}+\sum_{n=c}^{K}\frac{\gamma^n}{c!c^{n-c}}} = \frac{\frac{2^{10}}{2!2^{10-2}}}{\sum_{n=0}^{1}\frac{2^n}{n!}+\sum_{n=2}^{10}\frac{2^n}{2!2^{n-2}}} \approx 0.0952$
- 로그 유실 확률이 1% 이하($p_K \le 0.01$)가 되기 위해서는 K를 1씩 증가하며 계산해봐야 함

서버 최대 수용량(K)	로그 유실률(p_K)
20	0.04878
…	…
99	0.01005
100	0.00995

- ∴ 로그 유실 확률은 약 9%, 현재 시스템에서 유실률을 1% 이하로 만들기 위해서는 버퍼의 수용량이 100개 이상이어야 함

추가예제 9

어느 시스템에서는 정상 사용자와 공격자가 섞인 형태로 초당 1,000건의 연결 요청이 유입되고 있다. 이 연결은 5개의 병렬 웹 서버에 의해 처리되며, 각 서버는 초당 200건의 요청을 처리할 수 있다. 요청은 최대 10건까지만 동시에 수용할 수 있으며, 초과되는 요청은 즉시 차단된다. 이때 해당 시스템이 포화상태일 경우의 손실율(Blocking Probability)을 구하고, 서버 수와 최대 수용 연결 수 중 어떤 값을 증설하는 것이 유실률을 더 효과적으로 낮출 수 있는지 분석하라.

- 평균도착률: $\lambda = 1{,}000$요청/초
- 평균서비스율: $\mu = 200$요청/초
- 서버 수: $c = 5$, 서버 최대 수용량: $K = 10$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = 5$
- 시스템 포화상태일 때 연결 요청 손실확률(얼랑-B 공식 변형):

$$p_K = \frac{\frac{\gamma^K}{c!c^{K-c}}}{\sum_{n=0}^{c-1}\frac{\gamma^n}{n!}+\sum_{n=c}^{K}\frac{\gamma^n}{c!c^{n-c}}} = \frac{\frac{5^{10}}{5!5^{10-5}}}{\sum_{n=0}^{4}\frac{5^n}{n!}+\sum_{n=5}^{10}\frac{5^n}{5!5^{n-5}}} \approx 0.1175$$

- 유실률을 낮추기 위해서는 연결 요청 손실확률(p_K)뿐만 아니라 실제 유효 도착률($\lambda_{eff} = \lambda(1-p_K)$)도 비교해야 함
  - 유효 도착률은 시스템이 실제로 받아들인 요청의 수임
  - 연결 요청 손실확률만 고려했을 때 A 시스템에서 $p_K = 10\%$, B 시스템에서 $p_K = 8\%$이더라도, 실제로 A 시스템에서 더 많은 요청을 처리했을 수 있기 때문임
- ∴ 연결 요청 손실율은 약 11%이고, 서버 수를 증설하는 것이 유실률을 효과적으로 낮출 수 있음

서버 수 (c)	최대 수용 연결 수(K)	손실확률 (p_K)	유효 도착률 (λ_{eff})
5	10	0.1175	882.5
6	10	0.0618	938.18
7	10	0.0369	963.14
5	**12**	0.0951	904.86
5	**14**	0.0799	920.07

- 위 표에서와 같이, 서버 수를 증설하는 것이 최대 수용 연결 수를 증설하는 것보다 연결 요청 손실확률도 낮고, 실제 유효 도착률도 높은 것을 확인할 수 있음

추가예제 10

어느 CDN 서버는 사용자들의 실시간 영상 스트리밍 요청을 처리한다. 사용자의 요청은 초당 평균 200건의 포아송 도착 과정을 따르며, 각 서버는 평균적으로 0.02초의 시간이 소요된다. CDN 서버는 4개의 병렬 처리 유닛을 가지며, 한 번에 최대 10개의 요청을 수용할 수 있다. 이때 사용자들은 서버 상태에 따라 접속 여부를 판단하며, 서버에 대기 중인 요청이 적으면 대부분의 사용자가 접속을 시도하지만, 그렇지 않은 경우에는 일부 사용자가 접속 시도를 포기하는 경우도 있다. 이 시스템에서 사용자가 접속을 시도하는 확률은 서버의 상태(n)에 따라 다음과 같이 정의되고, 사용자의 행동 파라미터는 $\alpha = 0.7$이다.

$$b_n = \begin{cases} 1, & n \le c \\ \alpha^{n-c}, & n > c \end{cases}$$

이때 사용자가 시스템에 진입할 확률을 구하고, 행동 파라미터 값이 작아질수록 어떤 효과가 있는지 설명하라.

- 평균도착률: $\lambda = 200$요청/초
- 평균서비스율: $\mu = 50$요청/초
- 서버 수: $c = 4$, 서버 최대 수용량: $K = 10$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = 4, \rho = \frac{\gamma}{c} = 1$
- 시스템 초기 상태확률: $p_0 = \left[\sum_{n=0}^{c} \frac{\gamma^n}{n!} + \frac{c^c}{c!}\sum_{n=c+1}^{K} \chi_n \rho^n\right]^{-1} \approx 0.0212$
- 고객 수가 K명일 상태확률: $p_K = \left(\frac{c^c \chi_K \rho^K}{c!}\right) p_0 = \frac{4^4 \alpha^{(10-4)+(9-4)+\cdots+1+1+1+1} 1^{10}}{4!} \times 0.0212 \approx 0.000122$
- ∴ 약 0.012%의 확률로 고객이 진입함. 또한, 행동 파라미터(α)에 기반하여 시스템 내 고객 수가 지수함수적으로 감소하는 구조임을 확인할 수 있고, 시스템이 혼잡 상태일수록 사용자가 진입을 급격히 포기함에 따라 과도한 요청이 유입되는 것을 억제하고 시스템을 안정적으로 운영할 수 있게 됨

추가예제 11

WAF (Web Application Firewall)은 초당 150건의 웹 요청을 받고 있으며, 평균적으로 요청 1건을 처리하는 데 0.01초가 소요된다. WAF는 병렬적으로 2개의 요청을 처리할 수 있고, 최대 6건의 요청을 수용할 수 있다. 이 시스템에서 공격자는 시스템 내 응답이 늦어지면 해당 경로를 포기하고 다른 엔드포인트로 우회하는 행동을 보인다. 공격자의 행동은 시스템 상태(n)에 따라 다음과 같이 정의된다.

$$b_n = \begin{cases} 1, & n \le c \\ \dfrac{1}{n-c+2}, & n > c \end{cases}$$

이때 공격자의 시스템 진입확률이 상태별 확률 $p_n(0 \le n \le 6)$을 구하고, 각 상태에서의 진입 확률을 이용하여 공격자의 실제 유효 도착률을 계산하라. 또한, 이를 기반으로 공격자가 우회하게 될 확률을 구하라.

- 평균도착률: $\lambda = 150$요청/초
- 평균서비스율: $\mu = 100$요청/초
- 서버 수: $c = 2$, 서버 최대 수용량: $K = 6$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = 1.5, \ \ \rho = \frac{\gamma}{c} = 0.75$
- 시스템 초기 상태확률: $p_0 = \left[\sum_{n=0}^{c} \frac{\gamma^n}{n!} + \frac{c^c}{c!}\sum_{n=c+1}^{K} \chi_n \rho^n\right]^{-1} \approx 0.2520$
- 전체 진입확률: $\sum_{n=0}^{6} b_n \times p_n = 0.9410$
- 실제 유효 도착률: $\lambda_{eff} = \lambda \times 0.9410 = 141.15$
- 공격자가 시스템 진입을 포기하고 우회하게 될 확률: $1 - 0.941 = 0.059$
- ∴ 공격자는 요청 중 약 5.9%를 시스템이 혼잡하다고 판단하여 우회하게 됨. 따라서, WAF 시스템이 일정 수준의 자체 방어 기능을 수행하며, 혼잡도가 높아질수록 공격 트래픽 일부가 차단되는 구조로 구성됨을 알 수 있음

추가예제 12

어느 스트리밍 서버는 초당 평균 80건의 사용자 재생 요청을 받고 있으며, 각 요청은 평균적으로 0.025초의 세션을 유지한다. 서버는 2개의 병렬 처리 유닛을 가지며, 최대 8건의 요청까지 수용할 수 있다. 이 서비스의 사용자는 서버 대기열이 길어질수록 접속을 포기하는 경향을 보이며, 시스템 상태(n)에 따라 다음과 같은 진입 확률은 가진다.

$$b_n = \begin{cases} 1, & n \le c \\ \dfrac{1}{n-c+2}, & n > c \end{cases}$$

이때 사용자의 전체 진입 요청 중 서버로 들어오는 평균 요청 수를 구하고, 서비스 이탈률을 예측하라.

- 평균도착률: $\lambda = 80$요청/초
- 평균서비스율: $\mu = 40$요청/초
- 서버 수: $c = 2$, 서버 최대 수용량: $K = 8$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = 2,\ \rho = \frac{\gamma}{c} = 1$
- 시스템 초기 상태확률: $p_0 = \left[\sum_{n=0}^{c} \frac{\gamma^n}{n!} + \frac{c^c}{c!}\sum_{n=c+1}^{K} \chi_n \rho^n\right]^{-1} \approx 0.1257$
- 전체 진입확률: $\sum_{n=0}^{8} b_n \times p_n = 0.7325$
- 서버로 들어오는 평균 요청 수, 즉 실제 유효 도착률: $\lambda_{eff} = \lambda \times 0.7325 = 58.6$
- 서비스 이탈률: $1 -$ 전체 진입확률 $= 1 - 0.7325 = 0.2675$
- ∴ 서버로 들어오는 평균 요청 수는 약 58개이고, 서비스 이탈률은 약 26.75%임

추가예제 13

IoT 센서들이 클라우드 서버로 데이터를 전송하는 시스템이 있다. 이 시스템에서는 초당 평균 50건의 데이터 전송 요청이 발생하며, 각 요청은 평균적으로 초당 25건의 속도로 처리된다. 서버는 병렬로 3개의 요청을 동시에 처리할 수 있으며, 대기열은 최대 4건의 요청을 추가로 수용할 수 있고, 시스템 전체의 수용 가능 요청 수는 7건이다. 이때 센서들은 서버 상태를 확인하고 대기열이 길어질수록 진입을 포기하게 되며, 행동 파라미터가 $\alpha = 0.6$일 때 진입 확률은 시스템 상태(n)에 따라 다음과 같다.

$$b_n = \begin{cases} 1, & n \le c \\ \alpha^{n-c}, & n > c \end{cases}$$

이때 시스템에 실제로 진입하는 유효 도착률을 계산하고, 시스템 포화상태에서의 평균 진입 거절 확률을 분석한 후, α값의 변화가 시스템 부하에 미치는 영향을 설명하라.

- 평균도착률: $\lambda = 50$건/초
- 평균서비스율: $\mu = 25$건/초
- 서버 수: $c = 3$, 서버 최대 수용량: $K = 7$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = 2,\ \rho = \frac{\gamma}{c} = \frac{2}{3}$
- 시스템 초기 상태확률: $p_0 = \left[\sum_{n=0}^{c} \frac{\gamma^n}{n!} + \frac{c^c}{c!}\sum_{n=c+1}^{K} \chi_n \rho^n\right]^{-1} \approx 0.1305$
- 전체 진입확률: $\sum_{n=0}^{7} b_n \times p_n = 0.9096$
- 서버로 들어오는 평균 요청 수, 즉 실제 유효 도착률: $\lambda_{eff} = \lambda \times 0.9096 \approx 45.45$
- 시스템 포화상태에서의 진입거절확률, 즉 서비스 이탈률: $1 - 0.9096 = 0.0904$
- ∴ 유효 도착률은 45.45(요청/초)이며, 표의 해석에 따라 행동 파라미터 값이 감소할수록 사용자가 급격히 포기하는 경우가 증가하며 시스템 부하가 감소하게 됨

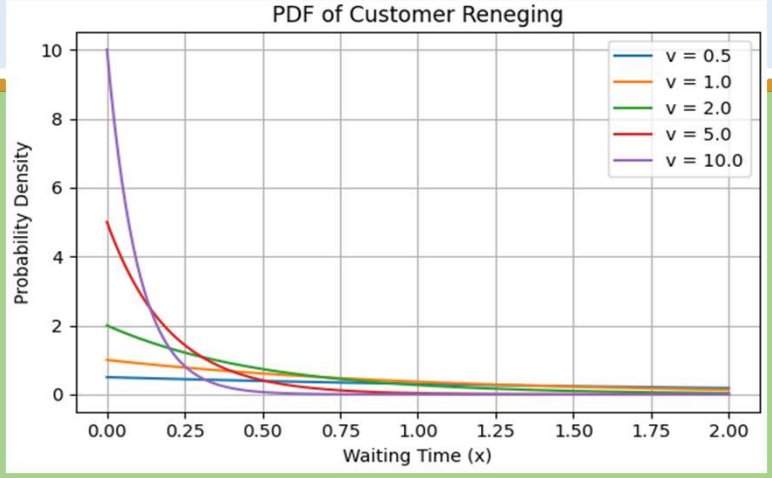
행동 파라미터(α)	진입거절확률 $(1 - \sum_{n=0}^{K}(1 - b_n) \cdot p_n)$
0.3	0.1207
0.6	**0.0904**
0.9	0.0367

추가예제 14

5G/6G URLLC를 위한 네트워크에서 사용자가 무선기지국에 접속 요청을 보낸다. 사용자는 일정 시간 이상 대기할 경우 자동으로 요청을 포기(Reneging)하는 이탈(Retry or Failover) 현상이 발생한다. 이 시스템은 초당 평균 10건의 요청이 도착하고 있으며, 각 요청은 평균적으로 초당 2건의 속도로 처리된다. 또한, 총 3개의 병렬 서버가 요청을 처리하고, 사용자는 대기열에서 평균 2초(이탈률 $v = 0.5$)를 대기한 후에도 처리가 되지 않으면 요청을 포기한다. 이때 이 URLLC 설계에서 사용자의 이탈 확률을 예측하고, v값 조절 시 시스템 안정성 및 이탈률의 변화를 분석하라.

- 평균도착률: $\lambda = 10$ 요청/초
- 평균서비스율: $\mu = 2$ 요청/초
- 서버 수: $c = 3$, 이탈률(Reneging Rate): $v = 0.5$
- 시스템 부하: $\gamma = \frac{\lambda}{\mu} = 5,\ \rho = \frac{\gamma}{c} = 1.667$
  (고객의 중도 포기가 존재하므로 시스템이 안정 상태에 이를 수 있음)
- 2초 이상 대기한 고객이 아직 이탈하지 않을 확률:
  $P(W_q > 2) = vdt + o(dt) = e^{-v \cdot 2} \approx 0.368$
- 사용자의 시스템 이탈률: $1 - 0.368 = 0.632$

- ∴ 사용자의 이탈 확률은 약 63.2%이고,
  v값은 커질수록 시스템은 안정적일 수 있으나
  사용자 이탈률이 증가하여 서비스 품질이 상대적으로
  저하될 수 있음



(그림 12) 추가예제 14

- 중도포기율 확률밀도함수를 활용하여 시간 t초 이후에 이탈할 확률 밀도를 그래프로 계산
- v가 클수록 사용자가 빠르게 이탈하는 것을 볼 수 있고, 시스템 부하가 감소하여 안정성이 향상됨. 그러나 서비스 품질 및 고객 만족도 저하 우려

감 사 합 니 다

김 지 혜 (jihye@pel.sejong.ac.kr)

V 부록 #1 - 주요용어

- 상태 전이: State Transition
- 연속시간 마코프 과정: Continuous-Time Markov Process
- 평균도착률: Average Arrival Rate
- 평균서비스율: Average Service Rate
- 단위 시간: Unit Time
- 평균도착간격: Average Interarrival Time
- 평균서비스시간: Average Service Time
- 생성률: Birth Rate
- 소멸률: Death Rate
- 상태확률: State Probability
- 상태전이방정식: State Transition Equation
- 평행상태: Steady State
- 평균 트래픽 강도: Average Traffic Intensity
- 도착과정: Arrival Process
- 서비스과정: Service Process
- 트래픽 부하: Offered Load
- 시스템 폭주 확률: Blocking Probability
- 유효 도착률: Effective Arrival Rate
- 체재 시간: System Sojourn Time
- 평균 체재 시간: Mean Sojourn Time
- 평균 대기 시간: Mean Waiting Time
- 대기 시간 분포: Waiting Time Distribution
- 시스템 내 평균 고객 수: Average Number of Customers in System
- 큐 내 평균 대기 고객 수: Average Number of Waiting Customers in Queue

V 부록 #2 - Erlang-C 표

- 대기행렬의 기초 책 pp. 237 (부록 C: Erlang-C 표) 발췌
  *단, 여기서 부하는 제4장에서 정의한 단일서버 환산부하임, 즉 γ를 나타냄

N	B=1%	B=10%	B=50%
1	0.010	0.100	0.500
2	0.147	0.500	1.281
3	0.429	1.040	2.116
4	0.810	1.653	2.977
5	1.259	2.313	3.856
6	1.758	3.007	4.747
7	2.296	3.725	5.647
8	2.866	4.463	6.553
9	3.460	5.218	7.466
10	4.077	5.986	8.383
15	7.394	9.970	13.021
20	10.973	14.116	17.717
25	14.721	18.364	22.449
30	18.588	22.684	27.207
35	22.546	27.059	31.984
40	26.577	31.478	36.777
45	30.666	35.932	41.583
50	34.804	40.416	46.399
55	38.985	44.925	51.224
60	43.202	49.456	56.057
65	47.450	54.006	60.897
70	51.728	58.574	65.743
75	56.030	63.156	70.594
80	60.356	67.752	75.450
85	64.702	72.361	80.311
90	69.067	76.981	85.175
95	73.450	81.612	90.044
100	77.849	86.252	94.915

V 부록 #3 - Python 코드

- (그림 5)

```
1.  import numpy as np
2.  import matplotlib.pyplot as plt
3.  from scipy.stats import erlang
4.  plt.rcParams.update({'font.size': 11})

5.  beta = 1.0
6.  x = np.linspace(0, 10, 500)

7.  plt.figure(figsize=(5, 4))
8.  for k in [1, 2, 3, 5, 10]:
9.      pdf = erlang.pdf(x, k, scale=1/beta)
10.     plt.plot(x, pdf, label=f'k={k}')

11. plt.title('Erlang Distribution PDF for Various k')
12. plt.xlabel('x (time)')
13. plt.ylabel('Probability Density')
14. plt.legend()
15. plt.grid(True)
16. plt.show()
```

V 부록 #3 - Python 코드

- (그림 6)

```
1.  import numpy as np
2.  import matplotlib.pyplot as plt
3.  from scipy.stats import erlang
4.  plt.rcParams.update({'font.size': 11})

5.  beta = 1.0
6.  c = 2
7.  k = 4
8.  shape = k - c + 1
9.  rate = c * beta

10. x = np.linspace(0, 10, 500)
11. pdf = erlang.pdf(x, shape, scale=1/rate)

12. plt.figure(figsize=(5, 4))
13. plt.plot(x, pdf, label=f'Erlang($k-c+1$={shape}, $\lambda$={rate})')
14. plt.title('PDF of Waiting Time $W_Q$')
15. plt.xlabel('x (Waiting Time)')
16. plt.ylabel('Probability Density')
17. plt.grid(True)
18. plt.legend()
19. plt.tight_layout()
20. plt.show()
```

V 부록 #3 - Python 코드

- (그림 7)

```
1.  import numpy as np
2.  import matplotlib.pyplot as plt
3.  plt.rcParams.update({'font.size': 11})

4.  c = 3
5.  alpha = 2
6.  beta = 1
7.  gamma = alpha / beta
8.  pc = 0.05

9.  def F_Q(t):
10.     coefficient = (c * pc) / (c - gamma)
11.     return 1 - coefficient * np.exp(-(c * beta - alpha) * t)

12. t_vals = np.linspace(0, 10, 200)
13. cdf_vals = F_Q(t_vals)

14. plt.figure(figsize=(5, 4))
15. plt.plot(t_vals, cdf_vals, label='$F_Q(t)$')
16. plt.title('CDF of Waiting Time $F_Q(t)$')
17. plt.xlabel('x (Waiting Time)')
18. plt.ylabel('Cumulative Probability')
19. plt.grid(True)
20. plt.legend()
21. plt.ylim(0, 1.1)
22. plt.show()
```

V 부록 #3 - Python 코드

- (그림 9)(1/2)

```
1.  import numpy as np
2.  import matplotlib.pyplot as plt
3.  from math import factorial
4.  from matplotlib.ticker import LogLocator
5.  plt.rcParams.update({'font.size': 11})

6.  c = 10
7.  K = 20
8.  alpha_values = [0.1, 0.3, 0.5, 0.7]
9.  rho_range = np.linspace(0.1, 0.9, 50)

10. def calc_chi(alpha, c, K):
11.     chi = [1.0]  # chi_0 = 1
12.     for n in range(1, K+1):
13.         if n <= c:
14.             chi.append(1.0)
15.         else:
16.             chi.append(chi[-1] * alpha**(n - c + 1))
17.     return chi

18. def calc_p0(chi, rho, c, K):
19.     sum1 = sum(chi[n] * rho**n / factorial(n) for n in range(c))
20.     sum2 = sum(chi[n] * rho**n / (factorial(c) * c**(n - c)) for n in range(c+1, K+1))
21.     return 1.0 / (sum1 + sum2)
```

- (그림 9)(2/2)

```
22. def calc_pK(chi, rho, c, K, p0):
23.     return (c**c / factorial(c)) * chi[K] * rho**K * p0

24. results = {}

25. for alpha in alpha_values:
26.     chi = calc_chi(alpha, c, K)
27.     blocking_probs = []
28.     for rho in rho_range:
29.         p0 = calc_p0(chi, rho, c, K)
30.         pK = calc_pK(chi, rho, c, K, p0)
31.         blocking_probs.append(pK)
32.     results[alpha] = blocking_probs

33. plt.figure(figsize=(5.5, 4))
34. colors = ['blue', 'red', 'green', 'orange']
35. for alpha, color in zip(alpha_values, colors):
36.     plt.plot(rho_range, results[alpha], label=f'α = {alpha}', color=color)
37. plt.yscale('log')
38. plt.xlabel('Offered Load')
39. plt.ylabel('Buffer Overflow Probability')
40. plt.title('M/M/c-B Queue Blocking Probability')
41. plt.grid(True, which='both', linestyle='--', linewidth=0.5)
42. plt.legend()
43. plt.tight_layout()
44. plt.show()
```

V 부록 #3 - Python 코드

- (그림 10)

```
1.  import numpy as np
2.  import matplotlib.pyplot as plt
3.  plt.rcParams.update({'font.size': 11})

4.  x = np.linspace(0, 5, 500)
5.  v_values = [0.2, 0.5, 1.0, 2.0]

6.  plt.figure(figsize=(5, 4))

7.  for v in v_values:
8.      f_x = v * np.exp(-v * x)
9.      plt.plot(x, f_x, label=f'v = {v}')

10. plt.title('PDF of Customer Reneging')
11. plt.xlabel('Waiting Time (x)')
12. plt.ylabel('Probability Density')
13. plt.legend()
14. plt.grid(True)
15. plt.tight_layout()
16. plt.show()
```

V 부록 #3 - Python 코드

- (그림 11)(1/2)

```
1.  import numpy as np
2.  import matplotlib.pyplot as plt
3.  from math import factorial
4.  plt.rcParams.update({'font.size': 11})

5.  c = 10
6.  K = 20
7.  mu = 1
8.  v_values = [0.1, 0.3, 0.5, 0.7]
9.  rho_range = np.linspace(0.1, 0.9, 50)

10. def calc_xi(v, c, K):
11.     xi = [1.0]  # xi_0
12.     for n in range(1, K - c + 1):
13.         xi.append(xi[-1] * (c * mu / (c * mu + v)))
14.     return xi

15. def calc_p0(rho, v, c, K):
16.     xi = calc_xi(v, c, K)
17.     sum1 = sum((rho ** n) / factorial(n) for n in range(c))
18.     sum2 = sum((rho ** (c + j)) * xi[j] / (factorial(c) * c ** j) for j in range(K - c + 1))
19.     return 1.0 / (sum1 + sum2)
```

- (그림 11)(2/2)

```
20. def calc_pK(rho, v, c, K, p0):
21.     xi = calc_xi(v, c, K)
22.     j = K - c
23.     coefficient = (c ** c) / factorial(c)
24.     return coefficient * xi[j] * (rho ** K) * p0

25. results = {}
26. for v in v_values:
27.     blocking_probs = []
28.     for rho in rho_range:
29.         p0 = calc_p0(rho, v, c, K)
30.         pK = calc_pK(rho, v, c, K, p0)
31.         blocking_probs.append(pK)
32.     results[v] = blocking_probs

33. plt.figure(figsize=(5.5, 4))
34. for v in v_values:
35.     plt.plot(rho_range, results[v], label=f'v = {v}')
36. plt.yscale('log')
37. plt.xlabel('Offered Load')
38. plt.ylabel('Buffer Overflow Probability')
39. plt.title('M/M/c-R Queue Blocking Probability')
40. plt.grid(True, which="both", ls="--")
41. plt.legend()
42. plt.tight_layout()
43. plt.show()
```

V 부록 #3 - Python 코드

- (그림 12)

```
1.  import numpy as np
2.  import matplotlib.pyplot as plt

3.  x = np.linspace(0, 2, 500)

4.  v_values = [0.5, 1.0, 2.0, 5.0, 10.0]

5.  plt.figure(figsize=(6, 4))

6.  for v in v_values:
7.      f_x = v * np.exp(-v * x)
8.      plt.plot(x, f_x, label=f'v = {v}')

9.  plt.title('PDF of Customer Reneging')
10. plt.xlabel('Waiting Time (x)')
11. plt.ylabel('Probability Density')
12. plt.legend()
13. plt.grid(True)
14. plt.tight_layout()
15. plt.show()
```