

Space and Time Priority Queues with Randomized Push-Out Scheme

Kilhwan Kim[†]

Management Engineering, Sangmyung University

확률적 밀어내기 정책을 가지는 공간-시간 우선순위 대기행렬

김 길 환[†]

상명대학교 경영공학과

In this study, we analyze a finite-buffer M/G/1 queueing model with randomized pushout space priority and nonpreemptive time priority. Space and time priority queueing models have been extensively studied to analyze the performance of communication systems serving different types of traffic simultaneously: one type is sensitive to packet delay, and the other is sensitive to packet loss. However, these models have limitations. Some models assume that packet transmission times follow exponential distributions, which is not always realistic. Other models use general distributions for packet transmission times, but their space priority rules are too rigid, making it difficult to fine-tune service performance for different types of traffic. Our proposed model addresses these limitations and is more suitable for analyzing communication systems that handle different types of traffic with general packet length distributions. For the proposed queueing model, we first derive the distribution of the number of packets in the system when the transmission of each packet is completed, and we then obtain packet loss probabilities and the expected number of packets for each type of traffic. We also present a numerical example to explore the effect of a system parameter, the pushout probability, on system performance for different packet transmission time distributions.

Keywords : Priority Queues, Space and Time Priorities, Push-Out Mechanism, Randomized Push-Out

1. 서 론

많은 통신 시스템은 이질적인 QoS(Quality of Service)를 요구하는 트래픽을 동시에 서비스한다. 대표적인 예가 TCP와 UDP 패킷을 동시에 처리하는 스위치나 라우터들이다[7, 34]. 일반적으로 UDP 패킷은 오디오나 비디오의 스트리밍 트래픽을 처리하는데, 이러한 트래픽은 어느 정도의 패킷 손실은 허용하지만 패킷 지연에는 서비스 품질이 매우 민감하게 변화한다. 반면 TCP 패킷은 파일 전송

등의 트래픽을 처리하는데, 이러한 트래픽은 어느 정도의 패킷 지연은 허용하지만 패킷의 손실에는 서비스 품질이 매우 민감하게 변화한다. 왜냐하면 TCP 패킷이 손실되면 TCP의 혼잡제어 메커니즘이 작동되기 때문이다[2].

아울러 통신 시스템은 패킷의 도착 과정과 처리 시간의 변동성을 흡수하기 위하여 처리가 지연되는 패킷을 버퍼에 저장한다. 버퍼는 일반적으로 메모리 상에 구현되고 유한한 용량을 가진다[12]. 따라서 서로 다른 QoS를 요구하는 트래픽에 대하여 서비스를 차별화하기 위해서는 서비스에 대한 시간적 우선권을 배분하는 것뿐만 아니라, 버퍼에 대한 공간적 우선권을 적절히 배분하는 것이 필요하다. TCP와 UDP 패킷을 같이 서비스하는 시스템의 경우, 지연에 민감한 UDP 패킷에게 TCP 패킷보다 우선하여 서비스받을 수

있는 시간 우선순위 정책을, 손실에 민감한 TCP 패킷에게 UDP 패킷에 우선하여 버퍼에 수용될 수 있는 공간 우선순위 정책을 적용하는 것이 바람직할 수 있다[7].

이러한 시간 및 공간 우선순위 정책이 적용된 통신 시스템의 성능분석을 위하여 우선순위 대기행렬 모형(priority queueing models)이 자주 이용된다. 통신시스템에 적용되는 우선순위 대기행렬 모형은 크게 세 범주로 분류할 수 있다. 첫번째 범주는 무한용량(시간) 우선순위 대기행렬로, 대기열은 무한으로 가정하여 이질적인 트래픽 간의 공간 상의 경쟁은 없다고 보고 서비스 처리 순서에 대한 시간 우선순위 정책만을 고려한다. 이러한 우선순위 대기행렬의 일차적 관심사는 이질적인 트래픽의 패킷 대기시간(waiting time)이며, 시간 우선순위 정책을 사용하여 트래픽별로 평균대기시간 등에 차등을 둔다. 선점 및 비선점 우선순위 정책 등의 고전적인 시간 우선순위 정책을 가지는 무한용량 대기행렬과 이를 통신 시스템의 성능분석에 적용한 연구는 Kleinrock[24]나 Takagi[32] 등을 참조한다. 좀 더 복잡한 혼합 시간 우선순위 정책을 가지는 무한용량 대기행렬에 대한 연구는 Kim[20, 21, 22, 23] 등을 참조한다.

두 번째 범주는 유한용량(공간) 우선순위 대기행렬로, 서로 다른 트래픽에 대한 시간 우선순위는 고려하지 않고 대기열에 접근할 수 있는 공간 우선순위만 트래픽별로 다르게 설정한다. 이러한 우선순위 대기행렬의 일차적 관심사는 이질적인 트래픽의 패킷 손실확률(loss probability)이며, 공간 우선순위 정책으로 트래픽별 패킷 손실확률을 조절한다. 대표적인 공간 우선순위 정책으로는 부분 버퍼공유(partial buffer-sharing) 정책과 밀어내기(push-out) 정책 등이 있으며, 이러한 정책을 적용한 유한용량 대기열과 이를 통신 시스템에 적용한 연구로는 Avrachenkov et al.[4] Kapadia et al.[16], Lin and Silvester[28], Sharma and Virtamo[30] 등이 있다.

그런데 첫 번째와 두 번째 범주의 연구는 이질적인 트래픽을 서비스하는 통신 시스템의 성능분석에 사용되기에는 제각기의 한계점이 있다. 무한용량 우선순위 대기행렬은 통신 시스템에서 버퍼의 크기가 유한이고 패킷의 손실이 주기적으로 발생하는 점을 반영하지 못한다는 점에서 모형의 정밀성에 한계를 갖는다. 유한용량 공간 우선순위 대기행렬은 트래픽 간의 시간 우선순위를 고려하지 않기 때문에, 지연에 대한 서로 다른 요구사항을 가지는 이질적인 트래픽에 대한 서비스 차별화(평균지연시간의 차별화 등)를 적용하기 어렵다.

우선순위 대기행렬의 세번째 범주는 이러한 한계점을 개선하여 시간과 공간의 우선순위를 모두 고려하는 우선순위 대기행렬 모형이다[1, 5-7, 11, 14, 17-19, 26, 28]. 그런데 이러한 연구들은 대부분 분석적 용이성을 위하여, 하

나의 클래스에 시간과 공간 우선순위를 모두 부여하거나 [5, 6, 14, 26, 28], 시간 우선순위와 공간 우선순위를 계층적으로 부여하거나[26], 아니면 공간 우선순위를 위하여 특정 클래스의 트래픽에는 무한용량의 대기열을 할당한다 [17, 19]. TCP와 UDP 트래픽을 동시에 서비스하는 예처럼, 지연과 손실에 대한 민감도가 서로 다른 트래픽에 대해 시간 우선순위와 공간 우선순위를 각기 다르게 부여하는 것이 더 바람직할 것이므로, 기존의 연구들은 이질적인 트래픽을 가지는 통신 시스템의 성능 분석에 적용하기에는 나름의 한계가 있다.

저자가 아는 한, 서로 다른 트래픽에 시간과 공간 우선순위를 제각기 부여한 연구로는 최근에 수행된 Carballo-Lozano et al.[7]과 Kim[18]의 연구가 있다. Carballo-Lozano et al.[7]은 유한용량 M/M/1 우선순위 대기행렬을 분석하였는데, 지연민감 트래픽에는 비선점 시간 우선순위를, 손실민감 트래픽에서는 확률적인 밀어내기(randomized push-out) 우선순위를 각각 부여하였다. 이 연구는 시간과 공간 우선순위를 제각기 부여한 최초의 연구였지만, 두 트래픽의 패킷 길이가 동일한 지수분포를 따른다고 가정하여 한계점이 있다. 통신 시스템의 패킷은 트래픽마다 서로 다른 길이를 가질 수 있고 일반적인 분포를 따르기 때문이다. Kim[18]은 Carballo-Lozano et al.[7]의 연구를 일반화하여 트래픽별로 패킷 길이가 서로 다른 일반분포를 따르는 유한용량 M/G/1 우선순위 대기행렬을 연구하였는데, 지연민감 트래픽에 비선점 우선순위를, 손실민감 트래픽에서는 비확률적 밀어내기(non-randomized push-out) 우선순위를 부여하였다. Kim의 연구는 클래스별로 패킷 길이가 서로 다른 일반분포를 가정하였다는 점에서 의의가 있으나, Carballo-Lozano et al.의 연구에 비해 공간 우선순위 규칙을 비확률적 밀어내기 정책으로 한정시킨 단점을 가지고 있다.

공간 우선순위를 구현하는 가장 단순한 방법은 버퍼를 분할하여 클래스별로 분리된 클래스 전용 버퍼 공간을 할당하는 것이다. 이러한 메커니즘 하에서는 공간 우선순위가 높은 클래스에게 버퍼의 더 많은 부분을 할당하여 해당 클래스의 손실 확률을 낮출 수 있다. 그러나 이러한 방법은 버퍼 전체를 모든 클래스의 패킷이 공유하는 것에 비해 버퍼 사용의 효율성이 저하된다. 왜냐하면 다른 클래스에 할당된 버퍼 공간이 비어있어도 이를 사용할 수 없기 때문이다.

부분 버퍼공유 정책은 클래스별 전용 버퍼 정책에서 나타나는 버퍼 사용의 비효율성을 줄이면서도 클래스별로 공간 우선순위를 부여하는 방법이다. 부분 버퍼공유 정책에서는 버퍼를 두 부분으로 나누어, 한 부분은 모든 클래스가 공유하고 다른 부분은 공간 우선순위가 높은 클래스만 전용으로 사용한다. 공용 버퍼 부분을 이용하여 전체 버퍼 사용의 효율성을 높이면서도, 전용 버퍼 부분으로 클

래스별 손실 확률의 차별화를 피하는 정책이라 할 수 있다 [13, 14]. 그러나 부분 버퍼공유 정책의 한 가지 문제점은 버퍼를 분할하는 적절한 비율을 결정하기가 쉽지 않다는 것이다. 특히 시스템에 도착하는 트래픽의 확률적 속성이 동적으로 변화하는 경우 전용 공간과 공용 공간의 비율을 환경 변화에 적응하여 변화시키는 것은 어려운 일이다.

밀어내기 정책은 버퍼의 이용률을 저하시키지 않으면서도 동적이고 적응적으로 공간 우선순위를 부여하는 정책이라 할 수 있다. 밀어내기 정책에서는 버퍼를 모든 클래스가 공유하지만, 공간 우선순위를 가지는 클래스의 패킷은 버퍼가 가득 찬 경우에도 버퍼에 있는 우선순위가 낮은 클래스의 패킷을 밀어내고 버퍼에 입장할 수 있다. 그렇기 때문에 버퍼에 비어 있는 공간이 있으면 도착하는 패킷은 항상 버퍼에 들어올 수 있어 버퍼 사용의 효율성을 높이면서도, 공간 우선순위가 높은 클래스의 트래픽이 증가하는 시기에는 밀어내기에 의해 자동적으로 해당 클래스의 버퍼 이용률이 자동적으로 증가한다. 이러한 환경에 대한 적응적 특성 때문에 밀어내기 정책을 사용하는 유한용량 대기행렬에 대한 연구가 많이 수행되었다[5-7, 9, 10, 18, 25, 27]. 그러나 앞서 언급한 바와 같이, 공간과 시간 우선순위를 별도의 클래스에 부여하는 대기행렬에 대한 연구는 Carballo-Lozano et al.[7]과 Kim[18]의 연구만 존재한다.

밀어내기 우선순위 정책은 이러한 환경 적응형 특징을 장점으로 가지지만 크게 두 가지 단점도 가지고 있다. 첫 번째 단점은 부분 버퍼공유 정책에 비해 통신 시스템에서 구현하기는 까다롭다는 것이다[3]. 왜냐하면 밀어내기를 수행한 후 버퍼의 순서를 재정렬해야 하는 문제가 있기 때문이다. 그러나 최근 SDN(Software-Defined Networking)의 발전으로 이러한 구현은 어렵지 않은 일이 되고 있다[33]. 두 번째 단점은 공간 우선순위가 높은 트래픽이 폭증하는 시기에는 해당 클래스의 패킷이 통신 자원을 모두 독점할 가능성이 높아진다는 것이다. 또한 손실에 민감한 TCP에 항상 밀어내기 우선순위를 주는 경우에, 버퍼가 TCP 패킷으로만 가득 찬 후에도 TCP 패킷의 손실이 발생한다. 그런데 TCP 트래픽은 이미 버퍼가 범람하기 시작할 때가 아니라 그 전의 적절한 시기에 패킷 손실이 사전적으로 발생하여 혼잡제어가 미리 수행되도록 하는 것이 전체적인 네트워크 성능에 좋은 것으로 알려져 있다[8]. 그러므로 Carballo-Lozano et al.[7]의 연구처럼 공간 우선순위로 확률적 밀어내기(randomized push-out) 정책을 사용하는 것을 고려할 필요가 있다. 확률적 밀어내기 정책에서는 공간 우선순위가 높은 클래스 패킷이 우선순위가 낮은 패킷을 밀어내기를 할 때 확률 α 로 성공하거나 확률 $1-\alpha$ 로 실패를 한다. 단, $0 \leq \alpha \leq 1$. 이렇게 하면 공간 우선순위가 높은 트래픽이 폭증하는 경우에도 해당 트래픽의 패킷이 버퍼를 완전히 장악하기 전에 손실이 발생하여 혼잡 제어 정책 등으로

트래픽 폭주를 조절할 수 있어서, 우선순위가 낮은 트래픽이 완전히 손실되는 것을 방지할 수 있다. 아울러 밀어내기 성공 확률 α 를 조절함에 따라 공간 우선순위 정책의 효과를 조절할 수 있는 장점도 있다. 예를 들어 $\alpha=0$ 으로 하면 공간 우선순위가 없는 완전 공유버퍼 정책이 될 것이며, $\alpha=1$ 이 되면 비확률적인 밀어내기 정책으로 버퍼에 공간 우선순위가 낮은 패킷이 있으면 항상 밀어내기에 성공할 것이다.

앞서 언급한 것처럼 Kim[18]의 연구는 클래스별로 패킷 길이가 서로 다른 일반분포를 가정하였다는 점에서 Carballo-Lozano et al.[7]의 연구를 확장하였다고 할 수 있으나, 공간 우선순위 규칙을 비확률적 밀어내기 정책으로 한정시킨 단점을 가지고 있었다. 본 연구에서는 Kim[18]의 연구를 확률적 밀어내기 정책으로 확장한다. 이러한 확장은 확률적 밀어내기 정책을 가정하였으나 클래스별 패킷 길이의 분포가 동일한 M/M/1 우선순위 대기행렬을 연구한 Carballo-Lozano et al.[7]과 달리, 대기행렬의 고객수 분포의 전이행렬과 대기시간 분석에 훨씬 더 큰 복잡성 야기시킨다. 그러므로 본 연구는 이론적으로 Carballo-Lozano et al.[7]과 Kim[18]의 연구를 종합하여 확장한다고 할 수 있다. 아울러 응용적으로는 확률적 밀어내기 정책의 모수 α 의 변화에 따라 패킷 길이의 분포가 다른 트래픽의 서비스 성능에 어떤 영향을 미치는지를 탐색함으로써, 확률적 밀어내기 정책과 서비스 분포의 관련성에 대한 이해를 도모한다.

서론 이후의 본 연구의 구성은 다음과 같다. 2 절에서는 본 연구에서 분석하고자 하는 공간-시간 우선순위 대기행렬 모형을 구체적으로 정의한다. 3 절에서는 내재점 마코프 체인(embedded Markov chain)을 사용하여 패킷 전송완료 시점의 패킷수 결합분포를 구한다. 4 절에서는 클래스별 패킷 손실확률을 구한다. 5 절에서는 임의 시점에 시스템에 머무는 평균 패킷수를 구한다. 6 절에서는 시스템의 주요 모수인 밀어내기 확률이 클래스별 서비스 성능에 어떠한 영향을 미치는지를 수치 예제를 사용하여 탐색한다. 마지막으로 7 절에서 본 연구의 결과를 요약하고 향후 연구 방향을 밝힌다.

2. 모형

본 연구에서는 다음과 같은 가정 하에 공간 및 시간 우선순위 대기행렬 모형을 다룬다. 시스템에는 손실에 민감한 트래픽의 패킷과 지연에 민감한 트래픽의 패킷이 도착한다. 손실에 민감한 트래픽을 클래스 L 이라 하고, 지연에 민감한 트래픽을 클래스 D 라고 하자. 클래스 L 과 D 의 패킷의 도착 과정은 각각 도착률이 λ_L 과 λ_D 인 독립적인 포아송 과정을 따른다. 그러므로 모든 클래스의 패킷의 도착률을 λ 라고 하면 $\lambda = \lambda_L + \lambda_D$ 이다.

시스템은 동일한 목적지로 전송되는 패킷을 하나의 전송 링크를 통해 한 번에 하나의 패킷을 전송하고, 다른 패킷이 전송 중일 때 도착하는 패킷은 용량이 K 인 버퍼에서 대기한다. 단, K 는 $K \geq 0$ 인 유한인 정수이다.

시스템에서 클래스 L 과 D 의 패킷 하나를 전송하는 시간을 각각 S_L 과 S_D 라고 하자. S_L 과 S_D 는 서로 독립인 i.i.d.(identical and independently distributed) 일반 분포를 따르고, S_L 과 S_D 의 CDF(cumulative distribution functions)와 LST(Laplace Stieltjes transformation)를 $S_L(t)$ 과 $S_D(t)$, 그리고 $S_L^*(s)$ 과 $S_D^*(s)$ 라고 하자. 단, $t \geq 0$ 이고 $\Re(s) \geq 0$. 그리고 $0 < E[S_L], E[S_D] < \infty$ 라고 가정한다.

손실에 민감한 클래스 L 의 손실확률은 낮추고, 지연에 민감한 클래스 D 의 지연시간을 줄이기 위하여 다음과 같은 공간 및 시간 우선순위 정책을 적용한다. 손실에 민감한 클래스- L 패킷에는 공간 우선순위로써 확률적 밀어내기 우선순위가 부여된다. 확률적 밀어내기 우선순위 정책을 엄밀하게 정의하기 위하여 다음의 확률과정을 먼저 정의하자. $\xi(t)$ 를 전송 링크의 상태를 나타내는 다음과 같은 확률과정으로 정의하자.

$$\xi(t) \triangleq \begin{cases} L, & \text{전송 링크가 유힬 상태} \\ L, & \text{클래스-}L \text{ 패킷을 전송} \\ D, & \text{클래스-}D \text{ 패킷을 전송} \end{cases}$$

$N_L^Q(t)$ 와 $N_D^Q(t)$ 를 t 시점에 버퍼에 있는 클래스- L 과 클래스- D 의 패킷 수라고 하자. 그리고 $N^Q(t) = N_L^Q(t) + N_D^Q(t)$ 를 버퍼에 있는 전체 패킷수라고 하자. 확률적 밀어내기 정책 하에서는, 공간 우선순위가 없는 클래스- D 패킷이 t 시점에 도착했을 때, 이 패킷은 다음처럼 처리된다.

- $\xi(t) = I$ 이면, 도착한 클래스- D 패킷 즉시 전송.
- $\xi(t) \neq I$ 이고 $0 \leq N^Q(t) < K$ 이면, 도착한 클래스- D 패킷은 버퍼에 들어가 대기한다.
- $\xi(t) \neq I$ 이고 $N^Q(t) = K$ 이면, 도착한 클래스- D 패킷은 즉시 손실된다.

반면 클래스- L 패킷에는 확률적 밀어내기 우선순위가 부여되어서, 클래스- L 패킷이 t 시점에 도착했을 때, 이 패킷은 다음처럼 처리된다.

- $\xi(t) = I$ 이면, 도착한 클래스- L 패킷 즉시 전송.
- $\xi(t) \neq I$ 이고 $0 \leq N^Q(t) < K$ 이면, 도착한 클래스- L 패킷은 버퍼에 들어가 대기한다.
- $\xi(t) \neq I$ 이고 $N^Q(t) = K$ 이면서 $N_D^Q(t) > 0$ 이면, 도착한 클래스- D 패킷은 확률 α 로 버퍼에 있는 가장 늦게 도착한 클래스- D 패킷을 밀어내고 버퍼에 들어가

나, 확률 $1-\alpha$ 로 즉시 손실된다. 단, $0 \leq \alpha \leq 1$ 이고, 밀어내기가 성공하면 밀어내기를 당한 클래스- D 패킷은 즉시 손실된다.

$\alpha = 1$ 이면 항상 밀어내기가 성공하여 Kim[18]에서와 같이 비확률적 밀어내기 정책이 된다. 반면 $\alpha = 0$ 이면 항상 밀어내기가 실패하여 공간 우선순위가 없는 대기행렬 모형이 된다. 어떠한 패킷도 자신과 동일한 클래스의 패킷을 밀어내기할 수 없다.

클래스- L 패킷에 공간 우선순위가 부여되는 대신, 지연에 민감한 클래스- D 패킷에는 다음과 같은 비선점 시간 우선순위가 부여된다.

- 전송 링크에서 패킷의 전송이 종료되었을 때 버퍼에 클래스- D 패킷이 있으면, 클래스- L 패킷이 먼저 도착하였더라도 클래스- D 패킷 중 하나가 다음으로 전송된다. 그러므로 클래스- L 패킷은 버퍼에 클래스- D 패킷이 없을 때만, 전송이 시작된다.
- 동일한 클래스의 패킷 사이에는 FCFS(First Come First Served) 정책이 적용되어 먼저 도착한 패킷이 먼저 전송된다.
- 시간 우선순위가 있는 클래스- D 패킷이더라도 이미 전송을 시작한 패킷의 전송을 중단시킬 수 없다.

3. 패킷 전송완료 직후 시점의 패킷수 결합분포

이 절에서는 패킷 전송이 완료된 직후 시점에 시스템에 남아있는 전체 패킷수와 클래스- D 의 패킷수의 결합분포를 EMC(Embedded Markov Chain)을 사용하여 분석한다.

3.1 전송 기간 동안의 도착 패킷수 분포

전송완료 직후 시점의 패킷수에 대한 EMC를 분석하기 위해, 먼저 패킷 하나를 전송하는 동안 도착하는 패킷수에 대한 분포를 고려해 보자. 이를 위해 다음과 같은 확률변수를 정의하자.

- $A(S_h)$: 클래스- h 패킷을 전송하는 동안 도착한 모든 클래스의 패킷수. 단, $h = L$ 또는 D .
- $A_g(S_h)$: 클래스- h 패킷을 전송하는 동안 도착한 클래스 g 의 패킷수. 단, $g = L$ 또는 D .
- B_n^g : 시도횟수가 n 이고 성공확률이 λ_D/λ 인 이항분포를 따르는 확률변수.
- $B_{L,s}^n$: 시행횟수가 n 이고 성공확률이 $\alpha\lambda_L/\lambda$ 인 이항분포를 따르는 확률변수.

그러면 정의에 의해 $A(S_h) = A_L(S_h) + A_D(S_h)$ 이다. 아울러 포아송 과정의 성질에 의해 $A_D(S_h) = B_D^{A(S_h)}$ 가 성립하고, 패킷 전송 시간 S_h 동안 도착하는 패킷수 분포는 다음과 같이 표현된다.

$$a_n(S_h) \triangleq \Pr\{A(S_h) = n\} \quad (1)$$

$$= \int_0^\infty \frac{(\lambda t)^n}{n!} e^{-\lambda t} dS_h(t)$$

$$b_{(n,j)} \triangleq \Pr\{B_D^n = j\} = \Pr\{A_D(S_h) = j | A(S_h) = n\} \quad (2)$$

$$= \binom{n}{j} \left(\frac{\lambda_D}{\lambda}\right)^j \left(\frac{\lambda_L}{\lambda}\right)^{n-j}$$

$$\alpha_{(n,j)}(S_h) \triangleq \Pr\{A(S_h) = n, A_D(S_h) = j\} \quad (3)$$

$$= a_n(S_h) b_{(n,j)}$$

$$= \int_0^\infty \frac{(\lambda_L t)^{n-j}}{(n-j)!} \frac{(\lambda_D t)^j}{j!} e^{-\lambda t} dS_h(t)$$

단, $h = L$ 또는 D 이고, $0 \leq j \leq n$. 포아송 과정의 성질에 의해 $b_{(n,j)}$ 는 전송시간 S_h 와 무관하다[29].

$A(z; S_h)$ 를 $A(S_h)$ 의 PGF(probability generating function)라고 하자. 즉,

$$A(z; S_h) \triangleq \sum_{n=0}^{\infty} a_n(S_h) z^n = S_h^*(\lambda - \lambda z).$$

클래스- h 패킷의 전송 시작 시점에 버퍼에 k 개의 빈 자리가 있었다고 하자. 만약 $A(S_h) < k$ 라면 클래스- h 패킷이 전송되는 동안 도착한 패킷은 손실 없이 모두 시스템에 입장한다. 그렇지 않고 $A(S_h) \geq k$ 라면, 클래스- h 패킷이 전송되는 동안 도착한 패킷 중 처음 h 개의 패킷은 버퍼가 가득 차기 전에 도착한 패킷이므로 모두 시스템에 입장한다. 반면 $A(S_h) - k$ 개의 패킷은 버퍼가 가득 찬 후에 도착한 패킷이므로, 이 패킷 중 클래스- D 패킷은 모두 손실되고, (버퍼에 밀어낼 클래스- D 패킷이 항상 충분히 있다고 가정하면) 클래스- L 패킷은 α 의 확률로 밀어내기에 성공하여 시스템에 입장하거나 $1 - \alpha$ 의 확률로 손실된다. 그러므로 포아송 과정의 성질에 의해, 버퍼가 가득 찬 후에 도착한 패킷 중, 밀어내기에 성공한 클래스- L 패킷수는 $B_{L,s}^{A(S_h)-k}$ 로 표현되고, 다음과 같은 이항분포가 된다.

$$\Pr\{B_{L,s}^{A(S_h)-k} = i | A(S_h) \geq k\}$$

$$= \frac{(A(S_h) - k)!}{i!(A(S_h) - k - i)!} \left(\frac{\alpha \lambda_L}{\lambda}\right)^i \left(1 - \frac{\alpha \lambda_L}{\lambda}\right)^{A(S_h) - k - i}$$

단, $A(S_h) \geq k \geq 0$, $0 \leq i \leq A(S_h) - k$.

또한 포아송 과정의 성질에 의해 버퍼가 가득 차기 전에 도착한 k 개의 패킷 중 클래스 D 인 패킷수는 이항확률 변수 B_D^k 로 표현할 수 있다. 이러한 사실들을 이용하여 클래스- k 패킷의 전송 시작시 버퍼에 k 개의 빈 자리가 있을 때, $A(S_h) \geq k$ 이고 $B_D^k = j$ 라는 조건 하에 $B_{L,s}^{A(S_h)-k}$ 의 부분 PGF $A_{(k,j)}(z; S_h)$ 를 다음과 같이 구할 수 있다.

$$A_{(k,j)}(z; S_h) \quad (4)$$

$$\triangleq \Pr\{A(S_h) \geq k, B_D^k = j\} E\left[z^{B_{L,s}^{A(S_h)-k}} | A(S_h) \geq k, B_D^k = j\right]$$

$$= \sum_{n=k}^{\infty} \Pr\{A(S_h) = n, B_D^k = j\} E\left[z^{B_{L,s}^{n-k}} | A(S_h) = n, B_D^k = j\right]$$

$$= \sum_{n=k}^{\infty} a_n(S_h) b_{(k,j)} \sum_{i=0}^{n-k} \frac{(n-k)!}{i!(n-k-i)!} \left(\frac{\alpha \lambda_L z}{\lambda}\right)^i \left(1 - \frac{\alpha \lambda_L}{\lambda}\right)^{n-k-i}$$

$$= \frac{b_{(k,j)}}{z_0^k} \left(A(z_0; S_h) - \sum_{n=0}^{k-1} a_n(S_h) z_0^n \right)$$

단, $0 \leq j \leq k$ 이고,

$$z_0 = \frac{\alpha \lambda_L}{\lambda} z + 1 - \frac{\alpha \lambda_L}{\lambda}$$

식 (4)에서 다음과 같은 확률을 구할 수 있다.

$$\tilde{a}_{(k,j)}(S_h) \triangleq \Pr\{A(S_h) \geq k, B_D^k = j\} \quad (5)$$

$$= A_{(k,j)}(1; S_h) = b_{(k,j)} \left(1 - \sum_{n=0}^{k-1} a_n(S_h) z_0^n \right)$$

$$\tilde{a}_{(k,j)l}(S_h) \triangleq \Pr\{A(S_h) \geq k, B_D^k = j, B_{L,s}^{A(S_h)-k} = l\} \quad (6)$$

$$= \frac{1}{l!} \left[\frac{d^l}{dz^l} A_{(k,j)}(z; S_h) \right]_{z=0}$$

$$\tilde{a}_{(k,j)l+}(S_h) \triangleq \Pr\{A(S_h) \geq k, B_D^k = j, B_{L,s}^{A(S_h)-k} \geq l\} \quad (7)$$

$$= \tilde{a}_{(k,j)}(S_h) - \sum_{l'=0}^{l-1} \tilde{a}_{(k,j)l'}(S_h)$$

단, $0 \leq j \leq k$, $l \geq 0$. 다음 절에서 (1)-(3)과 (5)-(7) 등을 사용하여 EMC의 전이확률을 구한다.

3.2 패킷 전송완료 시점의 EMC

$N_g^d(n)$ 을 n -번째 패킷 전송완료 시점 직후에 시스템에 있는 클래스- g 패킷의 수라고 하자. 단, $g = L, D$, $n = 1, 2, \dots$. 그리고 $N^d(n) \triangleq N_L^d(n) + N_D^d(n)$ 으로 n -번째 패킷 전송완료 시점 직후의 모든 클래스의 패킷수라고 하자.

그러면 $\{(N^d(n), N_D^d(n)) \mid n=1,2,\dots\}$ 은 EMC를 형성한다. 버퍼의 크기가 K 로 제한되어 있기 때문에 다음이 성립한다.

$$0 \leq N_D^d(n) \leq N^d(n) \leq K$$

EMC의 극한확률(limiting probability)을 다음과 같이 정의하자.

$$\pi_{(i,j)} = \lim_{n \rightarrow \infty} Pr\{N_L^d(n) = i, N_D^d(n) = j\},$$

$$0 \leq j \leq i \leq K$$

그리고 $N^d(n) = i$ 인 상태, 즉, $(i,0), (i,1), \dots, (i,i)$ 을 통칭하여 레벨 i 라고 하면, 레벨 i 의 극한확률 벡터와 전체 극한확률 벡터는 다음과 같이 정의된다.

$$\pi_i = [\pi_{(i,0)} \pi_{(i,1)} \cdots \pi_{(i,i)}], \quad 0 \leq i \leq K$$

$$\pi = [\pi_0 \pi_1 \cdots \pi_K].$$

$p_{(i,j),(l,m)}$ 을 EMC의 상태 (i,j) 에서 상태 (l,m) 로의 전이 확률이라고 하자. 단, $0 \leq j \leq i \leq K$ 이고 $0 \leq m \leq l \leq K$. 그러면 레벨 i 에서 레벨 j 로 가는 부분 전이확률 행렬 $\mathbf{P}_{i,l}$ 는 다음처럼 정의된다.

$$\mathbf{P}_{i,l} = \begin{bmatrix} P_{(i,0),(l,0)} & P_{(i,0),(l,1)} & \cdots & P_{(i,0),(l,i)} \\ P_{(i,1),(l,0)} & P_{(i,1),(l,1)} & \cdots & P_{(i,1),(l,i)} \\ \vdots & \vdots & \ddots & \vdots \\ P_{(i,i),(l,0)} & P_{(i,i),(l,1)} & \cdots & P_{(i,i),(l,i)} \end{bmatrix} \quad (8)$$

두 연속된 내재점 사이에서 전체 패킷수의 감소는 전송완료 시점에만 발생한다(밀어내기에 의해서는 전체 패킷수는 변하지 않는다). 그러므로 $0 \leq i \leq K$ 이고 $0 \leq l < i-1$ 이면 $\mathbf{P}_{i,l} = \mathbf{0}$ 이고, 전체 전이확률 행렬 \mathbf{P} 의 구조는 다음과 같은 M/G/1 형식이 된다.

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{0,0} & \mathbf{P}_{0,1} & \cdots & \mathbf{P}_{0,K-2} & \mathbf{P}_{0,K-1} & \mathbf{P}_{0,K} \\ \mathbf{P}_{1,0} & \mathbf{P}_{1,1} & \cdots & \mathbf{P}_{1,K-2} & \mathbf{P}_{1,K-1} & \mathbf{P}_{1,K} \\ 0 & \mathbf{P}_{2,1} & \cdots & \mathbf{P}_{2,K-2} & \mathbf{P}_{2,K-1} & \mathbf{P}_{2,K} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \mathbf{P}_{K-1,K-2} & \mathbf{P}_{K-1,K-1} & \mathbf{P}_{K-1,K} \\ 0 & 0 & \cdots & 0 & \mathbf{P}_{K,K-1} & \mathbf{P}_{K,K} \end{bmatrix} \quad (9)$$

EMC가 기약(irreducible)이고 에르고딕(ergodic)이면, 극한확률 벡터 π 는 다음 관계를 만족한다.

$$\pi = \pi \mathbf{P}, \quad \pi \mathbf{e} = 1. \quad (10)$$

단, \mathbf{e} 는 모든 요소가 1인 열벡터이다.

지금부터 전이확률 $p_{(i,j),(l,m)}$ 를 도출해 보자. 클래스- D 패킷에 시간 우선순위를 부여하였기 때문에, $N_D^d(n) \geq 1$ 이면 n 과 $(n+1)$ 번째 내재점 사이에 클래스- D 패킷이 전송된다. 반면 $N_D^d(n) = 0$ 이고 $N_L^d(n) \geq 1$ 이면, n 과 $(n+1)$ 번째 내재점 사이에 클래스- L 패킷이 전송된다. 마지막으로 $N_D^d(n) = N_L^d(n) = 0$ 이면, n 과 $(n+1)$ 번째 내재점 사이에 유희기간(idle period)이 발생하고 유희기간 동안에 도착한 패킷이 전송된다. 따라서 EMC의 상태 공간 $\{(i,j) : 0 \leq j \leq i \leq K\}$ 를 서로 배타적인 세 개의 부분집합으로 나눌 수 있다. 각 부분집합에서의 전이확률 $p_{(i,j),(l,m)}$ 를 고려해 보자.

먼저 $1 \leq j \leq i \leq K$ 인 경우의 $p_{(i,j),(l,m)}$ 를 구해보자. $N^d(n) \geq N_L^d(n) \geq 1$ 이면, 시간 우선순위에 의해 n 번째 내재점 직후에 버퍼에 있는 클래스- D 패킷의 전송이 시작되고 n 과 $(n+1)$ 내재점 사이의 시간은 S_D 가 된다. 클래스- D 패킷의 전송이 시작되었으므로 전송 시작 직후에 버퍼에 있는 전체 패킷수는 $N^d(n) - 1$ 이고 버퍼에 있는 클래스- D 패킷은 $N_D^d(n) - 1$ 이다. 그런데 버퍼의 크기는 K 로 제한되어 있고 n 과 $(n+1)$ 내재점 사이에 $A(S_D)$ 패킷이 도착하므로, $(n+1)$ 시점의 전체 패킷수 $N^d(n+1)$ 은 다음과 같다.

$$N^d(n+1) = \min\{N^d(n) - 1 + A(S_D), K\}$$

만약 $A(S_D) < K - N^d(n) + 1$ 이면, S_D 동안 도착한 모든 패킷은 손실 없이 시스템에 입장한다. 그러므로 새로 도착한 $A(S_D)$ 패킷 중 클래스 D 인 패킷의 수는 $B_D^{A(S_D)}$ 가 된다. 따라서 $A(S_D) \leq K - N^d(n) + 1$ 일 때, $(n+1)$ 시점의 클래스- D 패킷수 $N_D^d(n+1)$ 는 다음과 같다.

$$N_D^d(n+1) = N_D^d(n) - 1 + B_D^{A(S_D)}$$

반면 $A(S_D) \geq K - N^d(n) + 1$ 이면 버퍼가 가득 차는 사건이 발생한다. 버퍼가 가득 차기 전에 도착한 $K - N^d(n) + 1$ 패킷은 모두 손실없이 입장하고, 이 중 클래스- D 패킷의 수는 $B_D^{K - N^d(n) + 1}$ 이다. 그러므로 버퍼가 가득 차는 순간에 버퍼에 있는 클래스- D 패킷의 수는 다음과 같다.

$$N_D^d(n) - 1 + B_D^{K - N^d(n) + 1}$$

버퍼가 가득 찬 후에 도착한 패킷수는 $A(S_D) - K + N^d(n) - 1$ 이다. 그러므로 버퍼에 클래스- D 패킷이 충분

히 있다면 버퍼가 가득 찬 후에 도착한 패킷 중 밀어내기에 성공하는 클래스- L 패킷수는 $B_{L,s}^{A(S_D)} - K + N^d(n) - 1$ 이 된다. 그런데 밀어내기는 버퍼에 클래스- D 패킷이 있을 때만 가능하므로, 밀어내기에 성공하는 클래스- L 패킷수는 버퍼가 가득 찼을 때 버퍼에 있던 클래스- D 패킷수 $N_D^d(n) - 1 + B_D^{K - N^d(n) + 1}$ 를 초과할 수 없다. 따라서 $A(S_D) > K - N^d(n) + 1$ 일 때, $(n+1)$ 시점의 클래스- D 패킷수 $N_D^d(n+1)$ 는 다음과 같다.

$$N_D^d(n+1) = \max \{ N_D^d(n) - 1 + B_D^{K - N^d(n) + 1} - B_{L,s}^{A(S_D) - K + N^d(n) - 1}, 0 \}.$$

$(N^d(n), N_D^d(n)) = (i, j)$ 이고 $(N^d(n+1), N_D^d(n+1)) = (l, m)$ 이라고 하면, 위의 관계식과 $0 \leq B_D^k \leq k$ 이고 $0 \leq B_L^k \leq k$ 인 사실을 이용하여 $1 \leq j \leq i \leq K$ 일 때의 전이확률 $p_{(i,j),(l,m)}$ 를 다음과 같이 구할 수 있다.

$$p_{(i,j),(l,m)} = \begin{cases} a_{(l-i+1, m-j+1)}(S_D) & \text{if } i-1 \leq l < K, j-1 \leq m \leq l-(i-j), \\ \sum_{n=\max\{0, m-j+1\}}^{K-i+1} \tilde{a}_{(K-i+1, n); j-1+n-m}(S_D) & \text{if } l=K, 1 \leq m \leq K-(i-j), \\ \sum_{n=0}^{K-i+1} \tilde{a}_{(K-i+1, n); (j-1+n)+}(S_D) & \text{if } l=K, m=0, \\ 0 & \text{otherwise.} \end{cases} \quad (11)$$

다음으로 $1 \leq i \leq K$ 이고 $j=0$ 일 때의 전이확률 $p_{(i,j),(l,m)}$ 를 구해보자. $N^d(n) \geq 1$ 이고 $N_D^d(n)=0$ 이면 n -번째 내재점 직후에 클래스- L 패킷의 전송이 시작되고, n 과 $(n+1)$ 내재점 사이의 시간은 S_L 이 된다. 클래스- L 패킷이 전송되므로 전송 시작 직후 시점에 버퍼에 있는 전체 패킷수는 $N^d(n) - 1$ 이고 클래스- D 패킷수는 $N_D^d(n)=0$ 이다. 그러므로 n -번째 내재점과 $(n+1)$ -번째 내재점 간의 전체 패킷수의 관계는 다음과 같다.

$$N^d(n+1) = \min \{ N^d(n) - 1 + A(S_L), K \}$$

$A(S_L) < K - N^d(n) + 1$ 이면 S_L 동안 도착한 모든 패킷은 손실없이 시스템에 입장한다. 그러므로 n -번째 내재점과 $(n+1)$ -번째 내재점 간의 클래스- D 패킷수의 관계는 다음과 같다.

$$N_D^d(n+1) = N_D^d(n) + B_D^{A(S_L)} = B_D^{A(S_L)}$$

반면 $A(S_D) \geq K - N^d(n) + 1$ 이면 버퍼가 가득 차는 사

건이 발생한다. 버퍼가 가득 차기 전에 도착한 클래스- D 패킷의 수는 $B_D^{K - N^d(n) + 1}$ 이고, 버퍼가 가득 찬 후 도착하여 밀어내기에 성공한 클래스- L 패킷수는 $B_{L,s}^{A(S_L) - K + N^d(n) - 1}$ 이므로 다음 관계가 성립한다.

$$N_D^d(n+1) = \max \{ B_D^{K - N^d(n) + 1} - B_{L,s}^{A(S_L) - K + N^d(n) - 1}, 0 \}$$

$(N_L^d(n), N_D^d(n))$ 과 $(N_L^d(n+1), N_D^d(n+1))$ 의 관계를 사용하여, $1 \leq i \leq K$ 이고 $j=0$ 인 경우의 전이확률 $p_{(i,j),(l,m)}$ 를 다음과 같이 구할 수 있다.

$$p_{(i,0),(l,m)} = \begin{cases} a_{(l-i+1, m)}(S_L) & \text{if } i-1 \leq l < K, 0 \leq m \leq l-i+1, \\ \sum_{n=m}^{K-i+1} \tilde{a}_{(K-i+1, n); n-m}(S_L) & \text{if } l=K, 1 \leq m \leq K-i+1, \\ \sum_{n=0}^{K-i+1} \tilde{a}_{(K-i+1, n); n+}(S_L) & \text{if } l=K, m=0, \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

마지막으로 $i=j=0$ 일 때의 전이확률 $p_{(i,j),(l,m)}$ 를 구해보자. $N^d(n) = N_D^d(n) = 0$ 이면 n -번째 내재점 후에 $\text{Exp}(\lambda)$ 를 따르는 유희기간이 발생하고, 유희기간 동안 확률 λ_L/λ 로 클래스- L 패킷이 도착하여 S_L 길이의 패킷전송 기간이 발생하고, 확률 λ_D/λ 로 클래스- D 패킷이 도착하여 S_D 길이의 패킷전송 기간이 발생한다. 유희기간 동안 도착한 패킷은 $(n+1)$ -번째 내재점까지는 전송이 완료되어 시스템에서 이탈한다. 그러므로 유희기간 동안 도착한 패킷의 클래스가 h ($h=L, D$)라고 하면 앞의 두 경우와 유사한 논리로 다음과 같은 관계가 성립한다.

$$((N_L^d(n+1), N_D^d(n+1))) = \begin{cases} (A(S_h), B_D^{A(S_h)}) & \text{if } 0 \leq A_L(S_h) \leq K-1, \\ (K, B_D^K - B_{L,s}^{A(S_h) - K}) & \text{if } A(S_h) \geq K, 0 \leq B_{L,s}^{A(S_h) - K} < B_D^K, \\ (K, 0) & \text{if } A(S_h) \geq K, B_{L,s}^{A(S_h) - K} \geq B_D^K \end{cases}$$

따라서 $i=j=0$ 일 때의 전이확률은 다음과 같다.

$$p_{(0,0),(l,m)} = \begin{cases} \sum_{h \in \{L, D\}} \frac{\lambda_h}{\lambda} a_{(l,m)}(S_h), & 0 \leq m \leq l \leq K-1, \\ \sum_{h \in \{L, D\}} \frac{\lambda_h}{\lambda} \sum_{n=m}^K \tilde{a}_{(K,n); n-m}(S_h), & l=K, 1 \leq m \leq K, \\ \sum_{h \in \{L, D\}} \frac{\lambda_h}{\lambda} \sum_{n=0}^K \tilde{a}_{(K,n); n+}(S_h), & l=K, m=0, \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

식 (11)~식 (13)에서 모든 부분 전이확률 행렬 $\mathbf{P}_{i,l}$, $0 \leq i \leq K$, $\max(0, i-1) \leq l \leq K$, 그리고 전체 전이확률 행렬 \mathbf{P} 를 구할 수 있다. Kim[18]의 연구와 동일한 방식으로 부분 전이확률 행렬과 전체 전이확률 행렬의 구조를 사용하여 \mathbf{P} 가 기약이고 에르고딕 마코프 체인임을 증명할 수 있다([18, Section A] 참조). 따라서 (10)을 만족하는 극한확률 벡터 $\boldsymbol{\pi}$ 는 유일하게 존재하고 수치적으로 구할 수 있다[31]. (9)의 전이확률 행렬의 구조를 이용하여 효율적으로 벡터 $\boldsymbol{\pi}$ 를 구하는 방법에 대한 논의는 Kim[18]을 참조한다.

4. 패킷 손실확률

밀어내기 우선순위 정책 하에서는 두 가지 방식의 패킷 손실이 발생한다. 하나는 패킷이 도착했을 때 버퍼가 가득 차 있고 밀어내기도 불가능하여 도착 즉시 손실이 되는 것이고, 다른 하나는 클래스- L 패킷이 클래스- D 패킷을 밀어내어 버퍼에 있는 클래스- D 패킷 중 하나가 손실되는 것이다. $g \in \{L, D\}$ 일 때, P_g^b 를 클래스- g 패킷이 도착 즉시 손실될 확률이라고 하자. 그리고 P_L^{po} 를 클래스- L 패킷이 도착시 밀어내기를 할 확률이라고 하자.

2절에서 시스템의 전송 상태를 나타내는 $\xi(t)$, 버퍼에 있는 총 패킷수 $N^Q(t)$, 버퍼에 있는 총 클래스- D 패킷수 $N_D^Q(t)$ 를 정의하였다. 밀어내기 우선순위 정책 하에서 클래스- D 패킷은 버퍼가 가득 차 있으면 항상 도착 즉시 손실된다. 그러므로 포아송 과정의 PASTA 성질에 의해서 클래스- D 패킷의 손실확률 P_D^b 는 다음과 같다.

$$P_D^b = \lim_{t \rightarrow \infty} \Pr\{N^Q(t) = K, \xi(t) \neq I\} \quad (14)$$

밀어내기 우선순위 정책 하에서 클래스- L 패킷은 버퍼가 가득 차 있어도 클래스- D 패킷이 버퍼에 있으면 확률 α 로 밀어내기에 성공할 수 있다. 반면 버퍼가 클래스- L 패킷으로만 가득 차 있거나 버퍼에 클래스- D 패킷이 있어도 $(1-\alpha)$ 확률로 밀어내기에 실패하면 도착 즉시 손실된다. 그러므로 PASTA에 의해 클래스- L 패킷의 손실확률 P_L^b 와 밀어내기를 할 확률 P_L^{po} 은 다음과 같이 표현할 수 있다.

$$P_L^b = \lim_{t \rightarrow \infty} \Pr\{N^Q(t) = K, N_D^Q(t) = 0, \xi(t) \neq I\} \quad (15)$$

$$+ \lim_{t \rightarrow \infty} \Pr\{N^Q(t) = K, N_D^Q(t) \geq 1, \xi(t) \neq I\} \cdot (1-\alpha)$$

$$P_L^{po} = \lim_{t \rightarrow \infty} \Pr\{N^Q(t) = K, N_D^Q(t) \geq 1, \xi(t) \neq I\} \cdot \alpha \quad (16)$$

아울러 식 (14)~식 (16)에서 다음이 성립한다.

$$P_D^b = P_L^b + P_L^{po} \quad (17)$$

$g \in \{L, D\}$ 일 때, λ_g^b , λ_g^e , λ_L^{po} 를 각각 단위 시간당 클래스- g 패킷의 손실률, 전송률, 클래스- L 패킷에 의한 밀어내기 발생률이라고 하자. 도착시 손실되지 않는 클래스- L 패킷은 모두 전송이 완료되고, 도착시 손실되거나 밀어내기 당하지 않은 클래스- D 패킷도 모두 전송이 완료되므로 다음이 성립한다.

$$\lambda_g^b = \lambda_g P_g^b, \quad g \in \{L, D\} \quad (18)$$

$$\lambda_L^{po} = \lambda_L P_L^{po} \quad (19)$$

$$\lambda_L^e = \lambda_L - \lambda_L^b = \lambda_L(1 - P_L^b) \quad (20)$$

$$\lambda_D^e = \lambda_D - \lambda_D^b - \lambda_L^{po} = \lambda_D(1 - P_D^b) - \lambda_L P_L^{po} \quad (21)$$

$$= \lambda(1 - P_D^b) - \lambda_L(1 - P_L^b)$$

식 (21)의 마지막 등식은 식 (17)에 의해 성립한다.

P_D^b , P_L^b , P_L^{po} 를 구하기 위하여, 시스템이 유희상태일 확률, 시스템이 클래스- L 을 전송할 확률, 시스템이 클래스- D 를 전송할 확률을 EMC를 사용하여 구해보자. $T(n)$ 을 n -번째 내재점과 $(n+1)$ -번째 내재점 사이의 시간이라고 하자. 그러면 시간 우선순위 정책에 의해서 다음 관계가 성립한다.

$$T(n) = \begin{cases} S_D & \text{if } N^d(n) \geq N_D^d(n) \geq 1 \\ S_L & \text{if } N^d(n) \geq 1, N_D^d(n) = 0 \\ I + S_D & \text{if } N^d(n) = N_D^d(n) = 0, C(I) = D \\ I + S_L & \text{if } N^d(n) = N_D^d(n) = 0, C(I) = L \end{cases}$$

단, I 는 유희기간의 길이로 λ 의 발생률을 가지는 지수 분포를 따르고, $C(I)$ 는 유희기간에 도착한 패킷의 클래스이다. $T = \lim_{n \rightarrow \infty} T(n)$ 라고 하면, 앞의 관계식에서 다음이 성립한다.

$$E[T] = \lim_{n \rightarrow \infty} \sum_{i=0}^K \sum_{j=0}^i \left[\Pr\{N^d(n) = i, N_D^d(n) = j\} \times E[T(n) | N^d(n) = i, N_D^d(n) = j] \right] \quad (22)$$

$$= E[S_D] \left[\sum_{i=1}^K \sum_{j=1}^i \pi_{(i,j)} + \frac{\lambda_D}{\lambda} \cdot \pi_{(0,0)} \right]$$

$$+ E[S_L] \left[\sum_{i=1}^K \pi_{(i,0)} + \frac{\lambda_L}{\lambda} \cdot \pi_{(0,0)} \right] + \frac{1}{\lambda} \cdot \pi_{(0,0)}$$

$\xi(t) = I$ 는 시스템이 유희기간일 때만 발생한다. 그리고 유희기간 I 는 $N^d(n) = 0$ 일 때 시작하고, 유희기간 I 의 길

이는 $\text{Exp}(\lambda)$ 를 따르므로, 준-마코프과정(Semi-Markov process)에 대한 이론에 의해 다음이 성립한다[15].

$$Q_L = \frac{\pi_{(0,0)}}{\lambda E[T]} \quad (23)$$

마찬가지로 Q_L 과 Q_D 를 임의 시점에 시스템이 클래스 L 과 D 를 전송하고 있을 확률이라고 하면 다음이 성립한다.

$$Q_L = \frac{E[S_L]}{E[T]} \left(\sum_{i=1}^K \pi_{(i,0)} + \frac{\lambda_L}{\lambda} \pi_{(0,0)} \right) \quad (24)$$

$$Q_D = \frac{E[S_D]}{E[T]} \left(\sum_{i=1}^K \sum_{j=1}^i \pi_{(i,j)} + \frac{\lambda_D}{\lambda} \cdot \pi_{(0,0)} \right) \quad (25)$$

그런데 리틀의 법칙에 의해서 다음이 성립해야 한다.

$$Q_L = \lambda_L^c E[S_L], \quad Q_D = \lambda_D^c E[S_D]$$

그러므로 식 (17), 식 (20)~식 (21)를 위의 관계식에 대입한 후 손실 확률과 밀어내기 확률로 정리하면 다음 식을 얻는다.

$$P_L^b = 1 - \frac{Q_L}{\lambda_L E[S_L]} \quad (26)$$

$$P_D^b = 1 - \frac{Q_L}{\lambda E[S_L]} - \frac{Q_D}{\lambda E[S_D]} \quad (27)$$

$$P_L^{po} = P_D^b - P_L^b = \frac{\lambda_D}{\lambda_L} \frac{Q_L}{\lambda E[S_L]} - \frac{Q_D}{\lambda E[S_D]} \quad (28)$$

5. 임의 시점의 패킷수 기대치

안정상태에서 임의 시점의 전체 패킷수를 N , 클래스- D 패킷수를 N_D , 전송 상태를 ξ 라고 하자.

그리고 이 임의 시점에서 가장 직전의 내재점에서의 전체 패킷수를 N^d , 클래스- D 패킷수를 N_D^d 라고 하자. 그리고 $\xi=L$ 또는 D 이어서 임의 시점에 패킷이 전송 중이었다면, 임의 시점의 패킷 전송의 경과 시간(elapsed time)을 S_ξ^E 라고 하자. 임의 시점의 패킷수 (N, N_D) 의 기대치를 구하기 위하여, 직전 내재점의 패킷수 (N^d, N_D^d) 와의 관계를 고려해 보자.

만약 $N^d \geq N_D^d \geq 1$ 라고 하자. 그러면 시간 우선순위에 의해 $\xi=D$ 이고, 3절의 내재점 사이의 패킷수 분포를 구할 때와 유사한 방법으로 다음의 관계를 구할 수 있다. 단,

시스템에 머무는 패킷수에는 전송 중인 패킷의 수도 고려되어야 함에 유의한다.

$$(N, N_D) = \begin{cases} (N^d + A(S_D^E), N_D^d + B_D^{A(S_D^E)}) \\ \quad \text{if } 0 \leq A(S_D^E) < K - N^d + 1, \\ (K+1, N_D^d + B_D^{K-N^d+1} - B_{L,s}^{A(S_D^E)-K+N^d-1}) \\ \quad \text{if } \begin{cases} A(S_D^E) \geq K - N^d + 1 \\ 0 \leq B_{L,s}^{A(S_D^E)-K+N^d-1} < N_D^d - 1 + B_D^{K-N^d+1}, \end{cases} \\ (K+1, 1) \\ \quad \text{if } \begin{cases} A(S_D^E) \geq K - N^d + 1 \\ B_{L,s}^{A(S_D^E)-K+N^d-1} \geq N_D^d - 1 + B_D^{K-N^d+1} \end{cases} \end{cases} \quad (29)$$

마찬가지로 $N^d \geq 1$ 이고 $N_D^d = 0$ 이면, $\xi=L$ 이고 다음 관계를 구할 수 있다.

$$(N, N_D) = \begin{cases} (N^d + A(S_L^E), B_D^{A(S_L^E)}) \\ \quad \text{if } 0 \leq A(S_L^E) < K - N^d + 1, \\ (K+1, B_D^{K-N^d+1} - B_{L,s}^{A(S_L^E)-K+N^d-1}) \\ \quad \text{if } \begin{cases} A(S_L^E) \geq K - N^d + 1 \\ 0 \leq B_{L,s}^{A(S_L^E)-K+N^d-1} < B_D^{K-N^d+1}, \end{cases} \\ (K+1, 0) \\ \quad \text{if } \begin{cases} A(S_L^E) \geq K - N^d + 1 \\ B_{L,s}^{A(S_L^E)-K+N^d-1} \geq B_D^{K-N^d+1} \end{cases} \end{cases} \quad (30)$$

마지막으로 $N^d = N_D^d = 0$ 인 경우에는 유희기간 I 동안 $\xi=I$ 이고 이 동안 $(N, N_D) = (0, 0)$ 이다. 유희기간에 도착한 패킷의 클래스를 h 라고 하면, 유희기간 이후의 전송 시간의 임의 시점의 (N, N_D) 는 다음과 같이 표현된다.

$$(N, N_D) = \begin{cases} (1 + A(S_h^E), \Pi(h=D) + B_D^{A(S_h^E)}) \\ \quad \text{if } 0 \leq A(S_h^E) < K, \\ (K+1, \Pi(h=D) + B_D^K - B_{L,s}^{A(S_h^E)-K}) \\ \quad \text{if } \begin{cases} A(S_h^E) \geq K \\ 0 \leq B_{L,s}^{A(S_h^E)-K} < B_D^K, \end{cases} \\ (K+1, \Pi(h=D)) \\ \quad \text{if } \begin{cases} A(S_h^E) \geq K \\ B_{L,s}^{A(S_h^E)-K} \geq B_D^K \end{cases} \end{cases} \quad (31)$$

단, $\Pi(\cdot)$ 은 조건이 성립하면 1, 아니면 0이 되는 지시 함수이다.

임의 시점에 $\xi=L$ 또는 D 일 때, 전송 경과 시간 S_ξ^E 의 CDF를 $S_\xi^E(t)$ 라고 하자. 그러면 재생 과정 이론에 의해 $S_\xi^E(t)$ 는 다음과 같다[29].

$$S_h^E(t) = \frac{\int_0^t (1 - S_h(x)) dx}{E[S_h]}, \quad h \in \{L, D\}.$$

아울러 $a_n(S_h^E)$, $a_{(n,j)}(S_h^E)$, $\tilde{a}_{(n,j)}(S_h^E)$, $\tilde{a}_{(n,j);l}(S_h^E)$, $\tilde{a}_{(n,j);l+}(S_h^E)$ 를 $a_n(S_h)$, $a_{(n,j)}(S_h)$, $\tilde{a}_{(n,j)}(S_h)$, $\tilde{a}_{(n,j);l}(S_h)$, $\tilde{a}_{(n,j);l+}(S_h)$ 에 대한 정의에서 S_h 를 S_h^E 로 대체한 것으로 정의하도록 한다.

Kim[18]의 4절에서 사용한 준마코프 과정에 대한 이론을 이용하면, 식 (29)~식 (31)의 관계에서 (N, N_D) 의 분포를 EMC의 확률 $\pi_{(i,j)}$ 와 앞에서 정의한 확률들로 표현할 수 있다. 그러나 여기서는 논의의 단순화를 위해 준마코프 과정을 사용하여 (N, N_D) 의 분포가 아니라 기대치만을 구하도록 한다.

식 (29)에 의해 $N^d = i \geq 1$ 이고 $N_D^d = j \geq 1$ 일 때 다음 조건부 기대치를 구할 수 있다.

$$E[N|N^d = i, N_D^d = j] \quad (32)$$

$$= \sum_{n=0}^{K-i} (i+n)a_n(S_D^E) + (K+1) \left(1 - \sum_{n=0}^{K-i} a_n(S_D^E) \right)$$

$$E[N_D|N^d = i, N_D^d = j] \quad (33)$$

$$= \sum_{n=0}^{K-i} \sum_{m=0}^n (j+m)a_{(n,m)}(S_D^E)$$

$$+ \sum_{m=0}^{K-i+1+j+m-2} \sum_{l=0}^m (j+m-l)\tilde{a}_{(K-i+1,m);l}(S_D^E)$$

$$+ \sum_{m=0}^{K-i+1} \tilde{a}_{(K-i+1,m);(j-i+m)+}(S_D^E)$$

식 (30)에 의해 $N^d = i \geq 1$ 이고 $N_D^d = j = 0$ 일 때 다음 조건부 기대치를 구할 수 있다.

$$E[N|N^d = i, N_D^d = 0] \quad (34)$$

$$= \sum_{n=0}^{K-i} (i+n)a_n(S_L^E) + (K+1) \left(1 - \sum_{n=0}^{K-i} a_n(S_L^E) \right)$$

$$E[N_D|N^d = i, N_D^d = 0] \quad (35)$$

$$= \sum_{n=0}^{K-i} \sum_{m=0}^n m a_{(n,m)}(S_L^E)$$

$$+ \sum_{m=0}^{K-i+1} \sum_{l=0}^{m-1} (m-l)\tilde{a}_{(K-i+1,m);l}(S_L^E)$$

마지막으로 식 (31)에 의해 $N^d = N_D^d = 0$ 이고 유희기간 후에 클래스- h 패킷을 전송하는 동안의 N 과 N_D 의 조건부

기대치를 다음처럼 구할 수 있다.

$$E[N|N^d = N_D^d = 0, \xi = h] \quad (36)$$

$$= \sum_{n=0}^{K-1} (n+1)a_n(S_h^E) + (K+1) \left(1 - \sum_{n=0}^{K-1} a_n(S_h^E) \right)$$

$$E[N_D|N^d = N_D^d = 0, \xi = h] \quad (37)$$

$$= \Pi(h = D) + \sum_{n=0}^{K-1} \sum_{m=0}^n m a_{(n,m)}(S_h^E)$$

$$+ \sum_{m=0}^K \sum_{l=0}^{m-1} (m-l) \tilde{a}_{(K,m);l}(S_h^E)$$

따라서 준마코프 과정 이론을 이용하면 유희기간 동안에는 $N = N_D = 0$ 이라는 사실에 의해 임의 시점의 총 패킷수와 클래스- D 패킷수의 기대치는 다음과 같이 표현할 수 있다.

$$E[M] = \sum_{i=1}^K \sum_{j=0}^i \Pr\{N^d = i, N_D^d = j\} E[N|N^d = i, N_D^d = j] \quad (38)$$

$$+ \sum_{h \in \{L, D\}} \Pr\{N^d = N_D^d = 0, \xi = h\}$$

$$\times E[N|N^d = N_D^d = 0, \xi = h]$$

$$E[N_D] = \sum_{i=1}^K \sum_{j=0}^i \Pr\{N^d = i, N_D^d = j\} E[N_D|N^d = i, N_D^d = j] \quad (39)$$

$$+ \sum_{h \in \{L, D\}} \Pr\{N^d = N_D^d = 0, \xi = h\}$$

$$\times E[N_D|N^d = N_D^d = 0, \xi = h]$$

단,

$$\Pr\{N^d = i, N_D^d = j\} = \begin{cases} \frac{E[S_D]}{E[T]} \pi_{(i,j)}, & i \geq j \geq 1 \\ \frac{E[S_L]}{E[T]} \pi_{(i,0)}, & i \geq 1, j = 0 \end{cases}$$

$$\Pr\{N^d = N_D^d = 0, \xi = h\} = \frac{\lambda_h E[S_h]}{\lambda E[T]} \pi_{(0,0)}, \quad h \in \{L, D\}$$

그러므로 식 (32)~식 (37)을 식 (38)과 식 (39)에 대입하면 $E[M]$ 과 $E[N_D]$ 를 계산할 수 있다. 아울러 $N = N_L + N_D$ 이므로 임의 시점의 클래스- L 패킷수 N_L 의 기대치도 다음과 같이 구할 수 있다.

$$E[N_L] = E[M] - E[N_D] \quad (40)$$

클래스- L 의 시스템 평균 체류시간의 기대치 W_L 은 리틀의 법칙에 의해 다음과 같이 구할 수 있다.

$$W_L = \frac{L_L}{\lambda_L^e}$$

클래스- D 의 시스템 평균 체류시간의 기대치 W_D 는 리틀의 법칙에 의해서 계산될 수 없음에 유의한다. 왜냐하면 도착 시 바로 손실되지 않고 시스템에 합류한 클래스- D 는 밀어내기를 당하지 않으면 시스템에서 전송되지만, 밀어내기를 당하면 시스템에서 전송되기 전에 이탈한다. 그러므로 L_D 에는 시스템에서 전송된 패킷뿐 아니라 밀어내기로 이탈된 패킷의 기여분도 포함되어 있다. 그러므로 밀어내기를 당하지 않고 서비스를 완료한 클래스- D 패킷의 대기시간을 분석하려면 또 다른 분석이 필요하다. 본 연구에서는 클래스- D 패킷의 체류시간의 기대치는 구하지 않도록 한다. 클래스- D 패킷의 기대치의 분포를 구하는 방식에 관심있는 독자는 Kim[18]을 참조하기 바란다.

6. 수치 예제

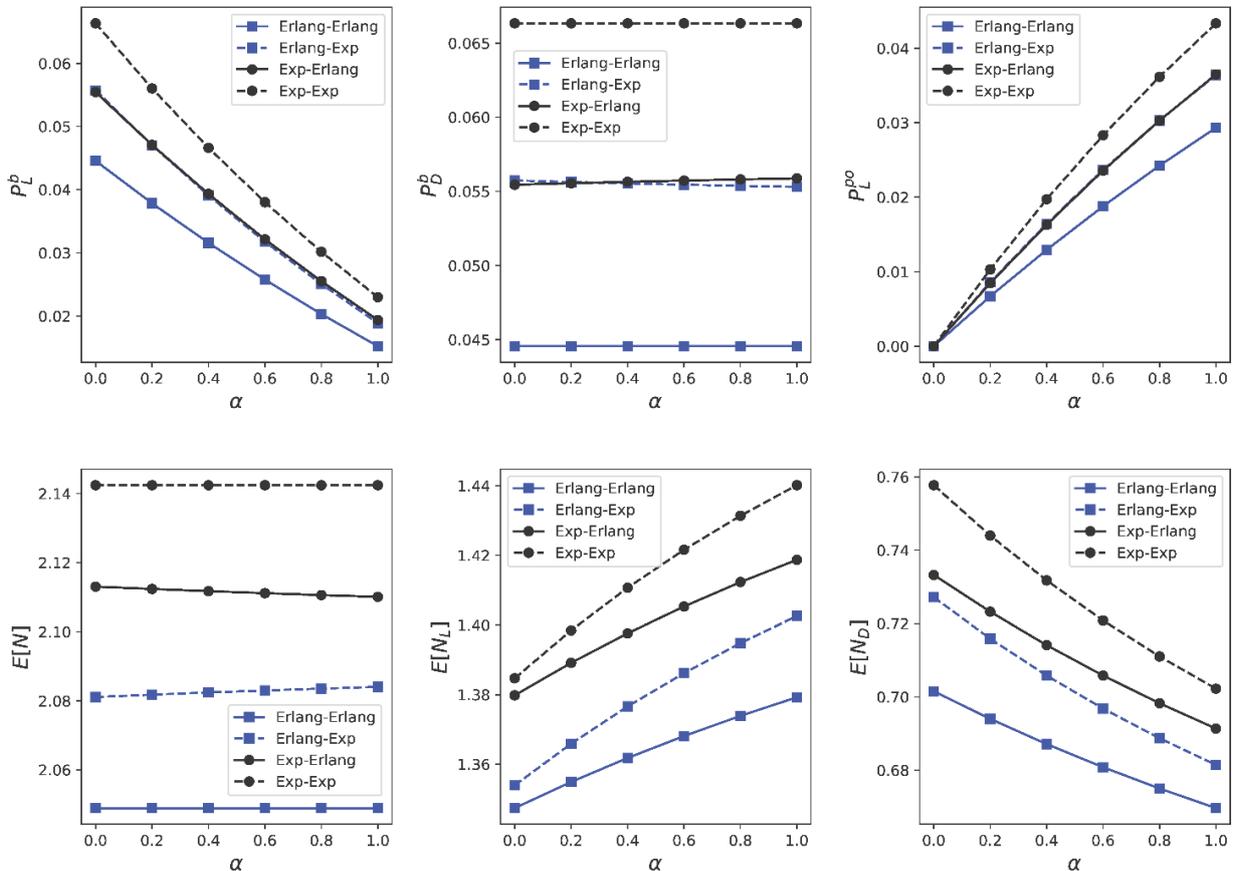
이 절에서는 확률적 밀어내기 정책에서 밀어내기 성공 확률 α 가 변화하면 시스템 성능에 어떠한 영향을 미치는지를 수치 예제를 통하여 살펴본다.

수치 예제는 TCP 같은 손실 민감 트래픽과 UDP 같은

지연 민감 트래픽을 동시에 서비스하고 있는 출력 큐 스위치의 한 출력 전송 링크에서의 전송 성능을 분석한다. 이 전송 링크에 대해 다음과 같은 가정을 하였다. 버퍼의 크기는 $K=5$ 로 가정하였다. 두 트래픽 클래스의 도착률은 $\lambda_L = \lambda_D = 1$ 로 동일하다고 가정하였다. 두 클래스의 패킷 전송의 평균 시간도 $E[S_L] = E[S_D] = 2/5$ 로 동일하다고 가정한다. 패킷 전송시간의 분포에 따른 시스템 성능 차이를 확인하기 위하여, 패킷 전송시간의 분포로 $Exp(5/2)$ 지수 분포와, $Erlang(5,2)$ 인 얼랑분포를 사용하였다. 이 두 분포는 평균은 $2/5$ 로 동일하나 분산이 다르므로, 변동계수 (coefficients of variations)가 1과 $1/\sqrt{2}$ 이 된다.

본 수치 예제의 분석은 매우 복잡한 적분과 미분을 사용하여 EMC의 전이확률 등을 계산하여야 한다. 이를 위하여 기호적 미적분을 수치적 미적분으로 변환하여 수행해줄 수 있는 Python의 sympy 모듈이 계산에 사용되었다. 아울러 수치 배열의 효율적 계산을 위해 numpy가 같이 사용되었다.

<Figure 1>은 밀어내기 성공확률 α 의 변화에 따라 두 클래스의 손실 확률 P_L^b 과 P_D^b , 클래스- L 의 밀어내기 확률 P_L^{po} , 시스템에 머무는 평균 패킷수 $E[M]$ 과 두 클래스의



<Figure 1> Performance Measures over Various Values of Pushout Probability (legend: class-L distribution, class-D distribution)

평균 패킷수 $E[N_L]$ 과 $E[N_D]$ 의 변화를 보여준다. 패킷 전송시간 분포에 따른 차이를 보기 위하여, 클래스- L 패킷의 전송시간 분포별로 그래프의 점의 모양을 바꾸어 지수분포이면 원으로, 알량분포이면 사각형으로 표현하였으며, 클래스- D 패킷의 전송시간 분포별로 그래프의 선의 모양을 바꾸어 지수분포이면 파선으로, 알량분포이면 실선으로 표현하였다(범례의 레이블은 (클래스- L 분포, 클래스- D 분포) 형식으로 표현하였다).

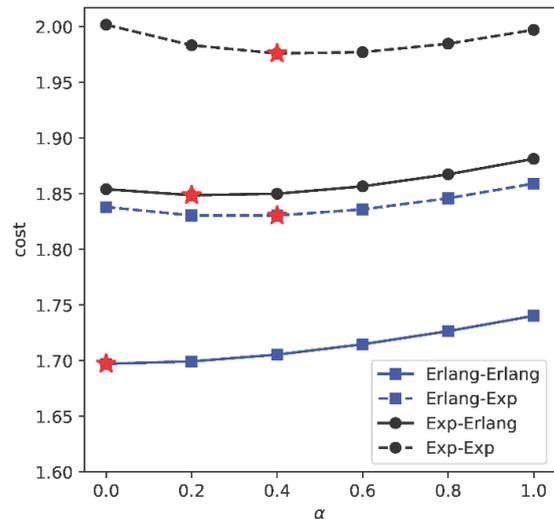
<Figure 1>의 상단 그래프를 보면, 전송시간의 분포와 상관없이 밀어내기 성공확률 α 가 커질수록 클래스- L 의 손실확률 P_L^o 은 감소하고 도착한 클래스- L 이 클래스- D 를 실제로 밀어내는 확률 P_L^p 은 증가하는 것을 볼 수 있다. 밀어내기 성공확률 α 의 증가는 클래스- L 의 공간 우선순위를 더 강하게 부여한 것이므로 이러한 결과는 직관적인 예상에 부응한다.

반면 클래스- D 의 도착시 손실 확률 P_D^b 는 거의 변화가 없다. 특히 두 클래스의 패킷 전송시간 분포가 동일하면 밀어내기 성공확률이 증가하더라도 버퍼에 있는 클래스- D 패킷이 동일한 전송시간 분포를 가진 클래스- L 패킷으로 바뀐 것이므로, 미래에 도착하는 클래스- D 패킷의 도착시 손실확률은 전혀 변화가 없게 된다(<Figure 1>의 상단 가운데 그래프에서 Erlang-Erlang과 Exp-Exp 경우를 참조.) 이러한 사실은 <Figure 1>의 하단 원편의 $E[N]$ 그래프에서도 확인할 수 있다. 두 클래스의 패킷 전송시간 분포가 동일하면, 밀어내기가 증가하더라도 시스템의 전체 패킷수의 기대치는 변화하지 않는 것을 확인할 수 있다.

두 클래스의 패킷 전송시간 분포가 다른 경우에는 밀어내기 성공확률이 증가하면 P_D^b 는 미세하게 변화한다. 밀어내기에 의해 버퍼에 있는 총 패킷의 수는 변화하지 않지만, 버퍼에 있는 패킷의 전송시간의 분포가 달라지므로 미래에 버퍼가 가득 찰 확률이 크지 않지만 변화하기 때문이다. 클래스- L 패킷이 지수분포이고 클래스- D 패킷이 알량분포인 경우 α 가 증가함에 따라 P_D^b 가 미세하게 증가하는 것을 볼 수 있다. 밀어내기 성공이 늘어날수록 좀 더 변동성이 큰 지수분포의 패킷이 시스템에 증가하게 되고, 이는 다시 버퍼가 가득 찰 확률을 증가시켜 클래스- D 패킷의 도착시 손실확률을 증가시키게 된다. 그런데 이러한 효과가 미세한 이유는 버퍼의 가득 찰 확률이 증가하면 클래스- D 패킷의 시스템 입장이 줄어들어 버퍼의 패킷수를 낮추는 역의 효과가 발생하기 때문이다. 그렇기 때문에 <Figure 1>의 하단 원편의 $E[N]$ 그래프를 보면 전체 패킷수의 기대치는 α 가 증가함에 따라 미세하게 감소하는 경향을 보였다.

두 클래스의 시스템에 머무는 패킷수의 기대치는 α 가 증가함에 따라 $E[N_L]$ 은 증가하고 $E[N_D]$ 는 감소한다. 이는

α 의 증가가 클래스- L 패킷의 공간 우선순위를 증가시키기 때문에 발생하는 효과이다. 아울러 모든 분포에서 클래스- D 패킷수의 기대치가 클래스- L 패킷수의 기대치보다 작는데, 이는 공간 우선순위 때문에 시스템에서 손실되는 클래스- D 패킷이 많을 것에 기인한 것뿐 아니라, 시간 우선순위에 의해서 시스템에서 머무는 시간도 더 적기 때문에 기인한 점도 있다.



<Figure 2> Costs over Various Values of Pushout Probability (legend: class- L distribution, class- D distribution)

마지막으로 본 논문의 방법이 Carballo-Lozano et al.[7]과 Kim[18]의 방법을 어떻게 종합적으로 확장하고 있는지를 <Figure 1>의 수치 예제를 통하여 살펴보자. 확률적 밀어내기 정책의 장점은 밀어내기 확률 α 를 미세 조정함으로써 시스템의 성능을 최적화할 수 있다는 것이다.

시스템 성능의 최적화를 위해 다음과 같은 서비 품질 비용함수를 가정하자.

$$50\lambda_L(P_L^b)^2 + 5(\lambda_D P_D^b + \lambda_L P_L^o) + 0.5E[N_L] + E[N_D]$$

위의 비용함수는 손실에 민감한 클래스- L 트래픽은 손실확률의 제공에 비례하여 품질 비용이 발생하고, 손실에 상대적으로 둔감한 클래스- D 트래픽의 서비스 품질 비용은 손실확률에 선형적으로 비례한다고 가정한 것이다. 아울러 클래스- L 과 클래스- D 트래픽 모두 패킷의 지연 시간에 선형적으로 비례하여 서비스 품질 비용이 발생하는데, 지연에 민감한 클래스- D 트래픽이 클래스- L 트래픽보다 단위 시간당 지연에 따른 품질 비용이 2배 더 많이 발생한다고 가정되었다. 실제 문제에서의 비용함수는 사용자의 효용 함수에 대한 가정에 따라 달라질 것이다.

<Figure 2>는 <Figure 1>과 같은 성능 지표가 예상될 때, 두 트래픽의 패킷의 전송시간 분포별로 밀어내기 확률 α 가 변함에 따라 비용함수가 어떻게 변화하는지를 보여 준다. 패킷의 전송시간 분포별로 비용이 최소화되는 α 는 별표로 표시하였다.

Carballo-Lozano et al.[7]의 모형은 패킷의 전송시간이 지수분포일 때만 다루었다. 반면 본 연구는 패킷의 분포가 지수분포가 아닌 경우로 확장을 하였다. <Figure 1> 예에서 보듯이 패킷의 전송시간 분포가 지수분포((Exp-Exp)인 경우와 아닌 경우에 성능지표는 뚜렷한 차이를 보인다. 아울러 <Figure 2> 예에서 보듯이 패킷의 분포가 일량 분포 등의 다른 분포를 따르는데, 지수분포를 사용하여 근사하면 최적 밀어내기 확률 α 가 잘못 설정될 수 있다. 예를 들어 실제 패킷의 전송시간 분포가 일량분포를 따르며 밀어내기 확률 $\alpha=0$ 으로 설정하는 것이 최적이지만, 지수분포로 근사한 경우에는 확률 $\alpha=0.4$ 가 최적으로 잘못 판단하게 되고, 비용에 대한 예측도 차이가 발생하게 된다.

한편 Kim[18]의 모형은 패킷의 전송시간 분포를 모두 고려하였지만 항상 밀어내기를 실행하는, 즉, $\alpha=1$ 인 모형만을 다루었다. <Figure 2>에서 보듯이 패킷의 전송시간 분포에 따라 최적인 밀어내기 확률 α 가 달라짐을 볼 수 있다. 따라서 Kim[18]처럼 확정적 밀어내기를 하는 것보다, 본 연구의 확률적 밀어내기 정책을 사용하는 모형이 다양한 분포 하에서 시스템의 성능을 유연하게 최적화할 수 있음을 확인할 수 있다.

7. 결 론

본 연구에서는 기존의 확정적 밀어내기 정책을 가지는 공간-시간 우선순위 유한용량 M/G/1 대기행렬에 대한 연구를 확률적 밀어내기 정책을 가지는 모형에 대한 연구로 확장하였다. 이를 위해 제안한 대기행렬 모형의 패킷 전송 완료 시점의 패킷수 분포를 내재점 마코프 과정으로 구하였고, 준 마코프 과정 이론을 이용하여 클래스별 패킷 손실확률과 임의시점의 패킷수의 기대치를 구하였다. 그리고 제안한 분석의 유용성을 보이고, 확률적 밀어내기 정책이 시스템 성능에 미치는 영향을 파악하기 위하여, 네 가지 전송 시간 분포의 조합에 대하여 밀어내기 성공확률 α 의 변화에 따른 시스템 성능의 변화를 수치 예제로 탐구하였다.

서로 이질적인 요구사항을 가지는 트래픽을 서비스하는 통신 시스템에는 트래픽마다 시간과 공간에 대한 우선순위를 다르게 부여할 필요가 있다. 이러한 시스템에 대한 정밀한 성능 분석을 위해서는 공간-시간 우선순위를 가지는 대기행렬에 대한 연구가 필요하다. 그러나 기존의 연구

가 패킷 전송시간을 동일한 지수분포로 가정하거나, 일반 분포로 가정하더라도 확정적인 밀어내기 정책을 가정하여 공간 우선순위에 대한 정밀한 조정이 어려웠다. 반면 본 연구의 결과를 사용하면 밀어내기 확률 α 의 값을 조정하여 공간 우선순위에 대해 좀 더 미세한 조정을 할 수 있고, 그러한 조정이 시스템 성능에 미치는 영향을 추론해 낼 수 있게 된다. 그러므로 본 연구의 결과는 이질적인 트래픽을 서비스하는 시스템의 성능 분석을 도와 시스템 운영자가 더 최적의 시스템을 구축할 수 있도록 도울 수 있다는 데에 그 의의가 있다.

마지막으로 본 연구의 향후 발전 방향을 제시하도록 한다. 이 연구에서는 확률적 밀어내기 정책을 도입하여 공간 우선순위를 미세 조정할 수 있도록 하였다. 반면 시간 우선순위는 항상 비선점 우선순위 정책을 따르게 하였다. 그러나 시간 우선순위도 시스템에 머무는 각 클래스의 패킷수 등을 고려하여 동적으로 우선순위를 조정하는 것이 시스템 전체를 보았을 때 더 효율적일 수 있다. 그러므로 좀 더 복잡한 시간 우선순위 정책을 사용하여 클래스 간의 시간 우선순위를 조정할 수 있는 대기행렬 모형에 대한 연구가 진행된다면 더 현실에 유용한 연구가 될 것이라고 기대된다.

Acknowledgement

This research was funded by a 2022 research grant from Sangmyung University.

References

- [1] Al-Begain, K., Dudin, A., Kazimirsky, A., and Yerima, S., Investigation of the M2/G2/1/, N Queue with Restricted Admission of Priority Customers and its Application to HSDPA Mobile Systems, *Computer Networks*, 2009, Vol. 53, No. 8, pp. 1186-1201.
- [2] Allman, M., Paxson, V. and Blanton, E., TCP Congestion Control, RFC 5681, Internet Engineering Task Force, 2009.
- [3] Arpaci, M. and Copeland, J.A., Buffer Management for Shared-memory ATM Switches, *IEEE Communications Surveys & Tutorials*, 2000, Vol. 3, No. 1, pp. 2-10.
- [4] Avrachenkov, K.E., Vilchevsky, N.O., and Shevlyakov, G.L., Priority Queueing with Finite Buffer Size and Randomized Push-out Mechanism, *ACM SIGMETRICS Performance Evaluation Review*, 2003, Vol. 31, No. 1, pp. 324-325.
- [5] Avrachenkov, K.E., Vilchevsky, N.O., and Shevlyakov,

- G.L., Priority Queueing with Finite Buffer size and Randomized Push-out Mechanism, *Performance Evaluation*, 2005, Vol. 61, No. 1, pp. 1-16.
- [6] Avrachenkov, K., Shevlyakov, G., and Vilchevskii, N., Randomized Push-out Disciplines in Priority Queueing, *Journal of Mathematical Sciences*, 2004, Vol. 122, No. 4, pp. 3336-3342.
- [7] Carballo-Lozano, C., Ayesta, U., and Fiems, D., Performance Analysis of Space-time Priority Queues, *Performance Evaluation*, 2019, Vol. 133, pp. 25-42.
- [8] Cardoso, K.V., de Rezende, J.F., and Fonseca, N.L., On the Effectiveness of Push-out Mechanisms for the Discard of TCP Packets, *2002 IEEE international conference on communications*, Conference proceedings, ICC 2002 (cat. No. 02CH37333), 2002, pp. 2636-2640.
- [9] Chang, C.G. and Tan, H.H., Queueing Analysis of Explicit Policy Assignment Push-out Buffer Sharing Schemes for ATM Networks, *Proceedings of INFOCOM'94 Conference on Computer Communications*, 1994, Toronto, ON, Canada, pp. 500-509.
- [10] Cheng, X. and Akyildiz, A., A Finite Buffer Two Class Queue with Different Scheduling and Push-out Schemes, *[Proceedings] IEEE INFOCOM'92: The conference on computer communications*, 1992, pp. 231-241.
- [11] Demoor, T., Walraevens, J., Fiems, D., De Vuyst, S., and Bruneel, H., Influence of Real-time Queue Capacity on System Contents in DiffServ's Expedited Forwarding Per-hop-behavior, *Journal of Industrial and Management Optimization*, 2010, Vol. 6, No. 3, pp. 587-602.
- [12] Eugster, P., Kesselman, A., Kogan, K., Nikolenko, S., and Sirotkin, A., Admission Control in Shared Memory Switches, *Journal of Scheduling*, 2018, Vol. 21, No. 5, pp. 533-543.
- [13] Gravey, A., Boyer, P., and Hebuterne, G., Tagging Versus Strict Rate Enforcement in ATM Networks, *IEEE Global Telecommunications Conference GLOBECOM'91: Countdown to the New Millennium*, Conference record, 1991, Phoenix, AZ, USA, pp. 271-275.
- [14] Gravey, A. and Hebuterne, G., Mixing Time and Loss Priorities in a Single Server Queue, *Proceedings of 13th International Teletraffic Congress*, 1991, Copenhagen, Denmark, pp. 147-152.
- [15] Heyman, D.P. and Sobel, M.J., Stochastic models in operations research. 1, Stochastic processes and operating characteristics, McGraw-Hill, 1982.
- [16] Kapadia, A.S., Kazmi, M.F., and Mitchell, A.C., Analysis of a Finite Capacity Non Preemptive Priority Queue, *Computers & Operations Research*, 1984, Vol. 11, No. 3, pp. 337-343.
- [17] Kim, K., Delay Cycle Analysis of Finite-buffer M/G/1 Queues and its Application to the Analysis of M/G/1 Priority Queues with Finite and Infinite Buffers, *Performance Evaluation*, 2020, Vol. 143, pp. 102-133. DOI:<https://doi.org/https://doi.org/10.1016/j.peva.2020.102133>.
- [18] Kim, K., Finite-buffer m/g/1 Queues with Time and Space Priorities, *Mathematical Problems in Engineering*, 2022, Vol. 2022.
- [19] Kim, K., M/G/1 Preemptive Priority Queues with Finite and Infinite Buffers, *Journal of the Society of Korea Industrial and Systems Engineering*, 2020, Vol. 43, No. 4, pp. 1-14.
- [20] Kim, K., (N, n)-preemptive Priority Queues, *Performance Evaluation*, 2011, Vol. 68, No. 7, pp. 575-585.
- [21] Kim, K., (N, n)-preemptive Repeat-different Priority Queues, *Journal of Society of Korea Industrial and Systems Engineering*, 2017, Vol. 40, No. 3, pp. 66-75.
- [22] Kim, K., The Analysis of an Opportunistic Spectrum Access with a Strict T-preemptive Priority Discipline, *Journal of Society of Korea Industrial and Systems Engineering*, 2012, Vol. 35, No. 4, pp. 162-170.
- [23] Kim, K., T-preemptive Priority Queue and its Application to the Analysis of an Opportunistic Spectrum Access in Cognitive Radio Networks, *Computers & Operations Research*, 2012, Vol. 39, No. 7, pp. 1394-1401.
- [24] Kleinrock, L., Queueing systems, volume 2: Computer applications, Wiley, 1976.
- [25] Kroner, H., Hebuterne, G., Boyer, P. and Gravey, A., Priority Management in ATM Switching Nodes, *IEEE Journal on Selected Areas in Communications*, 1991, Vol. 9, No. 3, pp. 418-427.
- [26] Lee, Y. and Choi, B.D., Queueing System with Multiple Delay and Loss Priorities for ATM Networks, *Information Sciences*, 2001, Vol. 138, No. 1-4, pp. 7-29.
- [27] Lee, Y., Choi, B.D., Kim, B., and Sung, D.K., Delay Analysis of an m/g/1/k Priority Queueing System with Push-out Scheme, *Mathematical Problems in Engineering*, 2007, Vol. 2007.
- [28] Lin, A.-M. and Silvester, J.A., Priority Queueing Strategies and Buffer Allocation Protocols for Traffic Control at an ATM Integrated Broadband Switching System, *IEEE Journal on Selected Areas in Communications*, 1991, Vol. 9, No. 9, pp. 1524-1536.

- [29] Ross, S.M., Stochastic processes, John Wiley & Sons, 1996.
- [30] Sharma, V. and Virtamo, J.T., A Finite Buffer Queue with Priorities, *Performance Evaluation*, 2002, Vol. 47, No. 1, pp. 1-22.
- [31] Stewart, W.J., Numerical Methods for computing stationary distributions of finite irreducible markov chains, *Computational probability*, Springer, pp. 81-111.
- [32] Takagi, H., Queueing analysis, volume 1: Vacation and priority systems, part 1, North-Holland, 1991.
- [33] Xue, H., Kim, K.T., and Youn, H.Y., Packet Scheduling for Multiple-switch Software-defined Networking in Edge Computing Environment, *Wireless Communications and Mobile Computing*, 2018, Vol. 2018.
- [34] Zaborovsky, V., Zayats, O., and Mulukha, V., Priority queueing with finite buffer size and randomized push-out mechanism, *2010 ninth International Conference on Networks*, 2010, pp. 316-320.

ORCIDKilhwan Kim | <http://orcid.org/0000-0002-0577-7906>