

## OFDMA/TDD 시스템의 효용을 최대화하기 위한 패킷 스케줄러

노정현<sup>0</sup>, 최진구<sup>\*\*</sup>, 최진희<sup>\*</sup>, 최진영<sup>\*</sup>

<sup>\*</sup>고려대학교 컴퓨터학과  
{jhnoh<sup>0</sup>@formal, jhchoi@os, choi@formal}.korea.ac.kr,

<sup>\*\*</sup>KT 컨버전스 연구소  
jinchoi@kt.co.kr

## Packet Scheduler Maximizing the System Utility in OFDMA/TDD

Jung Hyun Noh<sup>0</sup>, Jin-Hee Choi<sup>\*</sup>, Jin-Young Choi<sup>\*</sup>  
<sup>\*</sup>Dept of Computer Science & Engineering, Korea University

Jin-Ghoo Choi<sup>\*\*</sup>  
<sup>\*\*</sup>KT Convergence Laboratory

### 요약

비례 공평성 스케줄러는 시간에 따라 변하는 무선 채널의 특성을 이용하는 기회주의적 패킷 스케줄링 방식으로 시스템 수율을 크게 증가시키는 장점이 있으나 1) 로그 형태의 효용 함수만을 지원하고 2) 사용자 QoS를 고려하지 않으며 3) 단일 채널 시스템에만 적용 가능한 한계를 갖는다. 본 논문에서는 WiBro에서 채택하고 있는 다중 접속 방식인 OFDMA/TDD 환경에서 이러한 한계를 극복한 스케줄러를 제안하고 모의 실험에 의해 그 성능을 평가한다.

### 1. 서론

인터넷과 무선 통신의 발달에 힘입어 최근 들어 cdma2000 1xEV-DO나 WCDMA-HSDPA 같이 고속의 패킷 서비스를 지원하는 셀룰러 망들이 소개되고 있다. 이러한 시스템은 각 사용자가 자신의 채널 상태를 측정하여 주기적으로 기지국에 회송(feedback)하는 채널 구조를 가지고 있으며 링크 적응 기술과 결합됨으로써 기회주의적(opportunistic) 패킷 스케줄링을 가능하게 한다. 기회주의적 스케줄링이란, 현재 채널 상태가 좋은 사용자를 우선적으로 서비스하는 방식으로 시스템 수율(throughput)을 크게 증가시킨다. 그러나, 지연에 민감한 서비스에는 적합하지 않으며 지연에 둔감한 데이터 서비스의 경우에도 사용자의 위치에 따른 공평성 문제가 있을 수 있다.

많은 기회주의적 스케줄러가 제안되어 왔는데 [1]-[3], 각 스케줄러는 시스템 효용 또는 수율을 극대화하려는 공통의 목적을 가지며 사용자에게 공평성을 제공하는 방식에서 서로 상이하다. 비례 공평성(proportional fair) 스케줄러는 로그 형태의 효용 함수를 도입함으로써 간접적으로 공평성 문제를 해결하였으며, [2]와 [3]은 각 사용자에게 최소한의 QoS를 보장하는 방식을 채택하였다. 제안하는 스케줄러는 사용자에게 QoS를 제공하는 동시에 시스템

효용을 극대화한다는 측면에서 [2]와 유사하지만 보다 일반적인 효용 모델에 기반 한다는 점에서 차별화 된다. 제안하는 스케줄러는 효용 모델에 따라 [2] 혹은 비례 공평성 스케줄러로 단순화될 수 있으나, [2]는 비례 공평성 스케줄러로 변형될 수 없다. 뿐만 아니라 기존의 스케줄러들이 주로 단일 채널을 공유하는 시스템을 대상으로 한 반면, 제안하는 방식은 다중 채널을 갖는 OFDMA/TDD 시스템을 지원한다. [3]은 이미 다중 채널 환경을 고려한 스케줄러에 대해 연구하였으나 최적 스케줄러의 형태는 제시하지 못했다.

이후의 전개는 다음과 같다. 먼저, 2장에서 OFDMA/TDD 시스템에 대해 간단히 소개하고 3장에서 최적의 패킷 스케줄러를 제안한다. 4장에서는 모의 실험을 통해 제안하는 방식의 성능을 보이고 5장에서 본 연구의 기여를 정리하면서 글을 맺는다.

### 2. 시스템 모델

본 논문에서 고려하는 OFDM 시스템은  $M \times S$  개의 부반송파를 가지고 있으며, 주파수 상에서 인접한  $S$  개의 부반송파들을 결합하여  $M$  개의 부채널을 구현한다.  $K$  개의 연속된 OFDM 심볼들이 모여 하나의 시간 슬롯을 구성하는데, 각 슬롯에서  $M$  개의 부채널은 서로 다른 사용자에게 할당될 수 있다. 각

부채널을 이루는 부반송파들은 응집 대역 (coherent bandwidth)에 속하며 슬롯의 길이는 응집 시간 (coherent time)에 비해 충분히 짧아, 슬롯과 부채널이 주어졌을 때 채널 이득은 일정하다고 가정한다. 또한, OFDM 심볼에는 충분한 파일럿 신호가 삽입되어 있어 수신자는 임의의 슬롯에서 각 부채널의 채널 이득을 정밀하게 측정할 수 있다. 측정 결과는 회송 채널을 통해 지속적으로 기지국 상의 패킷 스케줄러에게 전달되므로 스케줄러는 각 부채널을 특정 사용자에게 할당할 때 얼마의 속도로 전송할 수 있는지 알 수 있다.

슬롯  $i$ 에서 부채널  $j$ 를 사용자  $k$ 에게 할당했을 때 실현 가능한 최대 전송 속도를  $R_{kj}^i$ 로 표기하자. 무선 채널의 특성상  $\{R_{kj}^i\}$ 는 확률 과정이 되는데, 이 확률 과정이 정적이라고 가정하면 혼돈의 여지 없이 인덱스  $i$ 를 생략할 수 있다.  $N$ 명의 사용자가 서비스되고 있다고 가정할 때, 패킷 스케줄러  $Q$ 는 매 슬롯  $(R_{11}, R_{12}, \dots, R_{NM})$ 을 입력 받아 이를 고려하여 부채널 당 한 명의 사용자를 선택한다. 부채널  $j$ 를 할당 받은 사용자  $q_j$ 라 하면 임의의 스케줄러  $Q$ 는  $(q_1, \dots, q_M) = Q(R_{11}, \dots, R_{NM})$ 의 형태로 나타낼 수 있다. 또한, 사용자  $k$ 가 부채널  $j$ 를 할당 받으면 1, 그렇지 않으면 0의 값을 갖는 지시 함수  $I_{(q_j=k)}$ 를 도입함으로써, 스케줄러  $Q$ 에 의해 사용자  $k$ 가 받는 평균 전송 속도를  $\bar{R}_k^Q = \sum R_{kj}^Q = \sum E\{R_{kj} I_{(q_j=k)}\}$ 로 쓸 수 있다.

IP망에서 사용되는 전송 프로토콜인 TCP는 망의 지나친 혼잡을 피하기 위해 상황에 따라 전송 속도를 변화시키는데 이러한 성질을 탄성적이라고 표현한다. 탄성 연결의 경우 평균 전송 속도가 증가하더라도 그로 인한 만족감 (효용)은 선형적으로 늘어나지 않고 단지 대수적으로 증가하는 것으로 받아들여 지고 있다. 이러한 이론적 바탕 위에서, 비례 공평성 스케줄러는 각 사용자의 효용이 평균 전송 속도의 로그로 주어진다고 모델링하고 사용자 효용의 합을 최대화하였다. 비례 공평성 스케줄러는 구현이 간단하고 시스템 효용을 극대화할 뿐 아니라 각 사용자에게 비슷한 양의 슬롯을 할당하는 특성을 가져 이론적으로, 실용적으로 많은 관심을 끌어들였다. 본 논문에서 제안하는 스케줄러는 임의의 오목 (concave) 함수를 효용 함수로 지원하며 각 사용자가 원하는 만큼의 슬롯과 부채널을 보장하며 다중 채널 환경에서도 사용할 수 있으므로 비례 공평성 스케줄러를 일반화한 형태라고 볼 수 있다.

### 3. 패킷 스케줄러

#### 3.1 제안하는 스케줄러

사용자  $k$ 가  $a_k$  이상의 비율로 슬롯과 부채널을 할당해 줄 것을 요구한다고 가정하자. 스케줄러  $Q$ 에 의해 사용자  $k$ 가 얻는 효용을  $U_k(\bar{R}_k^Q)$ 로 표기하면 최적의 스케줄러를 찾는 문제는

$$\text{maximize } \sum U_k(\bar{R}_k^Q)$$

$$\text{subject to } \Pr\{Q=k\} = M^{-1} \sum \Pr\{q_j = k\} \geq a_k$$

로 정형화된다. 위 식에서 효용 함수  $U_k(\cdot)$ 는 미분 가능한 연속 함수로서 오목하다고 가정한다. 아래의 정리로부터 우리는  $U'_k(\bar{R}_k^Q)R_{kj} + \lambda_k$ 를 최대화하는 사용자  $k$ 에게 부채널  $j$ 를 할당하는 스케줄러가 이 문제의 해가 됨을 알 수 있다. 최적화 이론의 관점에서 볼 때  $\lambda_k$ 는 라그랑주 승수에 해당하며 다음의 세 가지 조건,

- 1)  $\lambda_k \geq 0$ ,
- 2)  $\lambda_k(\Pr\{Q=k\} - a_k) = 0$ ,
- 3)  $\Pr\{Q=k\} - a_k \geq 0$

을 만족하는 상수이다.

정리:  $q_j = \arg \max U'_k(\bar{R}_k^Q)R_{kj} + \lambda_k$ 인 스케줄러  $Q$ 는 적절한 상수  $\lambda_k$ 에 대해, 사용자  $k$ 에게  $a_k$  이상의 슬롯과 부채널을 보장하며 동시에 시스템 효용을 최대화한다.

증명:  $\Pr\{\hat{Q}=k\} \geq a_k$ 를 만족하는 임의의 스케줄러  $\hat{Q}$ 를 선택하자. 두 스케줄러  $Q$ 와  $\hat{Q}$ 에 대해

$$\begin{aligned} & \sum_{j=1}^M \sum_{k=1}^N (U'_k(\bar{R}_k^Q)R_{kj} + \lambda_k) I_{(q_j=k)} \\ & \geq \sum_{j=1}^M \sum_{k=1}^N (U'_k(\bar{R}_k^Q)R_{kj} + \lambda_k) I_{(\hat{q}_j=k)} \end{aligned} \quad (1)$$

이 성립한다. 양변의 기대값을 취하고 식을 정리하면,

$$\begin{aligned} & \sum_{k=1}^N (U'_k(\bar{R}_k^Q)\bar{R}_k^Q + \lambda_k M \Pr\{Q=k\}) \\ & \geq \sum_{k=1}^N (U'_k(\bar{R}_k^{\hat{Q}})\bar{R}_k^{\hat{Q}} + \lambda_k M \Pr\{\hat{Q}=k\}) \end{aligned} \quad (2)$$

$$\Leftrightarrow \sum_{k=1}^N U'_k(\bar{R}_k^Q)(\bar{R}_k^{\hat{Q}} - \bar{R}_k^Q) \leq -\lambda_k M \sum_{k=1}^N (\Pr\{\hat{Q}=k\} - a_k)$$

을 얻는다. 또한, 스케줄러  $\hat{Q}$ 는 사용자의 QoS 요구를 만족시키므로 우변은 0보다 작거나 같고, 따라서 다음의 부등식이 성립한다.

$$\sum_{k=1}^N U'_k(\bar{R}_k^Q)(\bar{R}_k^{\hat{Q}} - \bar{R}_k^Q) \leq 0 \quad (3)$$

평균 속도 벡터  $\bar{\mathbf{R}}_0 = (\bar{R}_1^Q, \dots, \bar{R}_N^Q)$ 와  $\bar{\mathbf{R}}_{\hat{Q}} = (\bar{R}_1^{\hat{Q}}, \dots, \bar{R}_N^{\hat{Q}})$ 을 정의하고 시스템 효용을  $U(\mathbf{r}) = \sum U_k(r_k)$ 로 표기하면 위의 부등식은

$$\nabla U(\bar{\mathbf{R}}_0) \cdot (\bar{\mathbf{R}}_{\hat{Q}} - \bar{\mathbf{R}}_0) \leq 0 \quad (4)$$

가 되며, 이 식은 함수  $U(\mathbf{r})$ 가  $\mathbf{r} = \bar{\mathbf{R}}_0$ 에서 최대값을 가짐을 의미한다.

#### 3.2 매개 변수 추정

제안한 스케줄러는 매개 변수  $\lambda_k$ 와  $\bar{R}_k^Q$ 를 필요로 한다.  $\bar{R}_k^Q$ 는 사용자  $k$ 의 평균 전송 속도를 나타내는 값으로서 이동 평균에 의해 쉽게 구할 수 있으며  $\lambda_k$ 는 이번 절에서 제시하는 알고리즘에 의해 추정한다.

먼저, 벡터 함수  $(f_1, \dots, f_N) = \mathbf{f}(\lambda_1, \dots, \lambda_N)$ 를 정의하고

$k$  번째 항을  $f_k(\lambda_1, \dots, \lambda_N) = \Pr\{Q(\lambda_1, \dots, \lambda_N) = a_k\}$  라고 하자. 제안하는 알고리즘은 함수  $f(\cdot) = \mathbf{0}$  으로 만드는 벡터  $\lambda = (\lambda_1, \dots, \lambda_N)$  를 음이 아닌 영역에서 반복적(iterative) 방법으로 찾는다.

1)  $\lambda^0 = \mathbf{0}$  으로 초기화한다.

2)  $i$  번째 슬롯에서 최적의  $\lambda$  에 대한 추정값인  $\lambda^i$  를 이용하여 스케줄러  $Q^i = Q(\lambda^i)$  를 구한다.

3) 스케줄러  $Q^i$  에 의한  $f_k(\lambda^i)$  는 스케줄링 확률  $\Pr\{Q^i = k\}$  을 포함하므로 정확히 알 수 없고 대신

$g_k(\lambda^i) = M^{-1} \sum_{j=1}^M I_{(q_j=k)} - a_k$  로 추정한다.

4)  $\lambda^{i+1}$  은  $\lambda_k^{i+1} = \max\{0, \lambda_k^i - \delta^i g_k(\lambda^i)\}$  에 의해 갱신되며,  $\delta^i$  는 관측 에러에 대한 가중치로서  $\delta^i > 0$ ,  $\delta^i \rightarrow 0$ ,  $\sum \delta^i \rightarrow \infty$  의 조건을 만족한다.

3)에서  $f_k(\lambda^i)$  와  $g_k(\lambda^i)$  의 평균적인 오차가 0이라는 것과 4)에서 에러에 대한 가중치 수열  $\delta^i$  가 위의 세 가지 조건을 만족한다는 것은 알고리즘의 수렴성을 보장하는 중요한 요소이다.

4. 모의 실험

OFDMA/TDD 시스템에서 각 부채널은 서로 영향을 주지 않으므로, 부채널 별로 독립적인 패킷 스케줄러를 운영하는 것이 가능하다. 제안하는 방식에서  $M=1$  로 설정하면 단일 채널 시스템에 대한 최적 스케줄러를 얻을 수 있다. 단일 채널 환경에서 최적인 스케줄러를 각 부채널 별로 병렬적으로 운영하는 방식을 부채널 별 최적 스케줄링이라고 부르며 제안하는 방식의 성능을 평가하기 위한 비교 대상으로 선정한다.

모의 실험을 단순화하고 제안하는 스케줄러의 특성을 명확히 하기 위해 부채널의 수  $M$  은 2, 사용자의 수  $N$  은 4로 선택한다. 사용자의 최대 전송 속도는 매 슬롯마다 지수 분포에 따라 독립적으로 결정되며 평균값은 부채널 1에서 각각 100, 100, 20, 180 (kbps), 부채널 2에서는 100, 100, 180, 20이다. 제안하는 방식과 부채널 별 최적 스케줄링의 차이를 극명하게 드러내기 위하여 사용자 3과 4는 두 부채널 사이에서 극단적인 비대칭성을 갖도록 설정되었다. 각 사용자는 25% 이상의 슬롯과 부채널을 할당해 줄 것을 요구하며 사용자의 효용은 평균 전송 속도와 동일하다고 가정하였다.  $\lambda$  의 적용에 필요한 가중치 수열  $\delta^i$  는  $1/\ln i$  로 선택하였고 모의 실험은  $10^7$  개의 슬롯에 걸쳐 수행되었다.

표 1과 2는 각각 부채널 별 최적 스케줄러와 제안하는 스케줄러의 성능을 나타낸다. 두 스케줄러는 모두 주어진 QoS 요구를 만족시키지만, 제안하는 방식이 부채널 별 최적 스케줄러보다 큰 효용을 보인다. 채널 조건이 비대칭적인 두 사용자 3과 4의 성능을 자세히 살펴보자. 사용자 3은 부채널 3에서의 상태가 매우 열악하며 부채널 4에서의 상태가 양호하다. 이 때, 부채널 별 최적 스케줄러는 부채널 3에서 25%의 슬롯을, 부채널 4에서도 25% 슬롯을 사용자 3에게 할당한 반면에 제안하는 스케줄러는 부채널 3에서는 49%의 슬롯을, 부채널 4에서는 1%의

슬롯만을 사용자 3에게 할당한다. 두 스케줄러 모두 전체적으로 같은 양의 슬롯과 부채널을 사용자 3에게 할당하지만 제안하는 방식은 상태가 양호한 부채널에서 집중 할당함으로써 효용을 높일 수 있었다.

표 1 부채널 별 최적 스케줄러의 성능

	부채널1		부채널2		계	
	슬롯 할당량	효용	슬롯 할당량	효용	슬롯 할당량	효용
사용자1	0.25	53.84	0.25	53.87	0.25	107.71
사용자2	0.25	53.85	0.25	53.83	0.25	107.68
사용자3	0.25	6.48	0.25	103.73	0.25	110.21
사용자4	0.25	103.77	0.25	6.48	0.25	110.25
계	1.00	217.94	1.00	217.91	1.00	435.85

표 2 제안하는 스케줄러의 성능

	부채널1		부채널2		계	
	슬롯 할당량	효용	슬롯 할당량	효용	슬롯 할당량	효용
사용자1	0.25	49.08	0.25	49.11	0.25	98.19
사용자2	0.25	49.11	0.25	49.12	0.25	98.23
사용자3	0.01	0.71	0.49	141.97	0.25	142.68
사용자4	0.49	141.97	0.01	0.71	0.25	142.68
계	1.00	240.87	1.00	240.91	1.00	481.78

사용자 4에 대해서도 동일한 설명이 가능하다. 즉, 제안하는 방식은 매개 변수에 의해 다른 채널의 상황을 간접적으로 반영함으로써 부채널 별 최적 스케줄러보다 우수한 성능을 나타낸다.

5. 결론

본 논문에서는 WiBro와 같은 OFDMA/TDD 시스템에서 동작하는 기회주의적 패킷 스케줄러를 제안하였다. 제안하는 방식은 시스템 효용을 극대화하는 동시에 사용자 QoS를 엄격하게 보장할 수 있으며 구현이 간단한 장점을 갖는다. 또한, 여러 부채널의 상황을 동시에 고려하기 때문에 부채널 별로 독립적인 스케줄러를 사용하는 것보다 우수한 성능을 보인다.

5. 참고문헌

[1] P. Viswanath, D. Tse, and R. Laroia, "Opportunistic beamforming using dumb antennas," IEEE Trans. Info. Theory, vol. 48, no. 6, pp. 1277-1294, Jun. 2002.  
 [2] X. Liu, E. Chong, and N. Shroff, "A framework for opportunistic scheduling in wireless networks," Computer Networks, vol. 41, no. 4, pp. 451-474, Mar. 2003.  
 [3] Y. Liu and E. Knightly, "Opportunistic fair scheduling over multiple wireless channels," in Proc. IEEE INFOCOM 2003, San Francisco, CA, pp. 1106-1115.