

# User-Centric Cell-Free 시스템에서 공정한 자원 할당을 위한 Dynamic Scheduling Algorithm

<sup>1,3</sup>신경호, <sup>2</sup>최윤주 <sup>1,3</sup>송형규\*

<sup>1</sup>세종대학교 정보통신공학과, <sup>2</sup>세종대학교 전자정보통신공학과, <sup>3</sup>세종대학교 지능형드론융합전공

shinkh1000@naver.com, jj010513@naver.com, \*songhk@sejong.ac.kr

## Dynamic Scheduling Algorithm for fairness resource allocation in User-Centric Cell-Free System

<sup>1,3</sup>Kyung-Ho Shin, <sup>2</sup>Yoon-Ju Choi, <sup>1,3</sup>Hyoung-Kyu Song\*

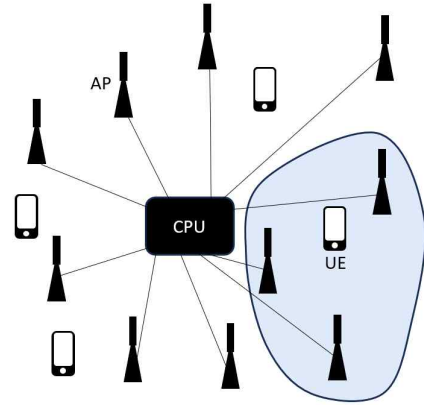
<sup>1</sup>Department of Information and Communication Engineering, <sup>2</sup>Electronic Information and Communication Engineering,  
and <sup>3</sup>Convergence Engineering for Intelligent Drone,  
Sejong University, Seoul, 209 Neungdong-ro, 05006, Korea

### 요약

본 논문은 user-centric scalable cell-free 시스템에서 Lyapunov optimization을 활용하여 네트워크 내의 UE(User Equipment)의 자원 할당을 하였다. Lyapunov optimization는 네트워크 성능을 최적화하면서 동시에 각 UE의 Queue를 안정화하는 기법이다. Lyapunov optimization에서 최적화할 수 있는 네트워크 성능은 power 할당과 data format 등 다양한 관점에서 응용할 수 있다. 본 논문은 Downlink 환경에서 시스템 spectral efficiency를 네트워크 성능으로 고려하였으며 queue 안정화와 trade-off 관계를 설정하였다. 파라미터  $V$ 는 trade-off 관계를 설정하는 변수이다. 본 논문에서는 각 UE의 queue backlog와 channel 상태를 고려하여 자원 할당을 진행하였다. 모든 time-slot마다 queue 안정화를 과도하게 고려할 필요가 없다. 따라서 queue 안정화를 만족하며 spectral efficiency를 최대화 할 수 있는  $V$ 값을 설정하는 것이 중요하다.  $V$ 를 각 time-slot마다 모든 UE의 queue backlog에 적응적으로 설정하였다.

### I. 서론

Cell-Free 시스템은 분산되어 있는 AP(Access Point)가 공동으로 UE를 지원하는 시스템이다. 5G에서 높은 주파수로 인한 짧은 커버리지 문제를 분산된 AP가 음영 지역의 UE를 지원할 수 있다.[1] 본 논문에서는 UE를 주변의 channel gain이 높은 AP에 의해 선별적으로 지원하는 user-centric scalable cell-free 환경을 고려하였다. 또한 최근 V2X와 같은 서비스를 5G에서 서비스하기 위해서 delay를 최소화하며 네트워크 성능을 보장하는 자원 할당이 필요하다. 이를 위해서 본 논문에서는 Lyapunov optimization을 활용하여 각 time-slot마다 CPU에서 UE의 QSI(Queue State Information)과 CSI(Channel State Information) 기반으로 스케줄링을 진행한다.[2] 그리고 각 AP에 스케줄링할 수 있는 UE는 최대 AP의 안테나 개수로 한정된다. CPU에서 네트워크 내의 모든 UE를 CPU에서 스케줄링하기 때문에 각 AP의 rank 자원을 고려하여 스케줄링하기 어렵다. 따라서 CPU에서 스케줄링한 후보 UE들이 각 AP에서 rank 자원을 초과한다면 SUS(semi-orthogonal user selection)을 활용하여 한 번 더 스케줄링을 진행한다.



[그림 1] user-centric scalable cell-free 시스템

### II. 본론

#### A. 시스템 모델

본 논문에서는 user-centric scalable cell-free 시스템을 고려하며  $L$ 개의 AP를 고려하며  $M$ 개의 안테나를 장착했다고 가정한다. 그리고 UE는  $K$ 개를 고려하며 CPU에서 스케줄링을 진행한 후보 UE는  $K_{act, CPU}$ 로 표시한다. 또한 채널을 시간-주파수 영역에서 coherence interval에서 채널이 변하지 않는다고 가정할 수 있는 표준 block fading model을 채택했다.

#### B. Queue 모델

각 CPU에서 QSI와 CSI를 알고 있으며 Lyapunov optimization를 활용하여 스케줄링을 진행한다. 각 UE별로 queue가 존재하며  $\mathbf{Q}(t) = \{Q_1(t), \dots, Q_K(t)\}$ 로 표기한다. 각 queue에는 time-slot마다 arrival rate  $\mathbf{a}(t) = \{a_1(t), \dots, a_K(t)\}$ 에 따라 쌓이는 데이터와 스케줄링이 되어 데이터가 전송되어 queue에서 제거되는 effective data rate  $\mathbf{r}(t) = \{r_1(t), \dots, r_K(t)\}$ 가 있다. 위의 과정은 [그림 2]로 나타나져 있다. UE  $k$ 의 time-slot  $t+1$ 의 queue는 수식 (1)과 같이 정해진다.

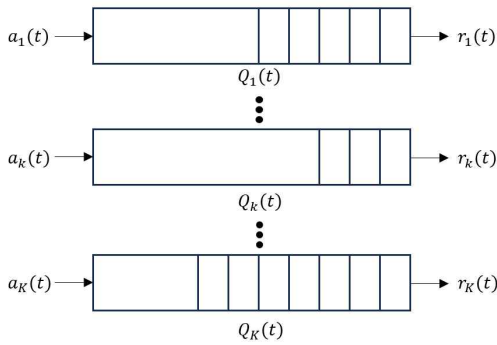
$$Q_k(t+1) = \max[Q_k(t) - r_k(t), 0] + a_k(t) \quad (1)$$

본 논문에서 outage rate를 고려하여 effective data rate을 구하였다. 수

식 (2)과 같이 CPU에서 각 UE마다 CSI를 통해 SINR을 활용하여 mutual information을 구할 수 있다. 그리고 에러 없이 전송 가능한 data rate인 각 UE의 allocated rate  $\hat{r}_k(t)$ 는 수식 (3) 같이 나타난다.  $R$ 은 mutual information의 한 샘플이며  $P_k(R < I_k(t))$ 은  $R < I_k(t)$  사건이 발생할 확률이다. 각 time-slot별로 각 유저의 mutual information이 allocated rate보다 높으면 해당 time-slot에 allocated rate가 effective data rate이 된다.

$$I_k(t) = \frac{|v_k h_k^H|^2}{\sum_{j \neq k} |v_j h_j^H|^2 + \sigma^2} \quad (2)$$

$$\hat{r}_k(t) = \arg \max_R R \times P_k(R) \quad (3)$$



[그림 2] K-queue model in CPU

### C. CPU optimization

CPU에서 각 UE의 스케줄링을 진행하기 위해 PF(Proportional Fairness)를 고려하여 drift-plus-penalty 목적함수를 만들어 최적화 문제를 (4)와 같이 만든다.

$$\begin{aligned} \max_{K_{act,CPU}} \sum_{k=1}^K Q_k(t) r_k(t) + V \log(1 + r_k(t)) \\ \text{subject to } \sum_{k=1}^K s_k(t) < K_{act,limit} \end{aligned} \quad (4)$$

수식 (4)에서  $s_k(t)$ 는 time-slot  $t$ 에서 UE  $k$ 의 스케줄링 여부 변수이며  $K_{act,limit}$ 은 CPU에서 설정한 최대 스케줄링 수이다.

trade-off 변수인  $V$ 는 값이 커질수록 data rate을 고려한 스케줄링이 실행되며 작아질수록 queue 안정화를 고려한 스케줄링이 실행된다. 수식 (5)에서 UE의 queue backlog을 고려하여 각 time-slot마다  $V$  값이 적응적으로 변화하도록 설정함을 보여준다.

$$V = V_{\max} \min \left( \frac{Q_{\max} - Q(t)}{Q_{\max}} \right) \quad (5)$$

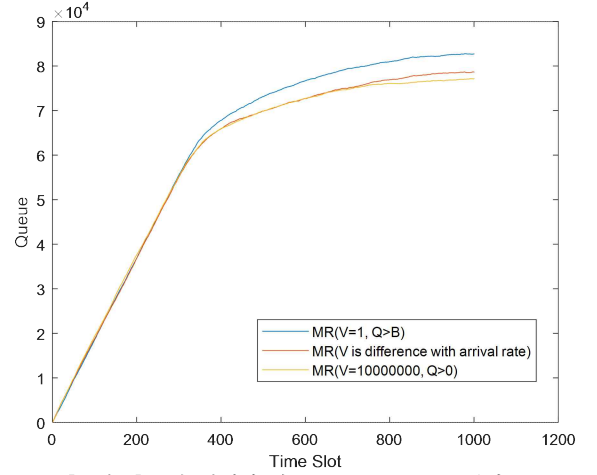
수식 (5)에서  $V_{\max}$ 와  $Q_{\max}$ 는 시스템에서 정한 상수이다.

### D. SUS algorithm in AP

CPU에서 스케줄링 된  $K_{act,CPU}$ 가 AP의 rank 자원을 넘어가면 데이터를 전송할 수 없다. 따라서  $K_{act,CPU}$  UE 중 AP 클러스터 내의 UE가 rank 자원을 넘어갈 경우 SUS 알고리즘을 통해 유저를 선정한다. SUS 알고리즘은 채널이 가장 좋은 UE 기준으로 상관도가 적은 UE를 선정하는 알고리즘이다. 따라서 각 AP에서 SUS 알고리즘을 활용하여 스케줄링된 최종 UE는  $K_{act}$ 로 표기한다.

### III. 시뮬레이션 결과

본 논문에서 고려한 시뮬레이션 환경은  $L = 45$ ,  $M = 4$ ,  $K = 200$ 으로 UE가 밀집된 환경을 고려했으며,  $K_{act,CPU} = \frac{LM}{2} = 90$ 으로 제한하여 스케줄링을 진행했다. coherence interval 당 심볼 개수  $\tau_c = 200$ , 그중 파일럿 심볼 개수는  $\tau_p = 10$ 으로 설정하였다. 각 UE의 arrival rate는  $a_k(t) = \text{Bernoulli}(p_k) \times A_{\max}$ 으로 설정하였으며  $\text{Bernoulli}(p_k)$ 는 time-slot마다 데이터가  $p_k$  확률로 queue에 들어오는 것을 의미하며  $A_{\max} = 5\tau_c$ 로 설정하였다.



[그림 3] V값 선택에 따른 queue backlog 변화

$V$ 값을 queue에 적응적으로 선택하여 time-slot에 따라 적응적으로 queue를 안정화할 수 있다.

### IV. 결론

본 논문에서는 user-centric scalable cell-free 시스템에서 Lyapunov optimization의 trade-off 파라미터인  $V$ 값을 queue backlog에 적응적으로 선택하는 방식을 제공한다. 시뮬레이션 결과에 따르면  $V$ 값을 적응적으로 선택한 queue backlog는  $V$ 값을 고정했을 때의 queue backlog를 절충할 수 있는 방식이다.

### ACKNOWLEDGMENT

이 논문은 2020년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2020R1A6A1A03038540)

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. NRF-2021R1A2C2005777).

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 정보통신방송혁신인재양성(메타버스융합대학원)사업 연구 결과로 수행되었음 (IITP-2023-RS-2023-00254529)

### 참고 문헌

- [1] Özlem Tugfe Demir; Emil Björnson; Luca Sanguinetti, Foundations of User-Centric Cell-Free Massive MIMO, now, 2021.
- [2] Z. Chen, E. Björnson and E. G. Larsson, "Dynamic Resource Allocation in Co-Located and Cell-Free Massive MIMO," in IEEE Transactions on Green Communications and Networking, vol. 4, no. 1, pp. 209-220, March 2020, doi: 10.1109/TGCN.2019.2953575.