Check for updates

Journal of KIIT. Vol. 21, No. 9, pp. 75-82, Sep. 30, 2023. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2023.21.9.75

IEEE 802.11ax 무선 랜에서 업링크 채널 성능 향상을 위한 동적 파라미터 제어 MAC 프로토콜

안태욱*, 김선명**

Dynamic Parameter Control MAC Protocol for Improving Uplink Channel Performance in IEEE 802.11ax WLANs

Taewook Ahn*, Sunmyeng Kim**

이 연구는 금오공과대학교 대학 연구과제비로 지원되었음(2022~2023)

요 약

IEEE 802.11ax에서 UORA(Uplink OFDMA-based Random Access) 프로토콜은 OCW(OFDMA Contention Window)와 OBO(OFDMA BackOff) 값을 유지 관리한다. 단말은 OCW 범위 내에서 무작위로 OBO를 선택하여 데이터 전송을 위한 RU(Resource Unit)를 결정한다. OCW가 작으면 충돌 확률이 높아지고, 반대로 크면 데이터 전송이 없는 RU가 많아져 성능이 저하된다. PCS(Parameter Control Scheme) 프로토콜은 UORA 프로토콜의 성능을 개선하기 위해 데이터 전송 성공/실패에 따라 OCW와 OBO 값의 증감 정도를 정적으로 제어한다. 그러나 정적으로 제어하기 때문에 성능 개선에 한계가 있다. 본 논문에서는 단말 수에 따라 OCW와 OBO 값을 동적으로 제어하는 방법을 제안한다. 시뮬레이션을 통해 제안된 방법의 성능을 기존 프로토콜과 비교 분석한다. 비교 분석결과, 제안된 방법이 충돌 확률은 낮추고 전송 처리율 및 평균 지연은 향상시킨다.

Abstract

In IEEE 802.11ax WLAN, the Uplink OFDMA-based Random Access(UORA) protocol manages the OFDMA Contention Window(OCW) and OFDMA BackOff(OBO) parameters. Stations randomly select an OBO value within the range of OCW values to determine the Resource Unit(RU) for data transmission. If the OCW value is low, the collision probability increases as more stations transmit data on the same RU. Conversely, if the OCW value is high, there are many RUs with no data transmission, resulting in channel waste. In this paper, we propose a method of dynamically controlling the OCW and OBO parameter values based on the number of stations. The proposed method's performance is compared and analyzed with the existing protocol through simulations. The comparative analysis shows that the proposed method reduces the collision probability and improves the transmission throughput.

Keywords
dynamic parameter, IEEE 802.11ax, MAC, RU, WLAN

* 금오공과대학교 컴퓨터소프트웨어공학과 학사과정
- ORCID: https://orcid.org/0009-0005-9248-232X
** 금오공과대학교 컴퓨터소프트웨어공학과 교수(교신저자)
- ORCID: https://orcid.org/0000-0001-6076-6401

· Received: Jun. 10, 2023, Revised: Jul. 04, 2023, Accepted: Jul. 07, 2023
· Corresponding Author: Sunmyeng Kim
Dept. of Computer Software Engineering, Kumoh National Institute of Technology, Korea
Tel.: +82-54-478-7547, Email: sunmyeng@kumoh.ac.kr

Ⅰ. 서 론

IEEE 802.11ax 표준은 사용자에게 고품질 서비스를 제공하기 위해 채택되었으며 OFDMA (Orthogonal Frequency Division Multiple Access)와 MU-MIMO(Multi-User Multiple-Input Multiple-Output) 기술을 이용한다[1]. OFDMA는 다중 사용자 채널 접속 기술로 여러 사용자가 동시에 데이터를 전송할 수 있도록 주파수 채널을 작은 단위로 나눈다. 나눠진 작은 단위를 RU(Resource Unit)라고 한다.

업링크(Up link) 채널 접근 방법에는 SA (Scheduled Access) 방식과 RA(Random Access) 방식이 있다. SA 방식에서는 단말들이 AP(Access Point)에게 자신의 데이터 및 채널 상태 정보를 제공한다. AP는 제공된 정보를 이용하여 각 단말에게 데이터를 전송할 RU를 할당하고 단말은 할당된 RU에서 데이터를 전송한다. 따라서 SA는 RU에서 데이터 충돌이 발생하지 않는 비경쟁 방식이다. RA 방식에서는 AP가 RU를 할당하지 않고 단말들간 경쟁을 통해 데이터를 전송할 RU를 획득한다. 본 논문에서는 SA 방식에 비해 효율성과 간단한 RU 할당 방법을 갖는 RA 방식에 초점을 맞춘다.

UORA(Uplink OFDMA-based Random Access)는 RA 방식으로 동작하는 대표적인 프로토콜이다[2]. UORA 프로토콜에서 단말들은 데이터 전송을 위한 RU를 선택하기 위해 OCW(OFDMA Contention Window)와 OBO(OFDMA BackOff) 카운터를 이용한다. OBO 카운터는 0과 OCW 범위 내에서 무작위로 선택된다. 단말들은 AP가 전송하는 TF(Trigger Frame)을 수신한 후, TF에 포함된 RU 수만큼 OBO 카운터를 감소시킨다. OBO 카운터 값이 0 이하인 단말들은 자신의 데이터를 전송할 RU를 임의로 선택하고 데이터를 전송한다. OBO 카운터 값이 0보다 큰 단말들은 다음 TF 수신을 대기한다.

UORA 프로토콜은 경쟁 단말 수가 증가할수록 성능이 저하되는 문제점을 갖고 있다. 이러한 문제점을 해결 및 완화하기 위한 다양한 프로토콜들이 연구되었다[3]-[8]. CRA(Cyclic Resource Assignment) 프로토콜은 RU와 단말 상태를 각각 두 종류로 구분한다[3]. RU는 경쟁과 비경쟁이고, 단말 상태는 데이터 전송 성공과 실패이다. 이전 데이터 전송에서 성공한 단말은 경쟁 RU를 통해 RU를 획득하고 데이터를 전송한다. 그러나 충돌로 인해 실패한 단말은 비경쟁 RU를 할당하여 성공적인 데이터 전송을 보장한다. H-UORA(Hybrid Uplink OFDMA Random Access) 프로토콜은 두 단계를 통해 데이터 전송을 수행함으로써 충돌 가능성을 줄인다[3].

단계 1에서는 기존 UORA 프로토콜과 유사하게 동작하여 RU를 획득한다. 그러나 UORA 프로토콜과 달리 바로 데이터 전송을 수행하지 않고 단계 2로 진입한다. 단계 2에 진입한 단말들은 한 타임 슬롯 시간 동안 채널을 감지한다. 채널이 유휴(Idle)이면 데이터를 전송하고, 비지(Busy)이면 전송하지 않는다.

RNACA(Retransmission Number Aware Channel Access) 프로토콜은 여러 성능 요소 중에서 데이터 전송 지연에 초점을 맞춰 성능 향상을 위해 제안되었다[4]. 이 프로토콜은 재전송 수와 RU 수, 단말 수 정보를 이용하여 계산된 확률에 기반하여 단말의 OCW를 다르게 설정한다. 즉, 일부 단말은 OCW를 두 배 이하로, 다른 단말은 두 배로 설정한다. 두 배 이하로 작게 설정된 단말은 다른 단말보다 데이터 전송을 위한 RU를 빠르게 취득하여 전송 지연을 감소시킨다. MORA(MU-MIMO enabled Uplink OFDMA Random Access) 프로토콜은 여러 안테나를 갖는 환경에서 한 RU에서 동시 전송을 위한 방법을 제안하였다[5]. 여러 단말이 한 RU에서 동시에 데이터를 전송하더라도 AP가 여러 안테나를 갖는 경우, 프리앰블 디코딩을 통해 여러 데이터를 충돌 없이 수신이 가능한 기술을 이용하였다. COSB(Channel Observation-based Scaled Backoff) 프로토콜은 충돌 확률에 따라 OCW 파라미터를 동적으로 변경한다[6]. PCS(Parameter Control Scheme) 프로토콜은 데이터 전송 성공 및 실패 여부에 따라 OCW 파라미터 값을 UORA 프로토콜과 다르게 설정한다[7]. 전송 실패인 경우, UORA 프로토콜은 OCW를 두 배로 증가시키지만 PCS 프로토콜은 두 배 이하로 작게 증가시킨다. 또한, 전송 성공인 경우, UORA 프로토콜은 OCW를 초기값으로 설정하지만 PCS 프로토콜은 초기값보다 큰 값으로 설정한다.

따라서 OCW 파라미터의 값의 변화가 크지 않게 적절하게 유지하여 성능을 향상시킨다. PCS 프로토콜은 OCW 파라미터 값을 정적으로 설정하기 때문에 적절한 단말 수에서는 좋은 성능을 유지하지만 단말 수가 많거나 적으면 UORA 프로토콜처럼 성능이 저하된다. 본 논문에서는 단말 수와 상관없이 항상 최적의 성능을 유지하기 위해 단말 수에 기반하여 동적으로 파라미터를 제어하는 DPC(Dynamic Parameter Control) 프로토콜을 제안한다. 이 프로토콜은 학술회의에서 발표한 논문을 개선한 것이다[9].

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구에 대해 간단히 설명한다. 3장에서는 제안된 DPC 프로토콜에 대해 자세히 설명하고 4장에서는 시뮬레이션을 통한 성능 분석 결과를 보여준다. 마지막으로 5장에서는 결론을 맺는다.

II. 관련 연구

2.1 UORA 프로토콜

UORA 프로토콜에서 AP는 데이터 전송 절차를 진행하기 위해 TF(Trigger Frame)를 단말들에게 브로드캐스팅한다. TF에는 각 RU에 대한 경쟁/비경쟁 여부 및 AID(Association Identifier) 등의 정보가 포함된다. 경쟁 RU의 AID는 0 또는 2045이다. AID 0번 RU는 AP와 연결 과정(Association)이 완료된 단말이 데이터를 전송하기 위해 채널 경쟁을 수행한다. AID 2045번 RU는 AP와 연결 과정을 진행하기 위해 단말이 사용한다. AID가 1~2044인 RU는 해당 단말에게 할당된 비경쟁 RU이다. 본 논문에서는 RA(경쟁) 방식에 초점을 맞춰 진행하므로 모든 RU는 경쟁 RU라고 가정한다.

단말들은 RU 경쟁을 위해 OCW와 OBO 카운터를 사용한다. OCW는 OCWmin과 OCWmax 사이 값을 갖는다. 초기 OCW는 OCWmin으로 설정되며 OBO 카운터는 [0, OCW-1] 범위 내에서 임의로 선택된다. TF를 수신한 후, 단말들은 OBO 카운터를 경쟁 RU 수만큼 줄인다. OBO 카운터 값이 0 이하인 단말들은 임의 RU를 선택하여 데이터를 전송하고, 0보다 큰 단말들은 데이터를 전송하지 않고 대기한다. AP는 RU에서 데이터를 수신한 후, 데이터 수신 성공 여부를 MU-BACK(Multi-User Block Acknowledgment)를 전송하여 각 단말에게 전달한다. 한 RU에 대해, 하나의 단말이 선택하여 데이터를 전송하면 전송 성공이고, 두 개 이상의 단말이 선택하면 충돌이 발생하여 전송 실패이다. 전송이 성공한 단말은 OCW를 OCWmin으로 초기화하고, 실패한 단말은 현재 OCW 값의 두 배로 증가시킨다.

2.2 PCS 프로토콜

UORA 프로토콜에서 단말의 RU 경쟁 수준을 제어하는 파라미터는 OCW와 OBO 카운터이다. 데이터 전송에 성공한 단말은 OCW를 OCWmin으로 초기화하여 작은 범위에서 무작위로 OBO 카운터 값을 선택하므로 확률적으로 작은 OBO 카운터 값을 갖게 되어 빠르게 다음 채널 접속 기회를 갖는다. 반면, 데이터 전송에 실패한 단말은 OCW를 두 배로 증가시켜 큰 범위에서 무작위로 OBO 카운터 값을 선택하므로 확률적으로 큰 OBO 카운터 값을 갖게 되어 천천히 다음 채널 접속 기회를 갖는다.

PCS 프로토콜은 UORA 프로토콜과 다르게 데이터 전송 성공/실패 여부에 따라 OCW와 OBO 카운터 파라미터 값의 증감 정도를 제어한다. 데이터 전송에 성공한 단말은 OCW를 현재 OCW의 절반으로 감소시키고, 실패한 단말은 (OCWmin / 2)만큼 증가시킨다. 또한, TF를 수신한 단말은 OBO 카운터를 (a * 경쟁 RU 수)만큼 감소시킨다. a는 접속 빈도 가중치로, 더 높은 채널 접속 기회를 제공하기 위해서는 1보다 크게 설정하고, 낮은 채널 접속 기회를 제공하기 위해서는 1보다 작게 설정한다.

2.3 PCS 프로토콜 분석

시뮬레이션을 통해 PCS 프로토콜을 분석한다. PCS 프로토콜에서는 접속 빈도 가중치 a 값을 설정하는 방법에 대해 구체적으로 제시하지 않고 0.5와 1.5를 사용하였기에 여기에서도 이 값으로 설정하여 분석한다.

그림 1은 UORA와 PCS 프로토콜의 단말 수에 따른 처리율의 변화를 보여준다.

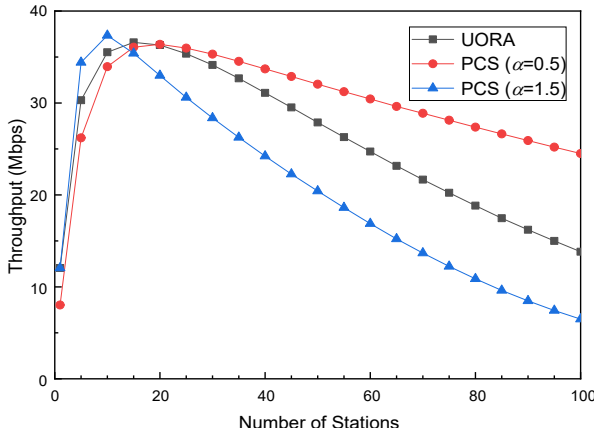

그림 1. UORA와 PCS 프로토콜의 단말 수에 따른 처리율

Fig. 1. Throughput according to the number of stations for UORA and PCS protocols

그림은 단말 수가 약 15개일 때 UORA와 PCS 프로토콜은 유사한 성능을 갖는다. PCS(α=1.5) 프로토콜은 단말 수가 15개보다 적을 때, 가장 우수한 성능을 갖는다. 이는 OBO 카운터 값을 빠르게 감소시켜 더 많은 단말이 채널 경쟁에 참여하고 데이터를 전송하기 때문이다. 그러나 단말 수가 증가함에 따라 경쟁에 참여한 많은 단말은 더 큰 충돌을 발생시켜 성능을 급격하게 감소시킨다. PCS(α=0.5) 프로토콜은 단말 수가 15개보다 많을 때, 가장 우수한 성능을 갖는다. 이는 OBO 카운터 값을 느리게 감소시켜 더 적은 단말이 채널 경쟁에 참여하고 데이터를 전송하기 때문이다. 그러나 단말 수가 증가함에 따라 경쟁에 참여한 더 적은 단말은 더 작은 충돌을 발생시켜 성능은 상대적으로 천천이 감소한다.

PCS 프로토콜의 문제점은 다음과 같다. 항상 고정된 파라미터 값을 사용하기 때문에 단말 수에 따라 성능 차이를 보인다. 또한 단말 수가 많은 경우에는 UORA 프로토콜처럼 성능 저하가 발생한다.

III. 제안된 DPC 프로토콜

DPC 프로토콜은 단말 수와 상관없이 항상 최적의 성능을 유지하기 위해 단말 수에 따라 동적으로 파라미터 OCW와 OBO 카운터 값을 제어한다.

현재 통신에 참여하는 단말 수를 파악하는 방법에 대해 많은 연구가 진행되었다[10][11]. 따라서 본 논문에서는 단말 수를 파악하는 방법에 대해서는 다루지 않고 정확하게 파악되었다고 가정한다.

AP는 주기적으로 통신 참여 단말 수(NSTA)를 파악하여 TF를 통해 단말들에게 전송한다. TF를 수신한 단말은 RU 경쟁 단말 수(N_{COM})를 계산한다. 업링크 통신에서 단말들이 이용할 수 있는 RU는 한정적이다. 단말들은 최대 경쟁 RU 수만큼 동시에 전송을 시도할 수 있는데, RU의 수가 적어지면 경쟁이 더 치열해지고 RU의 수가 많아지면 경쟁이 완화된다. 이는 전송을 위한 경쟁의 정도가 단말 수와 RU의 수에 의존한다는 것을 의미한다. 따라서 RU 경쟁 단말 수(N_{COM})는 다음과 같다.

$$N_{COM} = \begin{cases} 1, & \text{if}(N_{STA} \le N_{RU}) \\ 1 + N_{STA} - N_{RU}, & else \end{cases} \quad (1)$$

여기에서, N_{RU}는 경쟁 RU 수로 TF에 포함된 RU 중에서 AID가 0으로 설정된 RU 수이다. 통신 참여 단말 수가 경쟁 RU 수 이하이면, 모든 단말이 전송에 성공할 가능성이 있다. 즉, 모든 단말이 서로 다른 RU를 선택하여 데이터 전송 성공이 가능하다. 따라서, 각 RU에는 하나의 단말만 선택할 가능성이 있으므로 RU 경쟁 단말 수를 1로 설정한다. 반대로, 통신 참여 단말 수가 경쟁 RU 수보다 크면, 최소 두 개 이상의 단말이 한 RU를 선택하여 충돌이 발생할 수 있다. 각 RU를 선택한 단말이 하나 이상이라고 가정하면, 한 RU에서의 최대 경쟁 단말 수는 최대 ($1+N_{STA}-N_{RU}$)이다.

각 단말은 두 가지 데이터 전송 실패 정보를 유지 관리한다. 첫 번째는 n번째 데이터의 전송 과정에서 발생한 전송 실패 수(*Failn*)이다. n번째 데이터에 대해 전송을 처음 시도할 때 *Failn*은 0으로 초기화된다. 전송을 시도하였지만 실패한 경우, *Failn*을 1만큼 증가시킨다. 두 번째는 n번째 데이터의 전송 성공 후, 전송 성공된 모든 데이터에 대한 평균 전송 실패 수(E[*Fail*]n)이다.

그림 2는 n번째 데이터의 전송 과정을 보여준다. 그림에서 F, S, E로 표시된 하나의 네모는 UORA 프로토콜에서 한 번의 데이터 전송 과정을 의미한다. F와 S는 단말의 OBO 카운터 값이 0이하여서 데이터 전송을 시도하여 실패한 경우(Failure)와 성공한 경우(Success)를 각각 의미한다.

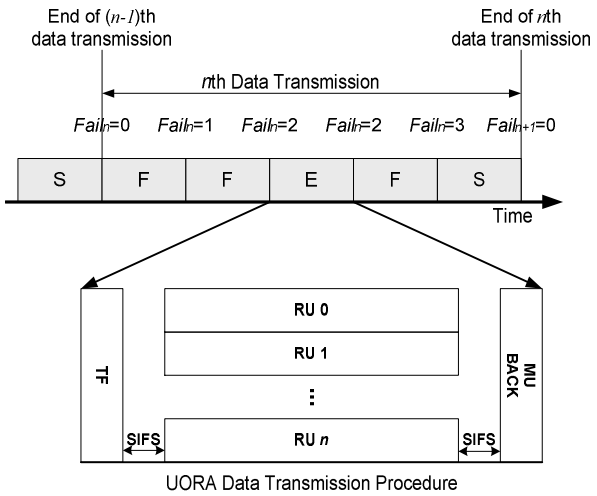

그림 2. 데이터 전송 구조

Fig. 2. Structure of the data transmission

E는 단말의 OBO 카운터 값이 0보다 커서 데이터 전송을 시도하지 않은 경우(Empty)를 의미한다. 데이터 전송을 처음 시도할 때 *Failn*은 0이고, 첫 번째와 두 번째 UORA 프로토콜 데이터 전송 절차에서 데이터 전송 실패이므로 *Failn*은 1과 2로 증가되었다. 그리고 세 번째 전송 절차에서는 데이터 전송을 시도하지 않았기 때문에 *Failn*은 변경되지 않고 2를 유지한다. 네 번째 전송 절차에서도 실패하였으므로 *Failn*은 3이다. 다섯 번째 전송 절차에서 성공하였다.

n번째 데이터의 전송 성공 후에, 전송 실패 수(*Failn*)와 이동 평균 윈도우(Moving average window)를 이용하여 평균 전송 실패 수(E[*Fail*]n)를 다음과 같이 계산한다.

$$E[Fail]_n = \delta \cdot E[Fail]_{n-1} + (1-\delta) \cdot Fail_n \quad (2)$$

δ는 평활 인수(Smoothing factor)이다.

(n+1)번째 데이터 전송을 위해, 각 단말은 단말 수에 따른 동적 접속 빈도 가중치 a_{n+1}를 다음과 같이 계산한다.

$$\alpha_{n+1} = \frac{N_{RU}}{N_{COM} + E[Fail]_n + Fail_{n+1}} \quad (3)$$

동적 접속 빈도 가중치는 경쟁 RU 수에 비례한다. 경쟁 RU의 수가 많을수록 큰 a_{n+1} 값을 갖는다. 그러나 RU 경쟁 단말 수 및 데이터 전송 실패 수에는 반비례한다.

제안된 DPC 프로토콜에서 단말은 AP로부터 TF를 수신한 후, 동적 접속 빈도 가중치를 계산하고 OBO 카운터 값을 다음과 같이 감소시킨다.

$$OBO = OBO - \alpha_{n+1} \cdot N_{RU} \quad (4)$$

동적 접속 빈도 가중치가 크면 OBO 카운터를 많이 감소시켜 더 많은 단말이 채널 경쟁에 참여한다. 그러나 가중치가 작으면 OBO 카운터 값을 조금 감소시켜 적은 단말이 채널 경쟁에 참여하도록 한다.

UORA와 PCS 프로토콜은 단말 수와 상관없이 데이터 전송 성공 여부에 따라 OCW 값을 조절한다. 단말 수가 많은 상황에서 전송 성공으로 OCW를 작게 설정하면 OBO 값도 작게 되어 충돌 확률이 높아진다. 또한 단말 수가 적은 상황에서 전송 실패로 OCW를 크게 설정하면 OBO 값도 크게 되어 채널 접속 빈도를 낮춘다. 따라서 전체적인 네트워크 성능이 저하될 수 있다.

제안된 DPC 프로토콜의 단말은 OCW 값을 단말 수와 동적 접속 빈도 가중치에 따라 증감을 조절한다. 데이터 전송에 성공하면 OCW를 다음과 같이 설정한다.

$$OCW = \left(1 + \frac{N_{RU} - \alpha_{n+1}}{2 \cdot N_{RU}}\right) \times OCWmin \quad (5)$$

단말은 데이터 전송에 실패하면 OCW를 다음과 같이 설정한다.

$$OCW = OCW + \left(\frac{N_{RU} - \alpha_{n+1}}{2 \cdot N_{RU}}\right) \cdot OCWmin \quad (6)$$

Ⅳ. 시뮬레이션 결과

본 장에서는 시뮬레이션을 통해 DPC 프로토콜이 기존 PCS 프로토콜보다 성능이 우수함을 보인다.

성능 분석을 위해 이벤트 기반 시뮬레이션 프로그램을 파이썬 언어를 이용하여 작성하였다. 표 1은 시뮬레이션에서 사용된 파라미터이다.

표 1. 시뮬레이션 파라미터
Table 1. Simulation parameters

Parameter	Value
OCWmin	32
OCWmax	1024
Data rate	1 *Gbps*
DATA	1000 bytes
Preamble	40 bytes
TF size	89 bytes
MU-BACK size	32 bytes
Beacon period	10 *ms*
SIFS	16 *us*
Number of RUs	8

주요 성능 요소는 충돌 확률(Collision probability), 처리율(Throughput), 평균 지연(Average delay)이다. 충돌 확률은 시뮬레이션 시간 동안의 전체 경쟁 RU 수와 충돌이 발생한 RU 수의 비율이다. 처리율은 성공적으로 전송된 초당 데이터 비트 수이다. 평균 지연은 전송을 위해 데이터가 큐 헤드에 도착한 시점부터 성공적으로 전송 완료될 때까지 시간이다.

그림 3은 단말 수에 따른 접속 빈도 가중치를 보여준다. PCS 프로토콜은 고정된 값을 사용하기 때문에 단말 수와 상관없이 항상 0.5 또는 1.5를 갖는다. 그러나 제안된 DPC 프로토콜은 단말 수에 따라 동적으로 설정되기 때문에 [0, 8] 사이의 값을 갖는다. 단말 수가 1일 때는, RU 경쟁 단말 수가 1이고 데이터 전송 실패가 발생하지 않기 때문에 식 (3)의 분모 값은 항상 1이다.

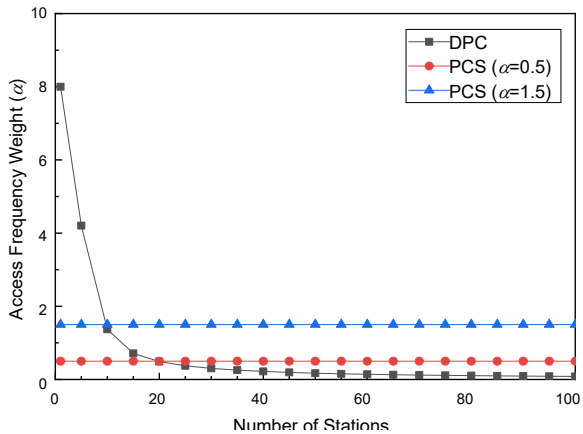

그림 3. 단말 수에 따른 접속 빈도 가중치(α)
Fig. 3. Access frequency weight (α) according to the number of stations

그리고 표 1의 시뮬레이션 파라미터에서 RU 수가 8이므로 식 (3)의 분자는 8이다. 따라서 동적 접속 빈도 가중치는 8의 값을 갖는다. 단말 수가 증가함에 따라 분모 값이 커지기 때문에 가중치는 점점 작아진다.

그림 4는 단말 수에 따른 충돌 확률 변화를 보여준다. PCS 프로토콜의 충돌 확률은 단말 수가 증가함에 따라 점점 커지는 것을 확인할 수 있다. 이는 많은 단말이 채널 경쟁에 참여하고 서로 충돌을 발생시키기 때문이다. PCS(α=0.5)의 충돌 확률이 PCS (α=1.5)보다 항상 낮다. 가중치가 클수록 더 빨리 OBO 카운터를 감소시켜 더 많은 단말이 채널 경쟁에 참여하기 때문이다. 제안된 DPC 프로토콜은 단말 수와 상관없이 낮은 충돌 확률을 보인다. 단말 수가 아주 적을 때는 DPC 프로토콜이 PCS(α=0.5)보다 약간 더 높은 충돌 확률을, PCS(α=1.5)와 비슷한 충돌 확률을 갖는다. 단말 수가 적을 때는 더 많은 단말이 채널 경쟁에 참여하는 것이 처리율 향상에 도움이 된다. 따라서 가중치가 클수록 좋으며 이에 따라 충돌 확률은 약간 증가한다. 단말 수가 약 10개 이상에서는 단말 수와 상관없이 항상 일정한 충돌 확률을 갖는다.

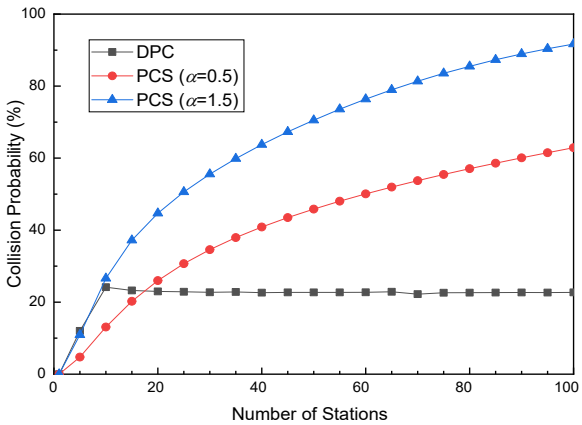

그림 4. 단말 수에 따른 충돌 확률
Fig. 4. Collision probability according to the number of stations

그림 5는 단말 수가 처리율에 미치는 영향을 나타낸 그림이다. PCS 프로토콜에서 단말 수가 증가함에 따라 충돌 확률이 커지기 때문에, 처리율은 급격히 감소한다.

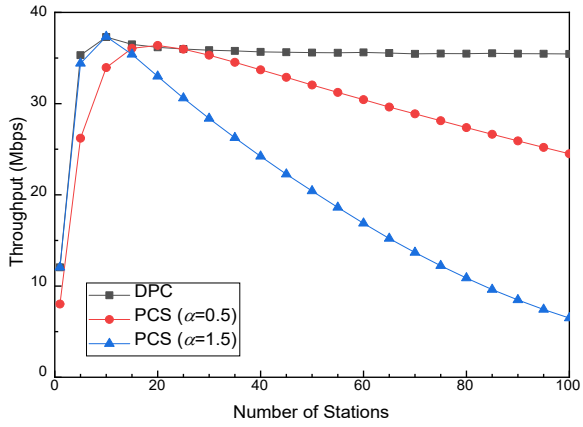

그림 5. 단말 수에 따른 처리율
Fig. 5. Throughput according to the number of stations

PCS(α=1.5)는 더 많은 단말이 채널 경쟁에 참여하기 때문에 처리율이 빠르게 낮아지고, PCS(α=0.5)는 더 적은 단말이 채널 경쟁에 참여하므로 상대적으로 천천히 감소한다. 제안된 DPC 프로토콜은 단말 수와 상관없이 항상 높은 처리율을 보여준다. DPC 프로토콜은 단말 수가 증가하더라도 파라미터를 동적으로 조절함으로써 적당한 단말들이 채널 경쟁에 참여하기 때문에 낮은 충돌 확률을 가지며 좋은 처리율을 보인다.

그림 6은 단말 수에 따른 평균 지연의 변화를 보여준다. 두 프로토콜 모두 단말 수가 증가함에 따라 평균 지연은 점점 커지는 것을 알 수 있다. PCS(α=1.5)는 단말 수가 증가함에 따라 더 많은 단말이 채널 경쟁에 참여하기 때문에 충돌 확률이 증가하여 평균 지연도 빠르게 증가한다.

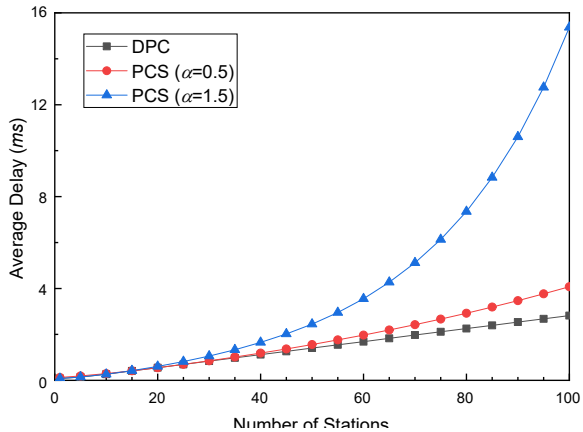

그림 6. 단말 수에 따른 평균 지연
Fig. 6. Average delay according to the number of stations

PCS(α=0.5)는 더 적은 단말이 채널 경쟁에 참여하므로 상대적으로 천천히 증가한다. 제안된 DPC 프로토콜의 평균 지연은 가장 천천히 증가한다. 이는 단말 수에 따라 동적으로 파라미터를 조절하여 충돌 확률이 낮아졌기 때문이다.

V. 결 론

IEEE 802.11ax 무선 랜은 고밀도 환경에서 사용자에게 서비스 품질을 제공하기 위해 표준화되었다. RU를 공유하기 위한 채널 접근 방식이 경쟁 기반으로 동작하기 때문에 단말 수가 증가하면 네트워크 성능이 하락하는 문제점을 갖고 있다. 본 논문에서는 단말 수에 따라 동적으로 OCW와 OBO 파라미터를 조절하여 항상 적절한 수의 단말이 RU 경쟁에 참여하도록 하였다. 제안된 방법에서 단말 수와 평균 전송 실패 수에 기반하여 동적 접속 빈도 가중치를 계산한다. 계산된 가중치를 이용하여 OCW와 OBO 파라미터 값을 동적으로 제어한다. 시뮬레이션을 통해 제안된 DPC 프로토콜이 PCS 프로토콜보다 낮은 충돌 확률과 좋은 처리율 및 평균 지연 시간을 갖는 것을 확인하였다. 단말 수가 50개일 때, PCS(α=0.5)와 PCS(α=1.5) 각각에 비해 제안된 DPC 프로토콜은 약 19%와 32% 낮은 충돌 확률, 약 10%와 75%의 향상된 처리율, 약 10%와 42%의 낮은 지연 성능을 보인다.

References

[1] IEEE P802.11ax, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment enhancements for high efficiency WLAN", IEEE Standard, May 2021. https://doi.org/10.1109/ieeestd.2021.9442429.

[2] Y. Daldoul, D.-E. Meddour, and A. Ksentini, "Performance evaluation of OFDMA and MU-MIMO in 802.11ax networks", Computer Networks, Vol. 182, Dec. 2020. https://doi.org/10.1016/j.comnet.2020.107477.

[3] L. Lanante, C. Ghosh, and S. Roy, "Hybrid

OFDMA random access with resource unit sensing for next-gen 802.11ax WLANs", IEEE Transactions on Mobile Computing, Vol. 20, No. 12, Dec. 2021. https://doi.org/10.1109/tmc.2020.3000503.

[4] Y. Zheng, J. Wang, Q. Chen, and Y. Zhu, "Retransmission number aware channel access scheme for IEEE 802.11ax based WLAN", Chinese Journal of Electronics, Vol. 29, No. 2, pp. 351-360, Mar. 2020. https://doi.org/10.1049/cje.2020.01.014.

[5] S. Joo, T. Kim, T. Song, and S. Pack, "MU-MIMO enabled uplink OFDMA MAC protocol in dense IEEE 802.11ax WLANs", ICT Express, Vol. 6, No. 4, pp. 287-290, Dec. 2020. https://doi.org/10.1016/j.icte.2020.04.007.

[6] R. Ali, N. Shahin, R. Bajracharya, B. S. Kim, and S. W. Kim, "A self-scrutinized backoff mechanism for IEEE 802.11ax in 5G unlicensed networks", Sustainability, Vol. 10, No. 4, Apr. 2018. https://doi.org/10.3390/su10041201.

[7] Y. Kim, L. Kwon, and E.-C. Park, "Channel access control scheme based on UORA in IEEE 802.11ax WLANs", KICS Winter Conference 2020, pp. 1005-1006, 2020.

[8] S. Kim, "Dynamic uplink channel access protocol based on collision probability in IEEE 802.11ax high density WLANs", Journal of KIIT, Vol. 20, No. 2, pp. 71-77, Feb. 2022. http://dx.doi.org/10.14801/jkiit.2022.20.2.71.

[9] T. Ahn and S. Kim, "Dynamic channel access protocol based on competition level in IEEE 802.11ax UORA", KIIT Summer Conference 2022, Jeju, Korea, pp. 751-753, Jun. 2022.

[10] G. Bianchi and I. Tinnirello, "Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network", in Proc. IEEE INFOCOM, San Francisco, CA, USA, pp. 844-852, Apr. 2003. https://doi.org/10.1109/INFCOM.2003.1208922.

[11] H. Choi, J. Lim, and D. Hong, "Game theory based H infinity filter approach for estimating the number of competing terminals in IEEE 802.11 networks", IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, NV, USA, pp. 953-957, Jan. 2011. https://doi.org/10.1109/CCNC.2011.5766648.

저자소개

안 태 욱 (Taewook Ahn)

2023년 7월 현재 : 금오공과대학교 컴퓨터소프트웨어공학과 학사과정

관심분야 : 무선 네트워크, 임베디드 소프트웨어

김 선 명 (Sunmyeng Kim)

2000년 2월 : 아주대학교 정보및컴퓨터공학부(공학사)

2002년 2월 : 아주대학교 정보통신공학과(공학석사)

2006년 2월 : 아주대학교 정보통신공학과(공학박사)

2008년 3월 ~ 현재 : 금오공과대학교 컴퓨터소프트웨어공학과 교수

관심분야 : 무선 네트워크