

## 수중 환경에서 효율적인 무선 통신을 위한 매체접근제어 메커니즘

정유진<sup>1\*</sup> · 신수영<sup>1</sup> · 박수현<sup>1</sup>

### Media Access Control Mechanism for Efficient Wireless Communication in Underwater Environments

Yoo-Jin Jeong · Soo-Young Shin · Soo-Hyun Park

#### ABSTRACT

Data transmission in poor environment like underwater has considerably serious delay rate and error rate. Like this environment usually has heavy fluctuation of error rate and limited wireless communication state. Therefore, mechanism using in such environment has to be efficient and simple. This paper suggests a new block ack mechanism, called the Pervasive Block ACK (PBA), which transmits aggregated ACKs. This mechanism takes effect on reducing number of traffic, decreasing overhead and delay rate in poor environment networks like underwater. Additionally, we can expect energy consumption. We verify propriety and efficiency of PBA through describing numerical result based analytical formula in this paper.

**Key words** : UW-ASN(Underwater Wireless Acoustic Sensor Networks), PBA(Pervasive Block Ack), Block Ack, ARQ

#### 요약

수중과 같은 에러의 변동이 크고 무선 통신에 제약이 따르는 환경에서 데이터 전송을 실행하게 되는 경우에는 상당히 심각한 지연율과 에러 발생률을 고려하지 않을 수 없다. 따라서 이러한 환경에서 사용되는 메커니즘은 가능한 한 효율적이면서 간단해야만 한다. 본 논문에서 제안하는 Pervasive Block ACK(PBA) 기법은 각 프레임마다 전송되던 ACK를 Set 단위로 묶어서 한꺼번에 보내주는 방법으로, 수중 등 열악한 환경에서 ACK 전송으로 인해 발생하는 트래픽 발생 빈도 감소를 위해 제안되었다. Pervasive Block ACK 기법은 전송 횟수를 최소로 줄임으로써 오버헤드를 감소시켜 네트워크의 효율성을 높여 지연율은 줄일 뿐만 아니라 그에 부가적으로 전력 소비 절감의 효과까지도 기대할 수 있다. 본 논문에서는 Numerical Result를 통하여 제안에 관한 효율성 및 연구 가치를 증명하였다.

주요어 : 해양 무선 음파 센서 네트워크, PBA, 블록 Ack, ARQ

## 1. 서론

무선 통신 기술이 발달함에 따라 점차 새로운 영역에서의 통신의 기회가 많아지고 있다. 다양하고 극히 열악한

환경에서의 무선 통신이 반드시 필요한 경우, 예를 들어 북극이나 남극과 같은 극한 환경에서도 통신은 필요하며, 태풍이 몰아치는 거친 날씨 속이나 깊은 바다 속에서 조차도 통신은 이루어질 수 있다. 물론 매질에 따라 전송의 품질은 상당한 차이를 가질 수밖에 없고 매체접속기술 또한 전송환경에 맞게 변형 될 것이다. 최근 물속에서의 통신에 대한 연구가 전 세계적으로 진행되기 시작하고 있는 것은 주목할 만하다.

수중 환경에서 무선 통신을 하게 되는 경우 주로 음파 신호를 사용하게 되는데, 음파 신호는 수중에서 공기보다 빠른 속도로 수 km 이상의 먼 거리까지 전파되는 특성을 가지고 있어서 원거리에서도 탐지할 수 있는 장점을 가지

\* 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 지원을 받아 수행되었음.  
(IITA-2006-C1090-0603-0044)

2006년 11월 27일 접수, 2006년 12월 8일 채택

<sup>1)</sup> 국민대학교 Business IT 전문대학원 비즈니스정보통신 연구실

주 저 자 : 정유진

교신저자 : 박수현

E-mail; shpark21@kookmin.ac.kr

고 있으나 계절, 수심 등 주변 환경에 많은 영향을 받기 때문에 열악한 환경으로 인해 발생하는 문제들을 고려해야 한다. 수중 환경에서는 대부분 나쁜 품질을 가지는 제한적인 대역폭의 물리적 링크를 제공하게 되며<sup>[1]</sup>, 이로 인해 높은 에러율과 지연율뿐 아니라, 각종 환경적 변수들로 인한 예측할 수 없는 채널의 상태 변화 여지와 도플러 효과로 인한 왜곡의 가능성도 가지게 된다. 특히, 천해(淺海)에서는 해면과 해저 바닥 면에서 음파의 반사와 산란이 심하고 담수에 의한 매질의 특성이 상이하므로 오·탐지 확률이 높다<sup>[2]</sup>.

아직까지 수중에서 이루어지는 무선 통신은 지상에서의 그것에 비해 많이 활성화 되지 않은 상태이다. 그러나 수중 환경에서도 해양학적 정보수집, 오염 정도 측정, 해양 탐사, 재해방지, 항해 정보 제공, 전술상의 탐지 등 다양한 분야에서 무선 통신을 응용할 수가 있으며<sup>[3]</sup>, 이 모든 과정에서 데이터 전송의 성공률을 향상시키고 에러의 감지 및 복구를 최소한의 비용으로 해결할 수 있도록 하는 것은 중요하다.

본 논문에서는 수중 환경에서뿐 아니라 다양하고 치명적인 약점을 가지고 있는 무선 환경에서도 활용할 수 있는 효율적이고 전송의 횟수를 획기적으로 줄인 방법을 제안하였다. ACK 트래픽 발생 빈도 감소를 위해 Block ACK(BA)를 도입, 수정하였고 Pervasive Block ACK(PBA)의 개념을 제안하여 BA의 시작요청, 데이터의 순차적 수집과 관리, BA의 전송요청을 별도의 패킷 전송 없이 수행하도록 하였다. 전송 횟수를 최소로 줄임으로써 오버헤드(overhead)를 감소시켜 네트워크의 효율성을 높임에 따라 지연율을 줄이고 그에 부가적으로 전력 절전의 효과까지도 기대할 수 있다.

2장에서는 관련 연구로서 ARQ 기법<sup>[4]</sup>과 최근에 표준화 된 IEEE 802.11e에서 사용하는 Block Ack 기법, 그리고 현재 연구가 진행 중인 수중에서의 MAC을 간단히 언급한다. 3장에서는 다양한 환경에서 특히 수중과 같은 무선 통신에 적용 가능한 Pervasive Block ACK 기법의 메커니즘을 제시하고, 4장에서 Pervasive Block Ack 의 이론적인 정립을 위한 Analytical Formula를 이용한 분석 모델을 정의하고자 한다. 5장에서는 4장에서 유도된 수식을 근거로 한 Numerical result를 통한 효율성 분석으로 PBA 기법 연구의 타당성 및 효율성을 증명하며 6장에서 요약하고 향후 연구 방향을 제시하며 결론을 맺는다.

## 2. 관련 연구

ARQ(Automatic Repeat Request) 메커니즘은 ACK와 관련된 가장 기본적인 메커니즘이다.

이 기법은 (Fig. 1)과 같이 각 패킷 마다 ACK를 보내어 패킷의 성공 여부를 알려주는데, 송신 노드(Sender)가 패킷을 전송한 후에 수신 노드(Receiver)가 보내는 Acknowledgement (ACK)를 받을 때까지 기다리게 된다.

송신 노드는 전송한 패킷을 자신의 버퍼(buffer)에 복사하여 둔다. 만일 ACK가 타임아웃(Time Out, 임의로 정한 데이터 전송 후 수신 측으로부터의 응답을 기다릴 수 있는 최대 시간) 내에 도착하지 않거나 혹은 Negative ACK(NAK)가 도착할 경우 버퍼에 저장해 둔 패킷을 재전송하고, ACK가 정상적으로 도착하면 그 패킷을 버퍼에서 제거시킨 후 새로운 다음 패킷을 전송 하게 된다. ARQ 기법은 통신 프로토콜이 단순하고 데이터전송의 신뢰성을 가진다는 장점을 가지지만<sup>[5]</sup>, 패킷의 크기가 커질수록 전파의 지연율도 높아서 링크의 효율이 낮으며, 수신 측으로부터 ACK/NAK를 수신하거나 타임 아웃이 발생할 때까지 다음 데이터 프레임을 전송할 수 없으므로 전송효율이 저하되어 정보를 연속적으로 보낼 수 없다는 단점을 가지게 된다.

현재에도 효율성을 높이기 위한 ARQ 연구와 실제 구현이 병행되고 있는데, 그 중 IEEE 802.11e에서는 ACK 전송에 따른 채널 낭비를 줄이기 위해서 적응적인 Acknowledge 기법인 Block Ack(BA) 기법을 사용하고 있다. 기본적으로 Block Ack 기법은 MPDUs(MAC Protocol Data Units)를 전송할 때에 SIFS(Short Inter Frame Space) period를 통해 분할된 Block 단위로 나누어 전송하도록 하며<sup>[6]</sup>, 각각의 Block마다 ACK를 보내지 않고 마지막 Block을 받은 후에 aggregating ACK를 Block Ack로 보내준다.

IEEE 802.11e 표준에서 사용되는 Block Ack 기법<sup>[7]</sup>은 다소 여유 있는 대역폭을 사용할 수 있는 802.11 계열에서 고안된 방법이다. 따라서 Block Ack를 주고받기 위한 셋업(Setup) 및 메시지 시퀀싱(Sequencing) 과정에서 컨트롤 패킷이 다수 발생하기 때문에 수중 환경에서 무선 통신을 하게 되는 경우 과도한 오버로드를 가지게 된다는 문제가 발생할 수 있다.

이외에도 현재 해양에서 사용 가능한 MAC 메커니즘 연구가 점점 활발하게 진행되고 있는데, 아직까지 ARQ

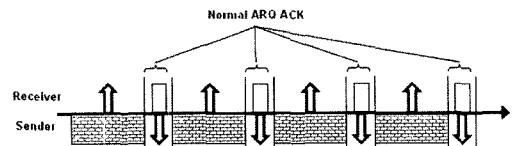


Fig 1. ACK in Normal ARQ

에 관련된 연구는 ARQ 기법을 기반으로 한 연구가 대부분을 차지하고 있다.

### 3. Pervasive Block Ack

본 논문에서 제안하는 Pervasive Block ACK(PBA)는 다양한 토폴로지에서 사용이 가능한 개념으로, infrastructure 토폴로지나 애드 혹 네트워크 등 토폴로지의 구성과 상관없이 모든 link-by-link 연결 상의 오류 제어에서 적용이 가능하다.

새로운 PBA 기법은 수중 환경과 같은 열악한 상황에 적용하기에 더욱 적합하다. 수중에서는 메시지 LQI(Link Quality Indication) 능력이 상당히 저하되어 메시지의 개수를 감소시켜야 할 필요성이 지상에 비해 더더욱 증대된다. 제시된 PBA 기법을 통해 전송을 하게 된다면, 기존의 IEEE 802.11e에서의 BA 기법을 사용할 때와 비교하여 트래픽 발생 빈도를 크게 줄일 수 있게 된다. 이러한 사실은 (Fig. 2)를 통해 증명할 수 있다.

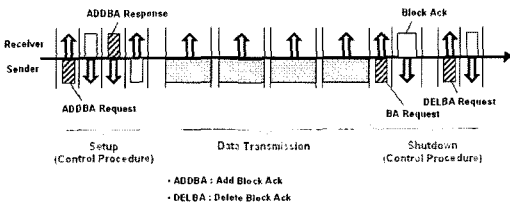
새로운 PBA 기법의 기본적인 흐름은 (Fig. 3)과 같은데, 다른 기법들과 달리 본 제안 기법에서는 초기화(Initiation)와 종료(Shutdown) 단계를 과감히 축소하여 기 전송 데이터 프레임의 일부에 포함시킴으로써 트래픽 전송량을 줄여 오버헤드를 최소한으로 감소시켰다.

또한 프레임 구성에 있어서는 불필요한 필드(field)들을 과감히 삭제하여 기존의 프레임 포맷보다 상당히 길이가 줄어들어 오버헤드의 부담이 적어지도록 하였다.

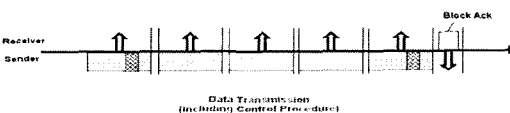
먼저 송신 노드는 자신의 큐에 저장된 데이터 프레임들을 전송하게 되는데, 초기화 절차가 없이 데이터 프레임들을 보내던 중에 데이터 프레임 내부에 Flag와, State 필드를 포함하여 전송한다. 이 필드들은 다음 패킷부터가 PBA의 대상이라는 표시로 PBA의 초기화 역할을 포함하게 되며, 이 필드들을 포함한 데이터 프레임은 초기화라는 중요한 역할을 하게 되므로, 수신 측면에서의 신뢰성 확보를 위하여 기존의 ARQ 기법을 사용하여 전송된다. 일단 이 PBA 초기화 정보를 담은 데이터 프레임에 대한 ACK가 수신 노드로부터 도착하면 다음 데이터 프레임들을 PBA기법을 사용하여 연속적으로 전송한다. 이런 식으로 연속적으로 전송된 데이터 프레임들은 수신 측에 위치한 큐에 저장된다. 이 기법에서는 처음에 첫 번째 데이터 프레임에 시작을 알리는 Flag를 실어 보냈던 것과 마찬가지로 부가적인 종료 절차를 가지지 않으며, 마지막으로 전송되는 데이터 프레임에 PBA를 요청하는 메시지를 나타내는 Flag를 포함하여 전송하게 된다. PBA를 요청하는 Flag를 포함한 마지막 데이터 프레임은 받은 수신 노드는 자신의 큐에 저장된 데이터 프레임들을 확인하여 PBA 혹은 PB-NAK(Pervasive Block NAK)을 통해 누적된 응답을 하게 된다.

본 제안에서 가장 핵심적인 역할을 수행하는 송신 노드 측에서의 주요 프로세스를 수도코드(Pseudo Code)로 나타내면 다음과 같다.

(Table. 1)은 위 수도코드에 사용된 변수들의 정의를 나타내고 있다.



(a) IEEE 802.11e Block Ack



(b) Pervasive Block Ack

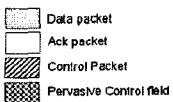


Fig 2. Comparison with Block Ack

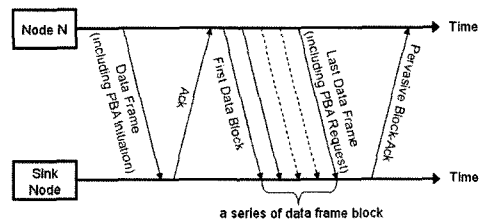


Fig 3. Sequence flow

Table 1. Terms Description (1)

Variable	Description
PBA_Max_Size	하나의 Ack Set에 들어갈 최대 프레임 개수
PBA_Min_Size	하나의 Ack Set에 들어갈 최소 프레임 개수
Blocked_Data	Queue_element 들을 여러 개의 Ack Set으로 분할하고 남은 데이터 프레임 들
Data_Block	큐 안에 쌓인 데이터 각각
Queue_element	큐 안에 있는 데이터들의 개수

```

START.
MEASURE 'Number of Queue_element'
IF
'Number of Queue_element ≥ PBA_Min_Size' THEN
IF
'Queue_element == First Data_Block' THEN
SET
'Flag Field = 1,
State Field = Number of Blocked_Data'
ARQ TRANSMIT
ELSE
IF
'Number of Queue_element = PBA_Max_Size' THEN
'int value = Number of Queue_element/PBA_Max_Size'
MOD (Number of Queue_element,PBA_Max_Size)
IF
'Blocked_Data = 0' THEN
PBA Process
PBA TRANSMIT
ELSE
IF
'Blocked_Data ≥ PBA_Min_Size' THEN
PBA Process
PBA TRANSMIT
ELSE
'State = 0000000'
PBA TRANSMIT
ELSE
PBA Process
PBA TRANSMIT
ELSE
SET
'Flag Field = 0,
State Field = 0000000'
ARQ TRANSMIT
    
```

#### 4. 효율성 평가를 위한 분석 모델(Analytical Model for Efficiency Evaluation)

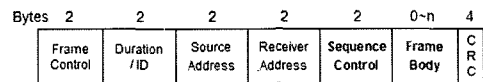
이 장에서는 제안한 Pervasive Block Ack 의 이론적인 정립을 위해 Analytical Formula를 정의하고자 한다. 사용되는 용어에 대한 정의는 (Table. 2)와 같다.

이 장에서 사용되는 수식을 산출하는데 있어 근거가 되는 프레임 형식의 예는 다음 (Fig. 4), (Fig. 5), (Fig. 6) 과 같다.

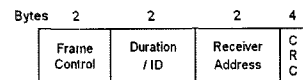
여기서 Source Address 및 Destination Address 필드의 경우는 기본 환경을 긴 주소가 필요 없는 센서네트워크로 정의했기 때문에 각각 2Byte씩으로 정의하였다.

Table 2. Terms Description (2)

$R$	프레임 전송율 (Data Rate)
$B$	Blocking 된 Ack의 개수
$data$	제어정보를 포함한 데이터 프레임
$SET$	Block Ack 의 시작과 끝을 알리는 프레임
$ACK$	한 번의 ACK 를 보내기 위해 필요한 프레임
$L_{total}$	데이터 전송 성공을 위한 프레임 길이의 합 (Data frame + Ack frame)
$L_{data}$	제어 정보를 포함한 데이터 프레임의 길이
$L_{payload}$	MSDU 즉, Payload의 길이
$L_{control}$	데이터 전송 성공을 위한 제어 정보의 총 길이
$L_{ack}$	Ack 프레임의 길이
$\sum L_{ack}$	해당 링크에서 발생된 Ack 프레임의 길이를 모두 합한 값
$N_{total}$	총 전송 횟수
$N_{data}$	데이터 프레임 전송 횟수
$N_{ack}$	Ack 프레임 전송 횟수
$N_{control}$	Ack 를 제외한 제어프레임 전송횟수 (BA association + deassociation)
$Len()$	프레임의 길이를 구하는 함수
$int()$	정수화 함수 (올림수 적용)

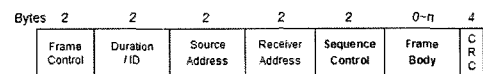


(a) Date Frame

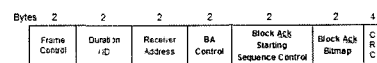


(b) Ack Frame

Fig 4. Frame Format in ARQ

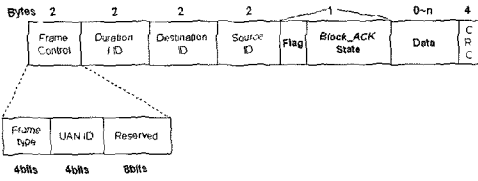


(a) Data Frame



(b) Ack Frame

Fig 5. Frame Format in Block Ack



(a) Data Frame



(b) Ack Frame

Fig 6. Frame Format in Pervasive Block Ack

채널의 사용성은 프레임 전송율을 전체 Bandwidth로 나눈 R/C로 표현할 수 있으며, 또한 이미 사용하고 있는 채널의 효율성은 전체 전송 프레임에서 순수한 데이터의 길이가 차지하는 비율을 말하는데 이는 프레임의 사용 bit 의 개수인 Length로 환산하여

$$\frac{L_{payload}}{L_{total}} = \frac{L_{total} - L_{control}}{L_{total}}$$

로 표현 가능하다. 전체 전송 프레임의 길이는 순수한 데이터인 Payload의 길이와 그 외 제어를 위한 정보들의 길이의 합이 되며 제어정보 길이인  $L_{control}$ 은 데이터 프레임에서 Payload를 뺀 길이와 Ack 전송을 위해 사용되는 길이의 합이 된다.(수식 (1)-(2))

$$L_{total} = L_{payload} + L_{control} \quad (1)$$

$$L_{control} = (L_{data} - L_{payload}) + L_{ack} \quad (2)$$

다음 정의들은 일반적인 Ack 와 데이터 전송 프레임의 내의 구성 필드들을 보여준다. 세 가지 종류의 Ack 길이는 Ack 정보를 제외한 필드가 동일할 때 식 (3)과 같은 관계를 갖는다. 식 (4)는 ARQ, 식 (5)는 Block Ack, 식 (6)은 Pervasive Block Ack의 구성과 (Fig. 4)-(Fig. 6)에서 제시한 각 프레임 길이 계산의 예를 보여준다.

$$Len(ACK_{ARQ}) < Len(ACK_{PBA}) < Len(ACK_{BA}) \quad (3)$$

$$ACK_{ARQ} = Frame\ Control + DurationID + DestinationID + CRC \quad (4)$$

$2+2+2+4 = 10\ bytes = 80\ bits/ ACK\_frame$

$$ACK_{BA} = Frame\ Control + DurationID + DestinationID + (BA\ Control + BlockAck\ Sequence\ Control + BlockAck\ Bitmap) + CRC \quad (5)$$

$2+2+2+(2+2+2)+4 = 16\ bytes = 128\ bits/ Block\_ack\_frame$

$$ACK_{PBA} = Frame\ Control + DurationID + DestinationID + (Flag + PBA\ Bitmap) + CRC \quad (6)$$

$2+2+2+(2)+4 = 12\ bytes = 96\ bits/ Pervasive\_Block\_ack\_frame$

전송 데이터 프레임의 경우 세 가지 기법에서 거의 비슷한 값을 갖게 되는데 아래 식 (7)-(9)과 같다. 이는 데이터 프레임의 길이보다는 같은 데이터 전송 시 Ack 프레임의 길이와 전송 횟수 등 Ack 방식과 프레임 길이의 차이에서 채널의 효율성이 결정된다는 의미가 된다.  $data.ARQ$ ,  $data.BA$ ,  $data.PBA$  는 각각 ARQ, BA, PBA를 위한 데이터 프레임임을 뜻한다.

$$L_{data.ARQ} = L_{data.BA} \geq L_{data.PBA} \quad (7)$$

$$data.ARQ = data.BA = Frame\ Control + DurationID + SourceID + DestinationID + Sequence\ Control + Payload + CRC \quad (8)$$

$2+2+2+2+2+Payload+4 = 14+Payload\ bytes$

$$data.PBA = Frame\ Control + DurationID + SourceID + DestinationID + (Flag + PBA\ State) + Payload + CRC \quad (9)$$

$2+2+2+2+(1)+Payload+4 = 13+Payload\ bytes$

다음 부분은 Ack 프레임과 컨트롤 프레임의 전송횟수와 이로부터 산출된 메시지별 총 길이를 설명한 것이다. 식 (10)-(11)은 ARQ 기법, 식 (12)-(13)은 Block Ack 기법, 식 (14)-(15)는 Pervasive Block Ack에 대한 부분이다.

$$N_{ack.ARQ} = N_{data} \quad (10)$$

$$\sum L_{ack.ARQ} = Len(ACK_{ARQ}) \times N_{ack.ARQ} \quad (11)$$

$\frac{N_{data}}{B}$ 는 데이터의 전송 횟수를 블록된 Ack의 개수인 B로 나누어 준 것으로, Block Ack 혹은 Pervasive Block Ack의 전송 횟수 계산을 위한 식이다. int()를 통해 (Pervasive) Block Ack 의 전송 횟수를 정수화 하였다. 식 (12)에서 세 배의 전송 횟수가 요구되는 이유는 Block Ack 전송 시에  $SET_{BA}$ , 즉 BA 시작과 BA 종료에 위한 두 개의 프레임 전송이 추가되었기 때문이다.

$$N_{ack.BA} = 3 \cdot \text{int}\left(\frac{N_{data}}{B}\right) \quad (12)$$

$$\begin{aligned} \Sigma L_{ack.BA} = & \text{Len}(ACK_{BA}) \times \text{int}\left(\frac{N_{data}}{B}\right) + \\ & 2 \cdot \text{Len}(SET_{BA}) \cdot \text{int}\left(\frac{N_{data}}{B}\right) \end{aligned} \quad (13)$$

BA의 경우와 비교해 PBA는 Ack를 위해  $SET_{BA}$  같은 추가적인 제어 프레임의 전송이 없다. 데이터와 Ack 내부의 최소한의 정보로 이러한 효율성 제고가 가능해 진다.

$$N_{ack.PBA} = \text{int}\left(\frac{N_{data}}{B}\right) \quad (14)$$

$$\Sigma L_{ack.PBA} = \text{Len}(ACK_{PBA}) \times \text{int}\left(\frac{N_{data}}{B}\right) \quad (15)$$

다음 절에서는 유도 된 수식에 근거하여 제어 프레임 및 전체 네트워크 프레임의 전송 횟수 및 프레임의 길이를 비교하여 분석하려 한다.

### 5. Numerical Result with Analytic Formula

5장에서 사용된 그래프를 도출하는데 있어서 고정된 Block의 개수가 필요할 경우는 10, 고정된 데이터 프레임의 개수가 필요할 경우는 100으로 통일하여 값을 산출하였음을 미리 명시한다.

(Fig. 7)은 데이터 프레임의 개수가 증가함에 따른 컨트롤 트래픽의 양을 나타낸 그래프이다. 데이터 프레임의 개수가 증가함에 따라 ARQ에서의 컨트롤 프레임 로드(Load)는 매우 큰 폭으로 데이터 프레임과 비례하여 늘어

나는 모습을 보이는데 반해, Block Ack나 PBA에서의 증가 폭은 점점 줄어드는 모습을 볼 수 있다. 특히 PBA의 경우는 자체적으로 컨트롤 프레임의 양이 적기 때문에 Block Ack와 비교해도 훨씬 적은 로드 증가를 보여준다.

(Fig. 8)은 하나의 Ack Set에 들어가는 Block Ack의 수가 증가함에 따라(즉, Block Ack가 더 많은 Ack를 포함함으로써 Ack의 횟수를 줄임에 따라) 컨트롤 트래픽의 로드를 감소시킬 수 있음을 나타내는 그래프이다.

특징적인 결과로써, ARQ의 경우는 균등한 값을 나타내는 것을 볼 수 있다. 여기서 ARQ의 경우는 Blocking 기능 자체를 제공하지 않기 때문이다.

(Fig. 9)는 데이터 프레임 개수 증가에 따른 전체 트래픽 로드를 나타낸 그래프이다.

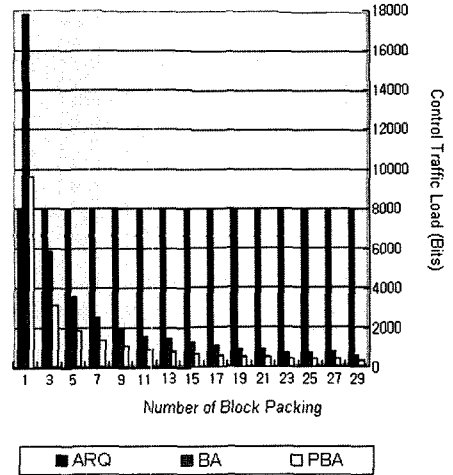


Fig 8. Control Traffic Load (2)

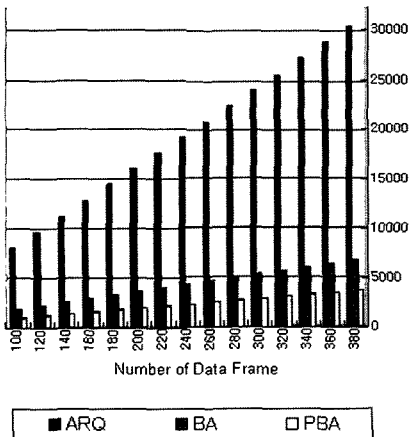


Fig 7. Control Traffic Load (1)

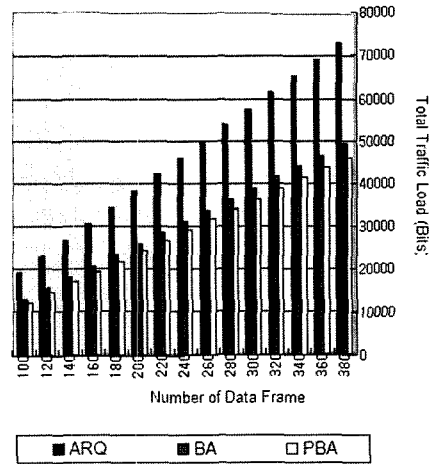


Fig 9. Total Traffic Load (1)

데이터 프레임 전송 개수가 증가함에 따라 ARQ와 Block Ack, 그리고 PBA 모두 증가하는 모습을 보이지만, 그 증가의 폭이 현저히 차이 나고 있음을 눈으로 볼 수 있다.

(Fig. 10)은 하나의 Ack Set에 들어가는 Block Ack의 수가 증가함에 따라 전체적으로 전송되는 트래픽의 로드 증가폭이 점점 감소하고 있음을 나타내는 그래프이다. 일대일로 통신하는 경우 Block Ack의 경우는 초기화와 종료료를 나타내는 컨트롤 프레임 때문에 다소 큰 Load를 보이지만, 하나의 Ack Block Set에 들어가는 숫자가 늘어날수록 전체적인 트래픽 Load가 현저히 줄어드는 모습을 보이고 있다. 마찬가지로 PBA의 경우도 일대일 통신의 경우 일반적인 ARQ에서 사용되는 ACK 프레임의 길이보다 2Bytes가량 길기 때문에 ARQ보다 약간 더 많은 로드를 보이지만, Ack Set 내의 Block 숫자가 증가할수록

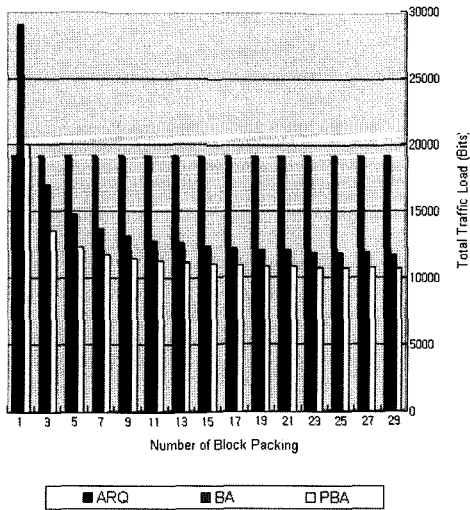


Fig 10. Total Traffic Load (2)

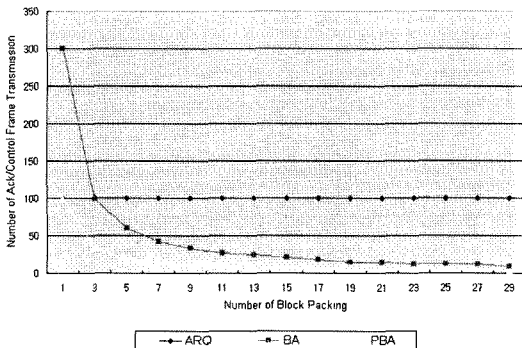


Fig 11. Number of Ack/Control Frame Transmission

Block Ack보다도 훨씬 더 감소하는 모습을 보여주고 있다. 여기에서도 ARQ의 경우에는 Blocking이 적용되지 않기 때문에 일정한 값을 나타내고 있다.

(Fig. 11)은 하나의 Ack Set 안에 들어가는 Block의 수가 증가함에 따라 Ack 프레임 및 컨트롤 프레임의 전송 횟수 변화를 나타내고 있는 그래프이다.

그림에서 ARQ는 역시 Blocking 기법이 적용되지 않기 때문에 Ack 및 컨트롤 프레임의 몇 개를 Block으로 묶든 전송횟수에 변화가 일어나지 않는다.

Block Ack의 경우 PBA에 비해 다수의 컨트롤 프레임을 가지고 있으므로 처음에는 월등히 높은 전송횟수를 보이지만 Block의 개수가 늘어날수록 점점 전송횟수가 감소하는 모습이 보인다.

이와 비교하여 PBA는 별다른 컨트롤 프레임이 없으므로 처음에는 ARQ와 같은 전송횟수를 보이다가, Block의 개수가 늘어날수록 Block Ack 보다도 더 적은 횟수로 감소하는 모습을 보여주고 있다.

## 6. 결론

본 논문에서는 변동이 심한 무선 환경이나 수중 등 열악한 전송환경에서 성공적인 데이터 전송이 가능하도록 지원 할 수 있는 오버헤드 감소 기법인 Pervasive Block ACK 방식을 제안하였다.

제안한 Pervasive Block ACK 기법은, 적정한 데이터 크기를 유지할 뿐 아니라 Ack를 비롯한 제어 프레임의 전송 횟수를 최소로 줄여 오버헤드를 감소시키고 네트워크의 효율성을 높였다. Numerical result 를 보면 ARQ와 비교해서 탁월한 성능향상을 보여주고 있으며, Block Ack 메커니즘과의 비교에서도 대폭 줄어든 전송 횟수를 볼 수 있었다.

향후 시간 개념을 추가하여 전송 시간에 따른 효율성을 증명하고, 시뮬레이션을 통하여 신뢰성 있는 다양한 각도의 성능 및 채널 효율성을 검증할 것이다.

## 참고 문헌

1. Stojanovic, "Optimization of a Data Link Protocol for an underwater acoustic Channel", Oceans-Europe 2005, pp. 68-73, 2005.
2. Robert J.Urick, Principles of underwater sound, Peninsula Publishing, Los Altos, pp. 172-182, 1983.
3. Ian F.Akyildiz et al, "Underwater acoustic sensor networks : research challenges", Ad Hoc Networks 3, pp.

- 257-279, 2005.
4. Shu Lin et al, "Automatic-repeat-request error-control schemes", IEEE Communications Magazine Vol. 22, No. 12, pp. 5-17, 1984.
  5. Deng-Lin LU et al, "Analysis of ARQ Protocols via Signal Flow Graphs", IEEE Transactions on Communications Volume. 37, pp. 245-251, 1989.
  6. Ilenia Tinnirello et al, "Efficiency Analysis of Burst Transmissions with Block ACK in Contention-Based 802.11e WLANs", Communications, 2005. ICC 2005. 2005 IEEE International Conference on Volume 5, pp. 3455-3460, 2005.
  7. IEEE Standard 802.11e-2005 (Amendment to IEEE Std 802.11, 1999 Edition (Reaff 2003)), 2005.
  8. M.Molins et al, "Slotted FAMA: a MAC protocol for underwater acoustic networks", MTS/IEEE OCEANS, 2006.
  10. X.Guo et al, "A Propagation-delay-tolerance Collision Avoidance Protocol for Underwater acoustic sensor network", MTS/IEEE OCEANS, 2006.
  11. F.Salva-Garau et al, Multi-cluster Protocol for Ad Hoc Mobile Underwater Acoustic Networks", MTS/IEEE OCEANS, 2006.



**정 유 진** (yj-best@kookmin.ac.kr)

2004년 정보통신공학 학사  
2005년~현재 국민대학교 BIT대학원 비즈니스정보통신 석사과정

관심분야 : 유비쿼터스 네트워크, 이동통신 시스템, MAC



**신 수 영** (sy-shin@kookmin.ac.kr)

1998년 방송통신대학교 교육학사  
2000년 덕성여자대학교 정보통신대학원 정보통신 전공 이학석사  
2004년~현재 국민대학교 BIT대학원 비즈니스정보통신 박사과정  
2003년~현재 덕성여자대학교 강사  
2004년~현재 국민대학교 강사

관심분야 : USN, 이동통신 시스템, MAC scheduling



**박 수 현** (shpark21@kookmin.ac.kr)

1988년 고려대학교 컴퓨터학과 이학사  
1990년 고려대학교 대학원 전산학 이학석사  
1998년 고려대학교 대학원 컴퓨터학과 이학박사  
1990년 (주)LG전자 중앙연구소 선임연구원  
1999년~2001년 동의대학교 공과대학 소프트웨어공학과 교수  
2002년~현재 국민대학교 비즈니스IT학부 교수  
2001년~현재 한국SI학회 이사

관심분야 : 유비쿼터스 네트워크, 이동통신 시스템, Active Network