

다중 채널 다중 인터페이스 능동형 RFID 리더 및 프로토콜

(Multi-Channel Multi-Interface Active RFID Reader and Protocol)

박현성[†] 김동현[†] 정상화^{**} 백윤주^{**} 김종덕^{***}
(Hyun-Sung Park) (Dong-Hyun Kim) (Sang-Hwa Chung) (Yun-Ju Baek) (Jong-Doek Kim)

요약 433Mhz에서 단일 채널을 사용하는 ISO 18000-7 능동형 RFID 표준은 낮은 전송 속도, 단일 채널에 따른 간섭 문제 등으로 최근의 응용 요구를 충족시키는데 어려움이 있다. 2.4Ghz 대역에서 다수 채널을 사용하는 새로운 능동형 RFID 시스템을 제안한다. 이 시스템은 하나의 리더에 여러 개의 인터페이스를 장착하여 성능을 개선하는 방안이 특징이다. 그런데 다수의 인터페이스를 장착하여도 대부분의 태그들이 특정 인터페이스로 몰린다면 기대하는 성능 개선이 어렵다. 이를 극복하기 위해 인터페이스간 부하 분배를 고려한 능동형 RFID 프로토콜이 필요하다. 우리는 단순결합, LP-조합, AP-균형이라는 세 가지 프로토콜을 제안하고 비교, 분석하였다. 태그의 수, 인터페이스 수, 태그의 부가 정보의 양 등을 변화시키면서 수행한 다양한 성능 평가 결과 AP-균형이 가장 우수한 성능을 보였다. AP-균형은 인터페이스 수에 거의 선형적으로 비례하여 성능이 증가하며 기대를 충족하였다.

키워드 : 능동형 RFID, 다중 채널, 다중 인터페이스, 부하 분배

Abstract The ISO 18000-7 Active RFID standard, a single channel system operating in the 433Mhz, faces technical difficulties in supporting some recently introduced application demands because of its low transmission rates and radio interference between readers. We propose a new multi-channel active RFID system operating in the 2.4Ghz. The special feature of the proposed system is that a reader makes use of multiple interfaces to improve its performance like a multi-core processor. However if only a small part of interfaces are actually used, the performance improvement would not meet the expectation. To overcome this problem, a new multi-channel multi-interface active RFID protocol, which balances communication loads among all available interfaces, is necessary. 3 protocols, named as "Aggregated", "LP-Combined", "AP-Balanced", are proposed. Through simulation, we compare them for various conditions by changing number of tags, number of interfaces, tag data size. AP-Balanced shows the best performance and its performance increases almost linearly as the number of interface increases, which meets our expectation.

Key words : Active RFID, Multi-Channel, Multi-Interface, Load Balancing Bilinear-pairing

* 본 연구는 지식경제부 지방기술혁신사업(B0009720) 지원으로 수행되었음

[†] 학생회원 : 부산대학교 정보컴퓨터공학과

jesse@pusan.ac.kr

dhkim1106@pusan.ac.kr

^{**} 중신회원 : 부산대학교 정보컴퓨터공학과 교수

shchung@pusan.ac.kr

yunju@pusan.ac.kr

^{***} 정 회원 : 부산대학교 정보컴퓨터공학과 교수

kimjd@pusan.ac.kr

논문접수 : 2008년 9월 24일

심사완료 : 2008년 12월 16일

Copyright©2009 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지 : 정보통신 제36권 제2호(2009.4)

1. 서론

능동형 RFID 시스템이란 태그가 배터리 및 송신 장치를 내장하고 있어 리더로부터의 에너지 공급 없이도 리더와 정보를 송, 수신할 수 있는 시스템이다. 능동형 RFID 시스템은 수동형 RFID 시스템에 비해 태그의 가격, 크기, 배터리 관리 등의 측면에서는 불리하지만 인식 거리, 인식률, 안정성, 부가 기능 등에서 훨씬 유리하다. 수동형 RFID 시스템이 인식률 등의 측면에서 기술적 문제가 있어 실용화 수준이 기대에 미치지 못하고 있는데 비해 능동형 RFID 시스템은 고가의 중, 대형 물품이나 컨테이너와 같은 대형 물류 용기 관리 등의 응용 분야에서 실용화가 빠르게 진행 중이다.

대표적 능동형 RFID 시스템으로 ISO 18000-7 능동형 RFID 표준이 있다[1]. ISO 18000-7 표준은 433Mhz 대역의 주파수를 사용하며 채널 대역이 200Khz인 단일 채널 시스템으로 데이터 전송 속도는 27.8Kbps 수준이다. 이 시스템은 미국 국방성의 군수품 보급 관리, 국토안보국의 컨테이너 보안 관리 등에 쓰이고 있다. 우리나라에서도 국토해양부의 항만자동화, 국방부의 군수품 관리 등의 공공 분야에 도입되어 사용 중이다.

능동형 RFID 응용 분야의 확대와 함께 기본적인 식별 기능 외에 사용자 정보의 저장 및 전달, 온·습도 등의 주변 정보 센싱 및 전달, 실시간 측위(RTLS: Real Time Location System) 등의 추가 기능 요구가 발생하고 있다. 그런데 이러한 기능을 수용하는데 있어 ISO 18000-7 기술은 한계가 있다. 첫째, 27.8Kbps 수준의 전송 속도는 대용량의 정보의 송, 수신이 필요한 현실 응용의 요구를 만족시키기에 충분하지 못하다. 둘째, 단일 채널 시스템인 이유로 시스템 도입이 늘수록 주변 리더들간의 채널 중복에 따른 간섭으로 인해 성능 저하 가능성이 증가한다. 셋째, 433Mhz는 다중경로전파(Multi-Path Propagation)나 파장 등의 주파수 특성 상 2.4Ghz와 같은 다른 주파수 대역에 비해 실시간 측위 측면에서 유리하지 못하다. 마지막으로 국내 전파법은 기존 아마추어 무선 통신 대역과의 충돌을 방지하기 위해 433Mhz RFID를 항만 지역에서만 사용할 수 있도록 규제하고 있으며 이는 433Mhz 시스템 도입에 주요한 걸림돌이 되고 있다.

ISO 18000-7 능동형 RFID 표준이 가지는 이러한 한계를 극복하기 위해 2.4Ghz ISM 대역을 사용하는 새로운 능동형 RFID 시스템을 제안한다. 2.4Ghz 대역은 별도의 사용 권한 획득이 필요 없고 실시간 측위 기술 측면에서도 433Mhz에 비해 유리하다. 활용 가능한 주파수 대역이 100Mhz 이상으로 하나의 채널에 ISO 18000-7 표준의 5배에 해당하는 1Mhz를 할당하여도

100여 개 이상의 다채널로 구성할 수 있다. 다수 채널을 사용하여 리더간 주파수 중복에 따른 성능 저하를 피할 수 있고 개별 채널의 대역폭이 늘어남에 따라 전송 속도도 증가시킬 수 있다.

2.4Ghz 주파수 대역을 활용함으로써 여러 측면의 장점을 얻을 수 있지만 우리는 전송 속도 증대에 중점을 두었다. 사용자 정보나 센싱 정보 전달의 요구가 높고 그 용량 역시 커지고 있기 때문이다. 전송 속도 증대를 위해 개별 채널의 대역폭을 늘이거나 변조율을 높여 속도를 증가시키는 접근도 있지만 과도할 경우 태그 비용 증대, 안정성 저하 등의 문제를 가져 올 수 있다.

우리는 리더에 복수의 인터페이스를 장치하고 각 인터페이스가 서로 다른 채널을 통해 태그들과 통신할 수 있도록 함으로써 성능을 개선시키는 방안을 제시한다. 컴퓨터 CPU 분야에서 멀티 코어를 이용한 성능 개선과 유사한 접근이라 볼 수 있다. 그런데 리더의 인터페이스가 다수라도 모든 통신이 한 인터페이스를 통해서만 이루어질 경우 단일 인터페이스 리더에 비해 성능 개선이 전혀 이루어지지 않을 수 있다.

이러한 다중 인터페이스 RFID 리더에서 인터페이스간 부하 분배를 통한 성능 개선은 이 논문에서 다루려는 핵심 문제이다. 이를 해결하는 방법론을 개발하기 위해서는 RFID 시스템의 인식 및 통신 방식에 대한 분석이 필요하다. 논문에서 제안하는 2.4Ghz 능동형 RFID 시스템은 인식 및 통신 과정의 기본 틀을 기존 ISO 18000-7 능동형 RFID 표준을 따르도록 하였다. 2장에서는 RFID 시스템에서 일반적으로 사용하는 알로하(Slotted Aloha) 기반 태그 인식 알고리즘을 포함한 ISO 18000-7 표준의 리더와 태그간의 통신 절차를 소개한다. 또 태그 인식 성능에 중요한 영향을 미치는 윈도우 값 설정 방법 및 관련 연구에 대해서도 설명한다. 3장에서는 제안하는 다중 채널 능동형 RFID 시스템을 개괄적으로 소개한다. 리더 간 간섭에 따른 성능 저하를 회피하기 위한 채널 할당 방법, 태그와 리더 간의 채널 동기화 방법 등을 설명한다. 4장에서는 다중 인터페이스 리더에서의 부하 분배 문제를 구체적으로 분석하고 이를 해결하기 위한 프로토콜을 제시하고 시뮬레이션을 통해 개발한 프로토콜을 검증, 평가한다. 5장은 논문의 결론을 포함한다.

2. 능동형 RFID 표준 기술 및 관련 연구

태그간 충돌 문제는 RFID 시스템에서 다루어야 하는 근본 문제이다. 이것은 리더가 다수의 태그를 대상으로 태그수집 명령을 전송했을 때 태그들이 이에 응답하는 과정에서 동시에 전송된 응답 메시지들이 서로 간섭을 일으켜 인식이 이루어지지 않는 상황을 가리키는 것이다.

태그간 충돌 문제 해결에 널리 쓰이는 알고리즘으로 트리워킹(Tree Walking)과 알로하가 있다. 트리워킹 알고리즘은 이진 값으로 표현되는 태그의 식별 번호를 비트 단위로 구분하면서 인식하는 기법이며 확률적 요소가 없는 성능이 고정적인 알고리즘이다[2]. ISO 18000-1, 18000-6A 등의 수동형 RFID 표준들에서 이 알고리즘을 사용하고 있다. 알로하 알고리즘은 통신 분야에서 잘 알려진 다중 접속 알고리즘을 활용한 것으로 개별 태그가 슬롯이라는 단위로 나뉜 윈도우라는 경쟁 구간에서 한 슬롯을 임의로 선택하여 충돌을 회피하는 알고리즘이다. 트리워킹 알고리즘과 달리 확률적 요소가 성능에 큰 영향을 미치는 알고리즘이다. 이 알고리즘은 Gen2라 불리는 최신 수동형 RFID 표준인 ISO 18000-6C와 ISO 18000-7 능동형 RFID 표준 등에서 쓰이는 대표적 태그간 충돌 회피 알고리즘이다.

알로하 알고리즘의 성능은 윈도우 크기와 밀접한 관계가 있다. 리더는 태그수집 명령을 전송하면서 인자 값으로 윈도우 크기 ω 를 포함한다. 태그수집 명령을 수신한 개별 태그는 Random(1, ω) 형태의 난수발생 함수를 이용하여 1과 ω 사이의 임의의 정수를 선택하고 그에 대응하는 슬롯에서 응답한다. 각 슬롯은 어떤 태그도 응답을 하지 않는 빈 슬롯, 하나의 태그만 응답하여 성공적 인식이 이루어지는 성공 슬롯, 하나 이상의 태그가 응답하여 충돌이 발생하는 충돌 슬롯 중 한 형태가 된다. 윈도우 크기가 ω 이고 태그의 수가 N 인 조건에서 어떤 슬롯이 빈 슬롯, 성공 슬롯, 충돌 슬롯이 될 확률을 각각 P_0 , P_1 , P_c 라고 할 때 이들은 식 (1)과 같이 유도할 수 있다.

$$\begin{aligned} P_0 &= \left(1 - \frac{1}{\omega}\right)^N \\ P_1 &= N \cdot \frac{1}{\omega} \cdot \left(1 - \frac{1}{\omega}\right)^{N-1} \\ P_c &= 1 - P_0 - P_1 \end{aligned} \quad (1)$$

태그 수가 N 일 때 어떤 슬롯이 성공 슬롯이 될 확률 P_1 은 ω 를 N 으로 할 때 최대가 됨을 미분을 통한 최대값 구하기로 알 수 있다. $\omega=N$ 일 때 확률 P_1 을 알로하 알고리즘의 최고 기대 성능으로 해석할 수 있고 식 (2)와 같이 유도할 수 있다. 기대 성능은 $N>1$ 정수에 대해 단조 감소하며 N 이 커짐에 따라 값은 $\frac{1}{e} = 37.5\%$ 에 수렴한다.

$$\therefore P_1(\omega=N) = \left(1 - \frac{1}{N}\right)^{N-1}, \lim_{N \rightarrow \infty} P_1(\omega=N) = 37.5\% \quad (2)$$

앞의 결과는 인식해야 하는 태그의 수 N 을 미리 알면 그를 바탕으로 윈도우 값을 정하여 성능을 개선할 수 있음을 보여준다. [3,4] 등은 알로하 알고리즘의 이러

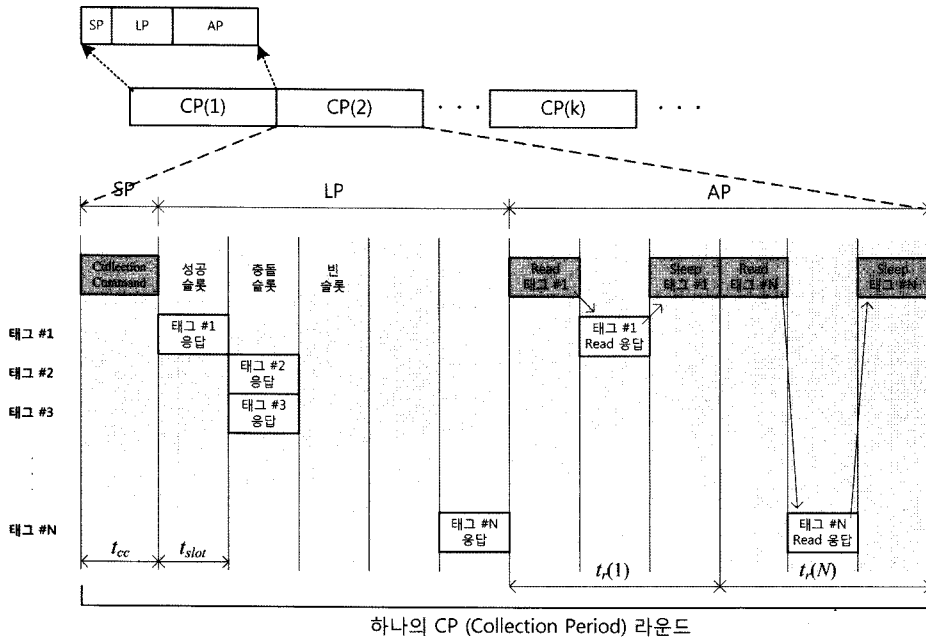
한 특성을 고려하여 태그의 수를 추정하는 방법을 제시한 연구들이다. 제시된 방법들은 비슷한 구조를 가지는데 먼저 내정 값 등을 이용하여 초기 윈도우 값을 설정하고 태그 수집 작업을 수행한다. 수행 결과 얻은 실제 빈 슬롯, 성공 슬롯, 충돌 슬롯의 수와 각각의 이론적 예측 값을 비교하여 N 을 추정한다. 이를 바탕으로 새 윈도우 값을 정하고 다시 태그 수집 작업을 수행한다. 모든 태그를 인식할 때까지 이 과정을 반복한다.

윈도우 값 결정을 위해 우리가 사용한 알고리즘은 Vogt가 제시한 방법을 기반으로 하고 있다[3]. Vogt의 방법은 다른 방법들에 비해 계산량은 많지만 태그 수 N 을 보다 정확히 추정할 수 있다. 논문에서 다룬 인터페이스간 부하 분배 방법의 적용을 위해 반드시 Vogt의 방법을 사용할 필요는 없다. 다만 태그 수 추정의 정확도가 성능에 미치는 영향을 최소화하기 위하여 이 방법을 사용한다. Vogt의 방법에서는 윈도우 크기를 ω 로 하고 시행한 태그수집 작업 결과 빈 슬롯, 성공 슬롯, 충돌 슬롯의 수가 각각 C_0 , C_1 , C_c 라면 태그 수를 식 (3)을 최소로 하는 N 값으로 추정한다.

$$D(N) = \sqrt{(C_0 - \omega \cdot P_0)^2 + (C_1 - \omega \cdot P_1)^2 + (C_c - \omega \cdot P_c)^2} \quad (3)$$

그림 1은 ISO 18000-7표준의 인식 과정을 개괄적으로 표현한 것이다. 인식과정은 세부적으로 SP(Synchronization Period), LP(Listen Period), AP(Access Period)로 이루어진 CP(Collection Period)를 반복하며 수행된다. SP는 "Collect"라는 태그수집 명령을 전송하는 구간이다. 이 명령은 이어지는 LP에서 사용할 윈도우 크기 값 ω 를 포함한다. LP는 알로하 방식으로 태그간 충돌을 회피하면서 태그가 리더에게 자신의 ID를 전송하는 구간이다. AP 구간에서 리더는 LP 구간에서 성공적으로 인식한 태그들에 대하여 부가 정보, 즉 사용자 정보나 센싱 정보 등을 읽어온 후 태그를 절전모드로 변경시키는 작업을 수행한다. LP 구간에서 성공적으로 ID를 전달하지 못한 태그는 다음 CP 라운드에 다시 참여하며 인식이 성공할 때까지 이를 반복한다.

현재 433Mhz의 ISO 18000-7 표준에 대응하는 형태의 2.4GHz 능동형 RFID 표준은 아직 없다. 다만 실시간 측위를 위한 표준인 ISO 24730에서 2.4Ghz 대역을 사용한다[5]. 이 표준은 중심 주파수가 2.44Ghz이고 채널 대역폭이 60Mhz인 단일 채널 시스템이다. 데이터 전송율은 DSSS를 사용할 경우 59.7Kbps, OOK를 사용할 경우 19.83Kbps이다. DSSS를 기준으로 보면 ISO 18000-7 보다 데이터 전송율은 높지만 최대 메시지의 크기가 152 비트로 제한되어 있어 정보 송, 수신에는 적합하지 않다.



하나의 CP (Collection Period) 라운드
 그림 1 ISO 18000-7 능동형 RFID 표준의 인식과정

표준과 별개로 2.4Ghz 대역을 사용하여 RFID 시스템을 구현한 예로 [6,7]이 있다. [6]은 상용 제품으로 개발된 것이고, [7]은 연구 결과물인데 RFID의 기술적 측면보다 응용 서비스를 강조하고 있다. 다만 두 관련 결과물 모두 채널의 대역폭, 다중 채널 사용 여부, 프로토콜 등에 관한 설명이 없어 연구와 직접 비교하기 어렵다. 이러한 관련 연구, 개발 상황을 고려할 때 2.4Ghz RFID 관련 연구, 개발, 특히 다중 채널이나 다중 인터페이스에 대한 고려는 매우 초기 단계에 있는 것으로 파악된다.

3. 다중 채널의 구성 및 관리

2.4Ghz 대역은 활용 가능한 주파수 대역이 100Mhz 이상이다. 따라서 ISO 18000-7 표준의 5배에 해당하는 1Mhz를 단일 채널용으로 할당하여도 100여 개 이상의 다채널로 구성할 수 있다. 본 논문에서 채널의 구체적 대역폭이나 변조 기법, 대역 확산 적용 여부 등 개별 채널의 물리 계층 수준 특성을 다루지는 않는다. 단지 활용 가능한 다수의 논리 채널이 있다는 전제 하에 리더와 태그간, 또는 리더와 리더간 프로토콜을 정의하고자 한다.

다중 채널 환경이므로 채널 할당과 채널 동기화라는 두 가지 문제에 대한 해결책이 필요하다. 채널 할당이란 인접한 두 리더가 서로 다른 채널을 사용케 하여 채널 중복에 따른 리더간 충돌 문제를 해결하기 위함이다.

채널이 하나더라도 시간을 나눠 사용함으로써 리더간 충돌을 회피하는 방법도 있지만 성능 저하가 불가피하기 때문에 바람직한 것은 아니다. 채널 동기화란 리더들이 서로 다른 채널들을 사용하므로 태그가 인접한 리더가 사용하는 채널을 알아내어 그 채널로 자신의 채널을 맞추는 것을 의미한다.

이 두 문제를 위해 활용 가능한 다수의 채널 중 하나를 제어 채널이라는 특수 채널로 지정한다. 이 채널은 리더들이 채널 할당 및 동기화 등을 위해 제어 정보를 인접한 리더나 태그들에게 전할 때 사용한다. 제어 채널을 제외한 나머지 채널들은 기존과 같이 리더와 태그간 통신을 통한 위해 사용하는 데이터 채널로 지정한다.

인접 리더들에게 서로 다른 채널을 할당하는 것은 본질적으로 그래프 색 지정 문제(Graph Coloring Problem)다[8]. 즉 임의의 리더 집합에 대해 충돌이 없도록 채널을 할당하고자 할 때 필요한 최소의 채널 수를 구하는 것은 NP-Complete 문제다. 리더간 충돌은 성능에 미치는 영향이 큰 중요 문제로 특히 최신 수동형 RFID 표준인 Gen2시스템과 관련하여 [9] 등의 여러 연구 결과들이 발표되었다.

그런데 제안하는 능동형 RFID 시스템의 경우 가능 채널 수가 100여 개로 대부분의 실용 환경에서 필요한 채널 수 보다 많다. 또 우리는 그래프 색 지정 문제와 달리 굳이 사용하는 채널의 수를 최소화 할 필요도 없다고 가정한다. 이에 따라 제안 시스템에서의 리더간 충돌

돌 문제는 간단해진다. 대신 개별 리더가 지역 정보만을 바탕으로 자신의 채널을 스스로 결정할 수 있는 분산형 채널 할당 알고리즘이 시스템의 효과적 관리를 위해 필요하다.

우리는 제안하는 시스템을 위한 분산형 채널 할당 방법을 개발하였는데 개발적 동작 방식은 다음과 같다. 리더들은 주기적으로 제어 채널에 접속하여 동기 정보를 방송하며 또 다른 리더들이 방송하는 동기 정보를 수집한다. 동기 정보에는 리더 자신이 사용하는 채널 정보, 그리고 자신이 직접 동기 정보를 수신한 이웃 리더들의 채널 정보를 포함한다. 결과적으로 어떤 리더의 채널 동기 정보에는 자신과 1-홉 이웃 리더의 채널 정보가 포함된다.

리더를 처음 구동하면 일정 시간 동안 제어 채널에서 방송되는 동기 정보를 수집한다. 이를 통해 리더는 자신의 2-홉 이내 이웃 리더들과 그 리더들이 사용하는 채널의 목록을 얻을 수 있다. 리더는 2-홉 이내의 이웃 리더들이 사용하는 채널을 피해 활용 가능한 나머지 채널 중 하나를 임의로 선택하여 사용한다. 2-홉 이웃 리더까지 정보를 수집하게 하는 까닭은 2-홉 이웃 리더의 경우 비록 리더 간에 제어 정보를 주고 받을 수 없는 위치에 존재하지만 리더의 동작 영역이 겹칠 수 있기 때문이다.

제어 채널을 사용함에 따라 채널 동기화 문제는 매우 간단한 문제가 된다. 제어 채널이 없다면 태그는 가능한 모든 채널을 순차적으로 검색(Scanning)하면서 동기화를 진행하여야 하는데 본 시스템과 같이 가용 채널이 많은 경우에는 많은 시간과 전력 소모를 필요로 한다.

그러나 제안하는 방법에서는 제어 채널을 통해 방송되는 동기 정보를 이용해 빠르고 효율적으로 동기화를 할 수 있다.

4. 다중 인터페이스리더 및 부하 분배

4.1 분석을 위한 상수 및 변수 정의

표 1은 성능 평가를 위해 정의한 각종 상수 및 변수들이다. W 형태의 값들은 인식과정 시작부터 종료까지 필요한 작업 시간 및 유휴 구간 등을 표시하기 위한 것이다. 단일 인터페이스에 대해 인식과정 시작부터 종료까지의 전체 작업시간을 구하면 식 (4)와 같다. 단일 인터페이스는 인식 종료 시까지 유휴 구간이 발생하지 않으므로 전체 작업 시간 W^1 은 $W_{SP}^1, W_{LP}^1, W_{AP}^1$ 의 세 가지 요소로만 구성된다. 세 가지 요소 중 W_{SP}^1, W_{LP}^1 은 알로하 알고리즘의 확률적 특성에 영향을 받는 가변적 요소이며, 나머지 W_{AP}^1 은 알로하 알고리즘에 영향을 받지 않고 부가 정보 처리 시간 $t_r(i)$ 에 의해 결정되는 고정적 요소이다.

성능 척도로 태그 당 처리 소요 시간 τ^M 을 사용한다. 추가로 τ_A^M 를 사용하는데 이것은 AP 구간을 제외한 처리 소요 시간으로 알로하 알고리즘이나 적용 프로토콜 등의 가변적 요소만으로 결정되는 값으로 적용 프로토콜의 차이를 보이는데 적합한 척도이다.

$$W_{SP}^1 = \gamma^1 \cdot t_{cc}, W_{LP}^1 = \sum_{k=1}^{r^1} \omega_k^1, W_{AP}^1 = \sum_{i=1}^N t_r(i) \cong N \cdot t_r, W_{IP}^1 = 0, W^1 = W_{SP}^1 + W_{LP}^1 + W_{AP}^1 \tag{4}$$

표 1 분석을 위해 사용할 상수와 변수

상수 및 변수	의미
$t_{slot} = 1$	상수, LP 구간 한 슬롯의 길이, 단위 시간으로 활용, 즉 값은 1, 그림 1 참조
t_{cc}	상수, Collect 명령 전송에 걸리는 시간 = 0.5 (ISO 18000-7 을 참조하여 결정), 그림 1 참조
$t_r(i), t_r$	상수, 태그 $i(1 \leq i \leq N)$ 의 AP 구간 소요 시간, 개별 태그의 부가 정보에 양에 종속적인 값. 평균 값 $E[t_r(i)]$ 를 t_r 로 표시, 그림 1 참조, 특별히 표시하지 않은 경우 $t_r=10$ 으로 하여 실험하였음
$W_{SP}^M(m, protocol)$	인식과정 시작부터 종료까지 다중 인터페이스를 위해 "protocol"을 사용하는 M 개의 인터페이스를 가진 리더의 m 번 인터페이스가 수행하는 모든 SP / LP / AP 구간들의 작업 시간의 합, $1 \leq m \leq M$, 그림 7 참조, m 값을 지정하지 않은 경우는 모든 인터페이스에 대한 합
$W_{LP}^M(m, protocol)$	
$W_{AP}^M(m, protocol)$	
$W_{IP}^M(m, protocol)$	인식과정 시작부터 종료까지 m 번 인터페이스의 유휴 구간(Idle Period)의 합, m 값을 지정하지 않은 경우는 모든 인터페이스에 대한 합, 그림 6 참조
$W^M(protocol)$	인식 종료까지 모든 인터페이스들의 모든 작업 시간 및 유휴 구간의 합 $= W_{AP}^M(protocol) + W_{LP}^M(protocol) + W_{SP}^M(protocol) + W_{IP}^M(protocol)$
$\tau^M(protocol)$	1개 태그를 처리하는데 필요한 평균 소요 시간 $= W^M(protocol) / (N \cdot M)$
$\tau_A^M(protocol)$	알로하 시간, τ^M 에서 AP 구간 소요 시간을 제외한 시간, $\tau_A^M(protocol) = \tau^M(protocol) - t_r / M$
$\gamma^M(protocol)$	N 개 태그 전체를 처리하는데 필요한 CP 라운드 수
$\rho^M(protocol)$	알로하 부하, 알로하 인식 과정에서 경쟁 윈도우 구간 외에 추가로 필요한 시간

4.2 경쟁 윈도우 크기 분석 및 개선 방안

앞서 태그 수 추정을 바탕으로 알로하 알고리즘의 윈도우 크기를 설정하는 기존 연구를 소개하였다. 그런데 이를 그대로 능동형 RFID 인식 프로토콜에 적용하는 것은 적절하지 않은데 그 이유는 경쟁 윈도우 구간, 즉 LP 구간 외에 알로하 인식 과정에서 발생하는 Collect 명령 전송을 위한 SP 구간과 같은 추가 부하를 고려하고 있지 않기 때문이다. 혼동을 방지하기 위해 이후 이를 알로하 부하 ρ^M 라고 정의하고 사용하겠다(표 1 참조).

단일 채널인 경우 알로하 부하로는 Collect 명령 전송 부하만 존재하므로 $\rho^1 = t_{\alpha}$ 이다. 이렇게 매 알로하 과정마다 경쟁 구간 LP 외에 알로하 부하 ρ 가 있을 경우 알로하 알고리즘의 성능을 최적화 시킬 수 있는 윈도우 크기 ω 는 식 (5)와 같이 구할 수 있다.

$$\omega = \frac{N + \sqrt{N^2 + 4 \times (N-1) \times \rho}}{2} \quad (5)$$

예를 들어 $N=256$ 인 조건에서 ρ 가 1, 8, 16, 32, 100 일 때 ω 값은 각각 257, 264, 271, 333이며 ρ 가 클수록 ω 값이 커진다. 이렇게 되면 한 LP에서 인식되는 태그의 수가 증가하여 CP 라운드 수 γ 를 줄이는 효과가 있다. 실제 ω 를 N 으로 했을 때는 평균 약 13번의 CP 라운드가 필요했지만 식 (5)와 같이 하였을 경우는 각각 12, 10, 9, 8, 7번의 CP 라운드가 필요하였다. 즉 CP 횟수를 줄여 알로하 부하 ρ 의 영향을 줄이는 것이다.

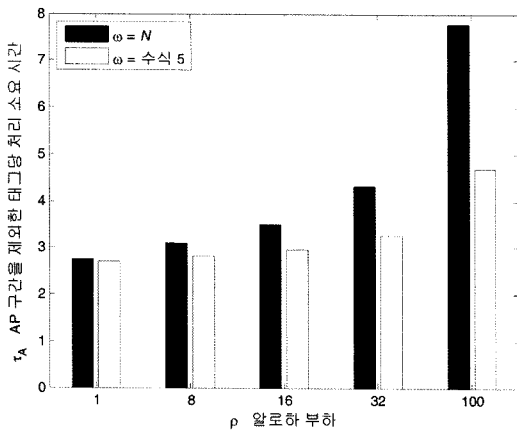


그림 2 부하 ρ 를 고려한 윈도우 크기 설정시의 성능 비교

그림 2는 앞의 ρ 들에 대해 $\omega=N$ 으로 한 경우와 ω 를 식 (5)와 같이 한 경우에 대한 알로하 시간 t_A 를 구한 것이다. ρ 가 클수록 둘 사이의 성능 차가 커짐을 확인할 수 있다. 이후 우리가 사용할 윈도우 크기는 알로하 부하 ρ 를 고려한 식 (5)를 이용하여 결정한다.

N 개의 태그를 모두 인식하는데 필요한 CP 라운드 수에 해당하는 γ^1 은 알로하 알고리즘에 영향을 받는 확률 변수이다. 계산을 통해 그 기대 값 $\overline{\gamma^1}$ 를 구할 수 있는데 윈도우 크기 결정 방법에 따라 달라진다. 우리가 제안하는 식 (5)의 윈도우 크기 결정 방법에 대해 $\overline{\gamma^1}$ 를 간단한 수식으로 표현하기는 어렵지만 이것의 상한은 식 (6)과 같이 구할 수 있다. 지면 관계상 유도 과정은 생략한다. 그림 3은 계산을 통해 얻은 $\overline{\gamma^1}$ 기대 값과 상한이다.

$$\overline{\gamma^1} < \frac{\log(N)}{1 - \log(e-1)} \approx 2.18 \cdot \log(N) \quad (6)$$

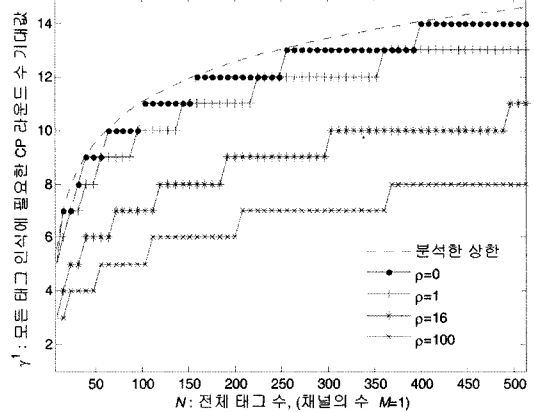


그림 3 태그 수에 따른 CP 라운드 수 기대 값

4.3 단순결합 프로토콜의 인터페이스 집중 문제

리더에 여러 개의 인터페이스를 장치하는 것은 속도 증대에 목적이 있다. 그러나 인터페이스간 부하 분배가 제대로 이루어지지 않으면 성능 개선이 기대에 미치지 못할 수 있다. 부하 분배를 위한 특별한 상호 작용 없이 각 인터페이스가 독립적으로 동작하는 방식을 단순결합(Aggregated)이라고 하고 이것의 성능을 분석하여 좀 더 부하 분배의 필요성을 설명하겠다.

단순결합 방식의 성능 분석을 위해 이론적 분석과 시뮬레이션을 병행하였다. 시뮬레이션은 동일한 인자에 대해 난수원(Random Seed)을 바꾸면서 100회씩 수행한 결과를 이용하였다. 이후의 시뮬레이션 결과들 역시 동일한 방식으로 도출하였다.

단순결합은 태그들의 초기 채널 동기화 분포에 따라 성능이 크게 달라진다. 예를 들어 모든 태그가 한 채널로 동기화할 경우 해당 채널을 담당하는 인터페이스에서만 실제 통신이 일어난다. 그림 4는 전체 태그 수 $N=256$, 이들 중 1번 인터페이스의 채널로 동기화한 태그의 수를 N_2 이라 할 때 N_2/N 을 50%~100%로 변화시키면서 구

한 단위 시간 당 평균 태그 처리 수 π 를 단일 인터페이스 리더와 비교한 것이다. 50%인 경우, 즉 두 인터페이스가 공평하게 부하를 나눠 가질 때는 성능이 2배정도 증가하지만, 100%인 경우, 즉 모두가 1번 채널로 동기화할 경우 성능 개선이 전혀 없음을 확인할 수 있다.

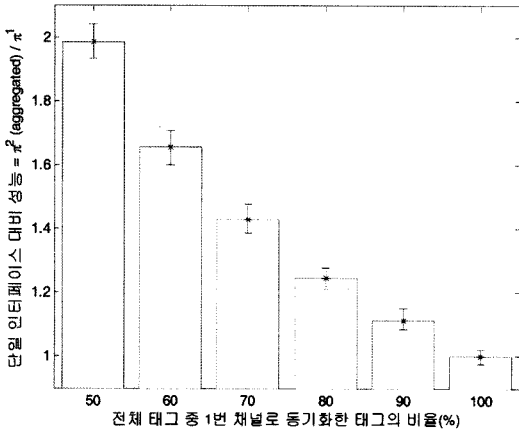


그림 4 단순결합에서 태그 초기 채널 분포에 따른 성능

물론 태그들이 리더가 지원하는 채널을 같은 확률로 선택할 경우 특정 채널에만 태그들이 몰리는 가능성이 높지는 않다. m 번째 인터페이스가 동작하는 채널로 동기화한 태그의 수를 N_m 이라고 하자. $M=2$ 일 때 N_2 과 N_2 의 차의 기대 값은 식 (7)과 같다.

$$E\{|N_2 - N_2|\} = \begin{cases} N \cdot \left(\frac{1}{2}\right)^N \cdot \binom{N}{n}, & N=2n, n\text{는 정수} \\ N \cdot \left(\frac{1}{2}\right)^{N-1} \cdot \binom{N-1}{n}, & N=2n+1 \end{cases} \quad (7)$$

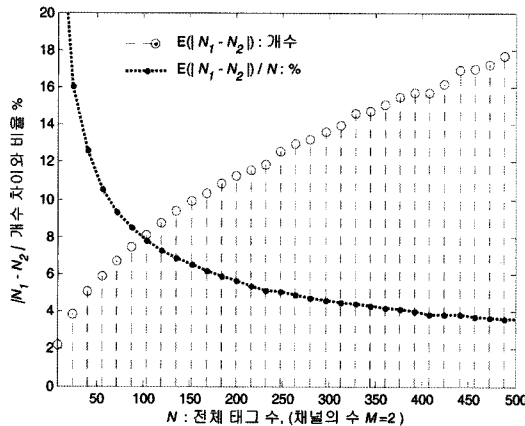


그림 5 채널 임의의 선택 시 채널 간 태그 수 차이 평균과 전체 태그 수 대비 비율

리더의 인식 과정은 자신의 모든 채널에서 인식을 완료할 때 끝난다. 개별 채널의 인식 시간은 해당 채널로 동기화한 태그 수와 비례 관계에 있다고 볼 수 있다. 따라서 가장 많은 태그가 동기화된 채널의 인식 시간 기대 값이 리더의 인식 시간 기대 값이라 할 수 있다.

그림 5는 다양한 전체 태그 수에 대해 $E\{|N_2 - N_2|\}$ 을 구한 것이다. 예를 들어 전체 태그가 256개라면 두 채널의 태그 수 차이 기대 값은 13개다. 즉 한 채널에는 134.5개 수준의 태그가 다른 한 채널에는 121.5개 수준의 태그가 동기화된다. 리더의 인식 시간은 134.5개의 태그 수에 비례한다고 볼 수 있고 128개씩 두 채널로 공평하게 나눠졌을 때에 비해 약 5% 정도의 더 오래 인식 시간, 즉 성능 저하가 평균적으로 발생할 수 있음을 뜻한다.

그림 6은 인터페이스 수를 늘리면서 시뮬레이션을 통해 구한 채널간 태그 수 차이의 평균값과 최대 값을 나타낸 것이다. 인터페이스 수가 늘어나도 인터페이스 간 태그 수 차이는 여전히 발생함을 확인할 수 있다.

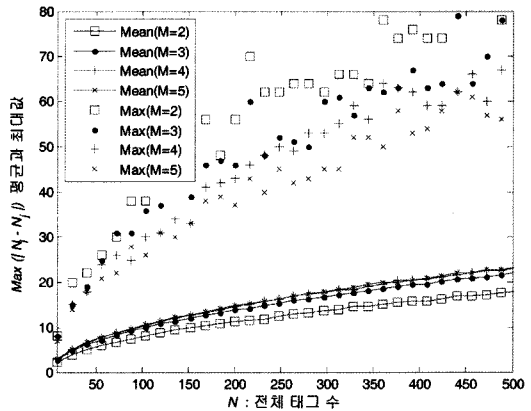


그림 6 채널 임의의 선택 시 태그 수 차이 평균과 최대값

태그 수 256개를 기준으로 평균 5% 성능 저하를 아주 나쁜 결과라고 평가하기 어려울 수 있다. 그러나 이는 평균 성능이며 그림 6의 최대 값에서 유추할 수 있듯이 성능이 매우 나빠질 수 있음을 고려하여야 한다.

또 위 분석은 매 인식 과정의 초기에 모든 태그가 리더의 채널 중 하나를 동일한 확률로 선택한다는 가정하에 구한 것이다. 태그들이 지난 인식 과정에 사용하였던 채널을 그대로 사용한다든지 등의 원인으로 특정 채널로 몰리는 현상이 가중될 경우 그림 4에서 확인한 것과 같은 성능 저하가 발생하고 이를 인식 과정 내에서 개선할 수 없다.

4.4 인식 과정 중의 부하 재분배 방안

단순결합의 초기 채널 치우침 문제를 극복하기 위해 인식 과정 중에 태그들이 스스로 또는 리더의 명령에 의해 채널을 변경하는 방법을 제안한다.

먼저 각 인터페이스의 SP, LP 구간을 동기화시키고 결합하여 사용하는 LP-조합(Combined) 프로토콜을 제안한다. 이것은 Collect 명령 전송 때 윈도우 값 외에 선택 가능한 채널의 수 M 과 동기화 정보를 포함시킨다.

모든 채널의 윈도우 값은 Ω 로 같고 태그는 Random ($1, M \times \Omega$)를 실행시켜 응답할 슬롯을 선택한다. 예를 들어 $M=2$ 이고 $\Omega=8$ 인 상황에서 Random 함수 값이 6인 경우는 현재 채널의 6번째 슬롯에서, 15인 경우 다른 채널의 7번째 슬롯에서 응답한다. 결과적으로 매 CP마다 랜덤 함수에 의한 부하 재분배가 이루어지며 초기 동기화 분포의 치우침에 따른 성능 저하를 해결할 수 있다.

그런데 LP조합 방식은 SP, LP 구간 동기화에 따른 비용이 발생한다. 예를 들어 1번 채널의 LP 구간에서 5개 태그를 인식했고 2번 채널서 1개의 태그를 인식 했을 때 5개를 처리하는 1번 채널의 AP 구간이 1개를 처리하는 2번 채널의 AP 구간보다 상당히 길 수 있고 그 차이만큼 2번 채널에 유휴 구간, 즉 IP가 발생한다. 그리고 이러한 IP는 매 CP 구간마다 발생할 수 있다(그림 7 참조).

이러한 IP는 일종의 알로하 부하로 간주할 수 있다. 이 IP의 길이를 수학적으로 정확히 분석해 내지는 못했다. 대신 다양한 인자 값에 대한 시뮬레이션을 통해 그

값을 구했고 이를 통해 매 CP 구간마다 평균 $(M-1) \times 2 \times t_r$ 수준의 IP가 발생한다고 단순화시켜도 많은 경우에서 별 무리가 없음을 확인하였다. 우리는 이렇게 유도한 IP를 알로하 부하에 반영하여 경쟁 윈도우 크기를 결정하였다.

LP-조합 방법이 가지는 알로하 알고리즘의 인식 결과 차이에 따른 문제점을 완화하기 위해 AP-균형(Balanced) 프로토콜을 제안한다. 이 방법에서는 채널변경(Channel Switch)이라는 명령을 추가하여 위 예와 같은 경우 LP 구간에서 2번 채널로 인식된 태그 일부를 1번 채널로 채널 변경시키고 처리하게 하였다(그림 7 참조).

태그에서 채널을 변경하는 데는 지연이 발생하지만 부가 정보 및 명령 전송에 걸리는 시간에 비해 매우 작기 때문에 성능에 미치는 영향이 미미하다.

채널 변경을 통해 AP 구간의 부하의 균형을 맞추려면 각 태그의 부가 정보 길이 $t_r(i)$ 가 필요하다. 부가 정보 길이를 알 수 없는 경우 각 인터페이스는 AP구간에서 매 태그 처리 시점마다 자신에게 남아 있는 부가 정보 처리가 필요한 태그의 수를 확인하여 2개 이상인 경우 다른 인터페이스의 상태를 확인한다. 유휴 상태에 있는 다른 인터페이스가 있을 경우 처리 대상 큐의 제일 앞에 있는 태그에게 채널 변경 명령을 보내어 그 채널로 변경하게 한다. 이렇게 하면 AP 구간이 가장 긴 한 인터페이스를 제외한 나머지 인터페이스들에서 최악의 경우 $2 \times t_r$ 만큼의 유휴 구간이 발생할 수 있다.

$t_r(i)$ 를 알 경우 유휴 구간을 보다 줄일 수 있다. $t_r(i)$ 를 알 때 유휴 구간 W_{IP}^M 을 최소화하는 것은 수 분리 문제(Number Partitioning Problem)와 같다. 수 분리 문제란 양의 정수 목록 a_1, a_2, \dots, a_N 에 대하여 어떤 부분 집합 A 를 정의하였을 때 $|\sum_{i \in A} a_i - \sum_{i \notin A} a_i|$ 의 값을 최소로 만드는 집합 A 를 구하는 것이다[10]. 이 문제는 앞서 소개한 그래프 색 지정 문제와 같이 NP-Complete 문제다.

수 분리 문제는 최적의 해를 구하기 위한 여러 휴리스틱 알고리즘들이 있으므로 유휴 구간을 최소화하기 위하여 이를 활용할 수 있다. 그런데 이런 알고리즘들은 복잡도가 매우 높다. 임베디드 시스템 기반의 리더의 성능 및 고속의 처리 속도를 고려할 때 이들을 적용하는 것이 어려울 수 있다. 더욱이 최적화를 통해 얻는 이익이 크지 않을 경우 이러한 알고리즘의 적용은 바람직하지 않다.

우리는 $t_r(i)$ 의 크기에 따라 태그들을 정렬하고 채널 간 AP 구간 차에 근접하는 태그를 선택하여 채널 변경하게 하는 간단한 알고리즘을 적용하였다. 지면 관계상이 알고리즘에 대한 설명은 생략한다.

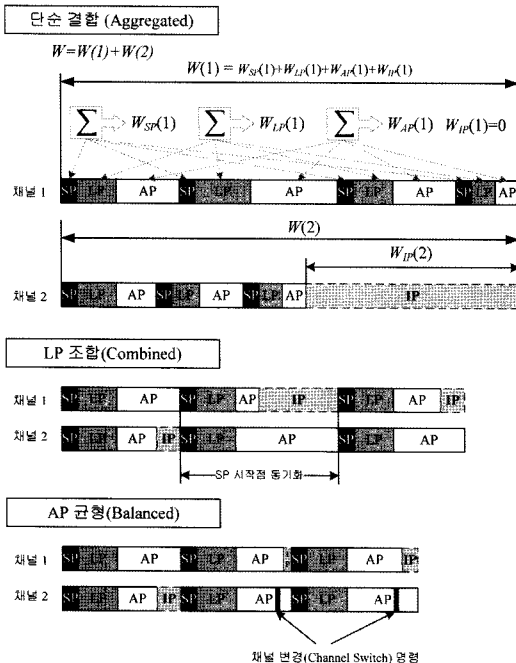


그림 7 프로토콜 별 SP, LP, AP, IP 구간 구성의 예

채널 변경으로 AP 구간의 차이를 줄일 수는 있지만 완전히 제거하지는 못한다. 예를 들어 채널 1에서 인식한 태그 수가 5개, 채널 2가 2개인 경우 두 채널의 인식 태그 수 차이가 홀수이므로 완전히 공평하게 나눌 수는 없고 한 쪽이 하나가 많아진다. 차이가 짝수라면 공평하게 나눌 수 있을 것이다. 결국 평균적으로는 1/2 개 수준의 차이가 날 것이다. 이러한 분석을 M개의 채널 수로 확대하면 AP-균형 방법의 경우 매 CP 구간마다 평균 $(M-1) \times t_r / 2$ 수준의 IP가 발생한다고 할 수 있다. 이러한 IP 예측 값을 알로하 부하에 반영하여 경쟁 윈도우 크기를 결정하였다.

예측한 평균 IP 길이를 기준으로 LP-조합의 IP 길이는 AP-균형의 4배에 이르며 그 만큼 AP-균형의 성능이 우수함을 예상할 수 있다. AP-균형은 LP-조합을 기초로 인식된 태그 수 차이의 악영향을 줄이는 채널 변경 기능을 추가한 것으로 LP-조합 보다 항상 좋은 성능을 가진다.

지금까지 소개한 세 가지 프로토콜의 SP, LP, AP, IP 구간의 구성 예는 그림 7과 같다. 그림 7을 보면 LP-조합과 AP-균형은 채널 간 SP 시작점을 동기화시키는데 비해 단순결합은 이러한 동기화를 하지 않는다. AP-균형은 채널 변경 명령을 통해 AP 구간의 차이를 줄일 수 있다는 점이 LP-조합과 다른 점이다.

4.5 성능 평가

앞서 소개한 세 가지 프로토콜을 비교하기 위하여 다양한 조건에 대해 시뮬레이션을 수행하였다. 성능 척도로는 τ^M 과 τ_A^M 을 사용했는데 특히 프로토콜이 미치는 영향을 평가하기 위해 프로토콜에 영향을 받지 않는 AP 구간을 제외한 τ_A^M 를 주로 활용하였다.

첫 째 실험은 $M=2$ 인 리더에서 태그 수를 변경 시키면서 τ_A^2 의 평균과 최대값을 측정된 실험으로 결과는 그림 8에 나타났다. 평균 성능도 중요하지만 최악 상황 성능도 중요한 지표라고 할 수 있기 때문에 평균뿐 아니라 최대 값도 표현하였다. 태그의 수가 늘수록 성능이 좋아지는 것을 확인할 수 있는데 이는 알로하 부하가 다수의 태그들에게 분배되기 때문이다.

세 방법 중 AP-균형 방법이 평균, 최대 모두에서 가장 우수한 성능을 보이며 태그 수 차이에 따른 성능 차이가 가장 적음을 확인할 수 있다. LP-결합 방법의 성능이 가장 나쁘는데 이는 서로 다른 채널들의 AP 구간 길이 차에 따른 유휴 구간이 상당히 크다는 것을 의미한다. 단순결합 방법은 최악 상황 성능이 매우 나쁜 것을 확인할 수 있다. 유의할 만한 특징은 AP-균형 방법의 최악 상황 성능이 단순결합 방법의 평균 성능과 유사하다는 점이다.

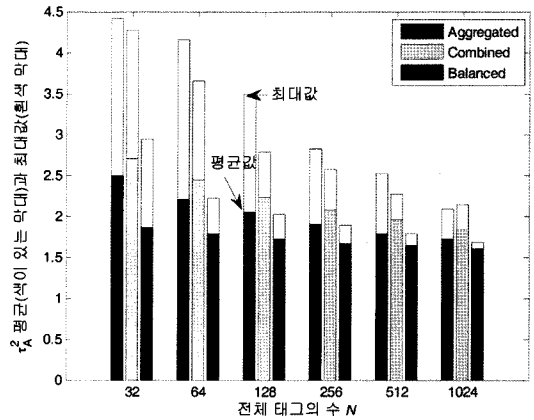


그림 8 다양한 태그 수에 대한 성능(알로하 시간) 비교

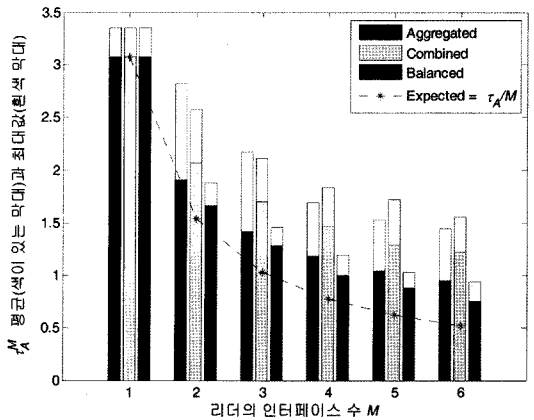


그림 9 인터페이스 수에 따른 성능 τ_A 비교

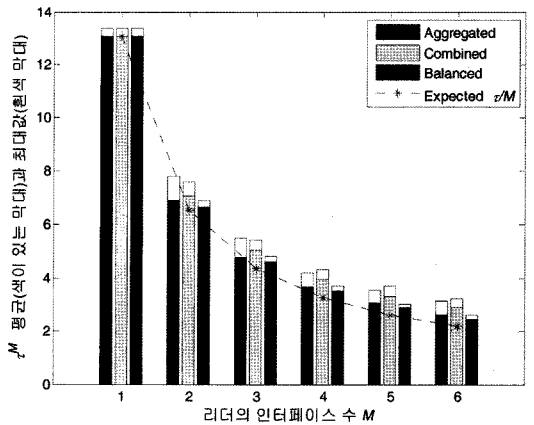


그림 10 인터페이스 수에 따른 성능 τ 비교

둘째 실험은 태그의 수 $N=256$ 인 조건에서 리더의 인터페이스 수 M 을 2에서 6까지 늘리면서 세 프로토콜의 성능을 비교한 것이다. 그림 9는 알로하 시간 τ_A 를 비

교한 것이고 그림 10은 태그 당 평균 처리 소요 시간 τ 를 비교한 것이다. 단일 인터페이스에서 하나의 태그를 처리하는데 1이라는 시간이 필요했다면 인터페이스가 M 개이면 $1/M$ 시간이 소요될 것으로 기대할 것이다. 점선 그래프는 이러한 기대 성능을 표시한 것이다.

세 방법 모두 기대 성능에는 미치지 못함을 확인할 수 있다. τ 에서의 기대 성능과 실제 성능의 차에 비해 알로하 시간에서 차가 더욱 두드러진다. 이는 알로하 시간에 다중 인터페이스 지원에 따른 부하가 모두 포함되어 있으므로 당연한 결과이다. 인터페이스 수가 늘더라도 AP-균형이 가장 우수한 성능을 보이며 최악 상황 성능이 단순결합의 평균 성능과 유사한 특성은 그대로 유지되었다.

셋째 실험은 태그의 부가 정보량이 성능에 미치는 영향을 살펴보기 위한 것이다. $M=2$ 이고 $N=256$ 인 조건에서 수행하였다. 이전 실험들은 모두 $t_r=10$ 으로 두고 한 것이지만 이번 실험에서는 t_r 을 1에서 100까지 다양한 값에 대해 수행하였다. 알로하 시간을 비교한 그림 11에서 확인할 수 있듯이 t_r 의 증가는 알로하 시간의 증가를 가져온다. 단순결합과 LP-조합의 t_r 증가에 따른 성능 저하는 두드러지지만 AP-균형은 상대적으로 그 영향이 적다. t_r 이 증가할수록 AP 균형의 성능 우위가 더욱 두드러지는데 예를 들어 $t_r=100$ 인 경우 AP 균형의 성능을 단순결합과 비교하면 평균 성능은 2배, 최악 상황 성능은 4배 수준에 가깝다.

프로토콜에 영향을 받는 알로하 시간의 성능 차는 매우 크지만 고정적인 AP 구간을 포함하여 살펴볼 때는 크지 않은 것으로 보일 수도 있다. 그림 12는 τ 를 비교한 것이다. t_r 이 증가함에 따라 τ 에서 AP가 차지하는 비중이 급격히 늘어난다. 결과적으로 전체 태그 처리 시간 측면에서는 AP 균형이 단순 결합에 비해 평균 기준

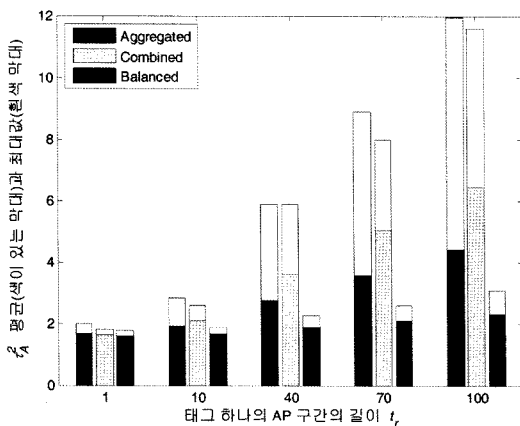


그림 11 태그의 부하 정보량에 따른 성능 τ_A 비교

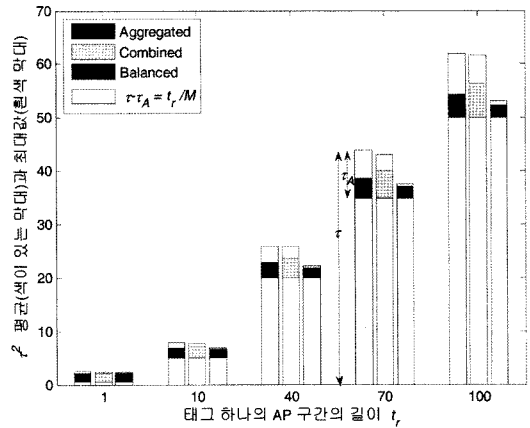


그림 12 태그의 부하 정보량에 따른 성능 τ 비교

으로 10% 미만 수준의 성능 개선만 이룰 수 있다. 물론 최악 상황 성능 측면에서 개선은 이 보다 크다.

태그 수 N , 인터페이스 수 M , 부가정보량 t_r 등을 변화시키며 살펴본 성능 평가를 통해 AP-균형이 가장 우수한 성능을 보임을 확인할 수 있다.

결과에서 유의할 특징은 AP-균형의 최악 상황 성능이 단순결합의 평균 성능 수준이라는 점이다. AP-균형은 매 알로하 인식 과정, 즉 매 CP 마다 최악의 경우 $(M-1) \times t_r$ 수준의 유휴 구간이 발생할 수 있다. 인식 완료까지 필요한 평균 CP 횟수는 식 (6) 등을 통해 살펴 보았듯이 $C \cdot \log(N)$ 수준이다. 따라서 전체 유휴 구간은 최악의 경우 $C \cdot \log(N) \cdot (M-1) \cdot t_r$ 이라고 할 수 있다.

이에 비해 단순결합의 유휴 구간 채널 별 태그의 차이에 의해 결정되며 $C \cdot E[|N_i - N_j|] \cdot (M-1) \cdot t_r$ 수준이다. 앞에서 채널 별 태그 수 차이의 평균을 식 (7)이나 그림 5, 6 등을 통해 도출하였다. 비교 결과 N 이 증가하면 $E[|N_i - N_j|]$ 는 $\log(N)$ 에 비해 빠르게 증가함을 확인할 수 있었다. 이러한 이유로 AP-균형의 최악 상황 성능이 단순결합의 평균 성능 수준이 되는 것이다.

5. 결론

능동형 RFID 시스템의 효율성이 물류 환경의 응용을 중심으로 확인되면서 이를 도입하려는 요구가 늘고 있다. 그런데 433Mhz에서 단일 채널을 사용하는 기존 ISO 18000-7 능동형 RFID 표준은 낮은 전송 속도, 단일 채널에 따른 간섭 문제 등으로 기술적 한계에 직면하고 있다. 이를 극복하기 위해 우리는 2.4Ghz 대역에서 다수 채널을 사용하는 새로운 능동형 RFID 시스템을 제안한다. 이 논문에서는 이 다중 채널 능동형 RFID 시스템에서 하나의 리더에 여러 개의 인터페이스를 부착하여 개별 리더의 성능을 증대시키는 방안에 대해 중

점적으로 다루었다.

하나의 리더에 다수의 인터페이스를 장착하여도 대부분의 태그들이 특정 인터페이스로 몰린다면 기대하는 성능 개선이 어렵다. 따라서 이를 극복하기 위해서는 인터페이스간 부하 분배를 고려한 능동형 RFID 프로토콜이 필요하다. 논문에서는 단순결합, LP-조합, AP-균형이라는 세 가지 다중 채널 다중 인터페이스 능동형 RFID 프로토콜을 제안하고 비교, 분석하였다. 태그의 수, 인터페이스 수, 태그의 부가 정보의 양 등을 변화시키면서 수행한 다양한 성능 평가 결과 AP-균형이 가장 우수한 성능을 보였다. 특히 AP-균형은 최악 상황의 성능이 단순결합의 평균 성능 수준으로 성능의 안정성 측면에서도 우수한 특성을 보였다.

결론적으로는AP-균형 프로토콜을 사용하면 인식 과정 중의 효과적 부하 분배를 통해 리더에 설치한 인터페이스 수에 거의 선형적으로 비례하여 처리 성능이 증가하는 다중 채널 능동형 RFID 리더를 구현할 수 있다.

부하 분배를 위한 다중 채널 능동형 RFID 프로토콜의 설계와 별개로 우리는 인접 리더간 채널 중복에 따른 간섭을 피하기 위한 분산형 채널 할당 방법, 태그와 리더간의 빠른 동기화 방법을 제안하였다. 그리고 알로하 기반 태그 인식 과정에서 경쟁 윈도우 구간 외의 부하를 고려한 효율적인 경쟁 윈도우 크기를 결정 방법을 제안하였다.

현재 본 연구진은 기존 ISO 18000-7 표준에 기반한 2.4GHz 단일 채널 능동형 RFID 시스템을 구현한 상태이다. 앞으로 이 논문에서 소개한 다중 채널 프로토콜을 구현하여 프로토콜의 실용성을 실증하고자 하는 연구 계획을 가지고 있다.

참 고 문 헌

[1] ISO/IEC 18000.7.2, "Information technology AIDC techniques - Radio frequency identification for item management - Air interface, Part 7: Parameters for active air interface communications at 433 MHz," 2008.

[2] C. Law, K. Lee, K. Siu, "Efficient memoryless protocol for tag identification," Proceedings of the 4th international workshop on Discrete algorithms and methods for mobile computing and communications, pp. 75-84, Boston, USA, 2000.

[3] H. Vogt, "Efficient Object Identification with Passive RFID Tags," Proceedings of International Conference on Pervasive Computing, Zurich, Swiss, 2002.

[4] 권성호, 홍원기, 이용두, 김희철, "저비용 RFID 시스템에서의 충돌방지 알고리즘에 대한 성능평가", 한국통신학회논문지, pp. 17-26, 제30권, 1B호, 2005.

[5] ISO/IEC 24730-2, "Real-time locating systems

(RTLS) -- Part 2: 2,4 GHz air interface protocol," 2006.

[6] 2.4Ghz Long Range Active RFID, SynoMetrix, <http://www.synometrix.com/>

[7] Open Beacon Project - a free active 2.4Ghz beacon design, <http://www.openbeacon.org>

[8] D. Engels, "The Reader Collision Problem," White Paper of Auto-ID Center, Nov. 2001, available at <http://www.autoidlabs.org/single-view/dir/article/6/98/page.html>

[9] K.S. Leong, M.L. Ng and P.H. Cole, "Synchronization of RFID readers for dense RFID reader environments," International Symposium on Applications and Internet Workshops, Phoenix, AZ., 2006.

[10] Stephan Mertens, "The Easiest Hard Problem: Number Partitioning," Computational Complexity and Statistical Physics, New York: Oxford University Press 2003, available at <http://arxiv.org/abs/cond-mat/0310317>



박 현 성

1999년 경성대학교 산업공학과 학사. 2007년 부산대학교 컴퓨터공학과 석사. 현재 부산대학교 컴퓨터공학과 박사과정. 관심분야는 무선통신, RFID/USN



김 동 현

1998년 동의대학교 정보통신공학과 학사. 2004년 동의대학교 정보통신공학과 석사. 현재 부산대학교 컴퓨터공학과 박사과정. 관심분야는 이동통신망, RFID/USN



정 상 화

1985년 서울대학교 전기공학과 학사. 1988년 Iowa State Univ. 컴퓨터공학과 석사. 1993년 Univ. of Southern California 컴퓨터공학과 박사. 1993년~1994년 Univ. of Central Florida 컴퓨터공학과 조교수. 1994년~현재 부산대학교 컴퓨터공학과 교수, 컴퓨터 및 정보통신연구소 연구원. 2002년~2003년 Oregon State Univ. 컴퓨터공학과 초빙교수. 관심분야는 클러스터 시스템, TOE, RDMA, WMN, RFID



백 윤 주

1990년 한국과학기술원 전산학과 학사
1992년 한국과학기술원 전산학과 석사
1997년 한국과학기술원 전산학과 박사
1999년~2002년 NHN 기술연구소 소장
2003년~현재 부산대학교 컴퓨터공학과
부교수. 관심분야는 임베디드시스템, 센

서 네트워크



김 종 덕

1994년 서울대학교 계산통계학과 학사
1996년 서울대학교 전산학과 석사
2003년 서울대학교 컴퓨터공학과 박사
2004년~현재 부산대학교 컴퓨터공학과
조교수, 컴퓨터 및 정보통신연구소 연구
원. 관심분야는 무선통신, 이동통신망,

RFID/USN